Post Adoptive ERP Use Behaviors: A Dynamic Conceptualization

Thomas D. Clark, Jr.
President, Strategy Associates, LLC
PO Box 270142
Littleton, Colorado 80127
720 236 6123
tom54clark@comcast.net

Mary C. Jones
Professor and Chair
Information Technology & Decision Sciences Department
College of Business Administration
University of North Texas
Denton, TX 76205
940 565 3167/Fax: 940 565 4935
jonesm@unt.edu

Robert W. Zmud
Michael F. Price Chair in MIS and George Lynn Cross Research Professor
MIS Division
Price College of Business
University of Oklahoma
Norman, OK 73019
405-325-0791

Abstract

A conceptual model of post-implementation enterprise resource planning systems (ERP) use is developed and operationalized using a dynamic systems perspective. The primary purpose is to enrich our collective understanding of how companies might facilitate ERP usage to enhance the business value of this technology investment after the initial installation. The conceptual model represents a dynamic information feedback structure that illustrates the relationships among several post-adoptive factors including software and work process training, experiential interventions to facilitate extended and deeper usage, user software and work process understanding, the extent of features implementation, the range of ERP system usage, and ERP benefits. The model is drawn from findings about post-ERP implementation described in the literature. Pragmatic insights are provided by the conceptual model and recommendations for future research are discussed.

Key Words: enterprise resource planning, post adoptive usage, information systems

Introduction

Enterprise resource planning (ERP) systems enable organizations to streamline operations, leverage common business processes, and manage multiple operations through an integrated suite of software modules and a centralized database (Scott and Kaindl 2000). By the turn of this century, at least 30,000 companies worldwide had implemented ERP (Mabert, et al. 2001), including over 70% of the Fortune 1000 and 80% of the Fortune 500 (Barker and Frolick 2003; Gattiker and Goodhue 2005). By 2004, companies were investing nearly $80 billion annually worldwide for their ERP initiatives (Gefen and Ragowsky 2005), and ERP investments are continuing to grow (Jacobson, et al. 2008). ERP is a large portion of the applications budget in large and medium size firms, and firms that implement ERP estimate that they devote about one-third of their information technology (IT) budgets to supporting this capability (Seewald 2002). Midsize firms that have installed ERP estimated that they would increase their ERP budgets by over 5% in 2008 (McGreevy 2007): IT executives across a variety of organizational sizes rank ERP as one of their top ten application and technology concerns (Luftman and Kempaiah 2007), and many continue to face an ERP skills gap in their organizations (Stiffler 2008).

In spite of heavy investments, mixed evidence remains regarding firms' success in leveraging their ERP installations. In the early days of industry installation of ERP, some firms had to restate financial results due to errors in implementation and some even sued their ERP vendors for losses suffered after implementation (Songini 2003a; 2003b). Unfortunately, this is still occurring nearly a decade after the initial surge in ERP adoptions. For example, Overstock.com recently restated its earnings for a 5 ½ year period, and cited problems with its ERP implementation project that dated back several years (Kanaracus 2008b). Waste Management has filed suit against its ERP vendor over a failed ERP implementation (Kanaracus 2008a). Some have achieved benefits, but not nearly to the extent that they had believed that they would (Swanton 2004b; Jones et al. 2008). Other firms, however, are realizing up to 80% of the expected benefits (Swanton 2004b), with many of these reporting having used ERP to cut significant costs from their operations and drive substantive changes throughout the entire organization (Thibodeau 2004).

Although stories of ERP successes and failures are widely discussed in both academic and practitioner publications, surprisingly little evidence exists about how well ERP has actually been assimilated in adopting organizations beyond the initial implementation. Little is known about how extensively or faithfully organizations use ERP functionalities (Sarkis and Sundarraj 2001) because much of the research to date focuses on data collected just prior to, during, or just after ERP software implementation (Hunton, et al. 2003; Jones, et al. 2006; Ke and Wei 2008; Zviran, et al. 2005). Although such research provides valuable insights, it overlooks what is arguably the most important aspect of an ERP implementation: the longer-term realization of benefits. Two notable exceptions are the research by Gattiker and Goodhue (2002; 2004; 2005) and Liang, et al. (2007). Gattiker and Goodhue (2002, 2004, 2005) focus largely on the role of structural factors (interdependence and differentiation among organizational subunits) in realizing ERP benefits. Thus, while this body of research provides valuable insights regarding those contexts most likely to benefit from ERP, it does not examine the endogenous drivers of ERP users' post-adoptive behaviors. Liang et al. (2007) examine post adoptive assimilation of enterprise systems from the perspective of institutional theory. While this work provides insights into post-adoptive ERP from the perspective of influences on assimilation that originate outside the organization, it again does not examine the internal interventions that drive of ERP users' post-adoptive behaviors.

The primary purpose of this paper is to further our collective understanding of how companies might facilitate usage of their installed ERP functionality in order to enhance the business value obtained from this organizational investment. It does so through a system dynamics view of post-adoptive ERP behavior and provides a theoretically grounded lens through which to examine the antecedents and outcomes of interventions taken to induce post-adoptive behaviors with a community of ERP users. While it is beyond the scope of this paper to conduct extensive (simulation-enabled) experiments to explore alternative management policies and their effects, establishing the stability of the theoretically-modeled web of relationships is a critical first step in most research programs that strive to gain rich insights into the behavior of complex systems (Burgess, et al. 1992; Repenning and Sterman 2002; Clark, et al. 2007).

The importance of understanding post-adoptive behaviors is made clear by examples of firms that experienced apparent ERP failures soon after installation but were then, after considerable effort, able to realize substantive improvements (Aakermans and van Heldon 2002; Hitt, Wu and Zhou 2002; Hoffman 2004; Swanton 2004a). Other firms achieved initial ERP success, but then displayed degraded outcomes over the longer term (Markus, et al. 2000a; Hitt, et al. 2002). Although several factors may contribute to these outcomes, one key driver is users' post-adoptive behavior as they grapple with the disruptions ERP brings to their environment (Beaudry and Pinsonneault 2005). Understanding these behaviors is crucial if firms are to realize continuing benefit streams from ERP implementations. We define post-adoptive behavior to include the extent to which users are making use of features in their ERP system, as well as the extent to which they are gaining understanding of both the software and work processes through training and experiential interventions.

The causal interrelationships among critical success factors that lead to initial ERP implementation success has been explored in a system dynamics framework (Akkermans and van Helden 2002), yet the findings have not been extended to post-adoptive behaviors. Therefore, our model represents a dynamic conceptualization in the post-adoptive context. We argue that rather than following a linear flow, system outcome (benefits) is a dynamic construct with complex feedback loops throughout the relationships among the elements that comprise it. If actual benefits are not as great as expected or desired, for example, then users will modify their behavior to achieve greater benefits. There is feedback from the end result to at least one element within the model and quite likely several factors that also interact with each other. Thus, we propose that examining the dynamic behavior over time provides a deeper and richer understanding of the relationships amongst the success variable and its antecedents.

The system archetype that provides a representation of the model that we discuss in this work is shown in Figure 1. It illustrates classic limits *to growth* structure that occurs in the efforts to balance actual achievements with a desired goal, hence the behavior is in the form of a balancing loop. Here, firms attempt to achieve desired ERP benefits through various interventions. The gap between the benefits they achieve and what they want to achieve then drives future interventions. This archetype provides a tangible lens through which to view post adoptive ERP behavior that managers often fail to articulate because they are caught up in the day-to-day struggle with accomplishing the required work and managing the processes the system was implemented to support (Wolstenholme 2004). In a best case scenario, the behavior over time would look something like that shown in Figure 2a. Benefits grow exponentially until they reach the desired state (goal).

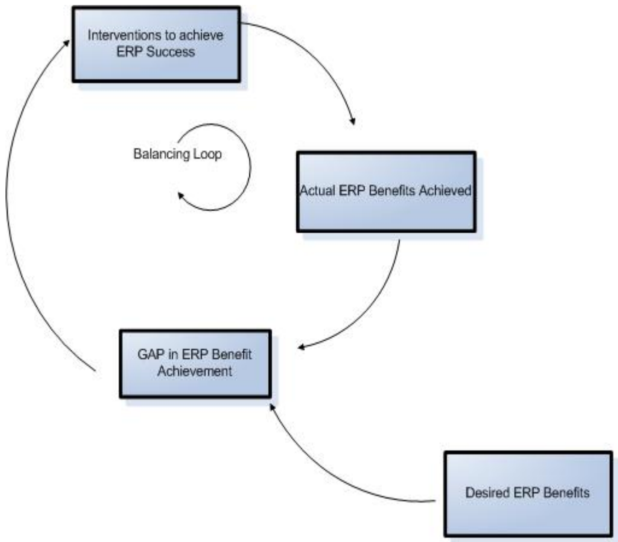

Figure 1: System Archetype of Healthy Post-Adoptive ERP Behavior

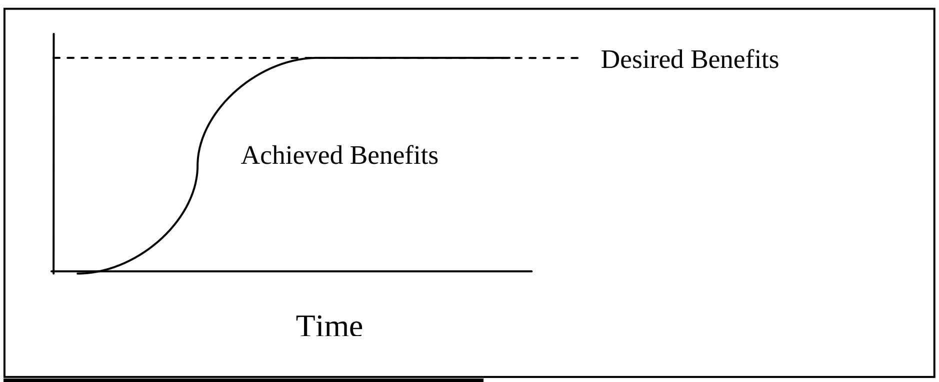

Figure 2a: Post-Adoptive ERP Behavior

However, research indicates that many companies do not reach their desired goals even several years after implementation (Jones et al 2008). In other words, the gap between desired and actual benefits is never sufficiently closed and behavior looks similar to that illustrated in Figure 2b.

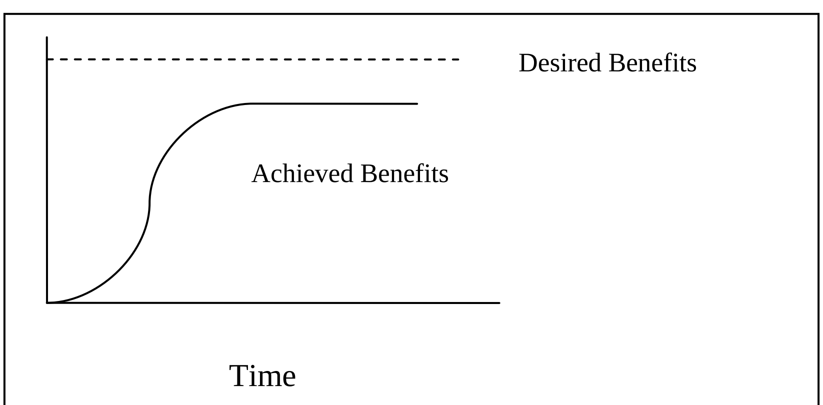

Figure 2b: Post-Adoptive ERP Behavior: Scenario 1

We propose a theoretically grounded conceptual model that addresses how to better close the gap for the behavior illustrated in Figure 2a. There are essentially two types of responses to the gap; one is to increase initiatives in order to close the gap and the other is to reduce the goal (Markus, et al. 2000b; Scott and Kaindl 2000). Our dynamic hypothesis is that attention to the interventions while maintaining desired goals will result in increased actual benefits. Rather than reducing the desired goals (unhealthy behavior that most likely leads to unintended negative consequences), we propose a model that forms a holistic basis for examining alternative managerial policies to narrow the gap between desired and achieved ERP benefits.

