Proceedings of Student-Faculty Research Day, CSIS, Pace University, May 2nd, 2008

Extending the Life of Legacy Software Systems: A Decision Process Model

David Fronckowiak

Computer Science Department, Pace University
1 Martine Avenue White Plains, New York 10606 (914) 422-4000

Abstract

This paper contains the research plan that will analyze the decision process that is used in replacing or extending the lifetime of a legacy system. A framework will be developed based upon the feedback from a Delphi study on a company in the microelectronics fabrication industry. The framework will be developed based upon known, well-established models that exist in the area of decision making and technology acceptance.

Introduction

In 2005, Zach Nelson and Joe Gentry [1] presented their views on whether a company should replace their legacy system or extend the lifetime of their legacy system. This is a perennial question which IT managers are confronted with daily. Nelson makes the case that companies should replace their legacy systems to remain competitive, comply with regulatory issues, and improve usability. Gentry, however, is an advocate for the "preserve and extend" philosophy whereby the system should continue to function as designed and implemented or be extended through the addition of "minor enhancements" such as web services with SOA in mind. Seacord et al. [2] dedicate the first chapter in their book to "The Legacy Crisis." The crisis relates to the concern that "the development of new software is outpacing our ability to maintain it." With the continued maintenance of these legacy systems, no resources will be left to develop new software systems. So in terms of the decision process, does the fact that a system is considered legacy change the process which is used to manage the system or is the same decision process used for all IT investment?

Both the "rip and replace" or "preserve and extend" strategies [1] may be appropriate for a given situation and possibly a given industry. Numerous articles have been written on legacy systems, but few have focused on the decision process and the key drivers that are critical to the lifetime extension or replacement decision. The literature can be divided into those works which present strategies that can be used in replacing or extending the lifetime of a legacy system, technologies that are applicable, legacy system assessment frameworks, and legacy system management tools which also include the legacy system extension or replacement decision process.

In terms of strategies which can be used to replace or extend the lifetime of a legacy system, Lavelle [3] mentions wait, wrap, renovate, replace, and outsource. If a product or business line is nearing end-of-life, then the wait strategy is ideal for no investment or decision is required. Wrapping a legacy system in middleware may enhance functionality, but increase complexity and maintenance costs. Renovate refers to the conversion of a legacy system into a more modern form. Lavelle uses the example of renovating a COBOL system into J2EE technology. Replacement is another strategy in which the current legacy system is replaced with newer technology, but this effort may be expensive and require several years given the complexity of the system. Finally, an outsource strategy may be incorporated where the business focuses on its core competence and outsources its IT needs to a company which specializes in that service.

As previously mentioned, the literature also contains numerous examples of the technologies which can be used in extending the lifetime of a legacy system or replacing the legacy system. XML, web services, and component-oriented programming with SOA in mind are the most common technologies presented for legacy system lifetime extension. The use of these technologies and their applicability to the legacy system extension model will not be explained here.

The technologies exist which can be used in replacing or extending the lifetime of a legacy system, but the real problem is in the decision process that is necessary in determining whether the owner of the system should wait and do nothing, extend the life of the system with an internally developed system, a COTS solution, or an outsourced agreement. As in most business environments, the decision is often based on cost and schedule with respect to the competitive market.

Definition of Legacy Systems

Numerous definitions of legacy systems can be found in the literature. Ransom, Sommerville, and Warren [4] define a legacy system as a system "developed sometime in the past which is critical to the business in which the system operates." Bennett [5], however, defines legacy systems as "large software systems that we don't know how to cope with, but that are vital to our organization." Brodie and Stonebraker [6] define a legacy system as "any information system that significantly resists modification and evolution to meet new and constantly changing business requirements." Gold similarly defines legacy software as "critical software that cannot be modified efficiently."

Most recently, Ullrich points out that "any production-enabled system regardless of language or platform is a legacy system." Based upon the continued improvements of technology, it is true that any system that is in production is considered legacy, but systems are often not considered for replacement or extension the moment that they are production enabled given that some of the technology that could extend the lifetimes of these systems has not been developed. DeLucia et al. [7] define legacy systems as "a mission critical software system developed sometime in the past that has been around and has changed for a long time without undergoing systematic remedial actions."

