



After Live Stage in Enterprise Software Implementations and Points to be Considered

Batuhan Kocaoglu¹

Abstract

The term enterprise software describes the applications that large companies use to conduct line-of-business operations such as accounting, business intelligence), communication and collaboration, customer relationship management and human resources. These tools are traditionally deployed in on-premise data centers, often as multi-faceted ERP suite. ERP is an industry acronym for Enterprise Resource Planning refers to automation and integration of a company's core business to help them focus on effectiveness & simplified success. Traditionally the process of controlling an investment like an ERP system ends with the delivery of the system to the users. But the realization of system benefits is reported to significantly lag expectations. The issue of successful implementations, as evaluated during the post-implementation process and the potential effects on the realization of potential system outcomes, provide a solid motivation for the present study. This kind of systems allows us lots of improvements. On the other hand, we must keep the system updated, regarding to business process changes. By using predefined indicators, we can measure the values in periods, and see the progress more easily and systematically. In this study we developed a framework for post-implementation studies and provided related indicators from the literature for a more effective management.

Keywords: Enterprise Software, Post Implementation, Usage, Indicator, Stage, ERP, MIS

Introduction

Enterprise software, most known type as **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 [1], [2].

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** [3], [4] because much of the research to date **focuses** on data collected just prior to, **during, or just after ERP software implementation** [5], [6], [7], [8], [2].

The primary purpose of this paper is to emphasize **how companies might facilitate usage of their installed ERP functionality**. This study addresses **maintenance activities** that take place **after an ERP system goes live**, classifies these activities into meaningful maintenance categories, then analyzes the indicators.

This study is relevant to both the **research and practitioner communities**. First, it contributes to knowledge on the **post-implementation stage** of ERP systems—an area where **little research** has been conducted. Second, the **findings** increase our understanding of **how maintenance** of software packages. Such understanding is **important and necessary for effective information systems management**. Third, with a **good understanding** of the indicators and sample alerts, usage of ERP systems can be better facilitated.

¹ Batuhan Kocaoglu, Okan University, Department of Logistics, Istanbul, Turkey, batuhan.kocaoglu@gmail.com



This paper covers a brief introduction to ERP software, after live activities, the methodology employed in this research, analysis of the results and findings.

Extent of ERP System Usage and ERP Systems Along Their Lifecycle

ERP like systems are designed around a process view of the business. It is widely reported in the literature that enterprise-wide applications promise seamless integration of all information flowing through a company: accounting and financial information, human resource information, supply chain information, and customer information [9], [10], [11].

Once the company successfully implements the ERP, the **attention moves forward** to the most efficient **use of the system**. Especially since considerable resources have been invested in the ERP implementation, the best possible **utilization** of the system is anticipated. Indeed, the value of an ERP system draws from its **effective and efficient usage** and not so much from the system itself [12].

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 [13], [14], [2]. 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 [15]. Even after the initial stabilization period, it often takes several years for firms to make a full transition to ERP [16], [17], [18]. 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 [19], [2].

Several **ERP life cycle models** have been reported in the literature to emphasize important phases and related activities during an ERP project. These models have phases comprising processes of 1) **Pre-implementation** phase (denoted as Project chartering phase) comprising selection of vendor and system, and signing a contract, 2) **Implementation** phase (denoted as Project phase), including the installation and configuration of the ERP-system, and 3) **Post-implementation phase** (denoted as Shakedown phase and Onward and Upward phase) involving solving of bugs, stabilization, further adaption, training, support and maintenance after the system is rolled out [16], [20], [21], [22].

In this study, we **utilized** the phases of an ERP life cycle to systemize our framework based on Esteves and Pastor [23]. This framework is structured in phases, which consist of the several stages that an ERP system goes through during its whole life within the hosting organization. The stages are: adoption decision, acquisition, implementation, **use and maintenance**, **evolution** and retirement phase. To simplify, we used the mentioned two sub-phases "use and maintenance", and "evolution" that correspond to "**post-implementation**" phase. In the following, we describe post phases.