Although our focus is on ERP, findings may also be useful in the broader context of other large scale integrated information systems, such as integrated supply chain management systems, customer relationship management (CRM), and enterprise wide business performance management systems. The core of this conceptualization is a dynamic information feedback model that focuses on relationships involving: primary interventions (software training interventions, work process training interventions, experiential interventions); transitional outcomes (software systems understanding, work process understanding); intermediate outcomes (extent of features implementation, system usage); and system outcome (system benefits). A conceptual overview of this model is provided in Figure 3.

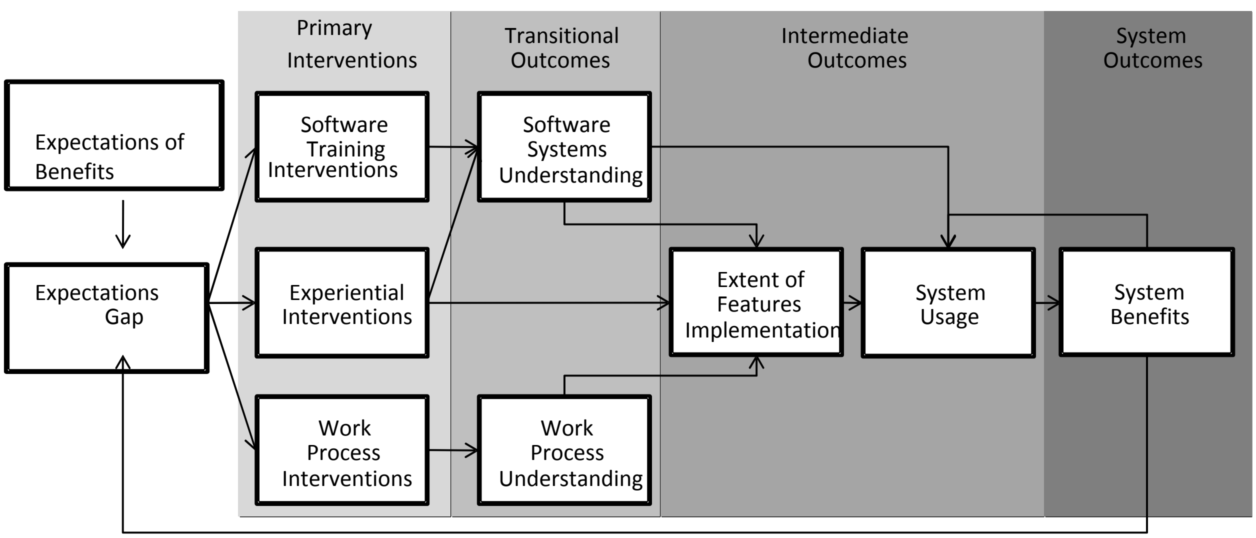

Figure 3: Conceptual Post-Adoptive Model

We use the structure inherent within this conceptual model to indicate the intermediate steps through which ERP benefits are engendered (Barua, et al. 1995; Gattiker and Goodhue 2005). To more clearly depict and describe the conceptual links within the model, we develop these relationships further via four sub-groupings. These are represented by the shaded areas in Figure 3 and are labeled primary interventions, transitional outcomes, intermediate outcomes, and system outcomes. These groupings conceptually represent the sets of elements through which post adoptive behavior progresses. Primary interventions precede transitional outcomes, which precede intermediate outcomes, which precede system outcomes. Then, system outcomes provide feedback to trigger further primary interventions.

Primary interventions are the starting point in our model. For example, software and work process training interventions lead to the better understanding of both ERP software and work processes (transitional outcomes). The term 'transitional outcome' is applied to software and work process understanding to indicate that, while greater understanding is necessary for ERP success, it is not sufficient. Enhanced understanding must be translated or transitioned into more directly observed and felt organizational outcomes. Specifically, we argue that enhanced understanding translates into a greater extent of both ERP features implementation and ERP system usage (intermediate outcomes). We further argue that it is these intermediate outcomes that then translate into system outcomes associated with the realization of system benefits. Because of the dynamic nature of our model, however, the realization of system benefits is not the end point of the conceptualization. Rather, we propose that when mangers perceive a gap between the benefits they expect and those they receive, they take steps to close the gap by revisiting the primary interventions.

The development of this model was based on a review of the literatures dealing with both ERP implementation and post-adoptive behavior. Elements in the model were also drawn from knowledge gained in a prior research project conducted by one of the authors where respondents represented both information technology and business perspectives (Jones and Price 2004; Jones 2005; Jones, et al. 2006) and validated via a ‘snapshot’ (Jones, et al 2008) from the current research program characterizing the stagnated states of successful ERP installations in six energy firms.

The modeling described herein involved first interrelating elements via a causal analysis, or influence mapping, approach that has been shown to be effective in describing complex systems in behavioral terms (Burgess and Clark 1990; Burgess, et al. 1992), and then translating the resultant causal map into a system dynamics structural model.

Face validity of the conceptual model is established by supporting proposed conceptual relationships with former research studies and with solicitation of input from people familiar with the system and its behavior. The latter is explicated in Jones et al. (2008), where 52 managers and 52 users of ERP in six organizations in the energy industry were asked extensive survey questions about the post-adoptive behavior of their own ERP systems. In addition, a focus group of ERP managers and users were asked interview questions about their ERP post adoptive behavior. Their responses lend credibility to the face validity of the conceptual model. In addition, the model is grounded in prior research so that the question “does the structure have support of prior research where possible?” is answered. Validity for a quantitative model (operational validity) is established when model simulations exhibit stability within the bounds of steady state behavior (Sterman 2000). A comprehensive development of the structure of the conceptual model, accompanied by substantiations of the model’s key elements and relationships, and suggestions for future research directions comprises the remainder of the article.

Post-Adoptive ERP System Structure

Our model includes both organizational-level and individual-level processes and cognitions. Both of these are included because individual behavior associated with installed technologies does not necessarily constitute *organizational* behavior or learning about the technologies (Robey, et al. 2000). Since an ERP initiative invariably involves business process standardization/integration across organizational units, it can prove dysfunctional for individuals to appropriate the technology in self-serving ways (Baskerville, et al. 2002; Robey, et al. 2002; Gosain 2004). Thus, embedding individual behaviors within a larger organizational context seems necessary in examining post-adoptive ERP use behaviors. The modeling accomplished here applies an organizational lens, with individual cognitions and behaviors implicitly aggregated across the organization. As technology usage unfolds in the post-adoptive context, the post-adoptive interventions, individuals' sense-making and action-taking, and the nature of the technology implementation evolve in a simultaneous fashion (Jasperson, et al. 2005). Each of these factors, in turn, can influence each other creating a complex, dynamic feedback structure.

We frame our work using an integrative systems approach that enables us to address the components of our system and their interactive behavior over time. The framework is based on the elements that have been identified in the literature as important to post adoptive behavior with regard to information systems in general and ERP specifically. This approach is employed to present the framework and to focus on the important relationships that interact through complex feedback mechanisms to produce behavior. The eight constructs that represent the basic structure of the system (Figure 3) form the basis for discussion of the structure of the conceptual model in the remainder of this section.

Primary Interventions

We posit three primary interventions available to influence the behavior of post-adoptive use. The first deals with software training in both the initial implementation phase and in subsequent periods, the second addresses training regarding work processes affected by the implemented ERP system, and the third addresses experiential interventions that users undertake on their own or with the encouragement/involvement of managers or co-workers.

SOFTWARE TRAINING INTERVENTIONS Prior to its implementation, many firms develop structured ERP training programs, which may include training a small set of users (called power users) who then train other users to roll out the training across the organization (e.g., Carte, et al. 2005). As training increases a person's ability to use and to feel comfortable using a software package, it indirectly increases his or her predilection toward acceptance of the software (Nelson and Cheney 1985; Compeau, et al. 1999). Poor or inadequate training of users often results in ERP implementation problems (Brown and Vessey 2003; Scott 2005). In fact, many firms rank ERP user training among their largest problems with their ERP (Duplaga and Astani 2003). Thus, training interventions may be key determinants of the long-term viability of ERP in a given organization (Markus, et al. 2000a; Yi and Davis 2003; Bajwa, et al. 2004). Unfortunately, training costs and tight implementation budgets can result in limited training (Jones 2001; Scott 2005).

The complexity and nature of installed software, however, can limit the amount of information that users are able to absorb prior to actual use (Yi and Davis 2003). Thus, there are gaps between what is expected to be encountered, based on pre-implementation training, and what is actually encountered in the use of an installed ERP system. After initial installation, for example, users may encounter features or be required to use features they were not exposed to during pre-implementation training. Consequently, users engage in efforts to make sense of the technology by comparing the outcomes of their post-implementation activities to their pre-implementation expectations (Griffith 1999; Orlikowski and Gash 1994). Users correspondingly engage in post-implementation sense-making (Bhattacherjee 2001; Jasperson et al. 2005), which ideally promotes learning about the installed ERP system. User adaptation to new systems can occur throughout the life of a system, and different users may adapt at different points (Beaudry and Pinsonneault 2005).

As users adapt and learn to use a system, they begin to perform certain tasks automatically because of their learning (Limayem, et al. 2007). This is referred to as habit, and as usage becomes habitual or routine, exposure to new features decreases, and user interest in exploration of the software typically wanes (Limayem and Hirt 2003; Nelson and Cheney 1985). Research indicates that once habit is formed, it may dwarf other determinants of usage in the absence of new disruptive interventions (Limayem, et al. 2007). Thus, as users apply software in repetitive ways, the size of perceived gaps between desired and perceived outcomes by users is likely to wane (Gersick 1991; Tyre and Orlikowski 1994).

The structure for software training shown in Figure 4 was developed on the basis of these ideas and concepts. In Figure 4, *Software Training Interventions* represents the on-going training users receive on how to use the ERP software. It is influenced by the extent of training received prior to implementation, software training costs, the gap between desired and actual system benefits.

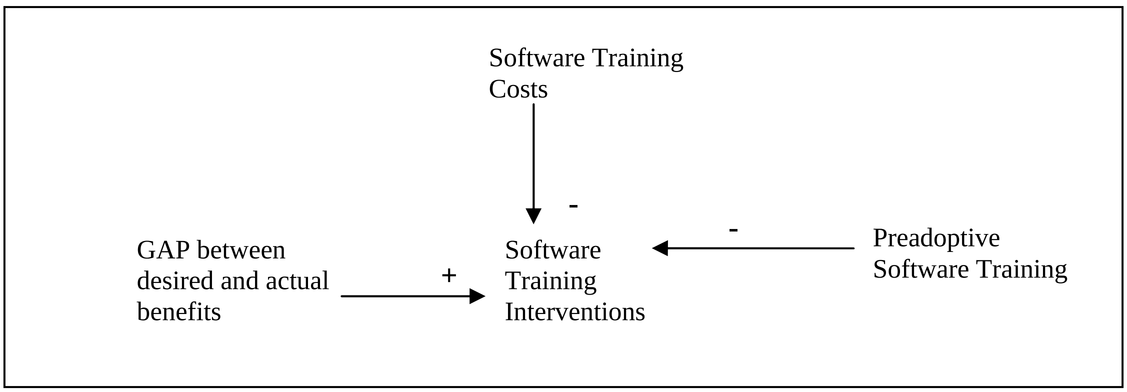

Figure 4: Software Training Construct

WORK PROCESS TRAINING INTERVENTIONS *Work Process Training Interventions* represent the training users receive in how their work processes have changed after ERP implementation and how these work processes are connected to other key work processes. As shown in Figure 5, the model structure for *Work Process Training Interventions* is similar to that just described regarding *Software Training Interventions*.

