The Dynamics of ERP Success

Meg Fryling, Ph.D. Student
Information Science and Policy
University at Albany
State University of New York
1400 Washington Ave
Albany, NY 12222 USA
Phone: +1 (518) 437-4528
E-mail: mfryling@uamail.albany.edu

Research Problem

Enterprise resource planning (ERP) commercial software packages exploded into the market during the 1990s as a popular way by which companies attempted to integrate their financial, human resource, operation, and customer information. The "seamless" integration of computer systems was appealing to organizations because it would allow real-time access to data, reduce redundant data elements and lower the costs associated with maintenance of multiple systems. ERP systems were intended to help organizations increase efficiency and provide a higher level of customer service.

Although ERP systems are capable of providing significant returns on investment, they can also cause havoc in an organization if not managed correctly. Unfortunately, the success rate of ERP implementations is only around 33% and approximately 90% of ERP implementations are late or over budget (Martin, 1998). ERP implementation articles consistently report that implementation failure or success is people-related (Peterson, 2003; Tapp, et al. 2003). It is often easier to blame the technology than to explore these deeper issues but in the end they are the controlling factors. It is important for managers to understand the non-technical complexities before embarking on a new ERP project.

Literature Review

Tapp, et al (2003) discusses four primary reasons that ERP implementations fail, they are: inadequate education/training, poor leadership from top management, resistance to change, and unrealistic expectations. What do all of these factors have in common? People. They have nothing to do with technology or the specific ERP software and everything to do with the people involved. As Peterson states (2003), "…nontechnical issues play a central role in the success of IT initiatives." This article stresses how important people are in an IT project. Both the Tapp and Peterson papers explore the challenges involved with implementing ERP information systems. The only real difference between these

two papers is that one refers to a successful implementation while the later refers to a failed one; the forces were the same. The inexperienced might initially believe the difference is related to the fact that the organizations employed a different ERP software package but when you read the literature it becomes obvious that the success/failure factors are much more complicated.

An ERP project can not be successful if the user community is not heavily involved in the project from its inception. User involvement hopefully leads to user commitment and this must last through the entire project lifecycle, even when times get tough. Associated with that aspect is the natural human reaction of resistance to change. Even open-minded individuals have a certain comfort zone in dealing with what they already know. Shaking things up with new IT initiatives, that will likely completely change the way they do their jobs, can be a very frightening prospect.

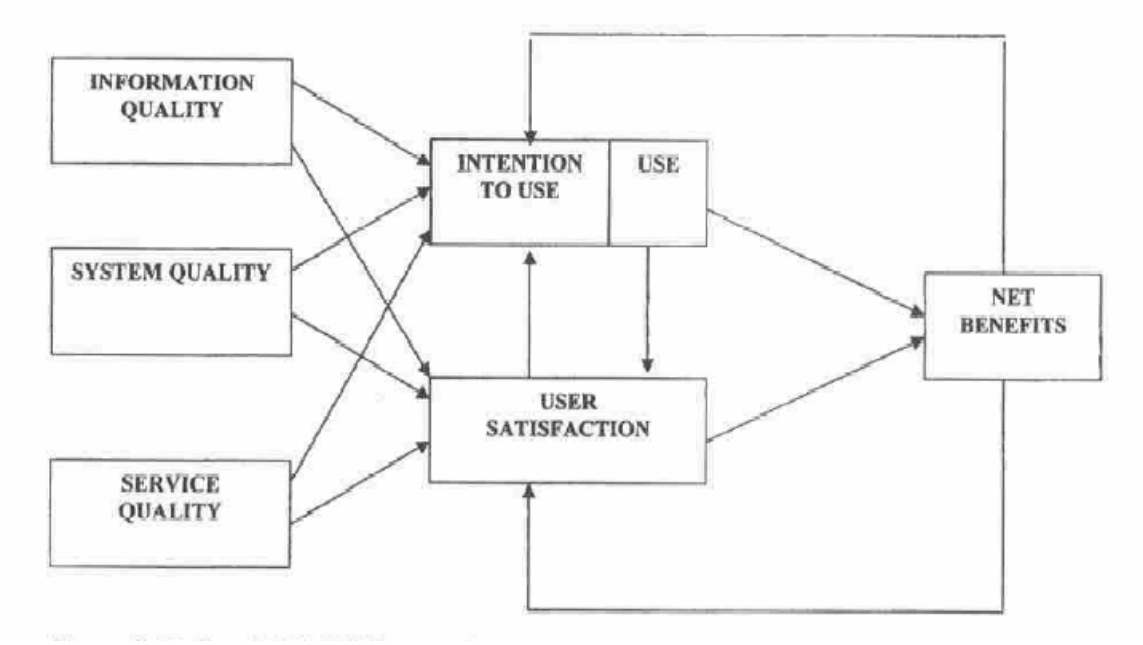
The fourth failure factor that the Tapp (2003) paper mentions is inadequate education and training. This is an often overlooked factor and it can have devastating effects. Education and training are one of those items that are not only time-consuming and costly but they are often delayed until the end of the project lifecycle. By the time individuals get some training they have a short window to learn and are even more resistant to the change because they have been left “out of the loop” for so long.

New IT projects, and particularly ERP implementations, will inevitably cause big changes in the way people do their jobs. In October 1999, Arkansas began planning the Arkansas Administrative Statewide Information System (AASIS). Unfortunately, the planning and implementation of AASIS was plagued with troubles from the beginning. The Peterson (2003) paper discusses the various stumbling blocks AASIS encountered during and after its implementation and how these challenges were primarily non-technical in nature. More specifically the paper addresses the fact that poor communication was the primary failure factor for the AASIS project. The managers of the project acknowledge that

"employee confidence was a critical factor in the system's success because AASIS was forcing the state to change how it did business." (Peterson 2003). This is typical of large-scale ERP projects; all business processes must be reevaluated and often modified dramatically. If education/training is provided early then not only does the likelihood of user-acceptance increase but it can actually uncover unnecessary processes, overlooked items and potential pitfalls.

Although emerging technologies have complicated the implementation of information systems, some basic principles have continued to haunt IT managers. The real fact behind the success or failure of information systems is not, interesting enough, the technology. The individuals involved with IT projects, both technical and functional, determine its success. Furthermore, it is the effective on-going communication of those individuals that keeps an IT initiative "on track". The user community must not be left out of any stage of the implementation.

Figure 1.1: The reformulated model of IS Success (DeLone and McLean, 2003)



A variety of literature exists on what constitutes Information System (IS) success. DeLone and McLean first introduced their IS Success Model (see Figure 1.1) in 1992 in an effort to provide a framework for measuring IS success/effectiveness.

There have been hundreds of articles referencing both the original and updated model. IS success is certainly a hot topic considering the large IT budgets and almost endless system implementation failure stories. Organizations make enormous investments in information systems and expect a return on their investment.