The **post-implementation stage** in a system's life cycle encompasses a number of processes that are critical for a system's success. Following the implementation of the system, an organization would engage in a number of activities, such as post-implementation review, support and maintenance (e.g., [24]). **The focus of this paper** is on the factors that drive successful implementations during the post-implementation process of an ERP system. The two sub-phases are:

- a) Use and maintenance phase:** This phase consists of the use of the product in a way that returns expected benefits and minimizes disruption. During this phase, functionality, usability, and adequacy to the organizational and business processes are important. Once a system is implemented, it must be maintained, because malfunctions have to be corrected, special optimization requests must be met, and general systems improvements have to be implemented.
- b) Evolution phase:** In this phase, the system is upgraded by new technology insertion and additional capabilities are integrated into the ERP system to obtain improved benefits. The extensions can be classified in two types: 1) Evolution "upwards": Functionality is oriented to decision making with applications such as advanced planning and scheduling, data warehouses, and business intelligence systems; 2) Evolution "outward" to the system's environment, with applications such as customer



relationship management, supply chain management, inter-organizational workflow, and electronic commerce [25].

Research Method

This study has the objective to explore the process of post-implementation in ERP systems and **identify the factors** that lead to post-implementation. Such “why” questions can be answered using the case study method [26]. A qualitative approach was used to analyze a series of events exhibiting some theoretical principles.

The proposed framework is based on the assumptions listed below. These assumptions are drawn from (1) the body of literature, (2) the authors’ experience by working in the field of enterprise information systems for small businesses. In this research, the **authors themselves were actively involved** as consultants to the ERP implementations. We focused to find an answer to these questions and develop a roadmap based on previous separate studies:

1. What kind of activities must be performed during the post-implementation phase of the system life cycle?
2. What tools can be used for their measurement, if any?

Framework

In ERP projects, companies try to build the picture of their desired business processes, named as TO-BE state. First step of this study is, to create a picture of the current situation, named as AS-IS state.

Developed framework in Figure 1 emphasizes the road to TO-BE state. First, we must define the indicators in AS-IS state and record the results, and when the time comes, we must compare the results of the indicators in post-implementation stage. In post implementation stage, we assume that the system becomes routine and the heavy work related with configuration settings and customization is minimized. We must consider and track, different indicators in this stage. And results of the indicators must be evaluated.

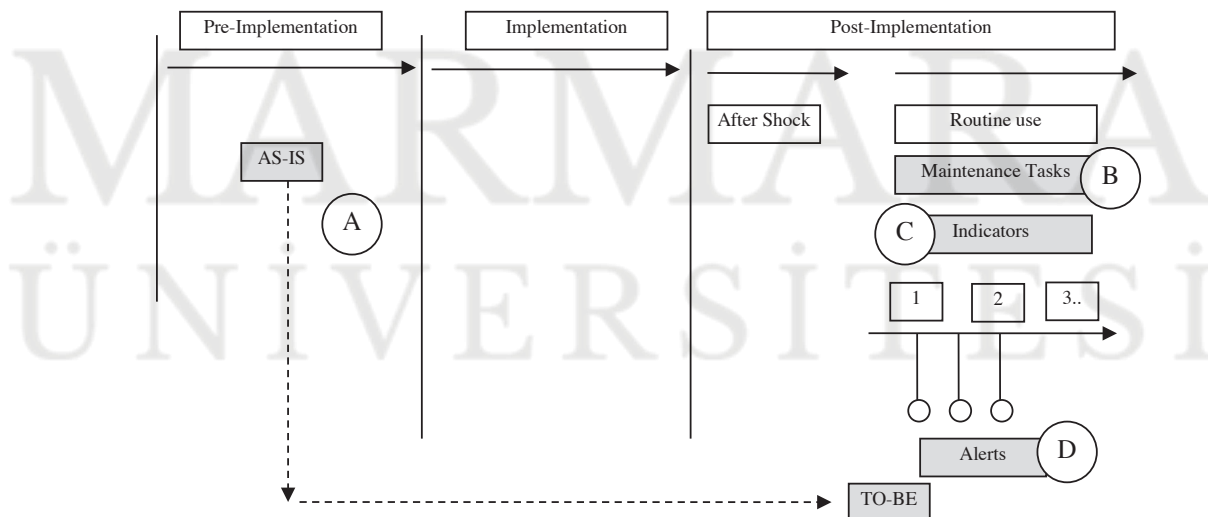


Figure 1. Developed Framework

As a result of the results of the indicators or audits, we may face some problems. In this study, we also emphasized some possible problems and their alerts, and their solutions. Using indicators, and alert-improvement cycles, companies can use their systems more efficient, and parallel with their business processes, up-to-dated.