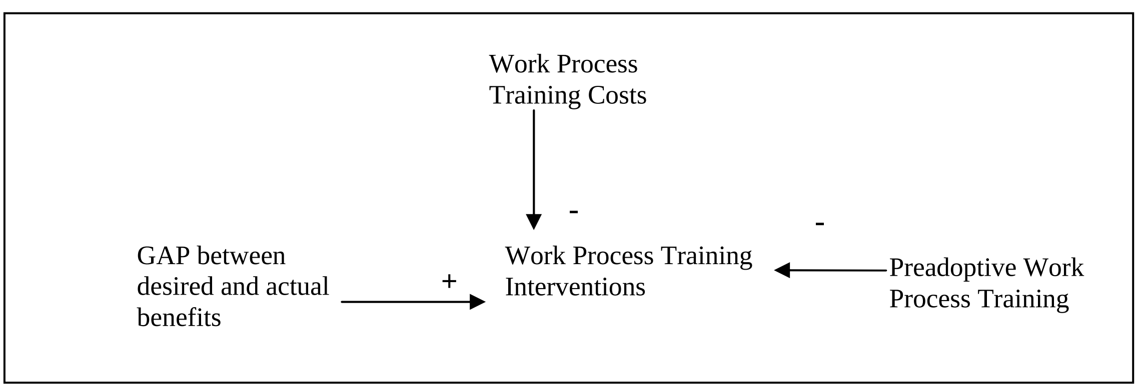

Figure 5: Work Process Training Construct

Because the work processes embedded within an ERP are typically based on industry best practices, most organizations significantly alter their work processes to fit the ERP software (Robey, et al. 2002). As a result, users must gain knowledge about the business rules and processes embedded in the ERP software (Lee and Lee 2000) and must understand the integrative nature of ERP in order to use it effectively (Hong and Kim 2002). In addition, conversion to an ERP work environment requires users to understand that they are no longer working in silos, and whatever they do now impacts someone else (Kawalek and Wood-Harper 2002; Welti 1999). Entire departments must be retrained with this in mind (Al-Mashari and Zairi 2000; Caldwell and Stein 1998; Roberts, et al. 2003).

Most pre-implementation training, particularly vendor supplied training, however, focuses on how to use the software rather than on how or why the work processes have changed (Jones 2001; Scott 2005). This is partly motivated by training costs, but is also a result of the implementing organization's failure to understand the magnitude of changes in the work processes required in the ERP environment (Jones 2001). Firms often indicate immediately after implementation that they wish they had understood the importance of training users on work process changes prior to implementation (Jones, et al. 2006). The gap between expectations regarding the nature of work processes and what is actually experienced is typically quite large for some time after an ERP implementation (Gattiker and Goodhue 2005; Ross 1999). Thus, it is desirable for users to engage in pre- and post-implementation work process training interventions in order to use the installed ERP functionality effectively in making decisions and managing processes (Bajwa, et al. 2004; Holsapple and Sena 2001).

Experiential Interventions

Experiential Interventions represent the proactive technology sensemaking efforts (use and experimentation) of individual users to learn about the new work environment (ERP software plus work processes). These interventions are often stimulated when users sense that more could be done with the system to increase its value to themselves and/or their organization (Jasperson, et al. 2005). Although ERP packages provide hundreds of standard reports, users often find that they need to develop customized or *ad hoc* reports for access to needed operational data or in response to manager requests for output or activities that necessitate use of additional ERP functionality (Baskerville, et al. 2000; Robey, et al. 2002). These needs prompt experiential interventions to discover ways to use the software to meet those needs. These experiential interventions then impact software and work process understanding as well as extent of features implementation.

Such experiential activity is likely to require an understanding of the ERP-enabled work environment beyond the initially understood functionality; all too often, however, insufficient training materials are available. Thus, users apply available documentation, their prior knowledge and knowledge gain from interacting with peers and support desk staff along with active self-experimentation to accumulate the understanding needed to perform more advanced tasks (Jasperson, et al. 2005). There are, of course, variable rates of uptake for both individuals and for work units as they experiment and extend the software functionality.

Figure 6 incorporates these issues and represents the rate at which experiential interventions occur to be a result of the current exposed functionality i.e. the number of ERP modules installed, of the system (*ERP Functionality)* and additional needs that surface as a result of the gap between desired and actual benefits.

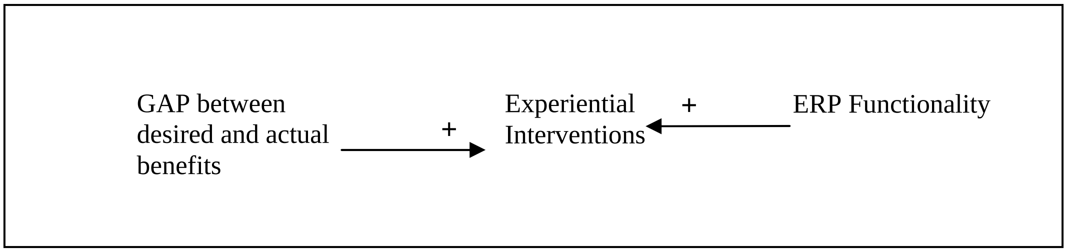

Figure 6: Experiential Interventions Construct

CUMULATIVE EFFECTS OF PRIMARY INTERVENTIONS These three types of primary interventions (software training, work process training and experiential interventions) are critical to system implementation success (Jasperson, et al. 2005). Little research, however, has focused on the interactive nature and cumulative effects of these factors in long-term ERP system success. In addition, once users feel comfortable with the software and their ERP-enabled work environment, they are more likely to explore the installed ERP functionality out of an innate desire to learn more about what it can do for them (Nelson and Cheney 1985). The structure that theoretically explains how they interact to create outcomes is introduced into our model, but much additional research in this area is needed.

Theory suggests that as use becomes institutionalized, interventions to stimulate new usage tapers off (Karahanna, et al. 1999); and, as knowledge decays over time, what was learned in training is lost if not exercised (Karuppan and Karuppan 2008; Yi and Davis 2003). In addition, theory suggests that the interventions put in place to facilitate a given set of activities may also, in time, come to inhibit other sense-making activities (Garud and Kumaraswamy 2005). While a technology is new, organizational members view it as separate from their routine, and they attempt to fit it in with their routines (Tyre and Orlikowski 1994). Simultaneously, and although it at first seems incongruous, user resistance to new technology also occurs (Barley 1986). Thus, it seems that an organization is subject to a paradox where users attempt to learn how to use the new technology while trying to preserve existing procedures.

The fact that almost no research has examined training interventions in terms of such self-reinforcing behaviors is likely because of the relatively short-term nature of most information systems training. Because it takes months, even years, for ERP to become stabilized in an organization (Markus and Tanis 2000; Nicolaou 2004b), it is important to examine how progress that has been made might inhibit on-going training interventions. Our model provides a way of conceptualizing such phenomena, because it incorporates feedback among its elements and because it allows for examination of relationships among these elements over time.

Transitional Outcomes

The primary interventions result in transitional outcomes that embody enhanced understandings of ERP software and work processes. These transitional outcomes are the bridges between the primary interventions and the intermediate outcomes associated with ERP adoption.

SOFTWARE SYSTEMS UNDERSTANDING The extent to which the software is understood *(Software Systems Understanding)* is partly determined by the software training provided (see Figure 7). Understanding is also affected by the *Extent of Prior Use* of ERP software and by how extensively *experiential interventions* have occurred. Over time, users' understanding will decay unless it is reinforced, and overall knowledge is lost from the organization as users who have acquired ERP related knowledge leave the organization (Ke and Wei 2008). Therefore, *Software Knowledge Decay* also is included as one of the antecedents of software systems understanding.

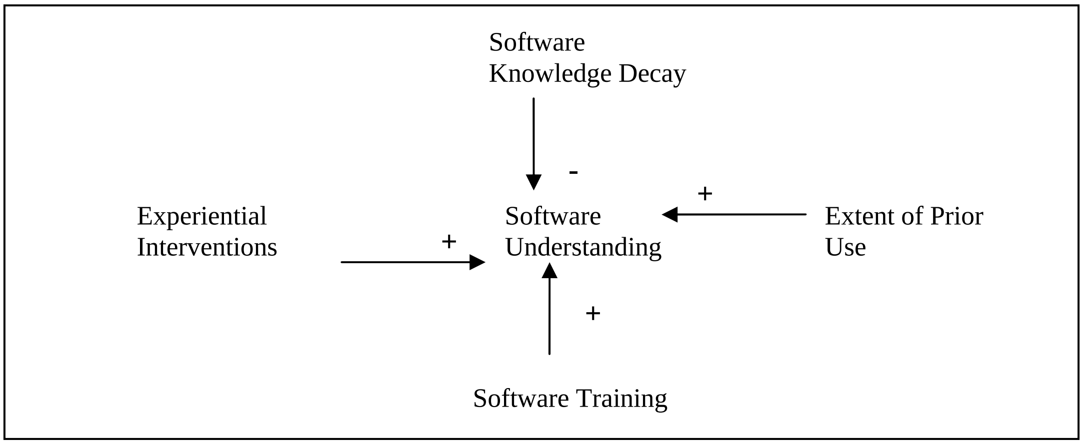

Figure 7: Software Systems Understanding Construct

Any given information technology has some room for flexibility in the way it is interpreted and adapted for organizational usage (Newell, et al. 2003). Although ERP use is mandatory in most organizations, the software has many more features than those typically mandated. Thus, there is always considerable flexibility in its application. For example, through experiential interventions users may discover new uses of existing features or new features to accomplish their tasks. Users may also use features in new or innovative ways (Griffith 1999), which can lead to unexpected consequences (Brown 1998; Robey and Boudreau 2000). All of these constitute experimentation and exploration that influences systems software understanding (Orlikowski 2000).

Prior experience with ERP also helps increase a user's understanding of the software he or she is currently employing. Users with prior ERP experience bring to the table what they have learned about a specific package or a general understanding of the ERP environment even when they have used a different package. As use becomes routinized, however, the likelihood of further software systems understanding decreases unless there are specific interventions to break the habitual use pattern (Karahanna, et al. 1999; Venkatesh, et al. 2003). Finally, as users settle into routine usage, they may forget about things they have learned that they do not apply on a regular basis (Yi and Davis 2003).

WORK PROCESS UNDERSTANDING The understanding of work processes depends heavily on training in how to perform these processes, experience with the work processes prior to ERP, and decay in work process knowledge. New technologies, such as ERP, interrupt established patterns of behavior and cause them to change (Barley 1986). The specific context in which technologies are implemented and in which they become embedded influence the outcomes of post adoptive technology adaptation (Barley 1986; Henfridsson and Söderholm 2000). Theory suggests that the more stable and institutionalized existing practices are, the more likely that new technology will conform to existing processes. At first glance, ERP does not fit this picture because it is implemented largely to help organizations disrupt ingrained practices and redesign existing processes to be more efficient (Lee and Lee 2000). Deeper analysis of post-adoptive ERP behavior, however, indicates that once new processes are established and become stabilized, they become yet another set of routinized, embedded practices (Newall, et al. 2003). When this happens, further and deeper change is inhibited, and experimentation, exploration, and the desire to learn tapers off (Newall, et al. 2003; Robey, et al. 2000). The window of opportunity that a firm has to effect change induced by the introduction of a new technology such as ERP may be short unless disruptive events, such as management-induced interventions to further alter processes, occur (Tyre and Orlikowski 1994).

Furthermore, users often do not apply all of what they have been exposed to in training, and the unapplied pieces of their work process understanding decay over time. As this happens, users' reliance on habitual or routinized practices is reinforced in their daily activities. The decay of work process understanding also reflects situations where users have applied the pieces of what they have learned only on the surface without a deeper understanding of how the pieces fit together (Barley and Tolbert 1997; Henfridsson and Söderholm 2000). People can appear to understand how an organizational process has changed without actually understanding how it functions. For example, some units retain their old master files or a separate parts tracking mechanism and use their ERP package primarily for data entry, thus defeating the purpose of using ERP to integrate processes across units (Jones and Price 2004). Training, experience, and decay in work process knowledge all influence work process understanding.