The D&M IS Success Model is a causal model that attempts to show the interrelationships between six dimensions of IS success. Many researchers have referenced, validated and expanded this model. The inclusion of “service quality”, which did not exist in the original model, includes not only the system itself but services offered by the IS department that supports the system. Unfortunately, a large, and often overlooked, aspect of systems implementation is user training and support. What starts as a highly successful IS implementation can end in IS failure if “service quality” is poor.

When DeLone and McLean developed their original IS Success Model in 1992, they did not validate it with empirical data but left that challenge to future researchers. In 2003, DeLone, et al conducted a literature review of articles referencing their original paper. Some articles reviewed validated and supported the original model, while others offered suggestions for modifications. DeLone and McLean used these findings to support, assess the value of and make improvements to their original model.

Besson, et al. (2001) developed a theoretical framework to describe the dynamics involved in ERP implementations. The model identifies distinct phases of an ERP implementation, all of which involved a variety of stakeholders, different levels of perceived understanding, and varied levels of perceived environmental leeway. They found that stakeholder involvement and perception changes as ERP implementations move through time (Besson, et al, 2001). Hawking, et al. (2004) found that there is an enormous disparity between the expected benefits from an ERP implementation and the actual benefits realized.

People-related issues, particularly change management difficulties, were the primary cause of lowered benefits.

Aladwani (2001) offers what appears to be a reasonable framework to prevent user resistance to change but performs no formal testing of the model. Future research with empirical data should be conducted to validate this model and offer suggestions for improvement. This research is similar to the original DeLone and McLean article where validation was left for other researchers. Although the Delone, et al. model has been validated and extended by the 2003 review, ways to measure soft variables like “quality of IS services” need to be developed in order to further test the extended model. Better operationalization of what constitutes IS success is also essential because variation can lead to divergent results among research studies. Both the DeLone, et al. (2003) and Hawking, et al. (2004) acknowledge that some variables in their research were intentionally discussed in broad terms and that future research might be needed to determine if there are additional factors that have not been identified.

So how does one know if an information system implementation was successful? Organizations, employees and customers will have very different views on whether or not an information system is considered successful. DeMarco (1982) explains that the frequency of project failures has pushed for a redefinition of the term “success”. Project Managers may consider an implementation successful even though cost and length of project are 30% over original targets and 25% of the system is left unused, while end-users/customers may have a very different perspective (DeMarco, 1982).

Research Purpose

Although there are many extensions of the IS Success Model, the basic concepts exclude ongoing user involvement with the implementation of IT initiatives. This exclusion may have been applicable to non-ERP IS but the unique characteristics of ERP systems lead to the notion that early and frequent user involvement is necessary for IS success. There may also be additional variables

related to ERP implementations that should be tested, including project scope, time and resources.

It appears that existing models do not include enough feedback behavior between the constructs. The IS Success Model (DeLone and McLean, 2003) is causal and does contain some feedback such as suggesting that user satisfaction increases net system benefits, which increases user satisfaction. However, there may be additional causal relationships not identified in the existing models that should be explored and validated.

Some of the research questions evolving from a literature review include:

- Are the IS Success frameworks appropriate for ERP implementations?
- How can the existing IS Success and system dynamics methodologies be combined to explain ERP success?
- What constructs missing from existing frameworks are necessary to explain ERP implementation success?

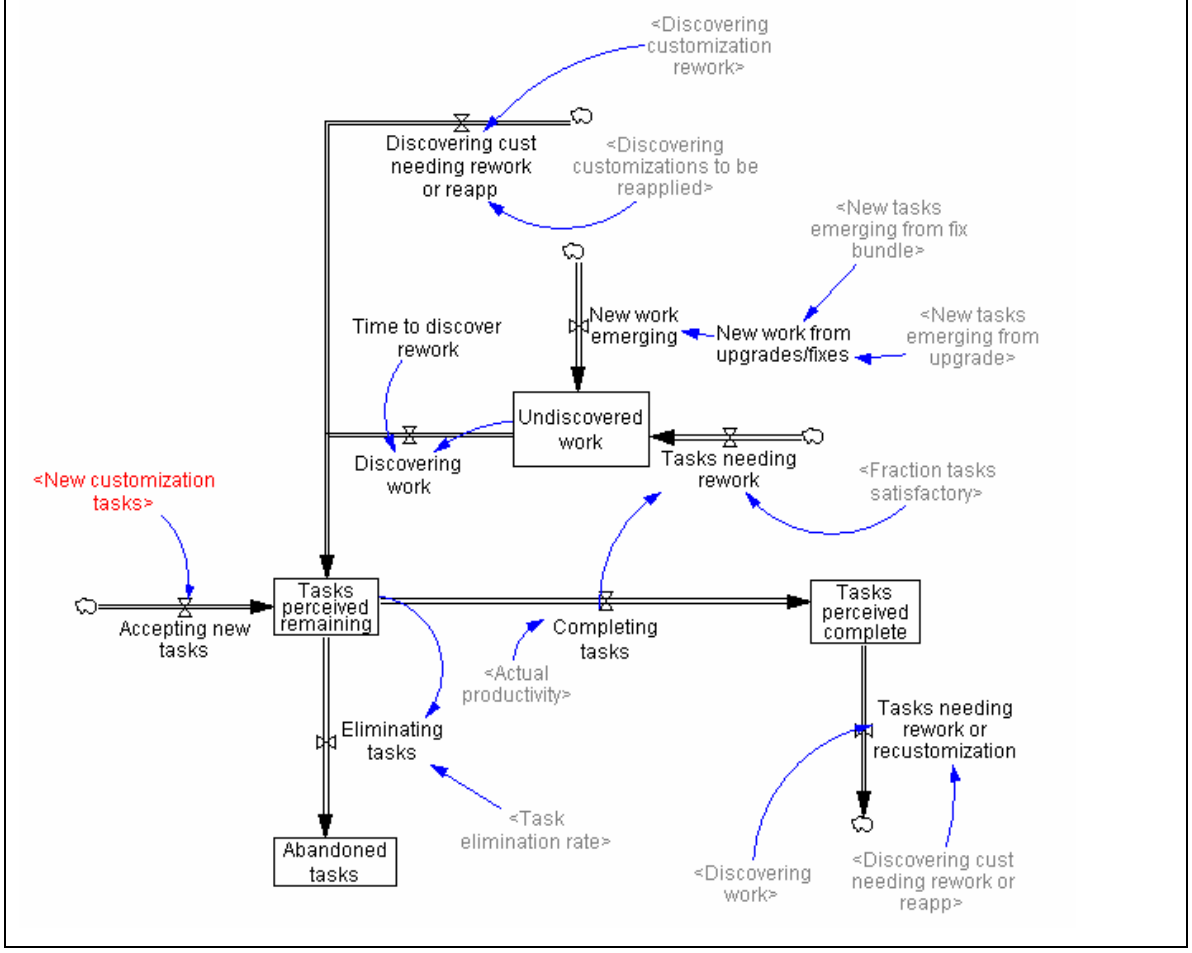
The original IS Success models were developed prior to the enormous growth of ERP implementations among organizations. This research will explore whether or not an extension of these models can be developed using system dynamics tools to explain ERP success.