Framework components namely A,B,C, and D will be discussed in following sections.

(A) Before (AS-IS) and After (TO-BE) Comparison

Companies must set some metrics in the pre-implementation stage, to **compare** with the desired state, and see if the targets have been reached. We must use the system for a period of time to see the results, for example after 1 year or 3 years. When the system comes **routine**, we can measure the new values of the indicators in the TO-BE situation, and compare with the values of the first AS-IS state.

Some sample metrics are given below. The following matrix is an example that sums up how the implementation to help companies to improve their operation in terms of intangibles. The number of metric can be increased.

Table 1. Indicators for AS-IS and TO-BE Comparison

Item	Before System (AS-IS)	Current System (TO-BE)	Resources
Quote to cash speed		Reduced	[27]
Offer realization time		Reduced	
Accurate cycle time	3 out of 4 (75%)	19 out of 20 (95%)	
Order processing speed in	in days	minutes	
Financial Close book data	> 9 days	few days / real time	
Paperwork		Less	[28]
Manual work		Less	
Paperwork associated with order processing		Less	
Time to process order and deliver		Less	

(B) After Activities (Maintenance)

In after live stage (post-implementations) systems is now routine. To make the system more sustainable, stable and updated, some task must be done. Some task must be executed periodically. In this study we mainly focused on list of Nah et. al (2001) [29].

Table 1 shows the two-tier classification of ERP maintenance. The first level comprises corrective, adaptive, perfective, and preventive maintenance as well as user support. External parties, such as the ERP vendor, third-party vendors, and consultants, may play a key role in ERP maintenance. Such activities may be sporadic or continuous and they take up a significant amount of time.

Table 2. Classification of Maintenance Tasks Into Maintenance Categories [29]

Maintenance categories	Maintenance tasks	Description of tasks
(1) <i>Corrective maintenance</i>	Application of hot packs	Incorporate system patches sent by ERP vendor (in cases where the problems could not be resolved by the IT department)
	Troubleshooting	Resolve anomalies reported by users
	Import new objects from ERP vendor	Incorporate objects (lines of code) sent by the vendor to solve problems (examples include new database structures, programs, and new reports)
(2) <i>Adaptive</i>	Transfers	Moving new features from development to test to production environment
	Testing	Integration testing after hot pack application and configuration changes
	Modifications and enhancements	In-house customization and modification performed by the IT personnel, and report development by the developer team
	Authorizations	System password maintenance, changes in access permissions as staff arrive, leave, or change positions
	Tuning of system interfaces	Tuning interfaces with other software
(3) <i>Perfective maintenance</i>	Version upgrade	Justification, planning, and actual implementation of new software versions
(4) <i>Preventive maintenance</i>	Routine administration	Monitoring average system response times and thresholds, file sizes, tape backups, and error logs



(5) <i>User support</i>	Monitoring workflow	Tracking flow of maintenance work items
	Training users	Training users on new or existing features
	Help desk	Answering user questions about the system
(6) <i>External parties</i>	Coordination and administration	Coordinating work and relations among ERP team members, vendors, contractors/consultants, and external user organizations
	Creation of Online Service System (OSS) notes	Online query or reporting of problems to the vendor, tracking vendor's progress towards resolution of problems reported

(C) After, Comparison Between Periods

The indicators are dedicated to an examination of the company's ability to effectively make use of the ERP system's functions as well as to enhance and improve it. Since that ability depends on the know-how of personnel, employee-centered measures covering both users and IT staff are called for. Table 3 lists the indicators that can be used in evaluating the system usage, among periods.

For example, a useful indicator is the level of training courses, measured by the amount of time or expenses spent. Specifically for system developers, their type of formal qualification can additionally be surveyed [30]. In some cases, it may have feed-forward value - negative deviations of actual training costs versus budgeted costs may indicate that the system's functions are not efficiently used by staff members. By contrast, a continuous increase in external consulting expenses may point to deficiencies in the internal training staff's competence [30].