Therefore, *Work Processes Understanding* is influenced by the following elements: training in how to perform these processes (*Work Process Training Interventions*), experience with the work processes prior to ERP implementation (*Extent of Prior Work Process Experience*), and decay in work process knowledge (*Work Process Knowledge Decay*). These elements and relationships are shown in Figure 8.

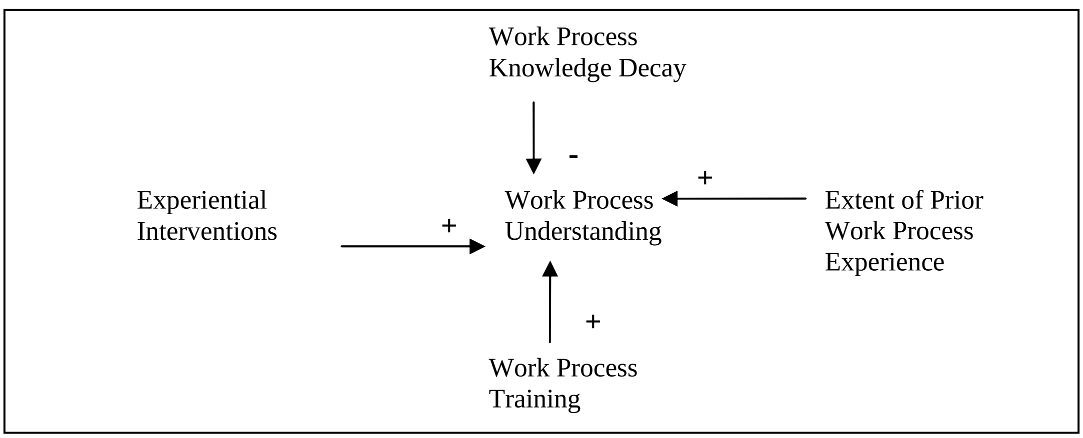

Figure 8: Work Process Understanding Construct

CUMULATIVE EFFECTS OF TRANSITIONAL OUTCOMES The role of software and work process understanding in post-adoptive ERP behavior is central to informing our knowledge about post-adoptive ERP use. The model posits that these both influence perceived usefulness. This issue is critical because theory indicates that perceived usefulness is influential in inducing individuals to use new technologies (Venkatesh, et al. 2003). Although ERP use normally is mandated, individual users have considerable leeway to selectively appropriate much of the functionality inherent in the software (Bradford and Florin 2003; Jasperson, et al 2005). The understanding gained from training and experience is essential for the software not only to be extended but also to achieve desired benefits.

It seems clear that software and work process understanding combine to drive the features implementation construct that, along with the amount of use, drives system benefits. What is not as clear, however, is the nature of the interactions among these two constructs. For example, Jones et al. (2008) inferred from their data that while some software understanding is needed to 'bootstrap' work process training; it is work process training that is more strongly associated with software understanding, work process understanding and the level of experiential intervention. Thus, it is critical to examine the interplay between software and work process understanding (as well as with their antecedents) in influencing perceived usefulness and ERP system use.

Intermediate Outcomes

The intermediate outcomes in the model are *Extent of Features Implementation* and *System Usage*. The result of the training interventions is enhanced understanding, which through perceived usefulness influences both the implementation of more features of the installed ERP system and system usage. In addition, experiential interventions directly impact extent of features implementation. Greater implementation of features and greater usage are both critical for the realization of system benefits.

EXTENT OF FEATURES IMPLEMENTATION Not all firms implement all modules of an ERP software package, and not all firms implement all of the functionality inherent in the modules that they do implement. This may be due to time and cost factors, difficulty with implementation processes, or perceived needs based on the nature of an organization's business processes (Jones 2001; Ranganathan and Brown 2006). However, research indicates that organizations having ERP implementations with greater functional scope have greater increases in operational performance (Karimi, et al. 2007) as well as in financial and market returns (Hitt, et al. 2002; Ranganathan and Brown 2006) than those with lesser functional scope.

Post-adoptive behavior is the result of a series of decisions to continue using a system, and exploration of the system may vary widely across time (Limayem, et al. 2007). Users may ask for, or the organization may decide to implement, additional functionality once the initial rollout is completed (Baskerville, et al. 2000; Robey, et al. 2002). Research indicates that the costs of drilling down deeper in the features of ERP are less of a barrier for firms that implement ERP broadly across the enterprise than for firms that implement in only one or two units or that implement only a few modules. This is because of an economy of scale effect of spreading the costs over a broader variety of activities (Gattiker and Goodhue 2002; Gattiker and Goodhue 2004). In addition, the costs of learning the software are spread over a broader variety of activities so that the former firms are likely to reap greater payback from extending features. Therefore, the cost of functionality should be considered in a model of the extent of features implementation.

Prior use of any specific information technology application also impacts the likelihood that its features will be more used (Bhattacherjee 1998; Igbaria et al. 1995; Venkatesh et al. 2002). Prior use increases not only specific skills, but also knowledge of the abstract principles on which the technology is based and the ability to apply those principles (Fichman and Kemerer 1997; 1999). However, repetitious prior use can lead to the routinization of technology. Theory indicates that as use becomes repetitive, users engage in less cognitive processing about whether to use the technology (Jasperson, et al. 2005; Ouellette and Wood 1998) and may inhibit feature extension. Thus, the manner in which prior use is modeled must account for both its facilitating and inhibiting influences.

Finally, perceived usefulness of the software impacts the extent to which its features are implemented (Venkatesh, et al. 2002). Perceived usefulness in our model is driven by the understanding of the software and work processes gained through training and experience. Even in situations where use is mandated, such as with ERP, users likely will appropriate, and further explore, the features they find most useful (Barki and Hartwick 1994). In extreme situations where users do not find the software useful, they are likely to apply workarounds that bypass the system. Many firms that have implemented ERP are currently dealing with the affects of this, which include inefficient use of resources and conflicting information about performance (Gosain 2004; Swanton 2004a).

In our model, *Extent of Features Implementation* is defined as the set of features a firm has exposed in its installed ERP system. *Extent of Features Implementation* is influenced by the *Extent of Prior Use* individuals have had with ERP, and the *Perceived Usefulness* of the ERP. The latter is a function of *Software Systems Understanding* and *Work Process Understanding. Extent of Features Implementation* is also determined by the *Extent of Experiential Interventions* and the *Cost of Features Implementation.* The structure is shown in Figure 9.

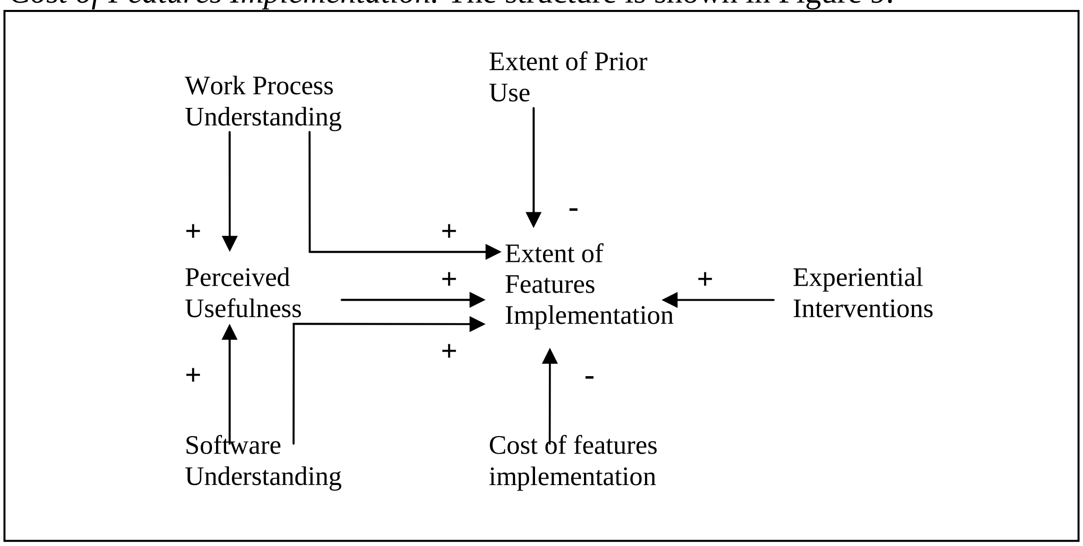

Figure 9: Extent of Features Implementation Construct

EXTENT OF SYSTEM USAGE It is estimated to take 12 to 18 months, even up to three years, after implementation for an organization to re-stabilize its processes and to stabilize its use of the ERP in order for it to begin to realize benefits from the installed ERP (Hitt, et al. 2002; Poston and Grabski 2001). It is not uncommon for a firm initially to require several days to perform processes in ERP that took only a few hours in the prior systems (Jones and Price 2004). Even after the initial stabilization period, it often takes several years for firms to make a full transition to ERP (Markus and Tanis 2000; Nicolaou 2004b; Ross 1999). One reason for this is the underlying cultural shift, in addition to the technical shift, that ERP often requires. For example, although ERP allows an organization to gain a more convergent view of its data and processes, this requires organizational members to understand a broader, more divergent set of activities within their own work processes (Robey, et al. 2002). Thus, we include a measure of time since implementation (*Software Stability)* in the model of system usage (Figure 10).

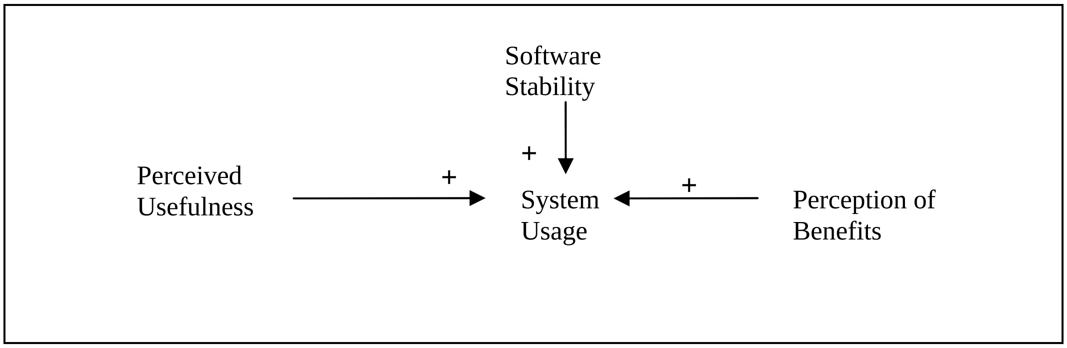

Figure 10: System Usage Construct

During periods of increasing stabilization, users develop strategies for using the system and perceptions about the system that are critical to continued usage. One of the perceptions already introduced is the *Perceived Usefulness* of the system, which has a strong influence on continued system usage (Venkatesh, et al. 2002). The underlying rationale for this is similar to that for the influence of usefulness on features implementation. Users form initial expectations about how useful a system will be to them, and as they use the system, these expectations are either confirmed or disconfirmed (Bhattacherjee 2001). Users who perceive the system as not useful will likely either stop using the system or find workarounds, even where initial use may be mandated.

The *Perceptions of Benefits* realized from ERP also impact the extent to which the system is used. Although the *System Benefits* construct is discussed more fully later, it is important to provide an initial discussion here to better describe the *System Usage* construct. Organizations often find that the benefits that they expected to achieve are not realistic in light of what they learn during implementation and the stabilization period (Markus, et al. 2000b; Scott and Kaindl 2000). Thus, managers adjust their perceptions of benefits, which in turn, may impact the way users' use behaviors. In addition, even organizations that believe that ERP can produce substantive benefits, yet have no mechanisms in place to first measure realized benefits and then motivate appropriate use behaviors based such measurement, often fail in their efforts (Al-Mashari and Zairi 2000; Ross 1999).