Problem Dynamics

One can easily find feedback behavior when reading articles regarding the challenges of implementing an ERP system. These feedback loops are essential to the story of what causes ERP failure. ERP implementations, like many IT projects, have some common factors that cause enormous headaches for project managers; these elements all have feedback behavior. Time, resources, and scope are three common elements of ERP implementations, which can conflict with one another. For instance, as the scope of a project increases, the time required and/or resources must be augmented. Further, as time is extended project scope inevitably increases (scope creep). Increasing the scope of a project unavoidably causes time and resources to grow. The causal nature of these influences is clear.

An ERP implementation has an initial scope (number of tasks) based on fit gap analysis between the business processes of the organization and the delivered functionality of the software product. This initial scope can vary depending on how well the ERP software selected fits the organization's needs and the complexity of existing business processes. In addition to the pool of tasks established by the initial scope, new tasks are constantly entering the system do to several factors. First, some of the work completed will be incorrect and will need to be redone. Also, as new gaps are discovered during the project implementation, requests for customizations will be made. Finally, as the ERP vendor delivers fixes/upgrades to the software, customizations will need to be reapplied and new work will emerge (see Figure 2.1).

Figure 2.1 – Tasks (Scope) Sector



The longer it takes to implement a project the more likely tasks will need to be revisited. ERP software fix bundles and upgrades are unavoidable but can be limited by implementing in a timely fashion. Fix bundles are provided by ERP vendors several times a year. While fixes are intended to correct issues with the current product, they often break other pieces. The only way to assess the extent to which fixes negatively impact an implementation is to retest all previously completed work. In addition, if there are many customizations to the delivered product the likelihood of rework increases appreciably. Customizations differ from other tasks in that they must be carefully tracked since the vendor may redeliver a new version; thus, wiping out the customization work. Each time the item is redelivered the customization must be reapplied (see Figure 2.2). Fixes and upgrades not only break tasks but they often introduce new tasks (see Figure 2.3).

Figure 2.2 - Customizations Needing Rework or Reapplication

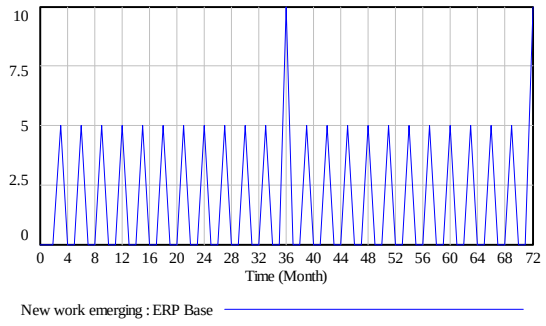


Figure 2.3 - Undiscovered Rework and New Work from Fixes/Upgrades

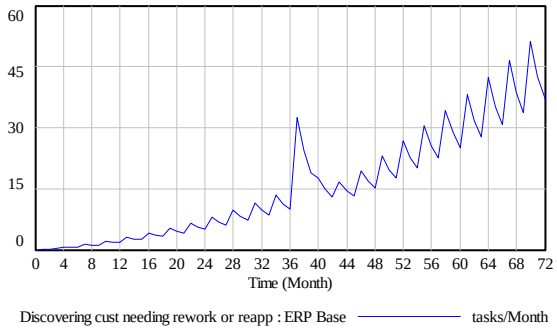


Figure 2.4 – Scope Flows

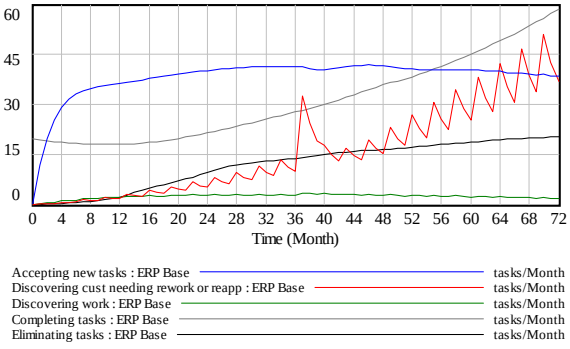
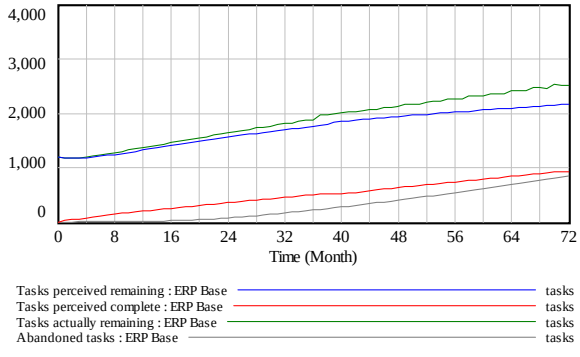
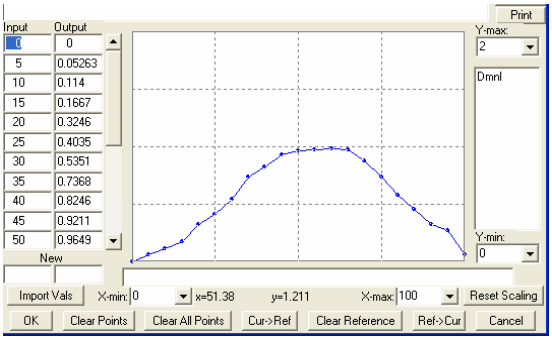


Figure 2.5 – Scope Stocks



If an ERP project is not completed quickly, then it is likely a full upgrade will occur in the middle of the implementation; this will cause a spike in customization rework and tasks actually remaining will increase in divergence from tasks perceived remaining (see Figures 2.4 & 2.5). Scope reduction is one way to counteract time and cost overruns. Pressure to eliminate tasks increases as an implementation passes its scheduled completion date. However, if a project is extraordinarily late then the elimination of tasks becomes difficult to justify (see Figure 2.6).