With the aforementioned definitions in mind, we propose the following modified definition of a legacy system:

Systems are considered legacy if any of their components, such as the hardware, operating system, utilities/libraries, and applications, are out of technical support or the system cannot be modified to meet all changing business requirements, but the system continues to be useful in terms of its functionality.

This definition begins to consider both functional and technical obsolescence which Sellers [8] details in his thesis. The hardware, operating system, and application libraries/utilities address the technical obsolescence, while the usefulness addresses the functional obsolescence.

Legacy System Decision Process Problem

Several works present a framework or method that can be used in assessing legacy systems for replacement or extension activities. Seacord et al. [2] present an overview of the legacy crisis along with methods to develop an integrated plan for legacy crisis mitigation. DeLucia et al. [7] present a decisional framework with a legacy system assessment model to model the maintenance interventions that a legacy system must undergo during its lifetime. This work also lists potential solutions to manage a legacy system, namely, ordinary maintenance, reverse engineering, restructuring or reengineering, migration, wrapping, replacement with COTS, and discarding. Bennett, Ramage, and Munro [5] present a two phased legacy system decision model where the first phase is a business process model with stakeholders while the second phase is the technical portion of the model where the extension or replacement decision is analyzed in terms of the technical and skill aspects of the decision which must be considered. Tahvildare [9] presents a multi-objective decision process that analyzes source-code transformation improvements.

Different frameworks and models have been presented on the decision process required to analyze a legacy system and determine how to manage its technology life cycle, be it through the use of lifetime extension techniques or replacement. Examples and case studies in insurance, banking, telecommunications, and healthcare have been published, but little has been published on the decision process itself.

This proposed study will focus on the legacy system decision process used in the extension or replacement of legacy systems in a microelectronics manufacturer. Known, well-accepted models will be modified to develop a framework for the legacy system management decision process.

Models that are appropriate for Legacy System Management

Two well accepted models are applicable in the legacy system extension or replacement decision process. Simon’s decision model [10], as shown in figure 1, can be used to model the overall decision process. Davis’ technology acceptance model [11], as shown in figure 2, can be used to model how technology is introduced and accepted in the legacy management decision process.