The reciprocal role of system benefits and system usage is inherent in the structure of the model. The model posits that system usage impacts system benefits, and simultaneously that managerial *perception of benefits* impacts *system usage*. The usage effects on benefits are shown in Figure 10. Theory and model structure indicate that as usage continues, organizations derive a deeper understanding of the benefits they can achieve (Ross 1999; Markus and Tanis 2000). If organizations discover during implementation and stabilization that the benefits they originally expected are not realistic, they tend to respond by adjusting benefit expectations downward (Markus, et al. 2000b; Scott and Kaindl 2000). If the original expectations were unrealistic, or if the expectations revisions are informed by learning more about ERP capabilities and organizational needs, then this may be appropriate. An alternate response is that managers seek to close the gap through seeking interventions that ultimately increase system benefits. Very little research has addressed the simultaneous and dynamic nature of system usage and benefits although it is a critical element in system behavior. The role of the reciprocal relationship between ERP system benefits and usage in post-adoptive ERP behavior, the process by which post-adoptive ERP usage clarifies or extends the organization's understanding of potential benefits, and the process by which organizations assess and adjust desired benefits in post-adoptive ERP environments all are key areas where further research is needed.

This literature and the theory surrounding system usage provide a sound basis for the structure shown in Figure 10. *System Usage* is defined as the extent to which users actually employ the features to which they have access. It is a function of how stable (*Software Stability*) the system is, *Perception of Benefits* of the system, and the *Perceived Usefulness* of the system. The literature supports the idea that it can take years to recognize the benefits of ERP and that usage varies greatly during the period. Both of these phenomena are inherent in the structure and parameters are assigned to the controlling variables.

System Outcome

The ultimate system outcome from the primary interventions, transitional outcomes and intermediate outcomes are the *System Benefits* realized by the organization. Such system benefits might, for example, be represented in the two dimensions of operational efficiency and strategic effectiveness. Operational efficiency relates to factors such as cost reduction, increased inventory turns, or productivity improvement (Shang and Seddon 2002). Strategic effectiveness refers to factors such as improved managerial decision making, improved business innovations, and the building of key external linkages (Shang and Seddon 2002). System benefits are presented as a function of the *Extent of Features Implementation*, which is an accumulation of the extent of features implemented and of *System Usage* (Figure 11*)*. The benefits resulting from an ERP implementation determine the *Gap between actual and desired benefits*, i.e., the difference between perceived and desired outcomes that ultimately drives the technology sense-making interventions discussed in prior sections. These elements create the model's major feedback loop because the gap leads back to the software training, work process training, and experiential interventions.

A basic tenet of the modeling of system benefits is that greater benefits flow from greater system use, all things equal. However, significant improvements in firm performance metrics are generally not realized until some time after an ERP installation (Nicolaou 2004a; Poston and Grabski 2001) and have been observed to decay with time (Hitt et al. 2002). Thus, while usage is tied to benefits, the link is likely much more complex than these findings imply.

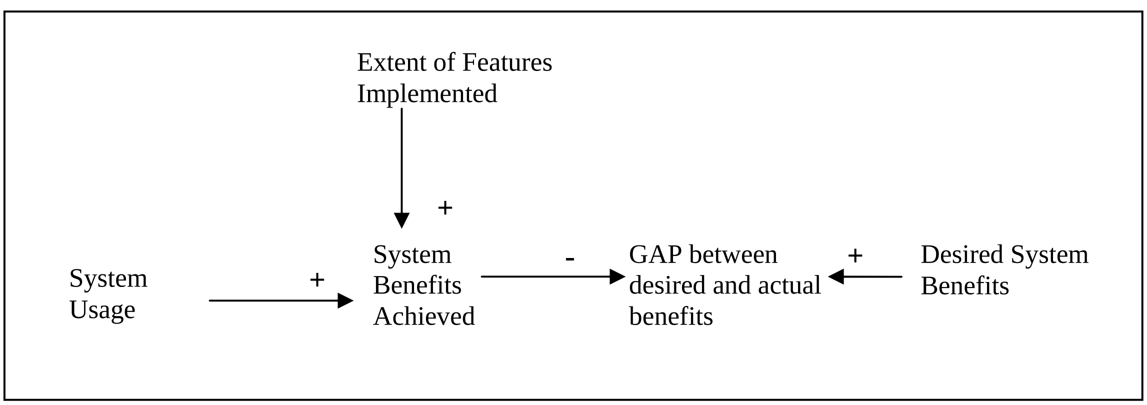

Figure 11: System Benefits Construct

Desired benefits are a key driver of the organizational adoption of an innovation (Rogers 1995). An organization's perceptions of the benefits that it can or should achieve with ERP often change as it learns more about itself and about the ERP-enabled work environment (Fox-Wolfgramm, et al. 1998; Scott and Kaindl 2000). It is tempting to assume that once attained, benefits continue at the initial rate at which they were attained. A constant rate of benefits is not realistic, however, because they are impacted by a variety of interventions. Theory indicates that when the gap between expected benefits and perception of actual benefits is large enough, action is taken to narrow it (Bhattacherjee 2001; Griffith 1999). This is also intuitively appealing. It is unclear, however, what 'large enough' means and how it is determined. Furthermore, users and managers may perceive the gap from very different frames of reference. Managers are likely to assess the gap in terms of whether desired organizational benefits are achieved (Shang and Seddon 2002), whereas users may assess the gap in terms of usage expectations formed in training and their early usage experiences (Bajwa, et al. 2004; Bhattacherjee 2001; Yi and Davis 2003). Therefore, the gap is quite a complex and critical part of the model of post-adoptive ERP behavior.

Extent of Features Implemented is represented as an antecedent of system benefits. At a basic level, the more ERP features that are implemented, explored, and used, the greater most organizations expect benefits to be (Markus, et al. 2000b). Much of the research about ERP benefits focuses primarily on the strategic benefits at the top level of the organization (e.g., Hitt, et al. 2002; Hunton, et al. 2003). These studies, however, overlook benefits at a more detailed level that can be identified through examining ERP at the features level (Shang and Seddon 2002). Research indicates that the perceived value of ERP investments are significantly better explained at the features level than at the overall enterprise level (Gattiker and Goodhue 2004; Gefen and Ragowsky 2005). For example, the financial module of an ERP may be implemented and used quite differently from the materials management module. The nature of benefits realized for each function may vary, and the processes impacted by various features are not likely to have equal strategic importance to the organization (Gattiker and Goodhue 2002; Shang and Seddon 2002). Finally, even though vendors often tout ERP packages as a one-size-fits-all solution, there is evidence that not all organizations benefit from broad-based features implementations (Markus, et al. 2000b). Deep usage of one or two modules may impact performance in some firms more extensively than shallow usage of several modules. Thus system benefits are driven by the *extent of features implemented* as well as by the extent to which the features that comprise the system are used (*System Usage*).

Assessing the Stability of the Quantitative Model

The material discussed in the preceding sections dealt with the conceptual structure of the dynamic model explaining ERP implementation post-adoptive behaviors. The causal map of these behaviors is presented in its entirety in Appendix A. However, while causal maps such as the ones developed here are very useful for providing conceptual insight, they can not be translated directly into the underlying quantitative model required for simulation in a system dynamics approach (Sterman 2000). It has been suggested that there are several phases of modeling that are inherent to the system dynamics approach, including conceptual, quantitative and comparative modeling (Sterman, 2000). We are concerned in this paper with the conceptual phase and following the guidance of Lyneis (1999) guidance, we have attempted to clearly define the problem of interest (in this case, failure to attain desired benefits in the post adoption stage of ERP) and identify possible causes of the problem. In the next phase, we develop a small, insight-based model that is essentially a prototype of the system at a highly aggregate level (Lyneis, 1999). This allows us to develop the basis for an operationally valid model, i.e. it is roughly right quantitatively to pursue the next level. The comparative phase is where the model is fully fleshed out in detail so that various alternative policies can be examined through simulation (Lyneis, 1999; Sterman, 2000) is beyond the scope of this paper. In this section, how our conceptual model was translated into a prototype quantitative model is briefly discussed. In addition, initial results from running the model to establish base-line stability are also discussed.

Using the iThink programming environment for dynamic models, the complete conceptual structure of the post adoptive organizational system is provided in Appendix B, Figure B1, as a stock and flow diagram. The stocks represent the eight key constructs in our model (software and work process training, experiential interventions, software and work process understanding, extent of features implementation, system usage, and system benefits). The model has been calibrated to produce an initial set of characteristic behaviors in the system. This means that the various input variables are set to their neutral values and the variables behave in a very narrow range. When it is operated over a five-year (260 weeks) time period, the output behavior produced shows virtually no increases or decreases. It is critical to establish the neutral, or baseline, model where little variation occurs in order to understand the extent to which driving forces are acting when we begin experimentation with varying levels of inputs.

As expected given the model structure, there is some initial small decay in the output of several variables because of decay functions in certain model elements in the absence of further training or when individuals do not extend their use beyond that which is initially set. An example of this behavior is shown in Figures 12a and 12b where several constructs and the system output construct (system benefits) are shown. Note that in the stock and flow diagram, system benefits are represented as efficiency benefits and effectiveness benefits. This helps to capture the complexity of benefits addressed earlier. In Figure 12a, the top variable is *Work Process Understanding,* which shows only very slight decline. The other two variables – *Software Systems Understanding* and *Extent of Features Implementation* – plot on the same line (under these initial neutral conditions) and show only very slight decline. The value of 100 (shown on the left axis of the graph) is the point at which the index value of the variables would be expected given neutral inputs for the primary intervention constructs. This behavior is produced when only initial software and work process training are provided and there are no experimental interventions. These primary interactions are translated through the transitional outcomes and intermediate outcomes to the system outcomes shown in the graph of Figure 12b. The model is calibrated to produce these base behaviors so that various policies regarding training or other interventions can be tested. Recall that testing policies involves changing the initial rules about the relationships. The two parts of *System Benefits, i.e., Efficiency* and *Effectiveness,* are equal under the initial neutral conditions and plot together. The very slight increase shown is created by a very slight increase in the *Extent of Features Implementation* resulting from the slight increase in the *Extent of Prior Use* some individuals have been presumed to posses. The increase is negligible on the overall behavior of the model.

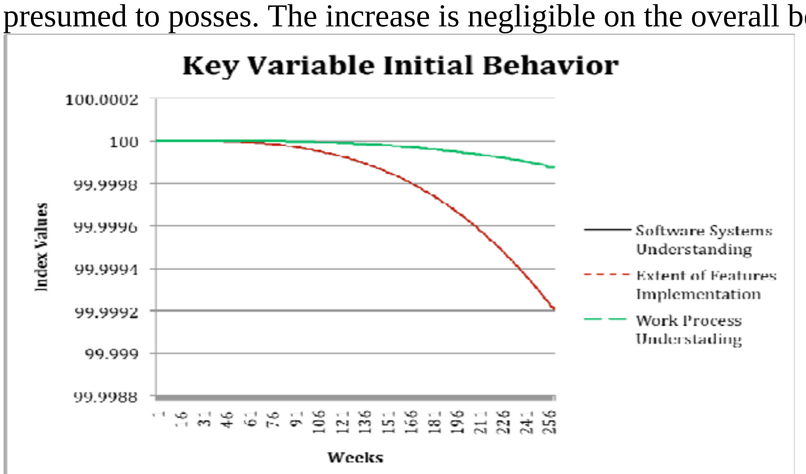

Figure 12a. Initial Base Input Model Behavior

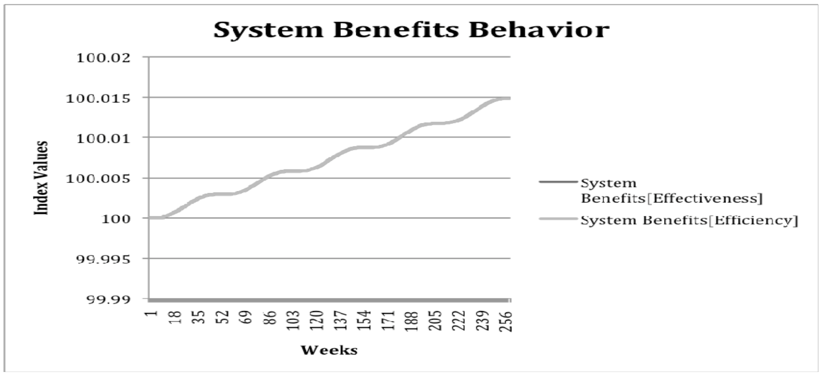

Figure 12b. System Benefits Initial Behavior

If the training provided initially is only slightly increased over the first year of implementation, system benefits begin to increase through the period. This behavior is shown in Figure 13 with the slight increase in the constructs *Software System Understanding* and *Extent of Features Implementation*. Various levels of initial and follow on training could be tested to determine the resulting effects on system behavior and the most productive and cost effective alternatives chosen. This also is true for the other primary intervention, experiential interventions. The number and types of experimentation are virtually endless and remain issues for further research employing the model.

