

ERP Implementation Dynamics

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

I. Abstract

ERP projects are often undertaken by project managers in an effort to solve a problem, increase efficiency, and/or provide a higher level of customer service. Although ERP systems can provide all of these benefits and more, they can also cause havoc in an organization if not managed correctly. There are far too many horror stories about organizations failed ERP initiatives. In fact, the success rate of ERP implementations is only around 33% and approximately 90% of ERP implementations are late or over budget (Martin 1998).

How can a wonderful thing such as ERP cause such heartache? Often the issue has nothing to do with technology and everything to do with the individuals involved with the project. ERP problems arise from unrealistic expectations regarding resources and cooperation required to implement an ERP system successfully.

ERP implementation articles consistently report that implementation failure or success is people-related (Tapp 2003 & Peterson 2003). It's 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 complexities of the people-related issues, relationships and office politics before embarking on a new ERP project. This research is intended to provide insight regarding ERP implementation dynamics through modeling; to build and explore theories regarding what causes ERP success/failure and ultimately aid project managers in avoiding common pitfalls.

Problem Dynamics

Some of the dynamics in ERP implementations include people's belief that the project will be successful. This belief may change over time and/or be influenced by various other factors. For example, if people believe there has been significant progress (perceived progress) then they might feel good about the project and have faith in its success.

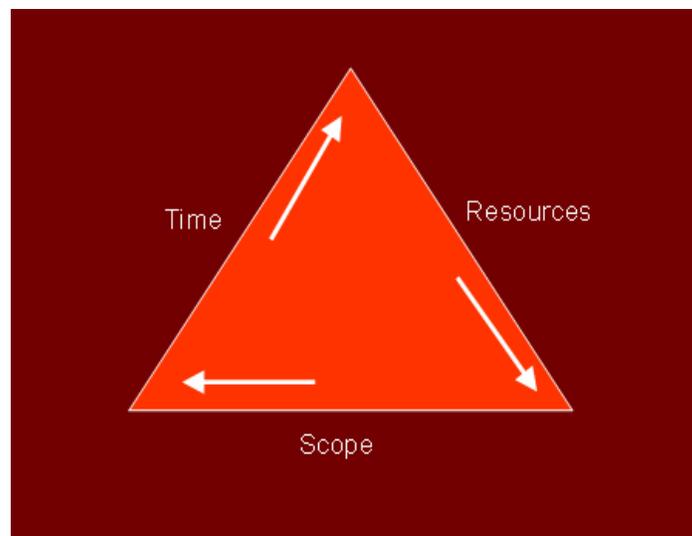
Additionally, as time passes management's willingness to invest additional funds or workforce might change. In the beginning of the project lifecycle resources might be

plentiful but as managers believe the project is close to completion they may be less willing to invest additional resources.

Justification for approaching problem with system 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 elements that cause enormous headaches for project managers; these elements all have feedback behavior. Figure 1-1 shows three common elements of ERP implementations. If any of the sides of the triangle (time, resources, & scope) are stretched, one or both of the remaining sides must stretch to accommodate the change. For instance, as the scope of a project increases, the time required and/or resources needed are 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.

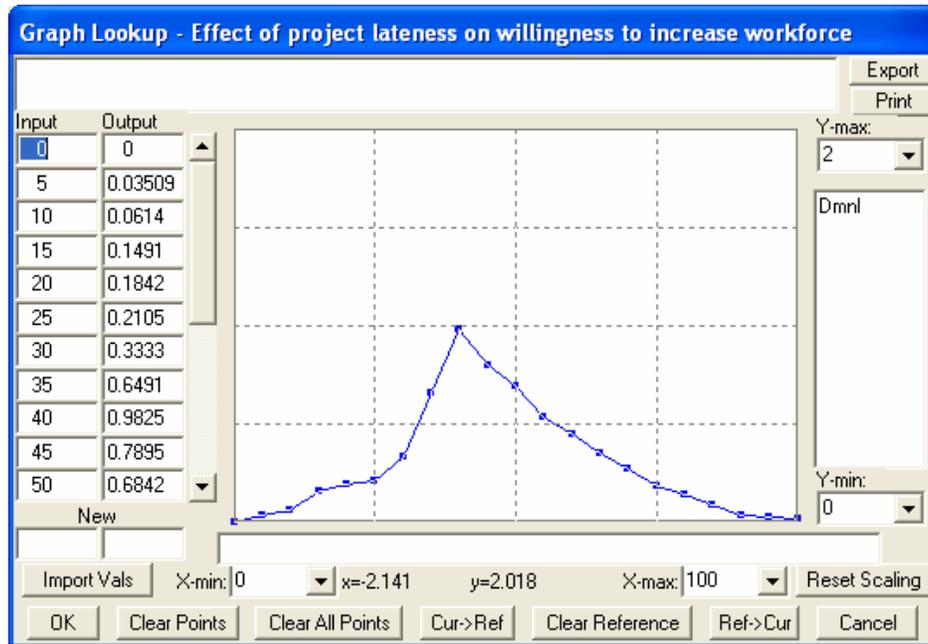
Figure 1.1



Reference Modes

As an ERP project moves through time, the willingness to invest additional resources (money/people) changes drastically. Initially when a project is late, management is willing to increase workforce in an effort to implement as soon as possible. Nonetheless, as more time passes thoughts of abandoning the project cause a decline in the willingness to increase workforce (Figure 1.2).

Figure 1.2 – Effect of Project Lateness on Willingness to Increase Workforce



Time and cost overruns influence several variables. For example, the longer it takes to implement a project the more likely tasks will need to be revisited. ERP software fix bundles and upgrades occurring during an implementation cause spikes in task rework. Unfortunately, these 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 (Figure 1.3). Fixes and upgrades not only break tasks but they often introduce new tasks (Figure 1.4).

Figure 1.3 – Customizations Needing Rework or Reapplication

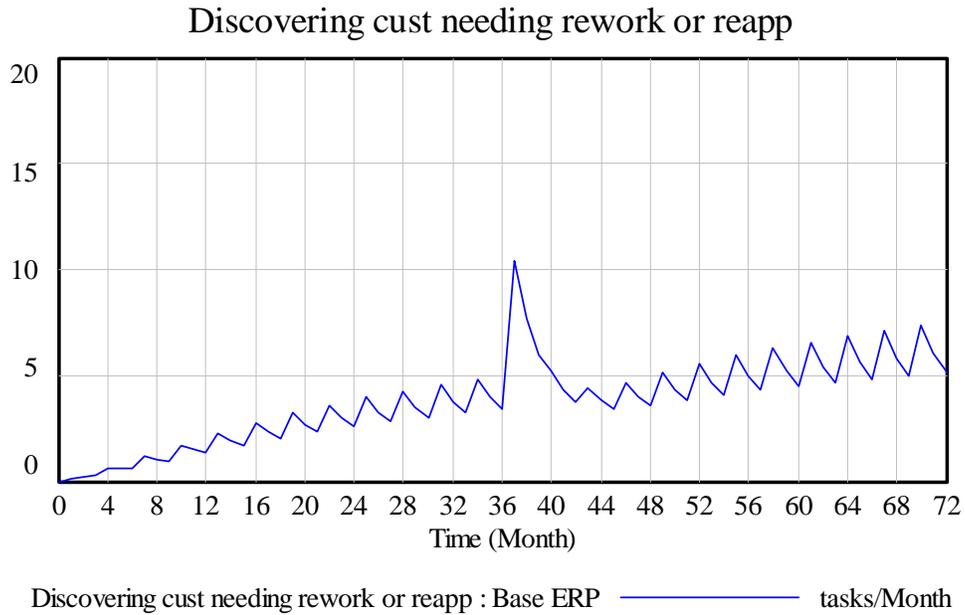
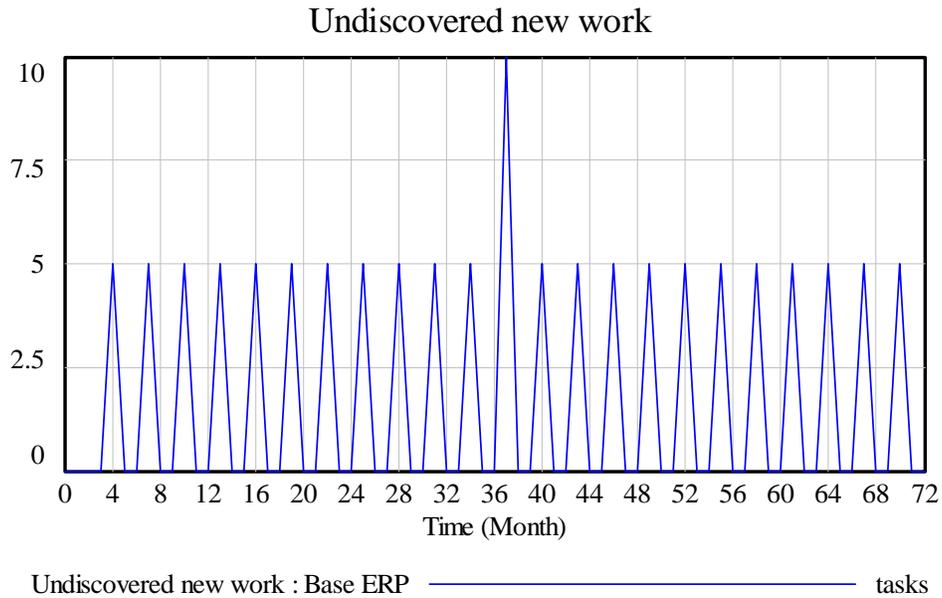
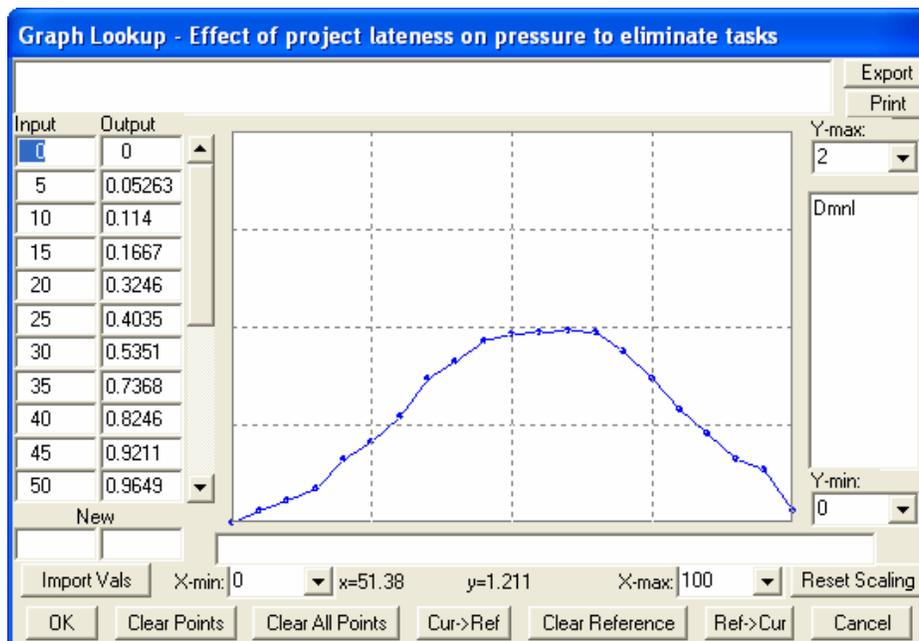


