Payoff

If IS is to play an active role in the effort to better control cost, then system refurbishment should be included as part of an overall cost-reduction plan. This article specifically targets legacy systems by presenting an approach for maintaining and repositioning these aging systems to make them more cost-effective and competitive. IS can similarly use this approach to achieve substantial benefits from refurbishing newer systems that have been developed using loose standards or that have experienced several modifications.

Introduction

In its effort to support the business need for more information to be available in less time, the IS department has implemented a variety of proprietary technologies, developed new systems, and added capabilities to existing ones. The amount of systems integration, development, and maintenance activity required to satisfy these demands has left IS with little time to properly care for the base of old and rapidly aging systems. These legacy systems are often in a state of disrepair, suffering from the use of outmoded technologies and years of changes at the hands of different IS personnel who used different programming styles and formats. For many companies, these systems have been left to age with few (if any) improvements to the program structure, complexity, or hardware and software technologies.

Currently, however, the IS department's new directives are to reduce cost and improve financial performance. Departmental budgets are being cut, business processes and computer systems are being reengineered, application systems are moving away from a centralized data center and distributed to the operating divisions, and IS is being outsourced or downsized. When the cost associated with maintaining legacy systems is evaluated, it is increasingly apparent that, perhaps more than ever, management must seriously consider system refurbishment as an integral component of cost containment and business process reengineering, and as a baseline for outsourcing, systems reengineering, and downsizing activities. This article presents an effective approach to maintaining and repositioning legacy systems to align them with the goal of systems that are cost-effective and competitive.

System Refurbishment and Business Process Reengineering

The increasing popularity of business process reengineering has placed greater pressure on IS to keep pace with similar systems reengineering projects. One study done by G2 Research, Inc., projected that the total market for business process reengineering will have grown from $5.3 billion in 1992 to $12.5 billion in 1997. The systems reengineering market for this same period is expected to grow from $17.6 billion to $39.8 billion. Among the companies that have implemented and realized the benefits of business process or systems reengineering projects are Shearson Lehman Brothers, KLM Engineering and Maintenance, and Nalco Chemical Company. Companies can adopt an approach to maintain and reposition their legacy systems to make them more cost-effective and competitive.
competitive so that they can support activities associated with business process reengineering.

Legacy systems share similar characteristics regardless of industry, business, or application focus. Usually, these systems are more than seven years old, may or may not be mission critical, use outmoded or different proprietary technologies, have poorly structured program code, have ineffective reporting systems, and use system and human resources inefficiently. To further complicate these systems, the original design and development team may have changed, leaving the current support team without a complete understanding of the detailed operation of the system. In other words, legacy systems are usually the systems that everyone fears and no one wants to support.

How do companies determine the strategic contribution of their legacy systems? How do they know which systems to keep? What if the operations or development staff is outsourced? Is it possible to convert a legacy system into a more competitive tool that allows a company to provide higher-quality products and services in less time? The answers to these questions can be found in a refurbishment approach that has been successfully implemented in a variety of businesses and industries.

The Refurbishment Process

The refurbishment process encompasses (encapsulates) an entire system, which may be of any size or complexity. The refurbishment process also provides IS with the ability to evaluate its functional and technical attributes and recondition the system to improve cost and maintainability. The process comprises five key phases: preliminary inventory analysis, encapsulation, application analysis, production standardization, and design recovery. Each phase is defined briefly as follows:

- Preliminary inventory analysis. This phase determines the scope of the refurbishment effort (i.e., which systems to include) and establishes the priorities of the systems to be refurbished.

- Encapsulation. This phase generates an inventory of the components of a system.

- Application analysis. This phase evaluates applications according to three aspects:
  - Functional fulfillment.
  - Technical quality.
  - Fit with the IS strategic plan.

- Production standardization. This phase eliminates many past mistakes and provides an understanding of the functional and technical aspects of the system. Design recovery. This phase provides detailed system documentation and positions the application software to be reverse and forward engineered.

The results of the refurbishment effort, together with other forces that influence a system's strategic direction, are used to determine the refurbishment strategy for a system. The key phases of refurbishment and the forces that influence the refurbishment strategy are illustrated in Exhibit 1. Application refurbishment improves the performance and maintainability of a system while also incorporating internal and external business forces.
Forces Influencing the Refurbishment Strategy

Preliminary Inventory Analysis

Before jumping into the refurbishment process, a preliminary analysis of the existing systems is performed to determine the overall scope of the refurbishment effort. This analysis is an abbreviated version of the encapsulation and application analysis phases. During this initial phase, the inventory of applications is quantified and analyzed to determine which systems should be included or excluded from the project. At this point, it is not necessary to develop a detailed inventory of each system's components. However, IS should determine the approximate number of executable jobs, procedures, and programs for each system selected.