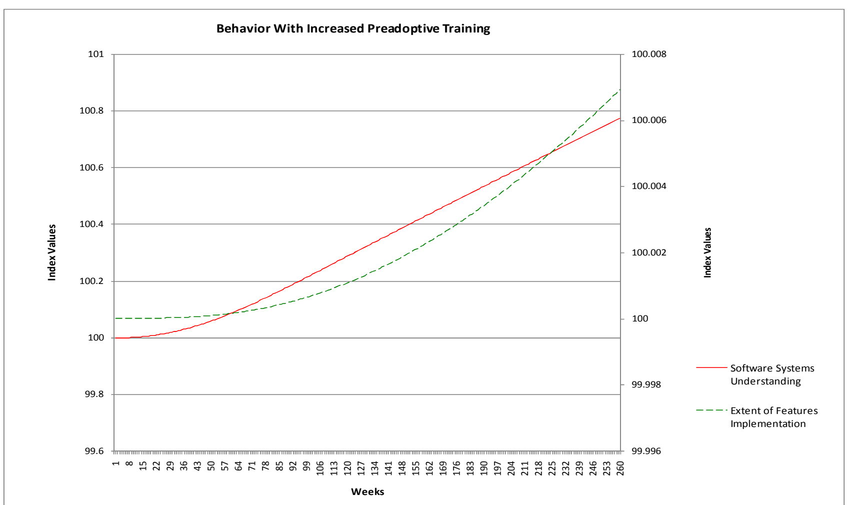

Figure 13. System Behavior Under Revised Training Policies

Summary of Model Structure

We have developed an integrated theory and structure for post-adoptive use behaviors associated with ERP implementations. The conceptual structure developed and presented as a dynamic information feedback model of a post-adoptive system structure is the main contribution of this work. This theoretical structure and the demonstration of its face and quantitative validity provides an extended, well-supported conceptual basis for further research about how companies might facilitate ERP usage in order to realize business value from their ERP investments. Thus, it provides a rich lens through which to study systems behavior in relationship to various post-adoptive factors in organizations that have implemented ERP. Using the structure of relationships among these pieces, we next provide a discussion of how this lens might be used to guide future research about ERP and how this lens might influence managerial decisions about post adoptive ERP.

Contributions and Conclusions

Our objective in developing and describing a system dynamics perspective of ERP post-adoptive behaviors is to provide a robust framing of the relationships that ultimately determine benefits realization from ERP implementations (along with other investments in IT-enabled business platforms). By explicitly surfacing and characterizing the complex feedback structures that underlie post-adoptive behaviors, it is hoped that both scholars who are interested in examining the organizational outcomes associated with enterprise system investments and managers who champion and are responsible for such investment initiatives will gain fresh insight into this increasingly important (and often exasperating) phenomenon.

Discussions pointing to numerous opportunities for furthering our collective understanding of post-adoptive behaviors regarding ERP implementations were present throughout the articulation of the model's subsections. Rather than repeat these discussions, we prefer to highlight two areas within the model for which a heightened research emphasis would be particularly valuable: relationships between the benefits gap and the stimulation of primary (training and experiential) interventions; and, relationships associated with the extent of features implementation. While both spheres of research are largely unexplored, each seems crucial in determining how an implementation effort plays out in practice.

The relationship between the training interventions and the gap between expected and actual benefits appears at first to be tautological and simplistic. However, closer inspection reveals that this relationship is quite intricate, and failure to understand and manage underlying influences properly could result in costly mistakes. Delays naturally intercede throughout the influence and feedback structures associated with the benefit gap's stimulation of software and work process training interventions. Gaps are not immediately detected after training occurs, and training is not immediately stimulated once a gap is recognized. If performance (perceived actual benefits) is decreasing, the gap could widen substantially from when it is recognized by a manager and when he/she takes action to increase one or both types of training. Because of the inherent delays associated with realized training-induced outcomes, the manager would most likely detect an even wider gap at the conclusion of the training intervention. This would likely lead to the erroneous interpretation by the manager that the training that had occurred was ineffective. This is a common, and often costly or disastrous, assumption with complex environments (Senge 1990; Sterman 2000). As a result, investments in training are likely to be increased beyond what is actually needed to close the gap. Alternatively, and worse, further training may be abandoned and other actions are taken to reduce the gap, such as lowering benefit expectations. Some companies have already pursued this in an effort to bring actual benefits closer to expected benefits (Markus, et al. 2000b; Scott and Kaindl 2000).

Research that applies our dynamic systems model as a theoretical lens could examine the relative merits of alternative managerial policies for closing the benefits gap. Questions that could be examined include: how much of what type of training is most beneficial, and how long does it take before training impacts are realized. One of the most powerful features of system dynamics models is that they allow such questions to be explored in the context of the complexities and feedback structures of the system within which the gap occurs.

The second issue we wish to highlight surrounds the relationship between extent of features implementation and actual benefits. On the surface, this relationship may be interpreted as "the greater the extent of features implemented, the greater the actual benefits." While individuals often find themselves intuitively thinking along such lines (Senge 1990; Stedman 2000), the relationship is far more involved because it is neither linear nor simplistic. Rather, it is subject to feedback and reinforcement. Benefits do not increase to infinity because there are indirect forces acting on the system through complex feedback structures that prevent this; hence, the relationship serves as a balancing loop. Research that surfaces the boundary conditions that stabilize this relationship would be particularly useful, as would research that characterizes the nature and effects of such boundary conditions.

In addition, consider the impact of perceived usefulness on the extent of features implementation. Again, linear thinking would suggest that the more useful a user perceives the system to be, the more features he/she will wish to see exposed in the installed ERP system explore. Indeed, research strongly suggests this is so (Venkatesh, et al. 2003). However, research also indicates that as users become comfortable with software, use becomes routinized, and exploration tapers off (Nelson and Cheney 1985; Tyre and Orlikowski 1994). This suggests that once users find a set of features to be particularly useful, their search for additional features may wane over time, thereby reflecting another balancing loop. Further, an ERP implementation would likely never implement every possible (or even every desirable) feature available in the software, thus reflecting a natural control on infinite installed-features expansion. Again, research examining the boundary conditions – both natural and imposed – that constrain this relationship would be useful in understanding how to best manage the process of exposing ERP systems features so as to align users' training, users' accumulating experience with already-exposed features, and managers' benefit expectations.

These examples illustrating the complex feedback structures inherent in the evolution of post-adoptive perceptions and behaviors reinforces the recognition that there are no single or simple solutions to be followed in order to better appropriate benefit streams from ERP implementations. The systems dynamic model described herein provides a rich and potentially insightful lens through which a variety of theoretically grounded and empirically validated factors can be systematically studied. Given the complexity and dynamism of the post-adoptive context, subsequent research needs to be multifaceted and reflective of the complex feedback structures that have been exposed. While linear models will continue to provide valuable understanding of ERP post-adoptive use behaviors, research grounded in a dynamic framework will likely be required in order to provide the richer insights into the complexities of these relationships that will accelerate enhancements in our collective understanding of how individuals and organizations evolve in their cognitions regarding and use of complex IT-enabled work systems.

References

Akkermans, H. and van Helden, K. 2002. Vicious and Virtuous Cycles in ERP Implementation: A Case Study of Interrelations between Critical Success Factors. *European Journal of Information Systems* **11**: 35-46.

Al-Mashari, M. and Zairi, M. 2000. The Effective Application of SAP R/3: A Proposed Model of Best Practice. *Logistics Information Management* **13**(3): 156-166.

Bajwa, D., Garcia, J.E., and Mooney, T. 2004. An Integrative Framework for the Assimilation of Enterprise Resource Planning Systems: Phases, Antecedents, and Outcomes. *Journal of Computer Information Systems* Spring: 81-90.

Barua, A., Kriebel, C.H., and Mukhopadhyay, T. 1995. Information Technologies and Business Value: An Analytic and Empirical Investigation. *Information Systems Research* **6(**1): 3-23.

Barker, T. and Frolick, M.N. 2003. ERP Implementation Failure: A Case Study. *Information Systems Management*, Fall: 43-49.

Barki, H. and Hartwick, J 1994. Measuring User Participation, User Involvement, and User Attitude. *MIS Quarterly*, **18**(1): 59-87.

Barley, S.R. 1986. Technology as an Occasion for Structuring: Evidence from Observation of CT Scanners and the Social Order of Radiology Departments. *Administrative Science Quarterly* **31(**1): 78-108.

Barley, S.R. and Tolbert, P.S. 1997. Institutionalization and Structuration: Studying the Links Between Action and Institution. *Organization Studies* **18(**1): 93-117.

Baskerville, R., Pawlowski, S., and McLean, E. 2000. Enterprise Resource Planning and Organizational Knowledge: Patterns Of Convergence And Divergence. *Proceedings of the 21st ICIS Conference*, Brisbane, Australia, 396-406.

Beaudry, A. and Pinsonneault, A. 2005. Understanding User Responses to Information Technology: A Coping Model of User Adaptation. *MIS Quarterly* **29(**3): 493-524.

Bhattacherjee, A. 1998. Managerial Influences on Intraorganizational Information Technology Use: A Principal-Agent Model. *Decision Sciences* **29(**1): 139-162.

Bhattacherjee, A. 2001. Understanding Information Systems Continuance: An Expectation-Confirmation Model. *MIS Quarterly* **25(**3): September: 351-370.

Bradford, M. and Florin, J. 2003. Examining the Role of Innovation Diffusion Factors on the Implementation Success of Enterprise Resource Planning Systems. *International Journal of Accounting Information Systems* **4:** 205-225.

Brown, J.S. 1998. Internet Technology in Support of the Concept of 'Communities-of-Practice': The Case of Xerox. *Accounting, Management, and Information Technologies* **8**: 227-236.

Brown, J.S. and Vessey, I. 2003. Leveraging ERP Learnings for the Next Wave of Enterprise System Projects. *MIS Quarterly Executive,* **2**(1): 65-77.

Burgess, G.M. and Clark, T.D., Jr. 1990. The defense acquisition system: A meta-organizational analysis. *Systems Research* **7**(3): 169-191.

Burgess, G.M., Clark, T.D., Jr., Hauser, R.D., Jr., and Zmud, R.W. 1992. The application of causal maps to develop a collective understanding of complex organizational contexts in requirements analysis. *Accounting, Management, and Information Technologies* **3**(2): 143-164.

Caldwell, B. and Stein, T. 1998. Beyond ERP - New IT agenda - A second wave of ERP activity promises to increase efficiency and transform ways of doing business. *InformationWeek* November, **30**: 34-35.