Figure 1.4 – Undiscovered New Work from Fixes/Upgrades



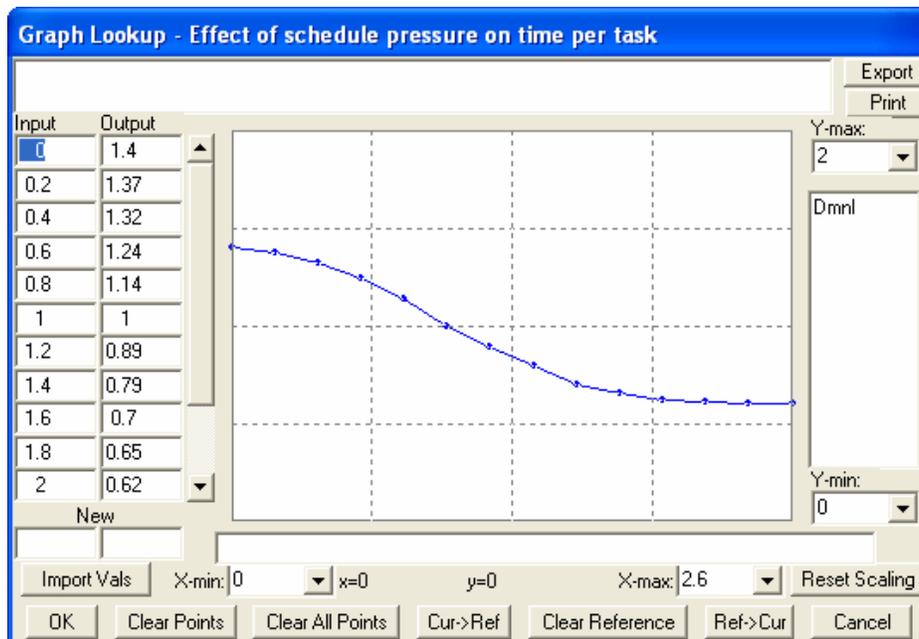
Scope reduction is one way to counteract time and cost overruns. Pressure to eliminate tasks increases as an implementation passes its scheduled completion date. However, if a project is extraordinarily late then the elimination of tasks becomes difficult to justify (Figure 1.5).

Figure 1.5 – Effect of Project Lateness on Pressure to Eliminate Tasks



The pressure of meeting implementation deadlines can have multiple effects. If schedule pressure is high then one way to reduce this pressure is to complete tasks more rapidly (Figure 1.6) (Serman 2000).

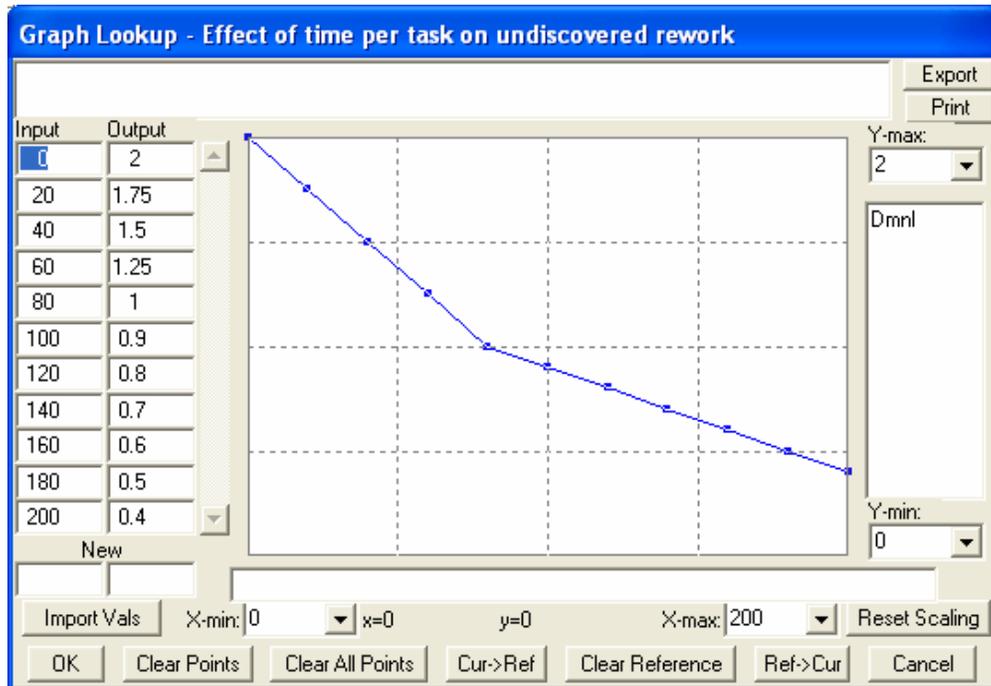
Figure 1.6 – Effect of Schedule Pressure on Time per Task



Data Source: Serman 2000

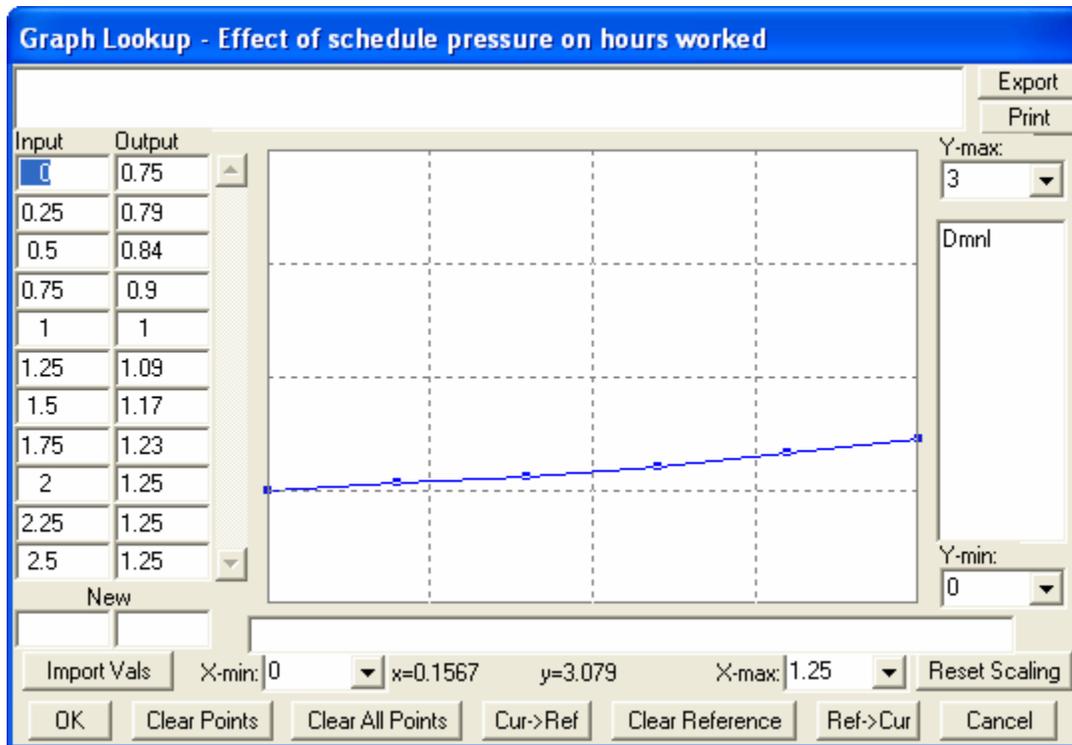
Shortening the time to complete tasks is often accomplished by reducing or skipping proper testing. The end result of task time reduction is that task quality decreases and it is likely that tasks will need to be revisited (Figure 1.7).

Figure 1.7 – Effect of Schedule Pressure on Time per Task



Schedule pressure also influences the normal hours worked per month (Figure 1.7).