Figure 2.6 – Effect of Project Lateness on Pressure to Eliminate Tasks

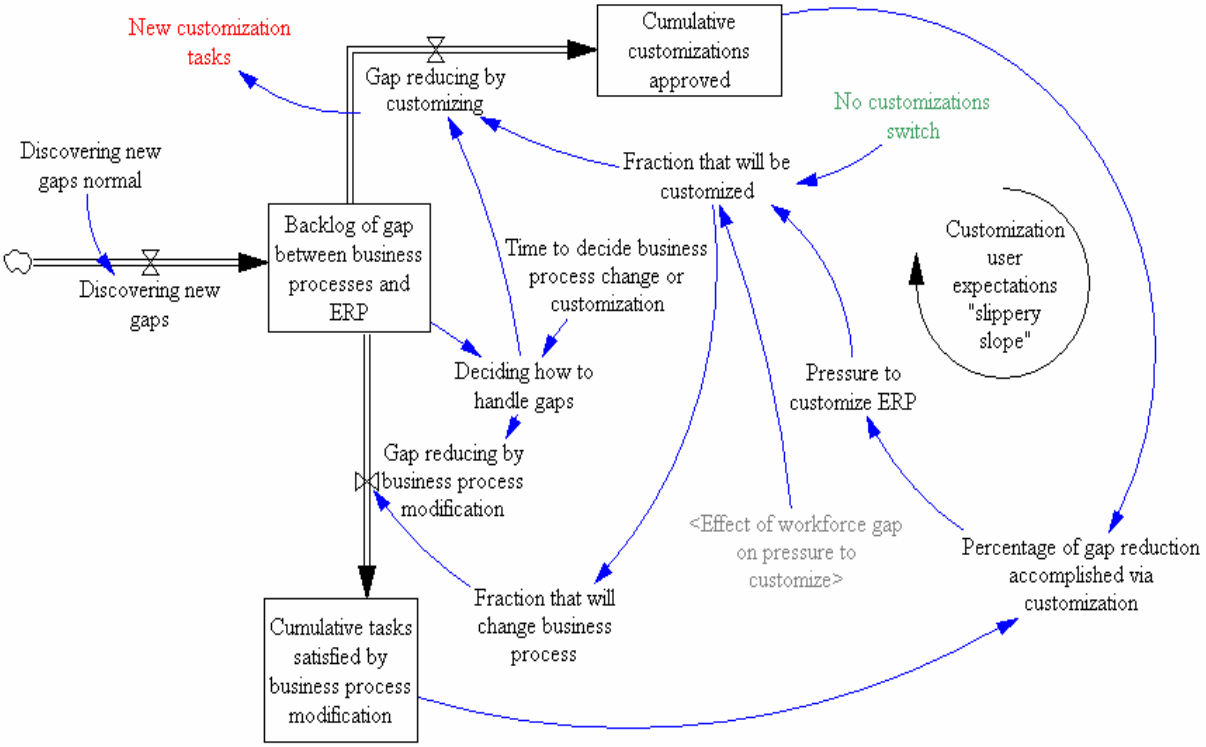


Shifting the Burden

ERP systems never completely match the business processes of any particular organization. Purchasing an ERP system that closely matches the business for which it is intended will help reduce functionality gaps but not eliminate them. The intention of ERP systems is that business processes will be redefined to match the product and not that the product will be customized to meet the existing business processes. Often the user community is resistant to this type of change and the adjustment can be extremely challenging. The gap between the product and business needs will cause pressure to customize the software. By approving customizations user expectations change and they become even

more likely to resist business process change (Shifting the Burden Archetype). Customizations are a “slippery slope” when it comes to ERP systems because although a few are necessary, once some are approved end-user expectations change and the pressure to customize increases (see Figure 2.7). Customizations may not sound like a bad option but there are many negative implications.

Figure 2.7 – Fit Gap Sector



Project lateness can have many affects on an ERP implementation as well (see Figure 2.8). The longer it takes to implement the more likely the software will need to be patched or upgraded. These changes can actually break previously completed work. Sometimes changes are so drastic that the tasks need to be completely redone. In addition, new work emerges each time a fix bundle or upgrade occurs and existing customization work often needs to be reapplied; this is particularly true for full upgrades.

Figure 2.8 – Time Sector

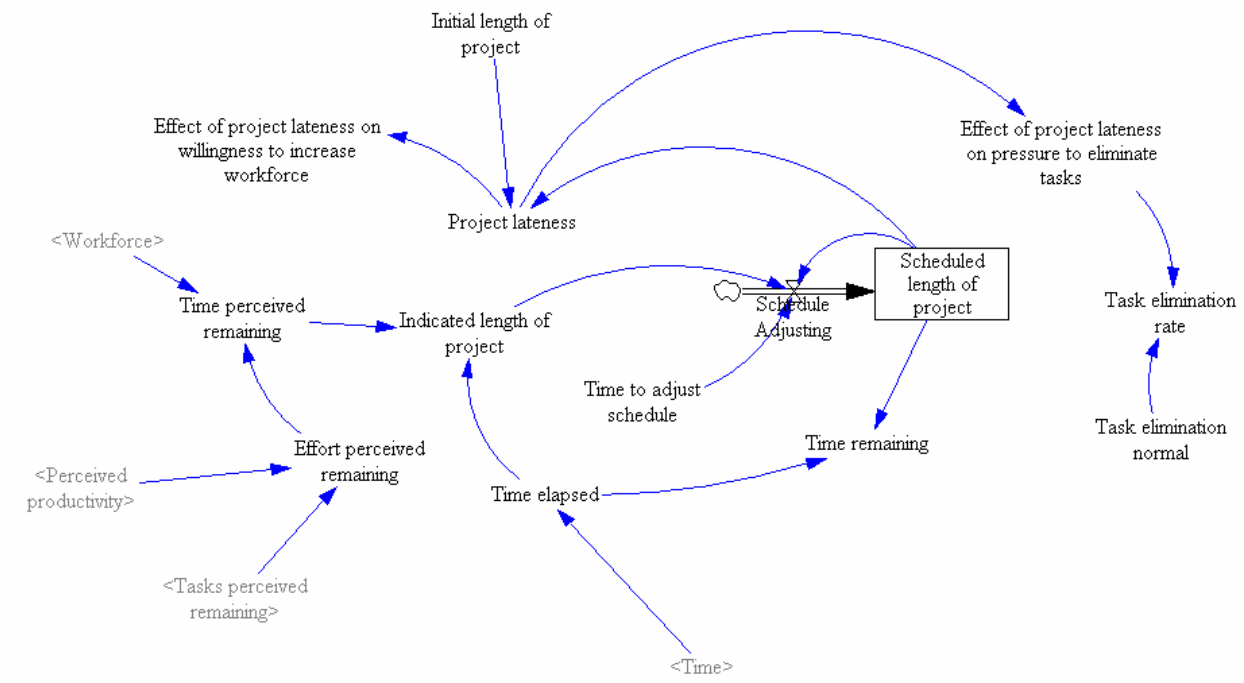
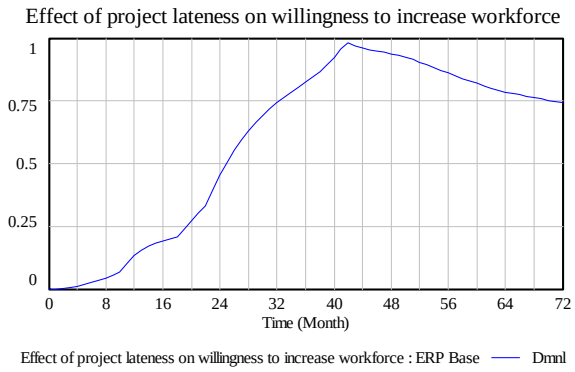