Carte, T.A., Schwarzkopf, A.B., Shaft, T.M., and Zmud, R.W. 2005. Advanced Business Intelligence at Cardinal Health. *MISQ Executive* **4**(4): 413-424.

Clark, T.D., Jones, M.C., and Armstrong, C.A. 2007. The dynamic structure of management support systems: Theory development, research, focus, and direction. *MIS Quarterly* **30**(3): 579-615.

Compeau, D. R., Higgins, C. A., and Huff, S. L. 1999. Social Cognitive Theory and Individual Reactions to Computing Technology: A Longitudinal Study. *MIS Quarterly* **23(**2): 145-158.

Duplaga, E.A. and Astani, M. 2003. Implementing ERP in Manufacturing. *Information Systems Management* Summer: 68-75.

Fichman, R.G. and Kemerer, C.F. 1997. The Assimilation of Software Process Innovations: An Organizational Learning Perspective. *Management Science* **43**(10): 1345-1363.

Fichman, R.G. and Kemerer, C.F. 1999. The Illusory Diffusion of Innovation: An Examination of Assimilation Gaps. *Information Systems Research*, **10**(3): 255-275.

Fox-Wolfgramm, S.J., Boal, K.B., and Hunt, J.G. 1998. Organizational Adaptation to Institutional Change: A Comparative Study of First-order Change in Prospector and Defender Banks. *Administrative Science Quarterly* **43**: 87-126.

Garud, R. and Kumaraswamy, A. 2005. Vicious and Virtuous Circles in the Management of Knowledge: The Case of Infosys Technologies.*MIS Quarterly* **29**(1): 9-33.

Gattiker, T.F. and Goodhue, D.L. 2002. Software-driven Changes to Business Processes: An Empirical Study of Impacts of Enterprise Resource Planning (ERP) Systems at the Local Level. *International Journal of Production Research*, **40**(18): 4799-4814.

Gattiker, T.F. and Goodhue, D.L. 2004. Understanding the Local-level Costs and Benefits of ERP Through Organizational Information Processing Theory. *Information and Management*, **41**(4): 431-443.

Gattiker, T.F. and Goodhue, D.L. 2005. What Happens after ERP Implementation: Understanding the Impact of Interdependence and Differentiation on Plant-level Outcomes. *MIS Quarterly*, **29**(3): 559-585.

Gefen, D. and Ragowsky, A. 2005. A Multi-level Approach to Measuring the Benefits of an ERP System in Manufacturing Firms. *Information Systems Journal*, Winter, 18-25.

Gersick, C. J. G. 1991. Revolutionary Change Theories: A Multilevel Exploration of the Punctuated Equilibrium Paradigm. *Academy of Management Review*, **16**(1): 10-36.

Gosain, S. 2004. Enterprise Information Systems as Objects and Carriers of Institutional Forces: The New Iron Cage?" *Journal of the Association for Information Systems*, **5**(4): 151-182.

Griffith, T. L. 1999. Technology Features as Triggers for Sensemaking. *Academy of Management Review*, **24**(3): 472-488.

Henfridsson, O. and Söderholm, A. 2000. Barriers to Learning: On Organizational Defenses and Vicious Circles in Technological Adaptation. *Accounting, Management, and Information Technologies*, **10**: 33-51.

Hitt, L. M., Wu, D.J., and Zhou, X. 2002. Investment in Enterprise Resource Planning: Business Impact and Productivity Measures. *Journal of Management Information Systems*, **19**(1): 71-98.

Hoffman, T. 2004. Saturation Stalls Enterprise Software Market. *ComputerWorld*, July 22, accessed on the WWW October 29, 2008 at http://www.computerworld.com/softwaretopics/crm/story/0,10801,94672,00.html

Holsapple, C.W. and Sena, M.P. 2001. Beyond transactions: The decision support benefits of ERP systems. *Journal of Decision Systems,* **10**(1): 65-85.

Hong, K.K. and Kim, Y.G. 2002. The Critical Success Factors for ERP Implementation: An Organizational Fit Perspective. *Information and Management,* 40: 25-40.

Hunton, J.E., Lippincott, B., and Reck, J.L. 2003. Enterprise Resource Planning Systems: Comparing Firm Performance of Adopters and Nonadopters. *International Journal of Accounting Information Systems,* **4**: 165-184.

Igbaria, M., Guimaraes, T., and Davis, G. B. 1995. Testing the Determinants of Microcomputer Usage via a Structural Equation Model. *Journal of Management Information Systems,* **11**(4): 87-114.

Jacobson, S, Shepherd, J., D'Aquilla, M. and Carter, K. 2008. The ERP Market Sizing Report. 2007-2012. *AMR Research,* June 30, accessed on the WWW October 29, 2008 at http://www.amrresearch.com/content/view.asp?pmillid=21595.

Jasperson, J., Carter, P.F., and Zmud, R.W. 2005. A Comprehensive Conceptualization Of The Post-Adoptive Behaviors Associated With It-Enabled Work Systems. *MIS Quarterly* **29**(3): 525-558.

Jones, M.C. 2001. The Role of Organizational Knowledge Sharing in ERP Implementation. *Final Report to the National Science Foundation Grant SES 0001998*.

Jones, M.C. 2005. Tacit Knowledge Sharing in ERP Implementation: A Multi-Site Case Study. *Information Resource Management Journal* **18**(2): 1-23.

Jones, M.C., Cline, M., and Ryan, S. 2006. Exploring Knowledge Sharing in ERP Implementation: An Organizational Culture Framework. *Decision Support Systems* **41**(2): 411-434.

Jones, M.C. and Price, R.L. 2004. Organizational knowledge sharing in ERP implementation: Lessons for industry. *Journal of End User Computing,* 1**6**(1): 21-40.

Jones, M.C., Zmud, R.W., and Clark, T.D., Jr. 2008. ERP in Practice: A Snapshot of Post-Installation Perceptions and Behaviors. Forthcoming Communications of AIS.

Kanaracus, C. 2008a. Waste Management Sues SAP Over ERP Implementation. *ComputerWorld,* March 27, accessed on the WWW October 29, 2008 at http://www.computerworld.com/action/article_do?command=viewarticlebasic&taxonomyName=erp/supply_chain&articleID=9072478&taxonomyID=121&intsrc=KC_top

Kanaracus, C. 2008b. Update: Overstock.com Restates Earnings, Cites ERP Implementation. *ComputerWorld,* October 27, accessed on the WWW October 29, 2008 at http://www.computerworld.com/action/article_do?command=viewarticlebasicc&articleID=00911205&intsrc=hm_list

Karahanna, E., Straub, D. W., and Chervany, N. L. 1999. Information Technology Adoption across Time: A Cross-Sectional Comparison of Pre-Adoption and Post-Adoption Beliefs. *MIS Quarterly*, **23**(2): 183-213.

Karimi, J., Somers, T.M. and Bhattacherjee, A. 2007. The Impact of ERP Implementation on Business Process Outcomes: A Factor-Based Study. *Journal of Management Information Systems* **24**(1): 101-134.

Karuppan, C.M. and Karuppan, M. 2008. Resilience of Super Users' Mental Models of Enterprise-wide Systems. *European Journal of Information Systems* **17**: 29-46.

Kawalek, P. and Wood-Harper, T. 2002. The finding of thorns: User participation in enterprise systems implementation. *The DATA BASE for Advances in Information Systems* **33**(1): 13-22.

Ke, W. and Wei, K.K. 2008. Organizational Culture and Leadership in ERP Implementation. *Decision Support Systems* 4**5**: 208-218.

Lee, Z. and Lee, J. 2000. An ERP Implementation Case Study from a Knowledge Transfer Perspective. *Journal of Information Technology* **15**: 281-288.

Liang, H., Saraf, N., Hu, Q., and Xue, Y. 2007. Assimilation of Enterprise Systems: The Effects of Institutional Pressures and the Mediating Role of Top Management. *MIS Quarterly* **31**(1): 59-87.

Limayem, M., Hirt, S.G., and Cheung, C.M.K. 2007. How Habit Limits the Predictive Power of Intention: The Case of Information Systems Continuance. *MIS Quarterly,* **31**(4): 705-737.

Limayem, M. and Hirt, S.G. 2003. Force of Habit and Information Systems Usage: Theory and Initial Validation. *Journal of the Association for Information Systems,* **4**: 65-97.

Luftman, J. and Kempaiah, R. 2007. Key Issues for IT Executives 2007. *MIS Quarterly Executive,* **7**(2): 99-112.

Lyneis, J.M. 1999. System Dynamics for Business Strategy: A Phased Approach. *System Dynamics Review* **15**(1), pp 37-70.

Mabert, V.A. Soni, A., and Venkataramanan, M.A. 2001. Enterprise Resources Planning: Common Myths versus Evolving Reality. *Business Horizons* **44**(3): 69-76.

Markus, L.M., Axline, S., Petrie, D., and Tanis, C. 2000a. Learning From Adopters' Experiences with ERP: Problems Encountered and Success Achieved. *Journal of Information Technology,* **15**: 245-265.

Markus, L.M., Tanis, C., and van Fenema, P.C. 2000b. Multisite ERP Implementations. Communications *of the ACM* **43**(4): 42-46.

Markus, L.M. and Tanis, C. 2000. The Enterprise Systems Experience - From Adoption to Success. in R.W. Zmud, ed., *Framing the Domains of IT Research: Glimpsing the Future Through the Past*, Pinnaflex Educational Resources, Inc.: Cincinnati, OH, 173-207.

McGreevy, M. 2007. AMR Research Reports Midsize Enterprises will Increase ERP Budgets by 5.1% in 2008. *AMR Research*, December 10, accessed on the WWW October 29, 2008 at http://www.amrresearch.com/content/view.asp?pmillid=20997.

Nelson, R.R. and Cheney, P.H. 1985. Training End Uses: An Exploratory Study. *MIS Quarterly*, December: 547-559.

Newell, S., Huang, J., Galliers, R.D., and Pan, S.L. 2003. Implementing Enterprise Resource Planning And Knowledge Management Systems In Tandem: Fostering Efficiency And Innovation Complementarity. *Information and Organization* **13**: 25-53.

Nicolaou, A.I. 2004a. Firm Performance Effects in Relation to the Implementation and Use of Enterprise Resource Planning Systems. *Journal of Information Systems*, 18(2): 79-105.

Nicolaou, A.I. 2004b. Quality of Post implementation Review for Enterprise Resource Planning Systems. *International Journal of Accounting Information Systems,* **5**: 25-49.

Orlikowski, W. J. 2000. Using Technology and Constituting Structures: A Practice Lens for Studying Technology in Organizations. *Organization Science,* **11**(4): 404-428.

Orlikowski, W. J. and Gash, D. C. 1994. Technological Frames: Making Sense of Information Technology in Organizations. *ACM Transactions on Information Systems,* **12**(2): 174-207.

Poston, R. and Grabski, S. 2001. Financial Impacts of Enterprise Resource Planning Implementations. *International Journal of Accounting Information Systems* **2**: *27*1-294.