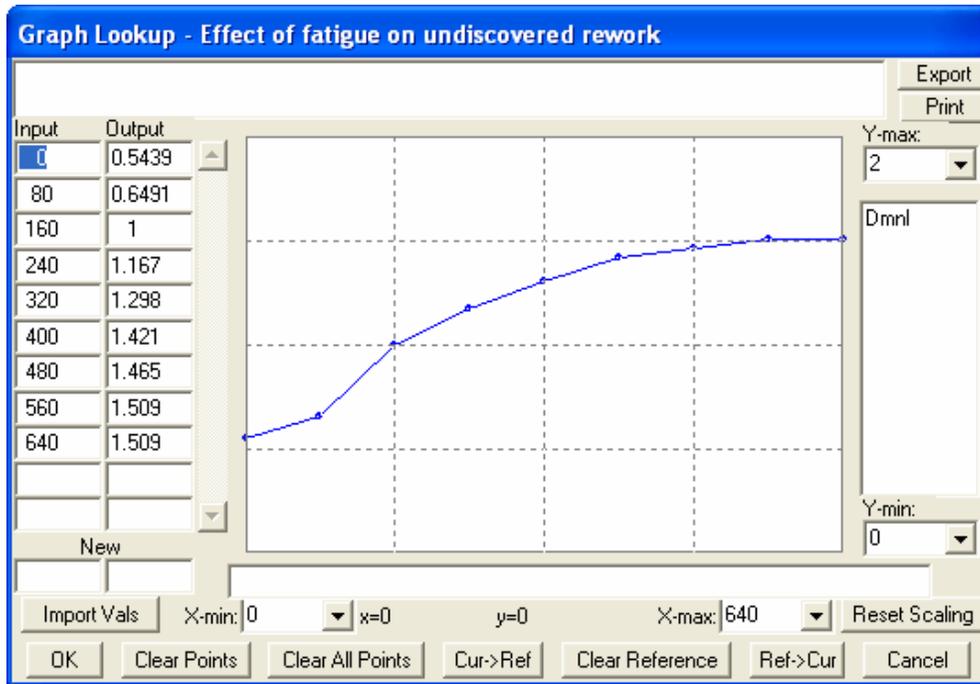
Figure 1.7 – Effect of Schedule Pressure on Hours Worked



Data Source: Sterman 2000

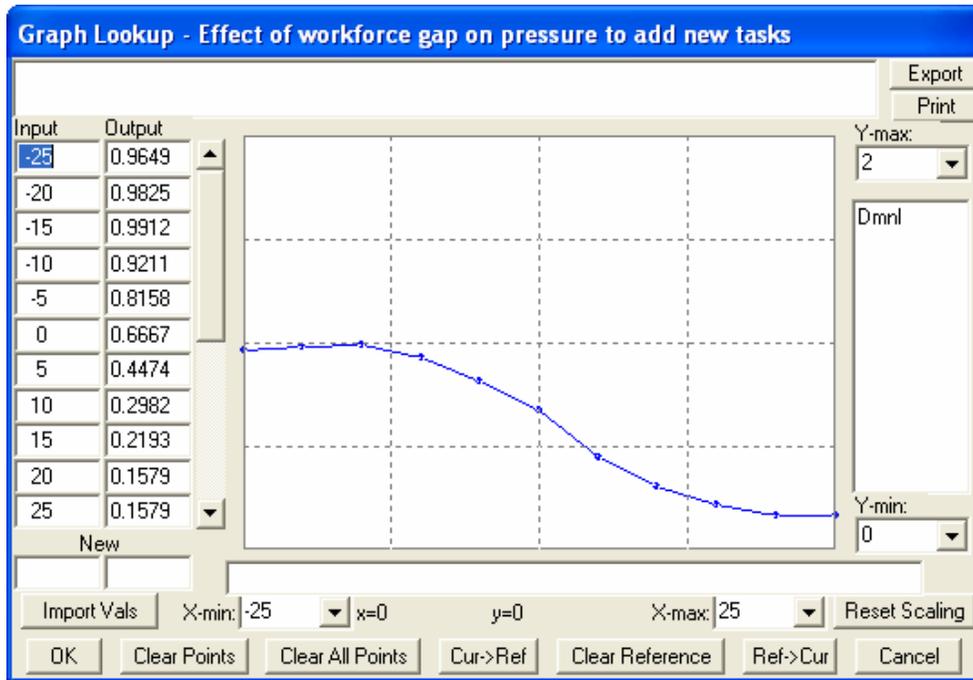
Increased work hours causes workforce fatigue, which ultimately affects the quality of work (Figure 1.8).

Figure 1.8 – Effect of Fatigue on Undiscovered Rework



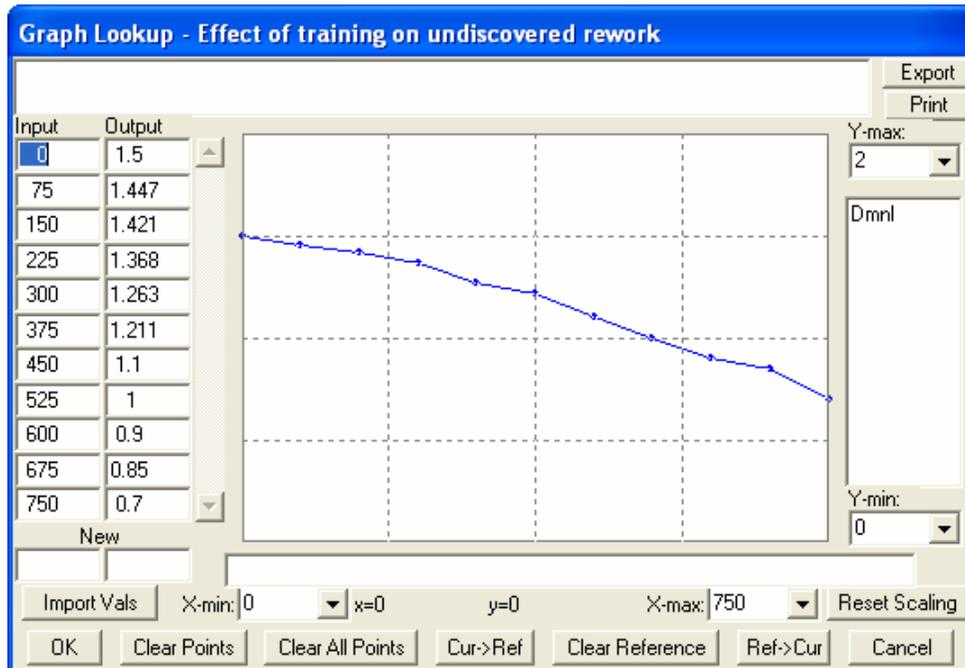
Workforce size impacts the pressure to change project scope (Figure 1.9). While a large gap between indicated workforce and actual workforce decreases the pressure to add new tasks, as this gap reduces belief that new tasks are justified increases. Consequently, both resources and scope increase together so anticipated time reduction benefits are not realized.

Figure 1.9 – Effect of Workforce Gap on Pressure to Add New Tasks



Training of the workforce can be costly and time consuming but a properly trained workforce would produce a better product that is less likely to require rework (Figure 1.10).

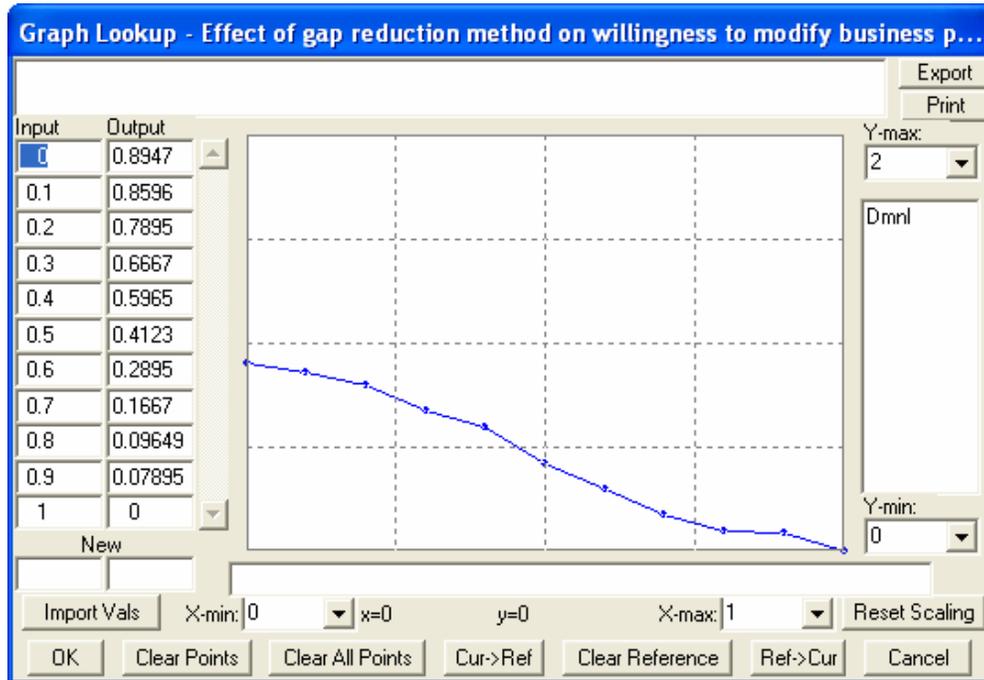
Figure 1.10 – Effect of Training on Undiscovered Rework



An appropriate ERP system should be purchased that closely matches the business for which it is intended. Nonetheless, a certain amount of fit gap will be necessary. There

are two ways in which to reduce gaps between an organization's business needs and the delivered ERP. The system can be customized (modified) to match current business processes or the business processes can be modified to better fit the delivered ERP system. Classic ERP implementation strategies suggest modifying business processes as much as possible to fit the product.