While conducting this analysis, IS has the opportunity to determine the relative state of each system (i.e., functional and Technical Quality) as well as to estimate the amount of time required to complete the four remaining project phases. From this analysis, a detailed project work plan can be developed for the remaining phases of the project.

Encapsulation

The encapsulation process is a key aspect of refurbishment because it is this process that ensures that the system components (e.g., job control program source, load modules, copybooks) are identified. An accurate component inventory must be developed before beginning the analysis. Although it is possible to include components in the inventory after the analysis has begun, it may be costly to reproduce the analysis a number of times.

After the inventory is completed, it may be necessary to verify that modules that are executed in production are the same as those identified in the inventory. There are several ways to verify the quality and integrity of the inventory, including reviewing source edit statistics, equating source modules to load modules, and systems testing.

The extent of the effort required to develop an accurate inventory is inversely related to the quality of the controls provided within the IS change management procedures. For example, a change management procedure that incorporates excellent controls requires less inventory and verification effort than one with few controls.

Encapsulation identifies all possible system components and shakes out those that are not a part of the system. Although several software tools are available to assist in the process of defining the component inventory, the use of a combination of both manual and automated analyses provides the most accurate inventory in the least amount of time. This is accomplished by digging through the system manually (system utilities may be used) to identify libraries that contain misplaced system components and by using the automated tools to piece together the remaining components.

By completing the encapsulation process, IS establishes a definitive inventory of all system components. The inventory alone may reduce the IS cost associated with analyzing the production and departmental libraries and any other additional activities that may be required to locate the components that satisfy a user's special request. In addition, the accuracy of the systems analysis required by an new development or maintenance activity may be improved because all of the system components have been identified and are easily located.
Application Analysis

Although refurbishment presents a significant opportunity for IS to effectively improve the performance, maintainability, and cost of legacy systems, it would be shortsighted not to include a more strategic review of these systems. This review should initially evaluate these systems according to three primary attributes:

- Ability to support the functional requirements of the system's users.
- System design and use of technology.
- Conformance with the IS strategy.

The evaluation of these three criteria provides insight about the value that the system provides to the business. A system that adequately supports the business needs of the user but employs outmoded technology is more valuable than one that provides little or no functional support to an organization but uses all the latest technology.

To complete the analysis, the functional and technical attributes of the system are mapped to the IS strategic systems plan to determine the refurbishment strategy for the system, as shown in the System Target Chart in Exhibit 2.

System Target Chart

The forces that influence the refurbishment strategy must be considered, because they may directly impact the order in which revitalization activities are performed. For example, if the business strategy is to outsource IS operations, management should focus the IS effort on those activities that will maximize resource use. This focus will achieve lower operational costs for production systems. Specific areas of focus would include Central Processing Unit utilization, disk and tape utilization, online performance, data base efficiency, system backup and restore procedures, and documentation. The sector in which a system is placed in Exhibit 2 provides a basis for formulating a system refurbishment strategy.

Targeting the Refurbishment Strategy

The System Target Chart is a single representation of two separate charts that have been blended to communicate refurbishment alternatives for one or more systems. One chart evaluates the technical and functional attributes of the system and the second chart evaluates the system's core competencies to determine the system's strategic importance. When combined, the charts form four quadrants (I to IV) and eight sectors (i.e., two sectors per quadrant, one which fits IS strategy and one that does not). Positioning a system within a sector provides a basis for formulating a refurbishment strategy.

The placement of a system within a quadrant represents the functional and technical capacity of the system. Within a quadrant, sector positioning is based on the system's strategic importance. Systems whose core competencies support the IS strategy are placed outside the circle (the bull's-eye) and those that do not support the IS strategy are placed inside the circle, regardless of the system's technical or functional rating.

Each system is rated on the basis of the information gathered during the preliminary inventory analysis and encapsulation phases of refurbishment. In general, systems that fall
into sectors I or II provide little functional support to an organization and therefore are considered less valuable than systems that fall into sectors III or IV.

**Quadrant I.**

Quadrant I of the Target Chart represents systems that provide little functional support to the business and use outmoded technologies or are poorly designed and constructed. Systems in this sector that provide functional support but are technically obsolete will be replaced, whereas the systems in this sector that are functionally obsolete (i.e., no longer support a business function) can be archived or deleted.