Figure 2.9 - Effect of Project Lateness on Willingness to Increase Workforce



Project lateness also impacts management's willingness to increase the workforce size. Initially lateness may increase willingness but eventually it peaks and declines. Project managers frequently have difficulty obtaining the necessary staff in a timely fashion (see Figure 2.10). At the beginning of the project, workforce is extremely understaffed so the gap between indicated workforce and actual workforce spikes (see Figure 2.11). Unfortunately, it takes

time to hire and even more time for new employees to become experienced; work is backing up during this time. Things seem to get better but then a major upgrade takes place around 36 months and this causes increased tasks. Then because the project is reaching scheduled length, the workforce needed to meet deadline shoots way up.

Figure 2.10 – Workforce (Resource)

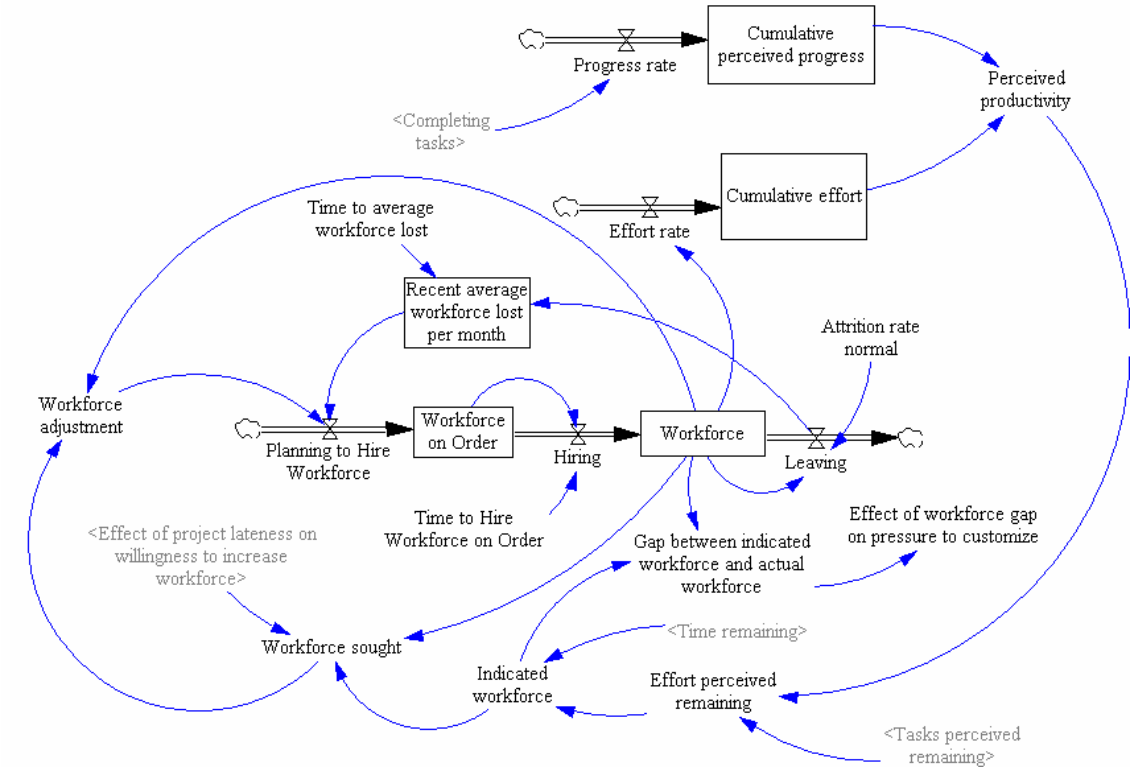
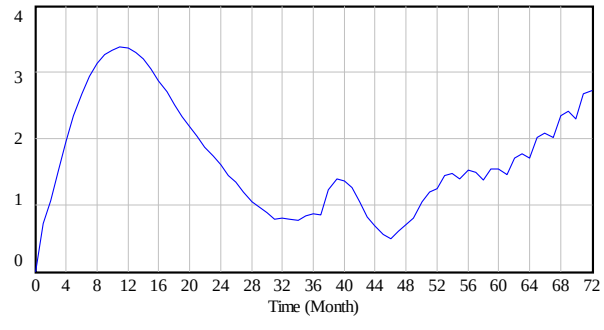


Figure 2.11 – Gap Between Indicated Workforce and Actual Workforce



Workforce size also impacts the pressure to change project scope. While a large gap between indicated workforce and actual workforce decreases the pressure to add new tasks, as this gap reduces belief that new tasks are justified increases. Consequently, both resources and scope increase together so anticipated time reduction benefits are not realized.

Productivity is affected by schedule pressure (see Figure 2.12). Pressure may increase work hours and decrease time spent on tasks in an effort to increase productivity. The negative effect is that workforce burnout can damage productivity (Workforce Burnout Loop) (see Figures 2.13 & 2.14). Furthermore, reducing time spent on tasks increases the likelihood that the tasks will require rework (Reduced Task Quality Loop).