Figure 1.11 – Effect of Gap Reduction Method on Willingness to Modify Business Processes



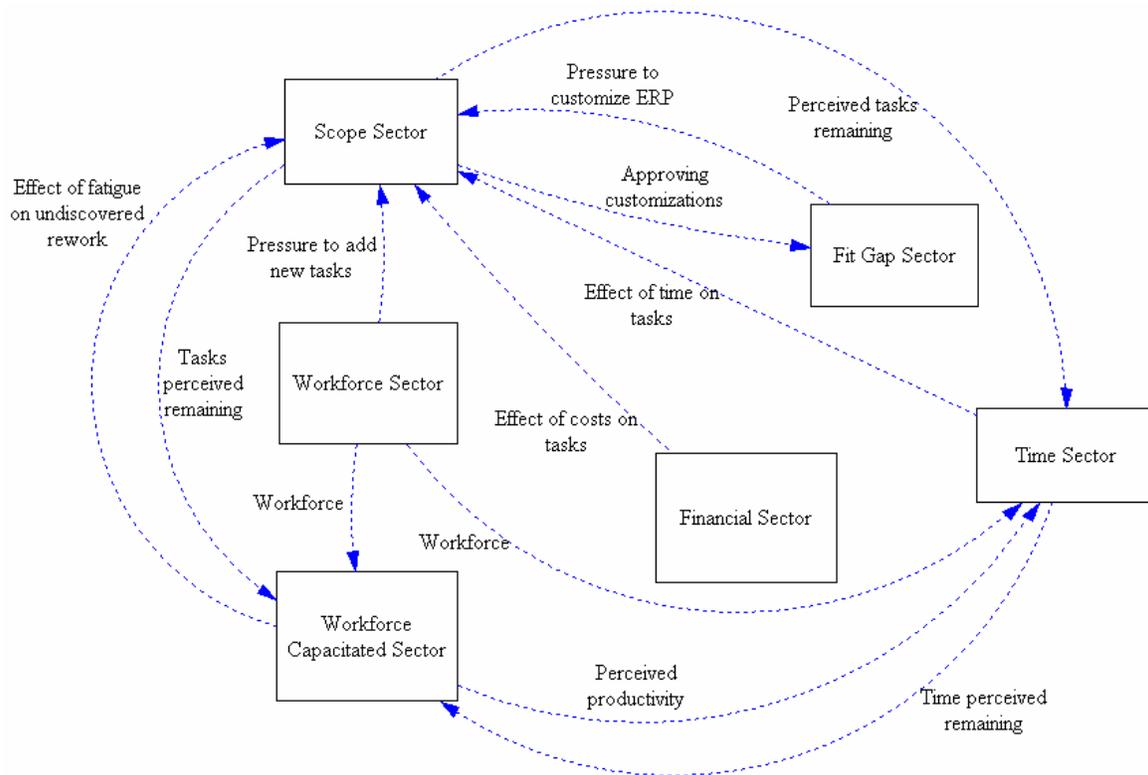
II. Model Structure

Some of the dynamics involved with typical ERP implementations are shown in the diagrams below. The development of these diagrams acted as building blocks toward the creation of the model for ERP implementation dynamics.

Sector Overview Diagram

The current model contains six sectors (Figure 2.1).

Figure 2.1 – Sector Overview Diagram



Causal-Loop Diagrams

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 (Figures 2.2 & 2.3).

Figure 2.2 - Causal-Loop Diagram with Explicit Stocks

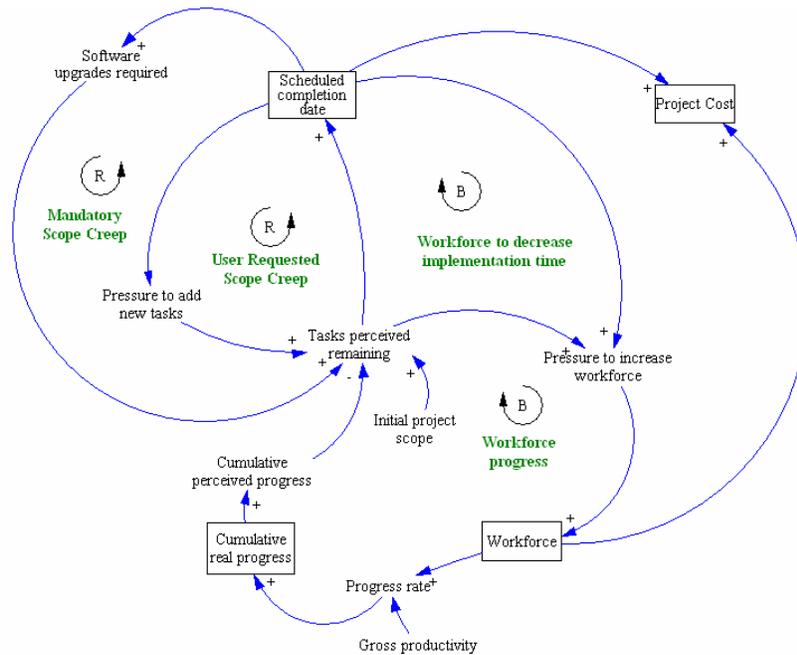
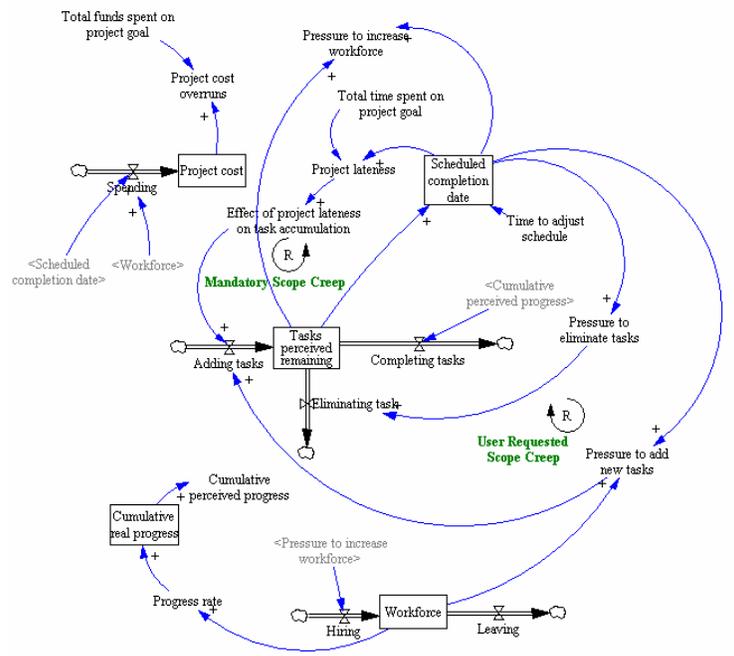


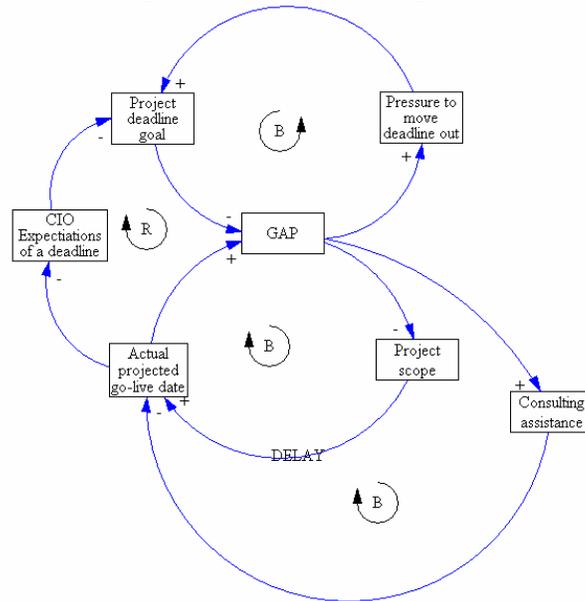
Figure 2.3 - Causal Loop Diagram with Explicit Stocks and Flows



System Archetypes

System archetypes help identify some additional structures that have been or will be included in the model.

Figure 2.4 – Scope Drifting Goals



ERP projects are notorious for not meeting original deadlines. As the gap between the actual projected go-live date and the deadline goal increases, project scope is reduced and/or consultants are hired to help reduce the deadline gap. Additionally, there are other variables such as CIO expectations putting pressure on maintaining the original deadline goal (Figure 2.4).

Shifting the Burden

ERP systems never completely match the business processes of any particular organization. It is for this reason that ERP's are often customized to fit the organization instead of examining the business processes and modifying to accommodate the new system. 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 so that more customizations are requested (Figure 2.6). Customizations may not sound like a bad option but there are many negative implications, some of which are explored in this research.