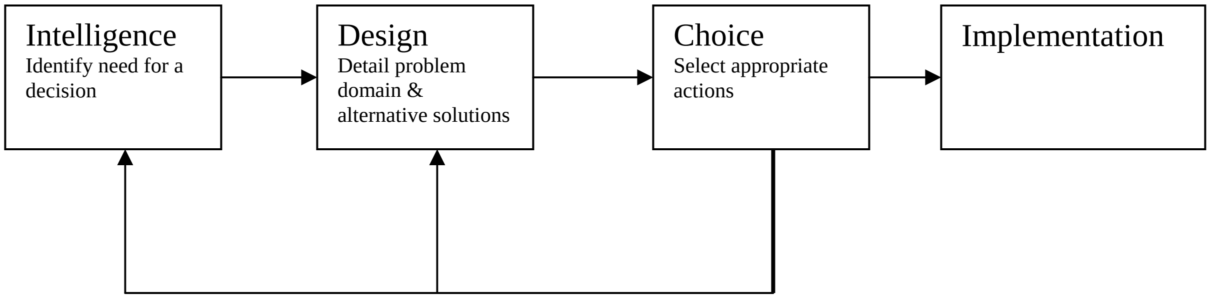

Figure 1. Simon's Descriptive Decision Model

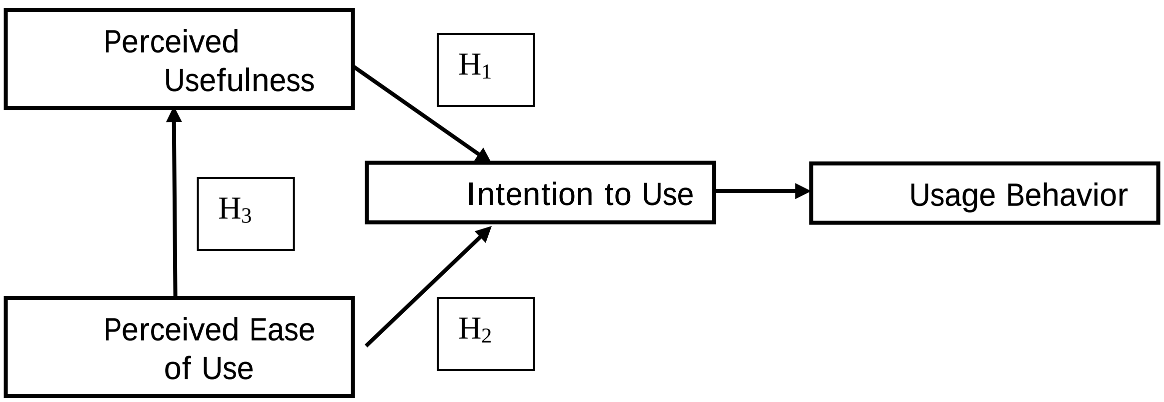

Figure 2. Davis' Technology Acceptance Model

Both models require further analysis as to how they can be used in the legacy system management decision process. The key, however, is the drivers or factors which influence the blocks in the model and the links between each of the blocks in the model. In Simon's decision model, the links indicate an order in which the decision process occurs along with an iterative factor that exists when new information becomes relevant. With this model, the drivers, as shown in figure 3, are critical in the development of the model. In Davis' model, however, the links between the concept blocks can be quantified through the testing of hypotheses which test the relationship of the concepts in each of the blocks. In general, these hypotheses are shown as H_1, H_2, and H_3 in figure 2.

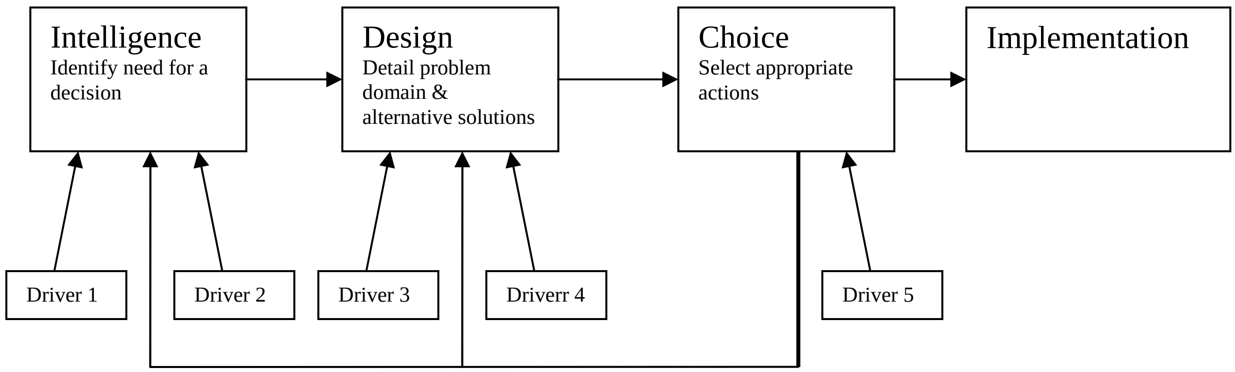

Figure 3. Simon's Descriptive Decision Model with Relationship Factors

Legacy System Research as Applied to Microelectronics Fabrication

Every industry experiences the need for legacy system management and the decision process that is required in the legacy system extension or replacement dilemma. The semiconductor industry which is comprised of fabricating semiconductor wafers and the packaging of the resulting chips into modules is no different. This industry is unique in that it covers numerous areas for IT system usage which include manufacturing execution systems (MES), automated material handling systems (AMHS), data analysis and diagnostic systems, accounting systems, human resource systems, ERP systems including order entry, customer information sites, and B2B systems which connect to customers and suppliers. Given this vast array of systems, legacy management becomes critical. The key in the decision process is to understand the key drivers in the legacy management decision process so as to be able to model the process for more efficient decision making. From the literature, table 1 summarizes the factors which must be considered in the decision making process. The proposed research section of the paper will discuss how legacy system management will be studied in relation to microelectronics fabrication.