Figure 2.12 – Workforce Capacitated

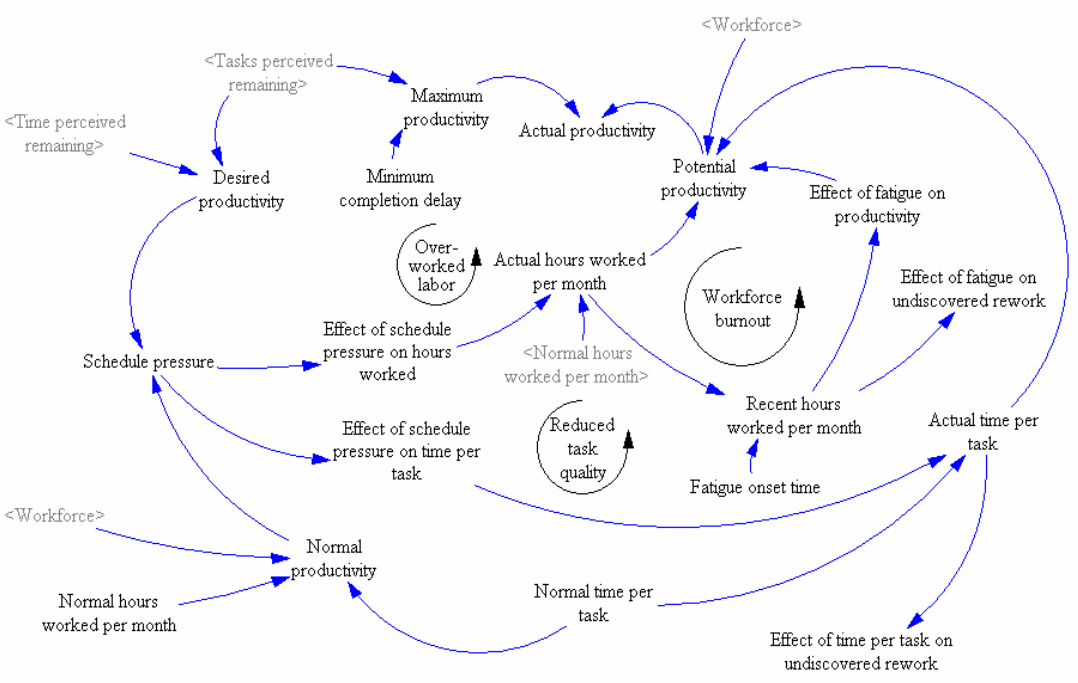


Figure 2.13 - Productivity Base Run

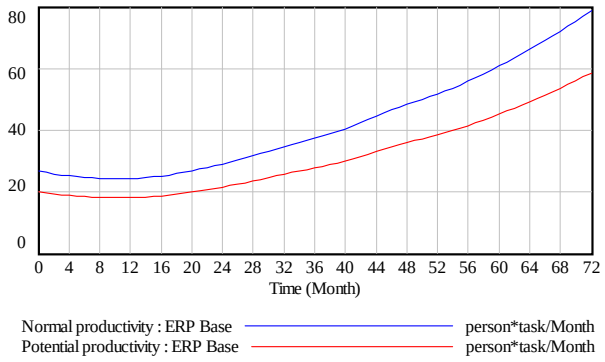
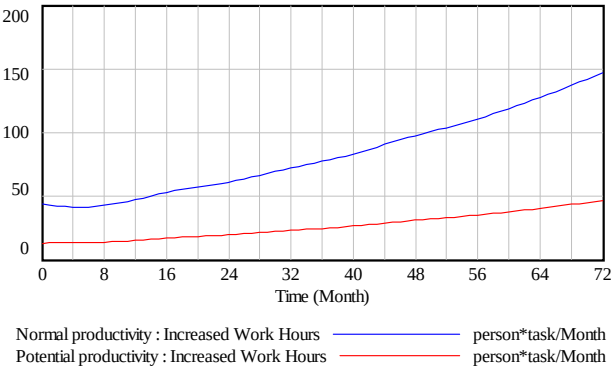


Figure 2.14 – Productivity increasing work hours from 40 per week to 65 per week



The pressure of meeting implementation deadlines can have multiple affects. If schedule pressure is high, then one way to reduce this pressure is to complete tasks more rapidly (Figure 2.15) (Sterman, 2000). Shortening the time to complete tasks is often accomplished by reducing or skipping proper testing. The end result of task time reduction is that task quality decreases and it is likely that tasks will need to be revisited. Schedule pressure also influences the normal hours worked per month (Figure 2.16). Increased work hours causes workforce fatigue, which ultimately affects the quality of work.

Figure 2.15 – Effect of Schedule Pressure on Time per Task

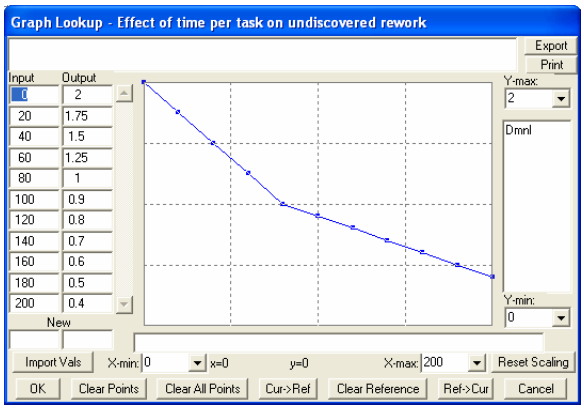
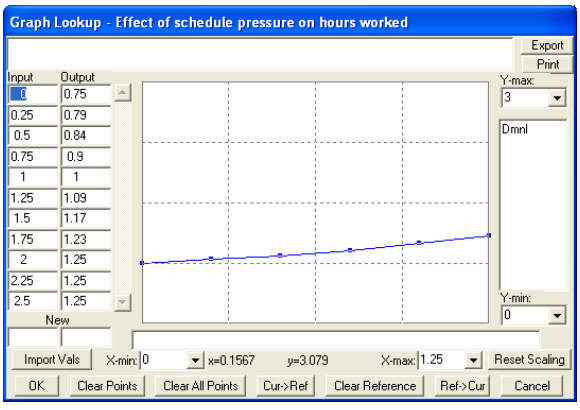


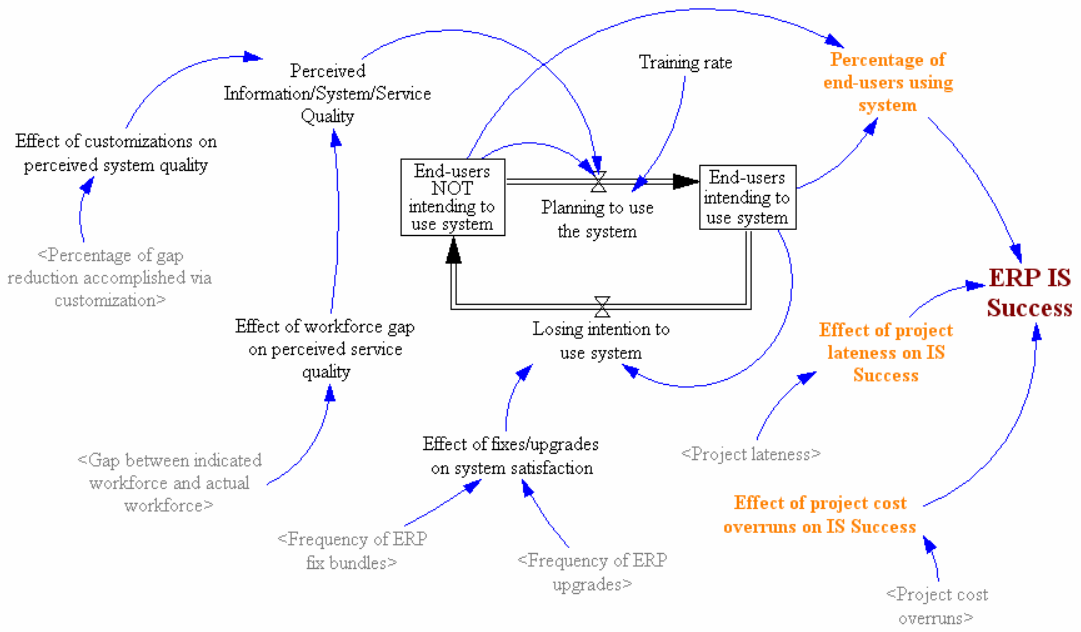
Figure 2.16 – Effect of Schedule Pressure on Hours Worked