Table 3. Indicators for Comparison Of The Different Period Of The Current System

Goal	Measure	Resource
Compliance with budget	Maintenance – Hardware : Budget/Actual	[30]
	Maintenance – Software : Budget/Actual	
	Maintenance - Consulting : Budget/Actual	
	Training – Hardware : Budget/Actual	
	Training - Software : Budget/Actual	
	Training – Consulting : Budget/Actual	
Coverage of business processes	% of covered process types	[30]
	% of covered business transactions	
	Percent of critical processes monitored	[31]
	% of covered transactions valued good or fair	[30]
Operational problems	% of transaction not finished on schedule	[30]
	% of cancelled telephone order processes due to non-competitive system response time	
	# of problems with customer order processing	
	# of problems with warehouse processes	
	# of problems with standard reports	
	# of problems with reports on demand	
Technical-Availability of the ERP-system	average system availability	[30]
	average downtime	
	maximum downtime	
	average response time in order processing	
	average response time in order processing in the peak time	
	average # of OLTP-transactions	
	maximum # of OLTP-transactions	
Actuality of the system upgrade	average time to upgrade the system	[30]
	release levels behind the actual level	
Development support	punctuality index of system delivery	[30]
	quality index	
	average workload per developer	
	rate of sick leave per developer	
	% of modules covered by more than 2 developers	



	# of training hours per user	
	# of training hours per developer	
	qualification index of developer	
	# of consultant days per module in use > 2 years	
	# of consultant days per module in use < 2 years	
Reliability of software vendor	# releases per year	[30]
	# of functional additions	
	# of new customers	
Technical-Using ERP system	How many employees use the system daily?	[32]
	How much time per day do employees work with the system? (%)	
	reports generated per day? (%)	
	Amount of use/duration of use	[33]
	Charge for system use	
	Number of reports generated	
	Number of inquiries	
	Amount of connect time	
	Number of different computer systems	[28]
Process improvement	Number of changes to targets for key processes effectiveness and efficiency	[31]
	Amount of satisfaction of management and the governance entity with ERP system	
	Amount of reduction in the number of outstanding process deficiencies	
	Amount of stakeholder satisfaction with the measuring process	
	Number of improvement actions driven by ERP system	
	Number of performance targets met	
Request management	Requirements Stability Index (RSI)=No. Of requirement changed (added / deleted / modified) / Total No. initial requirements	[34]
	Arrival Rate of Problems (ARP)=No. of problems reported by the users per month	
	Closure Rate of Problems (CRP)=No. Of problems closed per month	
	Age of Open Problems (AOP)=Sum of time (days per month) that problems have been open / No. Of open problems per month	
	Age of Closed Problems (ACP)=Sum of time (days per month) that problems have been closed / No. Of closed problems per month	
	Frequency of requested for special functions	[35]

(D) Alerts and Actions

The information system cannot support company strategy without being mastered as a “tool”. Certain situations are consequently impossible (control of the processes without mastery of the software). Alert criterias allowing recognising, and by the typical actions of improvement to be implemented at this level. These alert criteria and improvement actions are presented in Table 4.

Table 4. Alerts and actions (derived from [36])

Group	Level	Alerts	Actions
software maturity	Software mastery	Non-appropriation of the system by the users	Additional training of the users
		Unsatisfactory operational execution	Create a competence centre
		Insufficient speed/ability to react	Empowerment of the users on their role and on their duty (user's charter, quality indicators)
		Insufficient system response time	Stabilization of the execution