An example best illustrates this point. One organization maintained a system that relied on manual processes to collect revenue data in the form of paper receipts. It took 15 to 20 business days to enter the receipt information into the system using a data entry service bureau. The system was poorly designed and the programs were unstructured and difficult to maintain. The users complained that the system forced them to iteratively print reports, compute adjustments on a microcomputer before entry to the system, and reprint the reports to determine the adjustment's net effect. In addition, analyzing and modifying the system reports took a great deal of time, and in many cases, the users had to resort to building a new spreadsheet or data base system to produce their own reports.

Systems in a similar state of disrepair are fairly common. Quadrant I systems usually consume significant IS resources as well as increase user frustration. In the previous example, considerable time was required to create the manual receipts, audit the computerized reports to the manual receipts (what happens if a receipt is lost?), iteratively rerun and analyze reports after adjustments have been made, and develop new or maintain existing unstructured code to create or enhance reports. For this system alone, IS spent a total of one person-year responding to one-time user requests to modify or create reports. This effort was magnified by the fact that the system was supported by only three programmers. A thorough analysis would identify even more areas of cost associated with supporting such systems.

This system is currently being replaced with a new automated data collection system that uses a relational data base on the mainframe. A new conversational interface is also to be written to assist the users with the online processes, and all remaining programs are to be reengineered (using an automated program restructuring tool) to improve maintainability.

**Quadrant II.**

Quadrant II of the Target Chart represents systems that provide little functional support to the business but use current technologies or are well designed and constructed. Technology, no matter how advanced, has little value unless it supports a business requirement. Generally speaking, systems that provide functional support to the business should be replaced; those that do not can be archived or deleted. However, for those systems that do not fulfill functional requirements, the best alternative may be to salvage and redeploy technology components to new or other existing systems.

**Quadrant III.**

Systems that fall into Quadrant III or Quadrant IV (which is discussed in more detail in the next section) support the business functions performed by the organizations that use the system. They provide the greatest flexibility with regard to alternative system
strategies because they provide more business support than Quadrant I and Quadrant II systems. Although both Quadrant III and Quadrant IV systems support the functional requirements of the business, only Quadrant III systems use current technologies, are well-designed and constructed, and use IS resources efficiently.

Quadrant III represents systems that provide the best functional and technical support to the business. These systems are usually the most cost-effective users of IS resources. Supporting the strategic systems plan, Quadrant III systems are best positioned for the future and should continue to be maintained as usual. Those systems that fall within the bull's-eye are positioned to be reverse engineered to take advantage of other technologies (e.g., hardware platform, data base, or communications).

A simplified example of a Quadrant III system is an online purchasing system that satisfies the user's functional requirements, is well designed and coded (i.e., easy to maintain), and uses IS resources efficiently. If the system was based on an Integrated Data Management System data base using Canadian Independent Computing Services Association for communications that corresponded to the strategic systems plan, it would be positioned outside of the Quadrant III bull's-eye and therefore would be maintained as is. If, however, the strategic systems plan called for a broad sweeping conversion to data base 2 using CICS, the system would be placed within the bull's-eye and the system would be reverse engineered to convert the system to the new data base architecture.

Quadrant IV.

Quadrant IV systems satisfy the business functional requirements but do not score well on their design, code construction, or use of technology. The prognosis for systems in this sector is still positive, however. If a Quadrant IV system fits the strategic systems plan (i.e., its position is outside the bull's-eye), it would indicate that the technological components of the system support the strategy but that the system design or construction are difficult to maintain.

Systems that fall into this category are usually reengineered. In the context of the Target Chart only, computer systems reengineering differs from reverse engineering in that reengineering operates on a system at the source-code level whereas reverse engineering operates on a system at a higher technical level (e.g., hardware platform, data base architecture, or communication monitors). Reengineering implies improving the maintainability of a system by eliminating dead code, incorporating structured programming techniques, and adhering to IS standards for program development.

These program-level changes may be made using automated or manual methods. The use of automated tools speeds the modification process significantly. When using these tools, however, IS must review and test the regenerated source code to verify that system functionality is not altered. This method is the preferred approach for large source program because manually converting large programs is extremely time consuming and requires the same level of verification as the automated tools.

Quadrant IV systems that do not fit the strategic systems plan require more than just code-level modifications. These systems usually require a conversion to a new data base or other technical architecture. In this case, systems are usually reverse engineered to reposition the system for the new architecture.

Production Standardization

The goal of production standardization is to revitalize the existing system after years of touch-up work performed by different IS personnel using different programming
techniques and styles. The revitalization transforms the legacy system into one that performs better, is easier (i.e., more cost-effective) to maintain and operate, and uses resources more efficiently. Nearly every IS organization has at least one system that can benefit from this revitalization process.