Ranganathan, C. and Brown, C.V. 2006. ERP Investments and the Market Value of Firms: Toward an Understanding of Influen*tial ERP Project Variables. Information Systems Research*

17(2): 145-161.
Repenning, N.P. and Sterman, J.D. 2002. Capability traps and self-confirming attribution errors in the dynamics of process improvement. *Administrative Science Quarterly* **47**: 265-295.
Roberts, B., Jarvenpaa, S., and Baxley, C. 2003. Evolving at the Speed of Change: Mastering Change Readiness at Motorola's Semiconductor Products Sector. *MIS Quarterly Executive*, 2(2): 58-73.
Robey, D. and Boudreau, M. 2000. Accounting for the Contradictory Organizational Consequences of Information Technology: Theoretical Directions and Methodological Implications. *Information Systems Research*, **10**(2): 167-185.
Robey, D., Boudreau, M., and Rose, G.M. 2000. Information Technology and Organizational Learning: A Review and Assessment of Research. *Accounting, Management, and Information Technology* **10:** 125-155.
Robey, D., Ross, J. W., and Boudreau, M. 2002. Learning to Implement Enterprise Systems: An Exploratory Study of the Dialectics of Change. *Journal of Management Information Systems*, (19(1): 17-46.
Rogers, E.M. 1995), *Diffusion of Innovations*, 4th ed., The Free Press, New York.
Ross, J. 1999. Surprising Facts about Implementing ERP. *IT Pro*, July/August: 65-68.
Sarkis, J. and Sundarraj, R.P. 2001. A Decision Model for Strategic Evaluation of Enterprise Information Technologies. *Information Systems Management* **18**(3): 62-72.
Scott, J.E. 2005. Post-Implementation Usability of ERP Training Manuals: The User's Perspective. *Information Systems Management*, Spring: 67-77.
Scott, J.E. and Kaindl, L. 2000. Enhancing Functionality in an Enterprise Software Package. *Information and Management* **37**: 111-122.
Seewald, N. 2002. Enterprise Resource Planning Tops Manufacturers' IT Budgets. *Chemical Week*, **164**(35): 34.
Senge, P.M. 1990. *The Fifth Discipline: The Art & Practice of The Learning Organization.* New York, NY: Doubleday.
Shang, S. and Seddon, P.B. 2002. Assessing and Managing the Benefits of Enterprise System: The Business Manager's Perspective. *Information Systems Journal*, **12**: 271-299.
Songini, M.L. 2003a. Goodyear Hits $100M Bump with ERP System. *ComputerWorld*, November 3, accessed on the WWW October 29, 2008 at http://www.computerworld.com/industrytopics/manufacturing/story/0,10801,86744,00.html.
Songini, M.L. 2003b. J.D. Edwards User Wins Nearly $2M Over ERP Rollout Gone Bad. *ComputerWorld*, December 18, accessed on the WWW October 29, 2008 at http://www.computerworld.com/softwaretopics/erp/story/0,10801,88383,00.html.
Sterman, J. D. 2000. *Business Dynamics*, Boston, MA, Irwin McGraw Hill.
Stiffler, D. 2008. U.S. ERP Skills Gap Leaves Titans Vulnerable. *AMR Research*, August 25, accessed on the WWW October 29, 2008 at http://www.amrresearch.com/content/view.asp?pmillid=21762.
Swanton, B. 2004a. Phase III ERP Benefits: The Art of the Formerly Impossible. *AMR Research Outlook*, April 29, accessed on the WWW October 29, 2008 at http://www.amrresearch.com/content/view.asp?pmillid=17244&docid=11591
Swanton, B. 2004b. Consistent Process Execution Yields Millions in ERP Benefits. *AMR Research Outlook*, June 18, accessed on the WWW October 29, 2008 at http://www.amrresearch.com/content/view.asp?pmillid=17344&docid=11711&bhcp=1

Thibodeau, P. 2004. Computerworld Honors: Innovative IT Success Stories Touted. *ComputerWorld*, June 8, accessed on the WWW October 29, 2008 at http://www.computerworld.com/softwaretopics/erp/story/0,10801,93719,00.html.

Tyre, M. J. and Orlikowski, W. J. 1994. Windows of Opportunity: Temporal Patterns of Technological Adaptation in Organizations. *Organization Science* **5(**1): 98-118.

Venkatesh, V., Morris, M. G., Davis, G. B., and Davis, F. D. 2003. User Acceptance of Information Technology: Toward a Unified View. *MIS Quarterly*, **27**(3): 425-478.

Venkatesh, V., Speier, C., and Morris, M. G. 2002. User Acceptance Enablers in Individual Decision Making About Technology: Toward an Integrated Model. *Decision Sciences*, **33**(2): 297-316.

Welti, N 1999. *Successful SAP R/3 implementation: Practical management of ERP projects*, Addison-Wesley: Reading, MA.

Wolstenholme, E.F. 2004. Using Generic System Archetypes to Support Thinking and Modelling. *System Dynamics Review* **20**(4): 341-356.

Yi, M. Y. and Davis, F. D. 2003. Developing and Validating an Observational Learning Model of Computer Software Training and Skill Acquisition. *Information Systems Research* **14**(2): 146-169.

Zviran, M., Pliskin, N., and Levin, R. 2005. Measuring User Satisfaction and Perceived Usefulness in the ERP Context. *Journal of Computer Systems,* Spring, 43-52.

APPENDIX A. CAUSAL MAP OF POST ADOPTIVE ERP BEHAVIOR

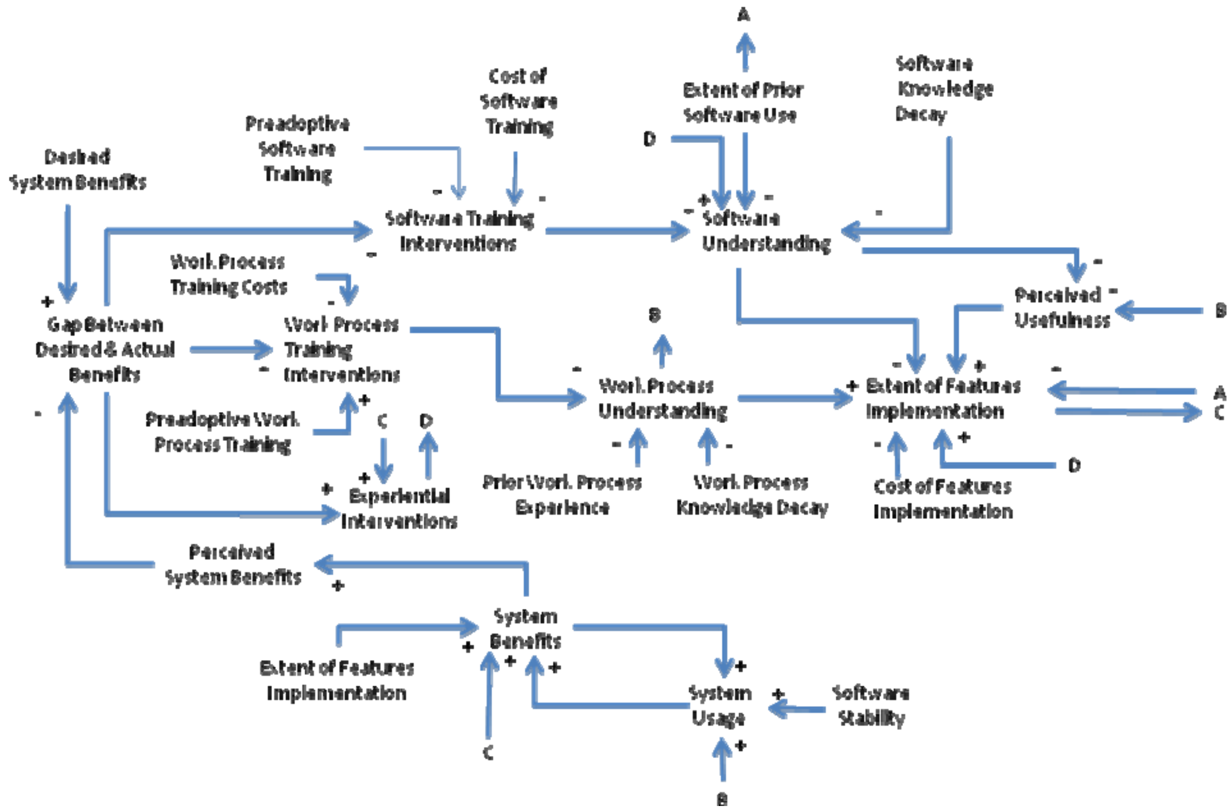

Figure A1. Post Adoptive ERP Model Causal Structure

APPENDIX B. SYSTEM DYNAMICS MODEL

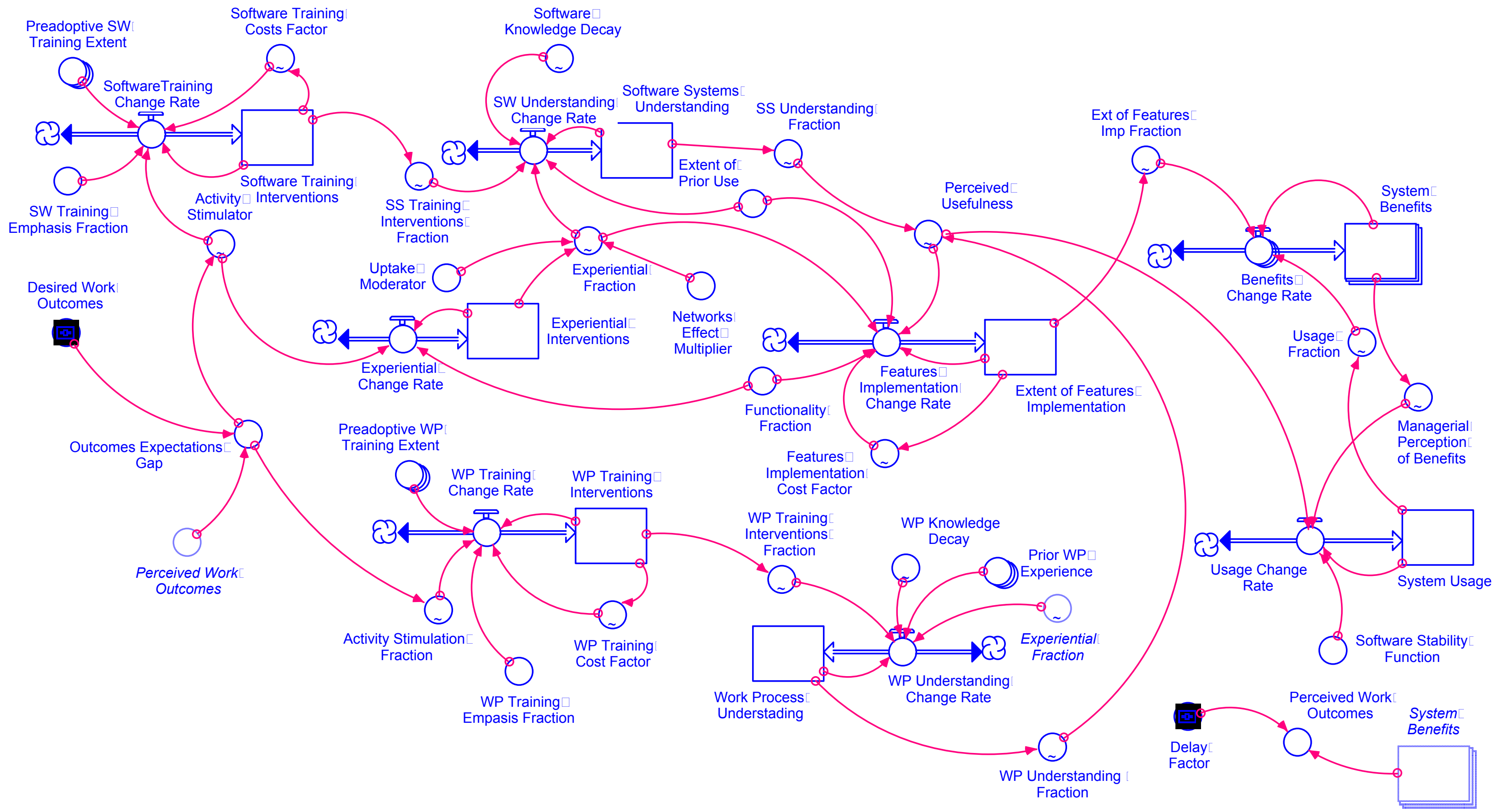

Figure B1. Post Adoptive ERP Model Dynamic Structure