Figure 2.6 – Shifting the Burden

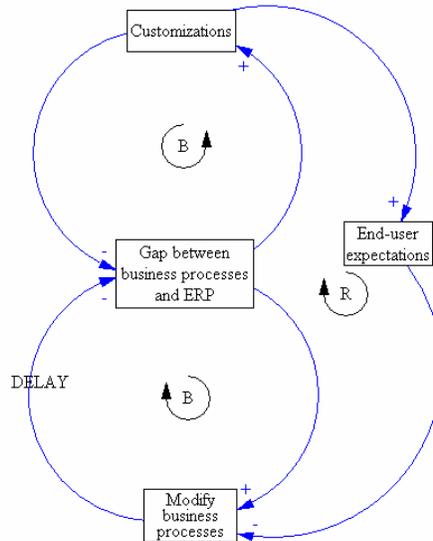
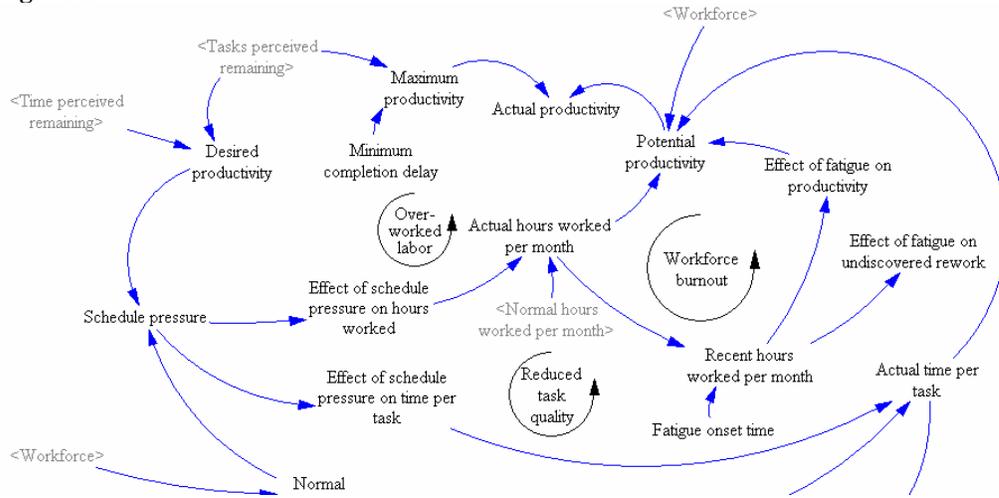


Figure 3.2



Project lateness can have many negative affects on an ERP implementation. 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 (Figure 3.3) and existing customization work often needs to be reapplied; this is particularly true for full upgrades (Figure 3.4).

Figure 3.3

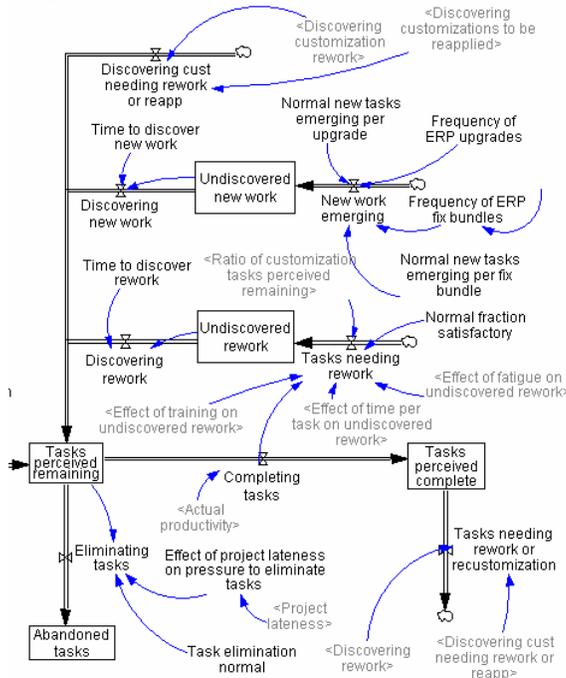
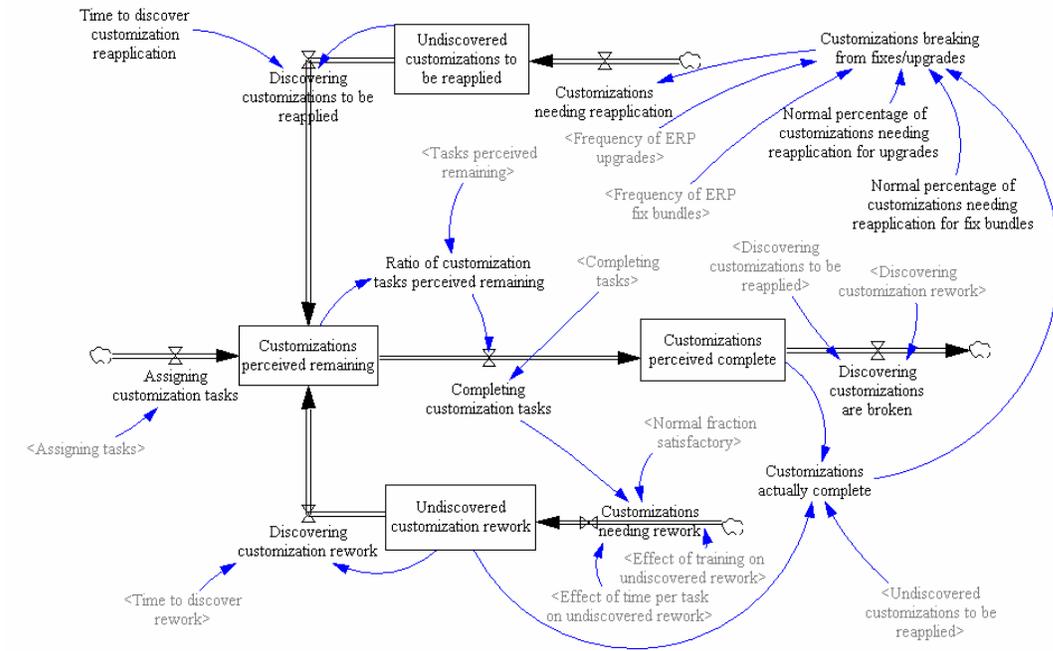
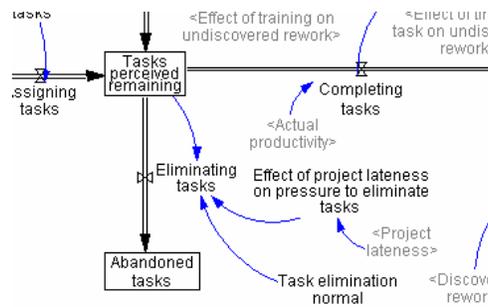


Figure 3.4



Project scope never stays constant during a project implementation lifecycle. For various reasons additional tasks are added to the project plan regularly. As new gaps are discovered between what the ERP system offers and user community needs, customization requests are introduced. This also adds complexity in that tasks actually remaining begin to differ greatly from tasks perceived remaining. Not only does this disparity include undiscovered rework but approved customizations that have not yet been given to the technical staff. Furthermore, there may be more customizations in the requested queue that will eventually be approved.