The refurbishment process allows IS to approach the system from two directions simultaneously—from both a functional and technical perspective. A functional knowledge of the system is required to identify and document the business attributes that are supported by the system. The technical aspect of this two-pronged approach provides IS with detailed knowledge of the processing within the system and the information necessary to improve the maintainability and performance of the system. Obviously, the refurbishment of legacy systems and, in particular, production standardization cannot be performed in a vacuum.

To obtain an accurate functional understanding of the system, IS should meet with the system users to determine the functions performed by the organization, the information that is required by the organization, and the information that the organization currently receives from IS. The objectives are to determine the value that the users place on the system and the positive and negative functional and technical attributes of the system.

Simultaneously, IS can initiate the activities that focus on analyzing and improving the technical aspects of the system. A preliminary analysis of the several systems in the inventory can be performed to determine the areas that may benefit most from the use of automated tools. A word of caution before purchasing the newest tools: it is essential to ensure that the tools perform the functions needed to perform the analysis. For example, some tools can restructure program source code to improve system maintainability, document the structure of the system, and document the system’s input and output.

For one insurance company, this revitalization approach, and the use of a purchased software package, improved the monthly financial closing time from 25 days to 5 days. For another company, converting tape data sets to disk and improvements in the disaster recovery process (for one batch job) improved the processing time by nearly 3 hours.

When the revitalization effort is completed, IS has a system that is more cost-effective to maintain and operate and is better positioned to react to internal and external business forces, such as outsourcing, downsizing, business process reengineering, or changes in government regulations. Production standardization provides significant benefits to IS and the business regardless of the priorities imposed by internal or external forces.

**Design Recovery**

The final phase of a refurbishment effort is the design recovery phase. Design recovery captures certain elements of the current system design, incorporates these elements into a Computer-Aided Software Engineering tool, and provides IS with the ability to accurately document the functional and technical aspects of the system. This repository of up-to-date system documentation improves systems analysis and maintenance time and cost, improves the learning curve for new IS personnel, and provides a basis for engineering these systems more competitively in a CASE environment.

The process of extracting and loading mainframe system design elements (i.e., program names and their relationships to other programs, data files and records, and data record attributes) to a microcomputer or mainframe-based CASE tool is not always straightforward. Automated analysis and documentation tools designed to pass this information to other products (e.g., CASE or data dictionary) are available; however, this is not necessarily a standard feature of these products. The element extraction process is simplified if this feature is available. Otherwise, an extraction program may be written to
perform the extraction function. Currently, the extraction process applies to data elements only. The processing or logic aspect of a system cannot yet be extracted and passed to the CASE tool. Some vendors are trying to develop this logic link between the system and the CASE tool, but such products are not expected to be commercially available in the near future.

The documentation produced from this phase provides insight into the functional purpose of the system, the major components of the system, the technology used to provide system functionality, the organizations that use the system, and the interfaces to other systems. This documentation can be used to support forward engineering, downsizing, and other IS refurbishment strategies.

**Conclusion**

If IS is to play an active role in the effort to better control cost, then system refurbishment should be included as part of an overall cost-reduction plan. Although this article has specifically targeted legacy systems, IS can achieve substantial benefits from refurbishing newer systems that have been developed using loose standards or that have experienced several modifications.

Systems refurbishment presents IS management with an effective approach to maintenance because it reduces the operating cost of systems, improves system maintainability, and positions systems to support the IS strategy as well as activities associated with business process reengineering, outsourcing, and downsizing. Refurbishment may also avoid the cost of replacing a system with a purchased package that is in a similar state of disrepair. The approach and tools provided in this article were designed through experience, practice, and a few painful lessons, and they provide IS with the road map for accomplishing its performance objectives.

**Author Biographies**

William F. Lenihan

William F. Lenihan is a senior manager with the Stamford CT practice of Deloitte & Touche Management Consulting, where he is a leading practitioner in the area of information technology reengineering and has responsibility for developing Deloitte & Touche's reengineering approach. He has consulted to corporations in the manufacturing, health care, and insurance industries as well as in the public sector on a wide range of operations and technology issues. Previously, he worked at McDonnell Douglas Automation (McAuto) Systems and at IBM. He is a graduate of the State University of New York at Oswego and has a degree in computer science.
The Chart shows four quadrants (I-IV) and eight sectors (two sectors per quadrant). Sector positioning is based on the system's strategic fit. Systems whose core competencies support the IS strategy are placed outside the circle (the bull's-eye); those that do not support the IS strategy are placed inside the circle, regardless of the system's functional or technical rating.

Notes:
*Functional obsolescence
**Technical obsolescence