Data Source: Sterman 2000

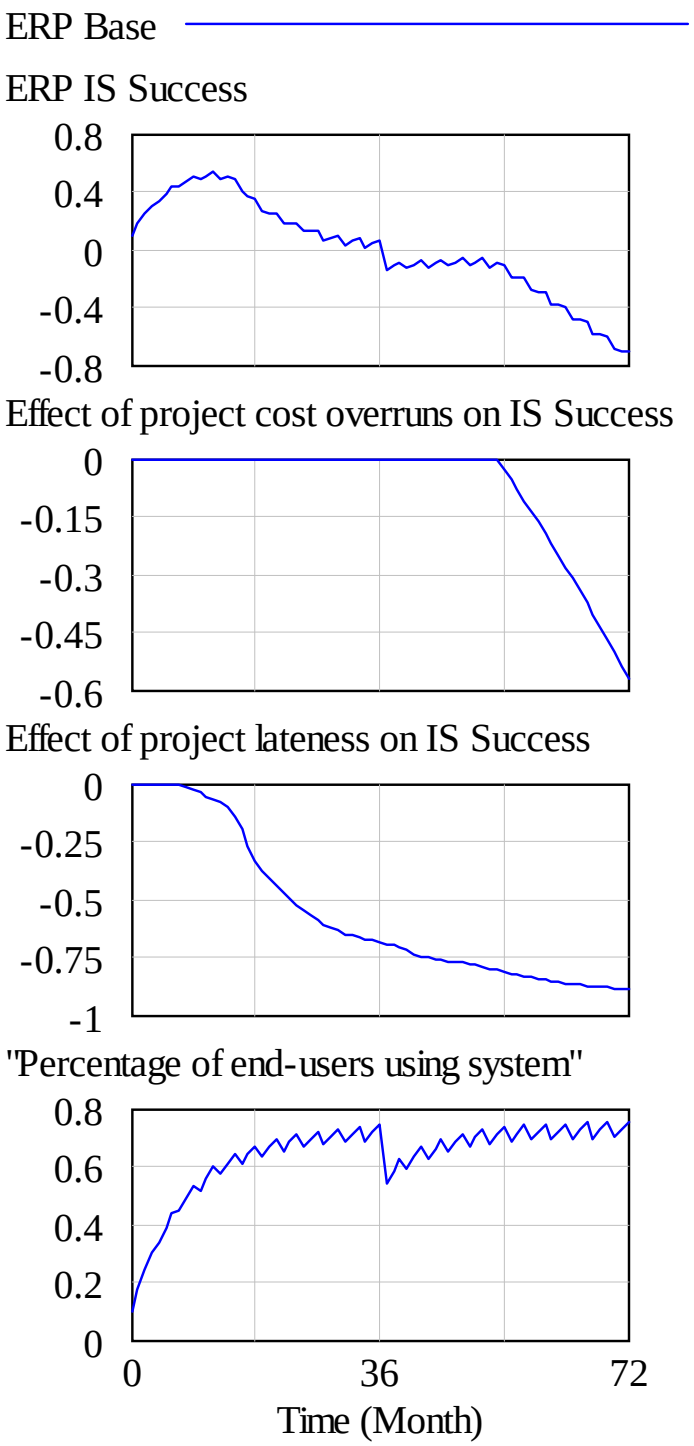
Time, project scope and resources (money/workforce) are competing variables in the model. As the time to implement an ERP extends so does the need to add additional tasks. Some tasks arise from changing user expectations; since the project time has been extended the tendency to ask for additional customizations increases. Other tasks occur from ERP batches and upgrades, which often break previously completed work. Although customizations increase scope, they also increase user satisfaction because the need to change the way they do business is reduced. In the IS Success sector of the model, customizations increase the likelihood that individuals will use the system; thus, positively affecting IS Success (see Figure 2.17).

Figure 2.17 – IS Success



ERP projects are notorious for not meeting original deadlines. As the gap between the actual projected go-live date and the original deadline increases, believe in the success of the system decreases. ERP projects are also infamous for exceeding budget goals. Budget overruns negatively impact IS success as well (see Figure 2.18).

Figure 2.18 – IS Success Causal Strip



Model Assumptions

- **Constant flow of discovering new gaps**
  Technically gaps should run out and would probably not be discovered at a constant rate. Nonetheless, there is an interesting phenomenon where gaps continue to appear no matter how long the project takes to implement; the fit gap never reaches 100%.

- **Entire workforce has same productivity**
  The workforce will have varied levels of experience and talent that will influence productivity. These factors are not included in this model.

- **ERP IS Success operationalization**
  IS Success in this model is determined by willingness to customize and the gap between indicated workforce and actual workforce; this drives user satisfaction. User satisfaction positively influences IS Success while cost and time overruns negatively impact it.

Policy Analysis

Increase Work Hours

In an effort to met project deadlines, management may opt to increase work hours. This policy change does not have the effect one might expect. "Tasks actually remaining" increases from the base run (see Figure 3.1) when workforce hours per week are increased from 40 to 65. This results from a fatigued workforce that is less productive and more apt to produce substandard work. In the end, this policy does not improve IS Success (see Figure 3.2).

Figure 3.1 – Tasks Actually Remaining

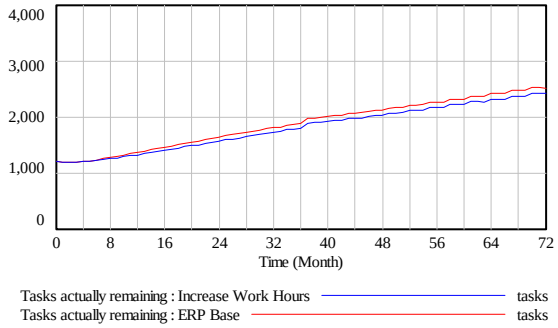
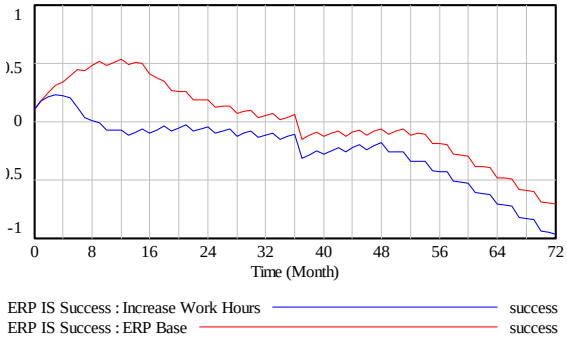


Figure 3.2 – IS Success



Eliminate Customizations

A policy of zero customizations has a dramatic affect on project scope (see Figure 3.3). Unfortunately, a no customizations policy is nearly impossible to implement as some customization will be necessary to met minimum institution requirements. Additionally, the user community is more likely to accept the system if some effort is made on the technical end to fit the system to better meet user needs. Nonetheless, customizations must be carefully considered since they pose an on-going maintenance issue.