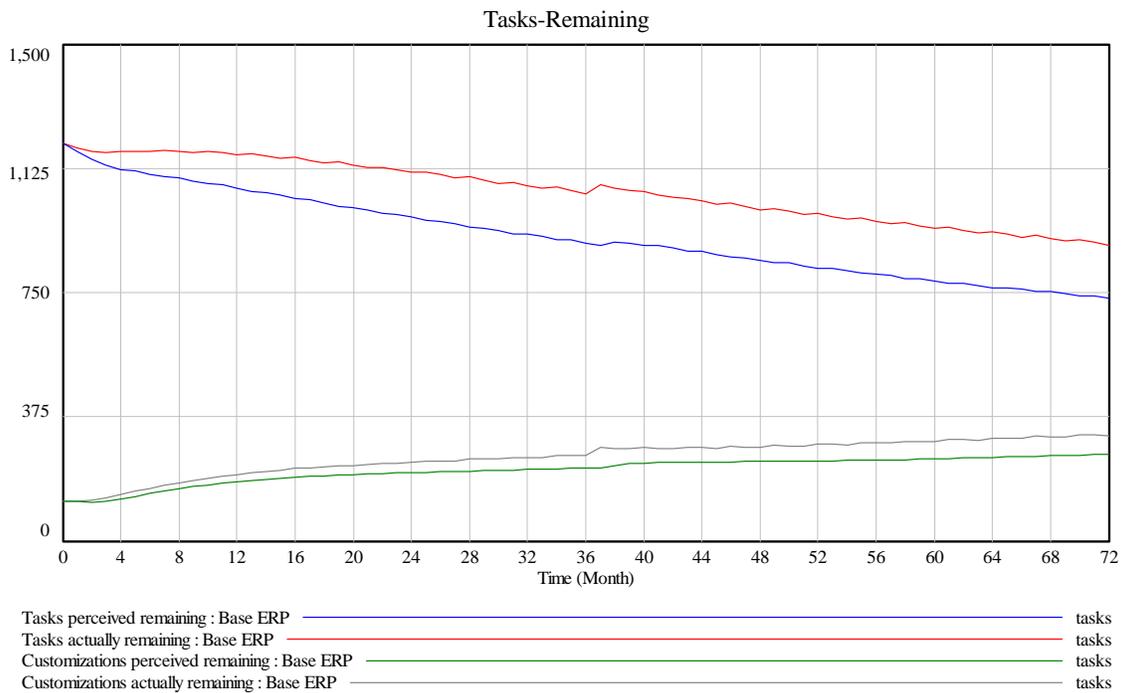
Figure 3.7



Model Behavior

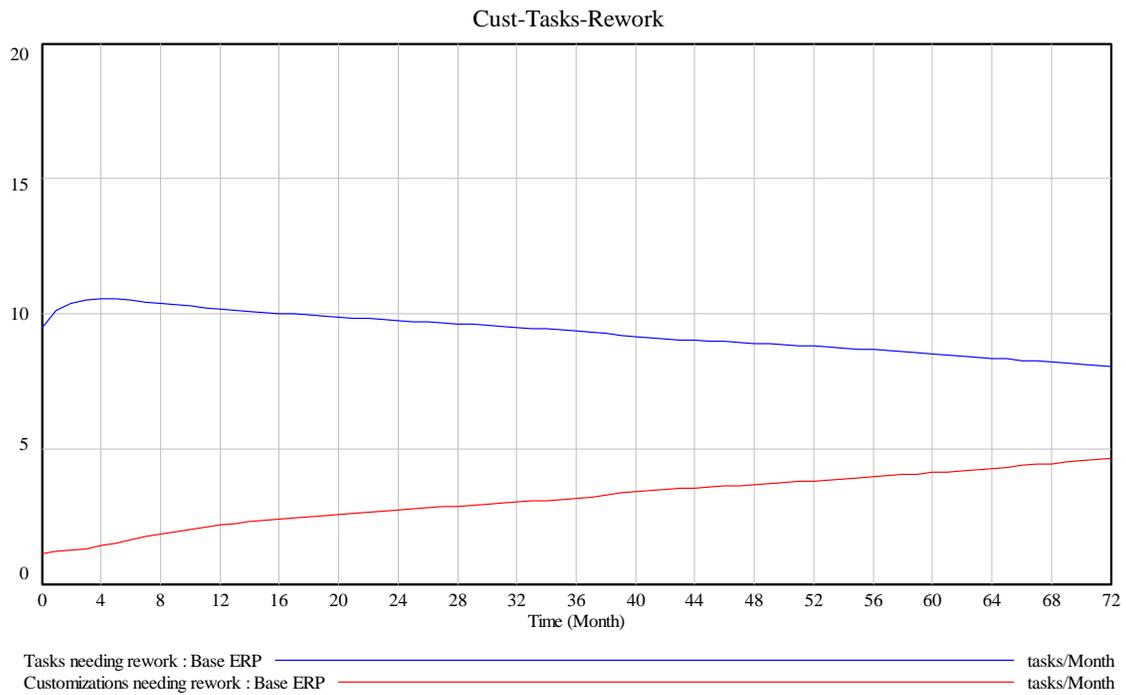
In the base run of the model tasks remaining declines over time but customizations actually increase. Customizations are the tasks that frequently need rework due to fix bundles and upgrades. As one might expect, actual and perceived tasks remaining differ fairly significantly. This difference causes incorrect estimation on time and resources needed to meet the implementation deadline.

Figure 3.8



The nature of customizations, together with project implementation delays, causes customization rework to slowly increase over time (Figure 3.9).

Figure 3.9



Policy Analysis

Increase Work Hours

In an effort to meet project deadlines, management may opt to increase work hours. This policy change does not have the effect one might expect. Tasks remaining actually increases (Figure 3.10) over the long term when workforce hours per month is increased from 160 to 260. This results from a fatigued workforce that is less productive (Figure 3.11) and more apt to produce substandard work (Figure 3.12).

Figure 3.10

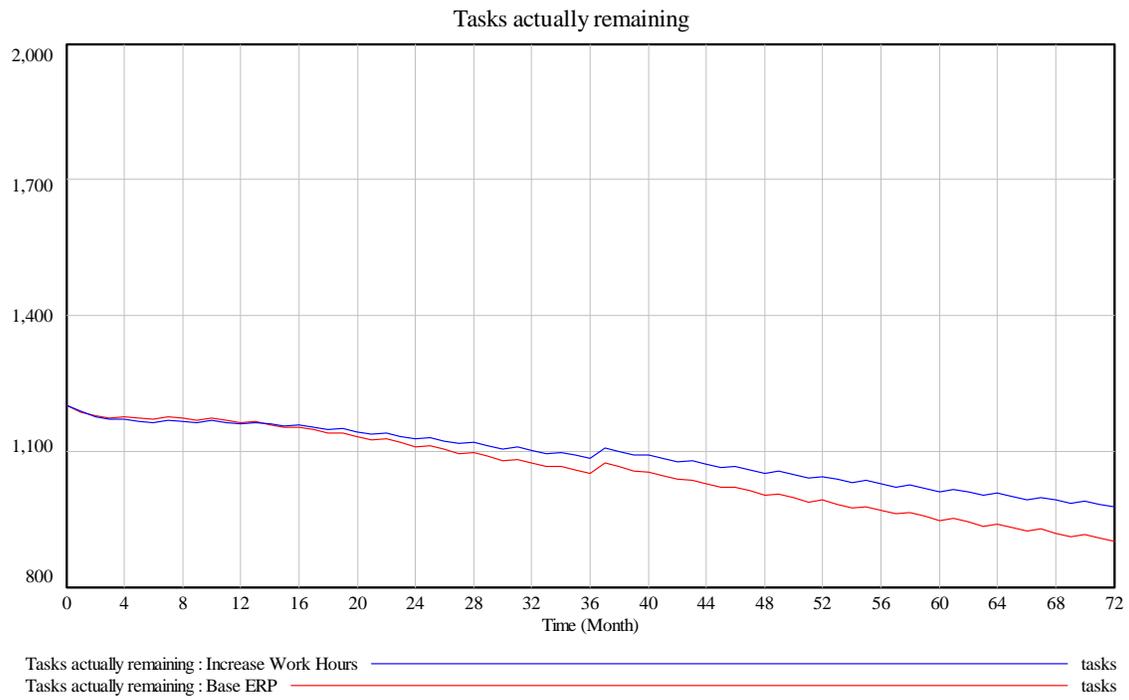


Figure 3.11

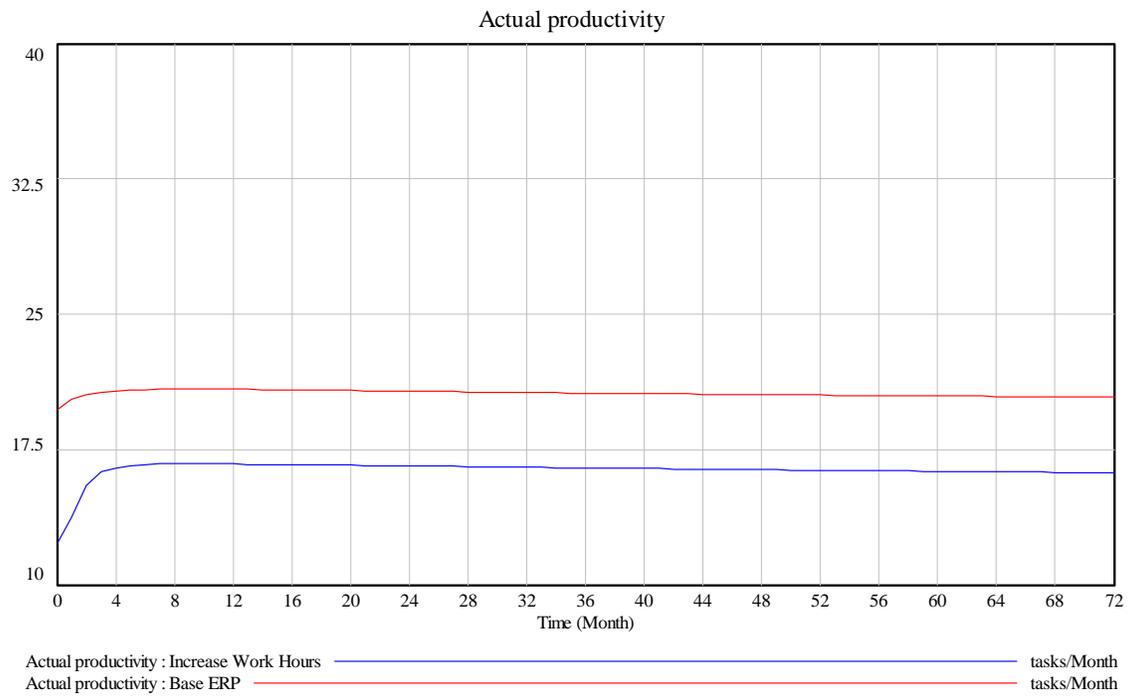
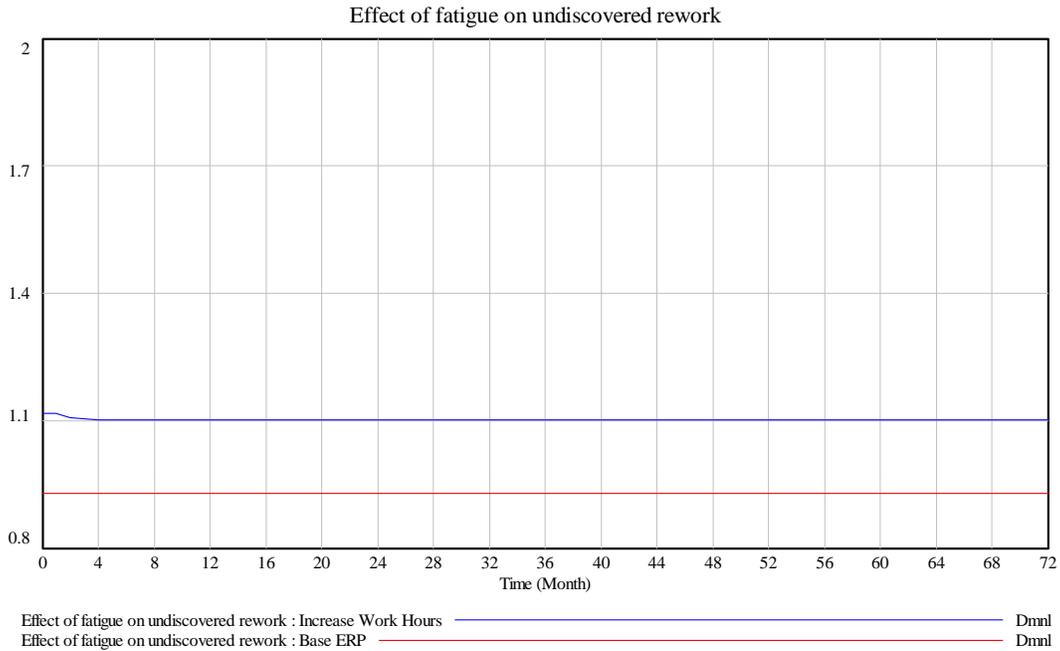


Figure 3.12



Decrease Work Hours

Decreasing work hours per month from 160 to 60 increases productivity slightly because significantly less time is spent on each task (Figure 3.13). Unfortunately, the decrease in time spent on tasks also decreases task quality (Figure 3.14). One positive is that workforce fatigue is lowered, which to some extent positively influences work quality (Figure 3.15).