	software maturity		(indicators with follow-up of objectives)
		The users create parallel procedures	
		No documentation on parameters, data, data management procedures	
	Improvement	The full ERP potential is not used	Implement the functions that are not yet used
		Results not reached, expectations unsatisfied	Definition of performance indicators, business indicators
		The standard system installed does not fit all requirements	
	software maturity	The number of office automation utilities increases	Improvement and automation of the reporting
		The procedures are too heavy	Rethink the roles to simplify the procedures
	Evolution	Context “multi-activities”, international firm	Standardization on several sites/activities
		Reorganizations, technological changes	Version upgrade
		Need of (analytics) reporting	Develop business intelligence systems
	software maturity	Outside integration: B to B	Address the ERP/environment technological evolution (EAI)
		Bar-code integration	Implementation of enterprise architecture (application mapping)
		Software maturity depreciation	Version upgrade
strategy deployment	Master-data control	Numerous erroneous technical data	Cleaning of the migrated data
		Messages of ERP not relevant (stock shortages, rescheduling in/out MRP, purchase proposals)	Maintain a business project team with a plan of action to master-data
		Numerous manual inventory corrections	Define responsibility for data
		Product lifecycle not improved, not integrated in the IS (revision ...)	Indicators of data control
			Assert the uniqueness of the data in the whole company
			Revise management rules in the company
	Process control	Conflicts between services on procedures	Verify the appropriateness of the tool to the organization
		Contradictions between local and global indicators	Rethink the roles to simplify the procedures
		Results not reached, needs unsatisfied	Define responsibility for processes
		Demands for improvement, for roles redefinition by the users	Strengthen the transverse responsibilities: indicators, communication
		Higher expectations of customers and top management	
		No return on investment calculated	
	Strategy support	Business objectives not reached	Modeling and optimization of the supply chain
		Higher expectations of customers and top management	External integration: B to B
		Changes of markets, of customer expectations	Implementation of application mapping
		International extension	Business Process Management (modeling, process performance measure)
		Management expectation concerning follow-up consultancy	IT associated to business strategies



Findings and Recommendations

Problems associated with enterprise resource planning (ERP) implementations become more rampant during the post implementation phase because once users learn their way around the system, they can **test its limits**, setting off shockwaves that can devastate an otherwise successful control environment. While extensive integration and acceptance testing flushes programming errors and exercises pathways through the system, post implementation brings an explosion of **bug reports** that hit the maintenance team, along with a slew of frenzied requests for new functions. As a result, the post implementation phase requires an **ongoing process of improvement** and fine tuning. It is the greatest challenge to audit, because an audit is expected to unravel the root of problems in the ERP processes and provide effective solutions [37].

To be effective, there must be a strong, across -the-board commitment to technology **audits**, including upper management. The audit also should take into account how well employees interact with the software. Does it help them feel more empowered to accomplish key tasks? How quickly do they feel they've mastered the software? Implementing **user surveys** immediately following training and again six months later can be advised [38].

The areas that are **designed to automate processes**, such as printing shipping labels and notifying customers, should be relatively easy to measure, as long as you've established benchmarks. More difficult to measure are areas that reveal an employee's lack of familiarity with the software. For example, if your audit reveals little improvement in overall shipping errors despite the software's automation capabilities, you may need to go back again to key employees. Ask them to show you precisely how they use the software [38].

When employees fail to use all of the ERP features because of inadequate training or because of a natural resistance to change, the software benefits will erode over time, until you're only using a fraction of its capabilities. At some point, you have to return to base-level training to protect your technology investment. A **continuous feedback loop** is necessary. You can go back in and talk to end users to discover any knowledge gaps, so that you can take corrective action. If end users have the perception that software is the problem, that perception tends to become a reality [38].

Natural **disasters**, security failures, or even a change in business cycle can affect your servers and applications. Start by defining what constitutes a disaster. Then you can plan for those things you believe are beyond your control, such as weather or theft [38].

In the post-implementation stage, when a critical mass of users is fully using the system, bringing to it all their ingenuity and creativity, when post-implementation aftershock strikes and bug reports start flooding the maintenance group. The bugs they report are rarely trivial and are difficult to track down. Many of these bugs result from users **doing tricky things** with the system in hopes of "faking it out". The records generated by these tricks cause problems of which the administrators are not aware in the subsystem. It takes **major detective work** to determine the source of these illegal records [37].

But this **slew of system** requests is **not a sign of bad design**; rather, it is a sign of the system's success. It shows that users have **accepted** the system and are putting it to use. It also demonstrates that the experienced users are confident enough to go beyond a cookbook approach to using the system and are becoming **creative** and innovative partners in the system's design [37]. Now, having digested the system that, these sophisticated users want to **push the system's limits further**.

It is essential to retain the services of the people who implemented the system, since it is at this stage that **users request** the **most useful** and, to them, necessary enhancements to the system.

The best way to track **aftershock** is to design a function into the new ERP system that monitors **system usage** in terms of transactions or function used and frequency of use, breaking that usage down by individuals. This information is usually available in the system already but is not presented in an accessible report format. This function should generate reports that go to ERP project management on a



weekly basis during the first months of the system's life, and it is a good idea to have a version of this information sent to user group management. These reports will give information that is helpful in several respects [37].