Business or technical driver	Decision Driver
Business	Operations costs
Business	Maintenance/Support costs
Business	Customer access /ability
Business	Legal requirements
Technical	Web Services
Business	Scheduled need date
Technical	Virtualization
Business / Technical	User Mobility
Business / Technical	COTS
Business	Risk
Technical	Internal development skills
Business	Benchmarking
Technical	Component redesign
Technical	SOA
Decision process	Technical champion
Decision process	Business champion
Decision process	Budget
Decision process	Review board

Table 1. Legacy System Management Decision Drivers

These drivers are then applied to each of the models. Figure 4 shows the drivers as applied to Simon's decision making model, while figure 5 shows the drivers in relation to the technology acceptance model (TAM).

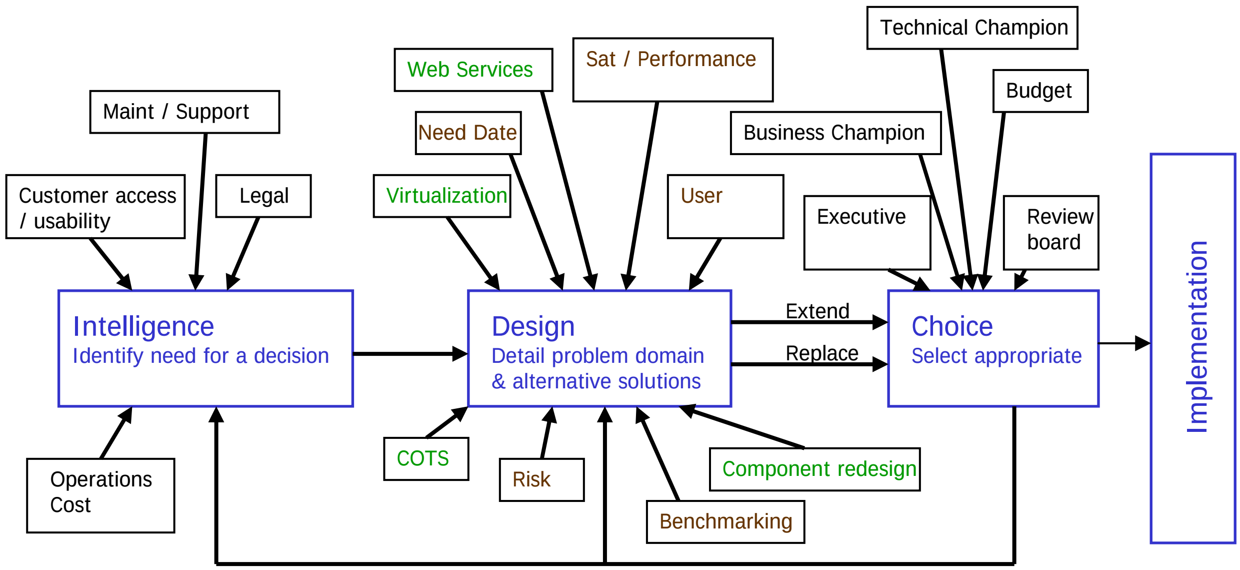

Figure 4. Legacy management decision drivers and Simon's decision making model

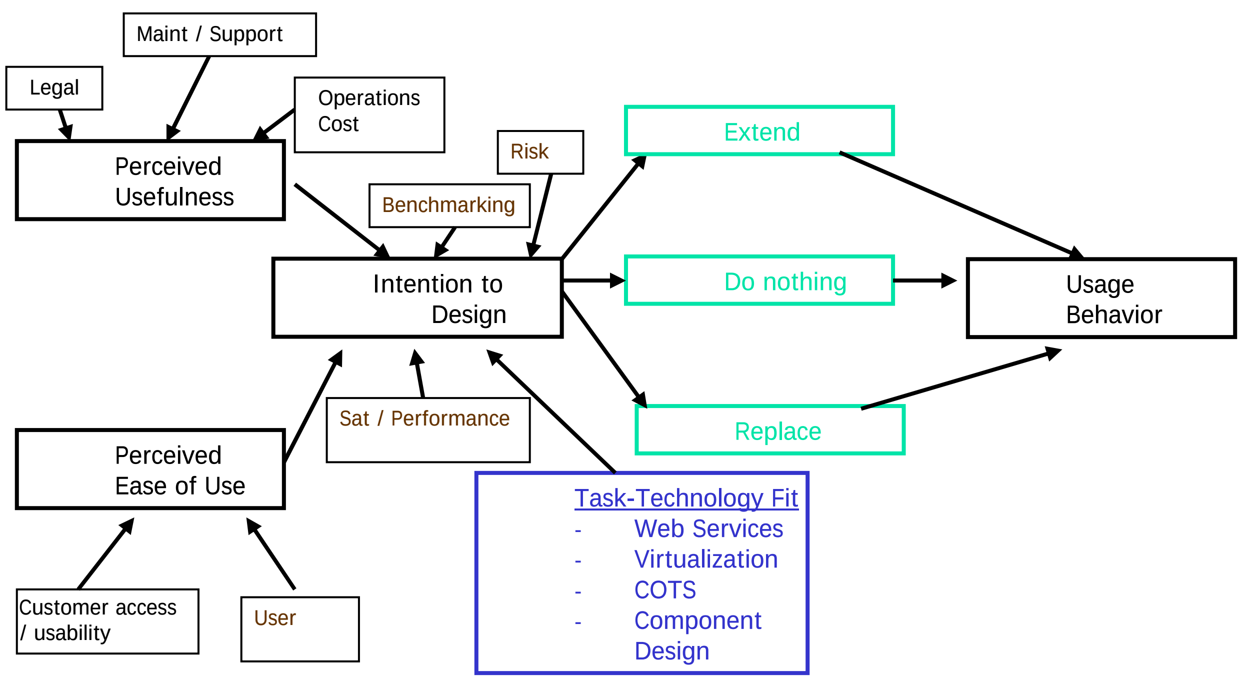

Figure 5. Legacy management drivers and Davis' technology acceptance model

Research Methodology

To analyze the legacy system management decision process, a qualitative study using the Delphi method [12] will be used between twenty-five IT professionals in the microelectronics fabrication industry at IBM. Linstone and Turoff [12] define a Delphi study as "a method for structuring a group communication process so that the process is effective in allowing a group of individuals, as a whole, to deal with a complex problem." The study will be used to determine the key drivers that must be considered in this decision process along with the decision process itself. This study will then be used to understand the applicability of Simon's decision making model and the technology acceptance model to the legacy system management problem.

This study will be open-ended in that a more rigorous definition of legacy systems will be solicited along with an analysis of the participating professional's level within the organization and their response.

The study will focus on the following questions which are more appropriate to Simon's decision model:

(1) What is a legacy system?
(2) Is their a formal decision process that is used to determine if a legacy system will be replaced or if its lifetime will be extended?
(3) Is the decision process based upon a business champion, technical champion, or both?
(4) Does the legacy system management decision process differ for non-legacy systems?
(5) What are the key drivers in a legacy system management decision process?
(6) What techniques are used to manage legacy system transformation?
(7) Site examples of a legacy system extension project.
(8) Site examples of a legacy system replacement project.

In considering the technology acceptance model, the hypotheses which are associated with the links between the blocks must be considered. An example of such a simplified hypothesis for H_1 in figure 2 may be as follows:

H_1: If legacy system extension is useful, then it will be designed into a plan with deliverables.