Figure 3.13

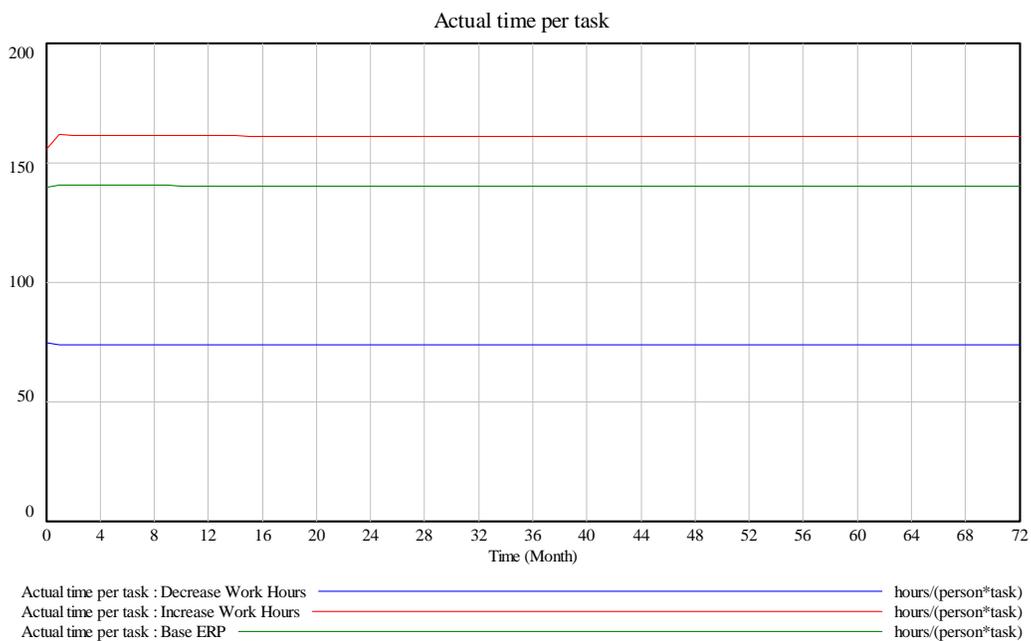


Figure 3.14

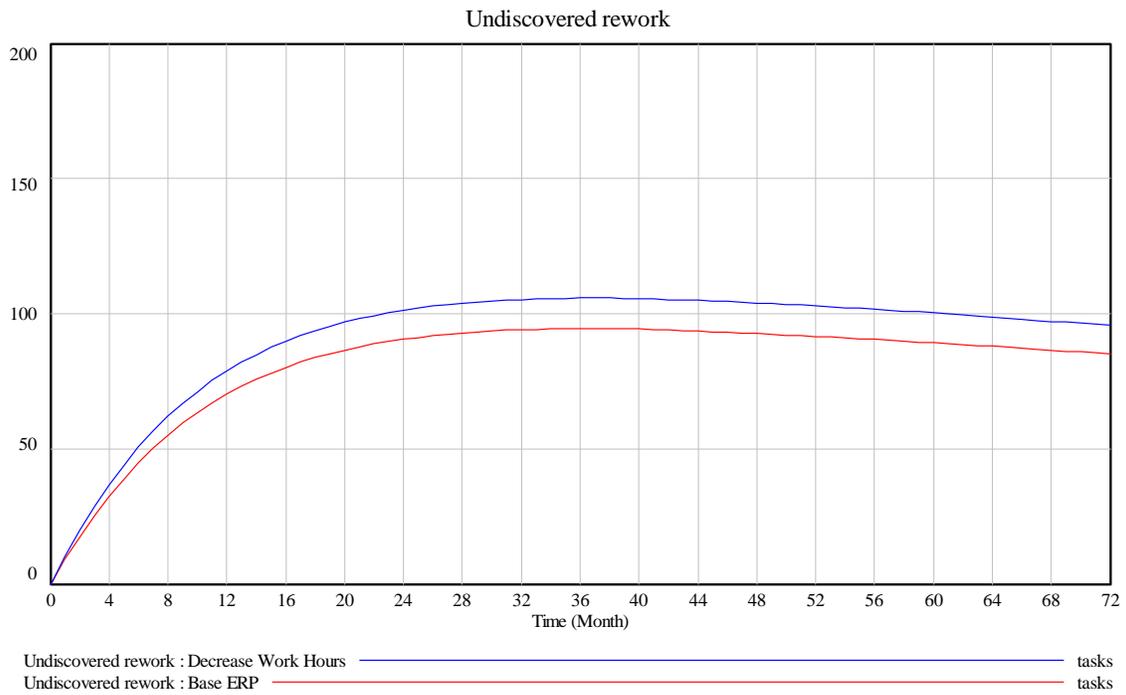
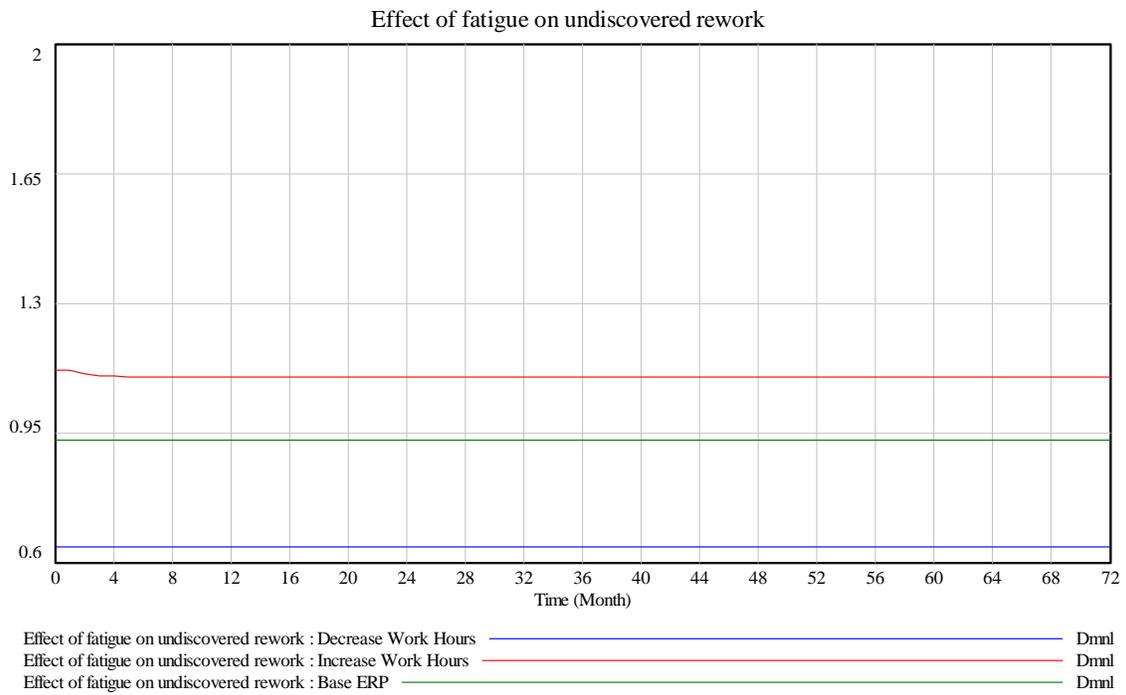


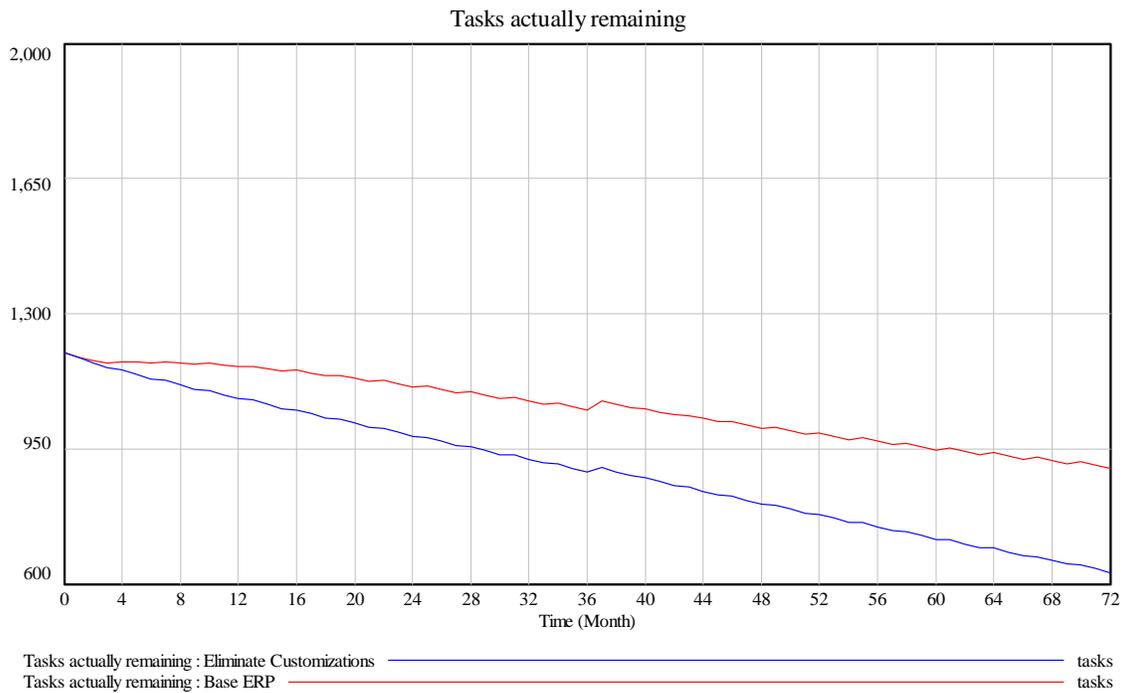
Figure 3.15



Eliminate Customizations

Closing out new customization requests by setting customization approval ratio to 0 instead of 75% slashes the project scope significantly (Figure 3.16). Unfortunately, this policy is nearly impossible to implement as some customization will be necessary to meet 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 3.16