Figure 3.3 – Tasks Actually Remaining

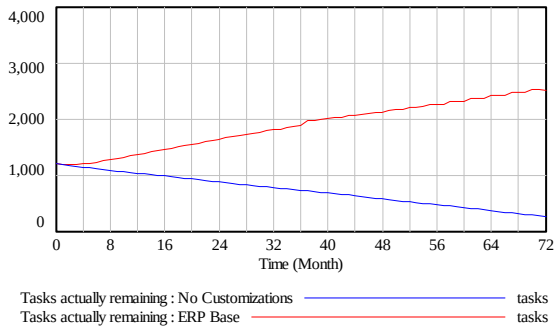
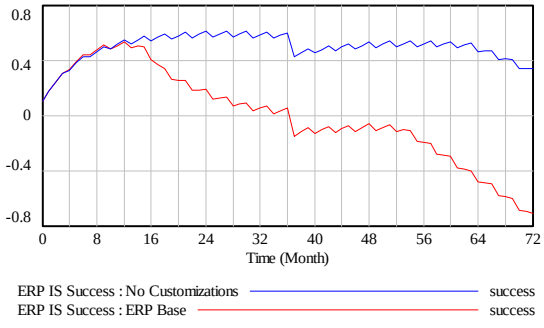


Figure 3.4 – ERP IS Success



Conclusions

Determining the factors that determine ERP implementation success is imperative in order to determine if existing frameworks can be used. On time and within budget implementations do not necessarily mean a successful implementation. If the end result is not well received by the user community or it does not ultimately provide a return on investment then an initially successful implementation can in many ways be perceived as a failure. IT project managers should attempt to provide an ERP solution that users find valuable and usable, while controlling scope, costs and timelines. Unfortunately, these can be conflicting goals so the trick is finding the right balance.

Project managers should assume a certain percentage of rework when determining time and resources needed. Having the proper workforce level from the beginning is particularly important for organizations where the time to adjust workforce is high; this is often true in the public sector. Controlling the number of customizations approved during an ERP implementation can have a dramatic effect on the implementation schedule. Allowing the project timeline to slip is particularly dangerous for ERP implementations because of the fix/upgrade schedule forced by ERP vendors.

IS success is highly influenced by the user community for which the ERP system is intended. The earlier a user is involved in the process the more likely they will ultimately be satisfied with the ERP and the more likely they will actually use the system.

Future Research

Focus groups categorized by functional project participants, technical project participants, and end-users will be conducted to review the model structure and behavior. One benefit of focus groups is that they help a "…researcher to develop an understanding about *why* people feel the way they do" (Bryman, 2004, pg. 348). This is precisely what this research wants to discover and model. The discussions among the focus group participants, along with the building of causal diagrams during these discussions, should help elicit in-depth perspectives about ERP implementation dynamics and success.

Sources

Abdel-Hamid, Tarek, et al. Software Project Dynamics: An Integrated Approach Prentice-Hall: New Jersey, 1991.

Bryman, Alan. Social Research Methods (2nd edition), New York: Oxford University Press (2004).

Davenport, Thomas H. "Putting the Enterprise into the Enterprise System" *Harvard Business Review*, July-August 1998.

DeLone, W. and McLean, E. "The DeLone and McLean Model of Information Systems Success: A Ten-Year Update" Journal of Management Information Systems, Spring 2003, Vol. 19, No. 4, pp 9 – 30.

DeMarco, T. Controlling Software Projects New York, NY: Yourdon Press, Inc., 1982.

Dennis, A, Venkatesh, V, and Ramesh V "Adoption of Collaboration Technologies: Integrating Technology Acceptance and Collaboration Technology Research" http://www.indiana.edu/~isdept/research/workingpapers.html, ID TR142-1, (Last updated February 2004)

Gefen, David, "What Makes an ERP Implementation Relationship Worthwhile: Linking Trust Mechanisms and ERP Usefulness", *Journal of Management Information Systems*, Summer 2004, Vol. 21, No. 1, pp 263-288.

Kaiser, K M and King, W R (1982), "The Manager-Analyst Interface in Systems Development", *MIS Quarterly*, March pp 49-59.

Martin, M.H. (1998) "An ERP Strategy", *Fortune*, February 1998, pp. 95-97

Peterson, S (2003), "Lost Signals: How Poor Communication and Other Nontechnical Issues Hampered Arkansas' Innovative Statewide ERP Implementation", Government Technology, February.

Senn, J A (1978), "A Management View of Systems Analysts: Failures and Shortcomings", *MIS Quarterly*, September pp 25-32.

Shanks, G. et al. (2000) Differences in Critical Success Factors in ERP Systems Implementation in Australia and China: a Cultural Analysis, European Conference on Information Systems

Shih, Hung-Pin, Extended Technology Acceptance Model of Internet Utilization Behavior, Information & Management, Volume 41, Issue 6, July 2004, Pages 719-729

Skok, W. et al (2001) Potential Impact of Cultural Differences on Enterprise Resource Planning (ERP) Projects, The Electronic Journal on Information Systems in Developing Countries

Sterman, John, Business Dynamics Boston, NY: McGraw Hill Companies, Inc., 2000.

Tapp, R M, Hesseldenz, J and Kelley, G (2003), “The Role of Project Acceptance in the Successful PeopleSoft Human Resources Management System Implementation for the Kentucky Community an Technical College System”, *Ninth Americas Conference on Information Systems*, pp 1380-1388

Taylor, S. and Todd, P.A. “Understanding Information Technology Usage: A Test of Competing Models” *Information Systems Research*, (2001), Vol. 6, No. 2, pp. 144-176

Venkatesh, V and Davis, F. “A Theoretical Extension of the Technology Acceptance Model: Four Longitudinal Field Studies” *Management Science*, Vol. 46, No. 2, Feb 2000, pp. 186-204

Venkatesh, V, Morris, M, Davis G and Davis F. “User Acceptance of Information Technology: Toward a Unified View” *MIS Quarterly*, Vol. 27, No. 3, September 2003

Vitale, M R (1986), “The Growing Risks of Information Systems Success”, *MIS Quarterly*, December pp 327-334.

Wysocki, Jr., Bernard, "Pulling the Plug: Some Firms, Let Down By Costly Computers Opt to "De-Engineer"" *The Wall Street Journal*, April, 30, 1998.

Zhang, Liang (2002) “Critical Success Factors of Enterprise Resource Planning Systems Implementation Success in China”.