The key in using the technology acceptance model is that a decision has been assumed and then this model can be used to take the next step which is to determine the technology that should be implemented. A critical part of this study and key research area will be in developing and grounding the appropriate questions to address the drivers of the blocks in the model and the relationship links between the blocks in the technology acceptance model. In many studies, questions based upon previous research are modified so as to adapt the model to the current research environment. An example can be found in the work of Lapczynski [13], where questions used by Vankatesh and Davis [14] for perceived usefulness in the technology acceptance model were modified to study the application of this model to mobile computing allowing for the development of the Integrated Technology Acceptance Model for Mobil Computing (ITAMM). Table 2 shows the questions for the two published works as they may relate to the study of perceived usefulness in legacy system management. The literature needs to be reviewed for more relevant studies that have applied the technology acceptance model. A pilot study may be more appropriate to ensure that the questions for the Delphi method in relation to the technology acceptance model are relevant and unambiguous.

TAM2 Questions [11]	**ITAMM Questions [13]**	**Legacy System Management Questions**
Perceived Usefulness • "Using the system improved my performance in my job"	Perceived Usefulness • Using the PDA improves my job performance [rate level of agreement]	Perceived Usefulness • Using web services will improve legacy system performance

Table 2. Perceived Usefulness Questions

The strength between the blocks of the models can then be determined by evaluated the strength and relative direction of the response to the questions associated with the block factors in the model.

In reviewing these two models, it becomes apparent that the first step is to decide if the legacy system should be left "as-is", its lifetime extended, or be replaced. During the design stage of the decision model, the technology acceptance model would be useful in determining the technical alternatives and the implementation strategy, be it XML with web services, component oriented programming, SaaS, and/or SOA.

Future Work

Since this study focuses on legacy system management in semiconductor fabrication, it would be of interest to determine if the same legacy system decision drivers exist in other industries such as insurance and banking and if the same decision process is employed given a totally different industry.

References

1. **Nelson, Z. and J. Gentry, *Face-off: Should your company replace its legacy systems?*, in *Network World*. April 25, 2005. p. 49.**
2. **Seacord, R.C., D. Plakosh, and G.A. Lewis, *Modernizing Legacy Systems*. 2003, Boston, Ma.: Addison-Wesley.**
3. **Lavelle, G., *Practical Strategies for Dealing with Legacy Systems.* Risk Management, Feb 2005. 52(2): p. 45.**
4. **Ransom, J., I. Sommerville, and I. Warren. *A Method for Assessing Legacy Systems for Evolution*. in *IEEE Proceedings of the Second Euromicro Conference on Software Maintenance and Reengineering*. March 8-11, 1998.**
5. **Bennett, K.H., M. Ramage, and M. Munro. *Decision Model for Legacy Systems*. in *IEEE Proceedings Software*.**
6. **Brodie, M.L. and M. Stomebraker, *Migrating Legacy Systems*. 1995, Burlington, MA.: Morgan Kaufmann Publishers.**
7. **DeLucia, A., A.R. Fasolino, and E. Pompella. *A Decisional Framework for Legacy System Management*. in *17th IEEE International Conference on Software Maintenance (ICSM '01)*. 2001.**
8. **Sellers, A., *Life Cycle Extension Strategies for Legacy Systems*, in *Vanderbilt University Master's Thesis*. 2004.**
9. **Tahvildari, L. *Evolving Legacy Systems through a Multi-Objective Decision Process*. in *Proceedings of the 2004 International Workshop on Software Technology and Engineering Practice (STEP '04)*. 2004.**
10. **Simon, H.A., *The New Science of Management Decision*. 1960, New York, New York: Harper and Row. 50.**
11. **Davis, F.D., *Perceived Usefulness, Perceived Ease of Use, and User Acceptance of Information Technology.* MIS Quarterly, 1989: p. 319-339.**
12. **Linstone, H.A. and M. Turoff, *The Delphi Method: Techniques and Applications*. 1975, Reading, Massachusetts: Addison-Wesley Publishing Company, Inc.**
13. **Lapczynski, P.A., *An Integrated Model of Technology Acceptance for Mobile Computing*, in *School of Computer Science and Information Systems*. 2004, Pace University: New York. p. 183.**
14. **Venkatesh, V. and F.D. Davis, *A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies.* Management Science, 2000. 42: p. 186-204.**