IV. Conclusions

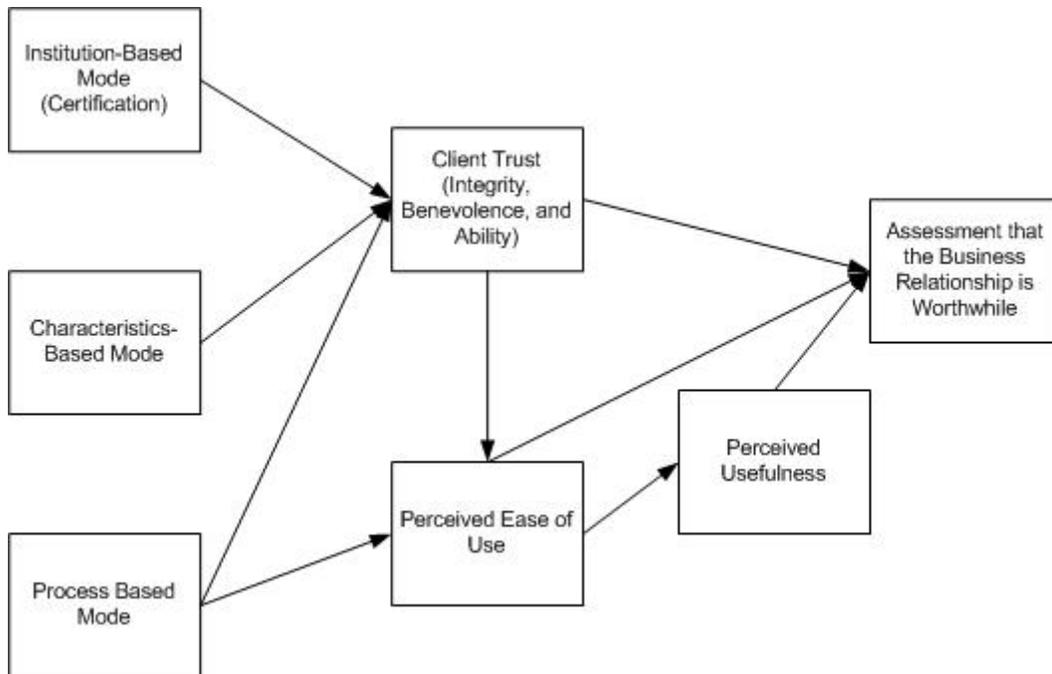
Project managers should assume a certain percentage of rework when determining time and resources needed. Having the proper workforce level from the beginning is particularly important for organizations where the time to adjust workforce is high; this is often true in the public sector. Controlling the number of customizations approved during an ERP implementation can have dramatic effects 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.

Future Research

The current model is exploring causes of scope, time and cost overruns. However, 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.

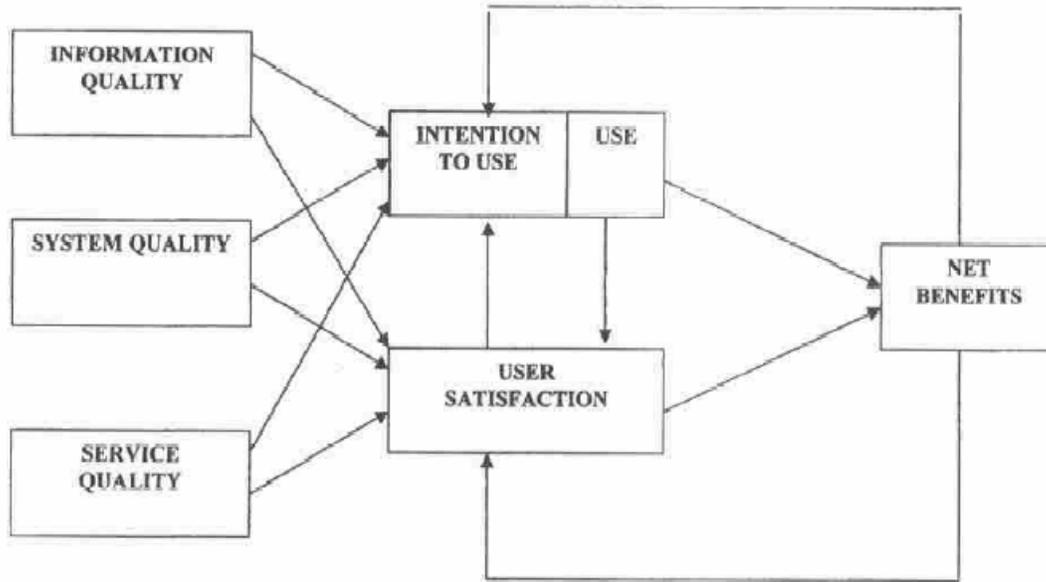
Future extension of this research will include influences from Technology Acceptance Models (TAM), which are often used to explain why technology is or is not successful. TAM influenced models (Figure 4.1) clearly point out the relationship between “Client Trust” and its effect on “Perceived Ease of Use” but do not include many obvious feedback loops such as the effect of “Perceived Ease of Use” on “Client Trust”.

Figure 4.1 - Research Model Dealing with the Contact Person's Perspective on Business Relationships (Gefen 2004)



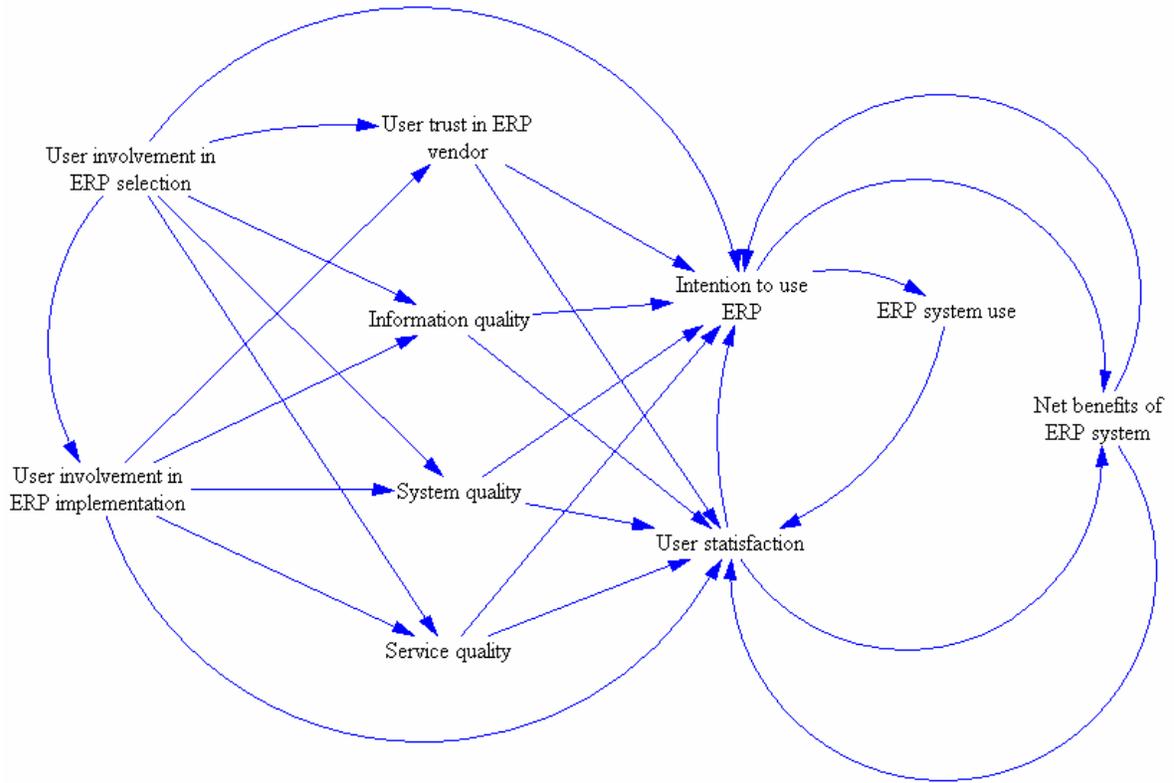
User acceptance and IS success are highly influenced by the user community for which the ERP system is intended. The earlier a user is involved in the process the more likely they will ultimately be satisfied with the ERP and the more likely they will actually use the system.

Figure 4.2: The reformulated model of IS Success (DeLone and McLean, 2003)



Expansion of this model will include an ERP Success sector (Figure 4.3); an extension of TAM and the D&M IS Success Model (Figure 4.2). Additionally, ERP Success could be expanded to include variables such as cost, organization culture, and top management attitudes.

Figure 4.3 – ERP Success Model



Sources

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

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.

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.

Richardson, George, *Modelling for Management: Simulation in Support of Systems Thinking*, The International Library of Management. Aldershot, U.K.: Dartmouth Publishing Company, 1996.

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*

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".