An **audit** can be performed for various reasons based upon the organization's needs. It might be a post-implementation **audit** or an ongoing audit performed to monitor system integrity or test controls based upon management's control objectives.

Some example issues to be checked in the audit process: Authorization Concept, Review the list of inactive users (An auditor can review user access logs and ensure that the user list contains only active users), Review the security of custom codes and tables (Check for users who have program execution access)

How often you measure will depend on the size of your organization and the depth of your software change [38]. Typically, an **audit** should occur halfway through the implementation, again at three-quarters of the way through the implementation, and when it's complete. Then, you should follow up at 3-, 6-, and 12-month intervals.

Due to the complexity and extremely technical aspect of the system, ERP reviews require a **person** who understands this environment and is trained to perform such reviews.

Conclusions

The implementation concerns of ERP do not end once the system becomes operational. As indicated by Davenport [39], 'an enterprise system is not a project; it's a way of life'. He argues against the traditional assumption of treating ERP as a project that has a termination date. Users need **on-going support and organizations** face a variety of issues such as fixing problems, upgrading to new versions of the software, and managing organizational performance with the system to achieve desired benefits [40]. Since ERP is continually evolving and is a fairly new phenomenon, little research has been done to study maintenance of these systems [29].

While many companies spend significant amounts of time and money researching, analyzing, and justifying an enterprise resource planning (ERP) purchase, they give only a token look (if any) at how well the application actually performs once it's installed. And that's a big mistake [38].

Realization of ERP system benefits is reported to significantly **lag expectations** [41], [42], [43]. As a result, the issue of successful implementations, as evaluated during the post-implementation process and the potential effects on the realization of potential system outcomes, provide a solid motivation for the present study.

Most of the companies focus on choosing the right product. To but if companies don't establish specific **performance metrics**, it will be very hard to gauge how well that product is working to meet their objectives, let alone correct any **performance gaps**.

All management and resources focus on going live. Most of the companies don't engage in a thorough, **after-the-fact audit**. Because there is not a roadmap, and no one agrees on what precisely constitutes a successful ERP program. Also it may be dangerous if the project fails for the responsible senior management.

In this study, we proposed a **roadmap** by a framework and related indicators. Identified metrics and activities can help companies.

Approaches that are related to the productive phase of the ERP system focus mainly on the financial effects in terms of costs. We also presented a set of interrelated **measures** that give a more complete insight into the **performance** of the ERP system **beyond financial key** performance indicators.



Limitations

We indicated these factors from literature and author's consulting experience. Although limited to few industries and companies, we believe these cases capture general information that would be expected to be found in many organizations. With better track of the indicators, ERP system management can be accomplished more successfully.

For Further Studies

For further studies, **number** of the indicators listed in tables may be increased. Developing a consensus on post-implementation and ERP usage metrics will help companies to analyze and benchmark using general accepted metrics. So, this study may help for generalization and standardization of the metrics, for measuring the ERP systems usage and maintenance.

Companies should calculate the values of these metrics in different periods and make evaluation. Results must be interpreted carefully to achieve useful results. Based on these results, it becomes possible to see some critical points and points to be improved. Also accepted metrics, can be measured using ERP logs and simple queries.

References

- [1] Scott, J.E., and L. Kaindl. "Enhancing Functionality in an Enterprise Software Package." *Information and Management*, 2000: 37, 111-122.
- [2] Clark Jr., Thomas D., Mary C. Jones, and Robert W. Zmud. "Post Adoptive ERP Use Behaviors: A Dynamic Conceptualization." July 26-30, *Proceedings of the 27th International Conference of the System Dynamics Society*. Albuquerque, New Mexico, USA, 2009.
- [3] Sarkis, J., and R.P. Sundarraj. "A Decision Model for Strategic Evaluation of Enterprise Information Technologies." *Information Systems Management*, 2001: 18(3), 62-72.
- [4] Sarkis, J., and R.P. Sundarraj. "A Decision Model for Strategic Evaluation of Enterprise Information Technologies." *Information Systems Management*, 2001: 18(3), 62-72.
- [5] Hunton, J.E., B. Lippincott, and J.L. Reck. "Enterprise Resource Planning Systems: Comparing Firm Performance of Adopters and Nonadopters." *International Journal of Accounting Information Systems*, 2003: 4, 165-184.
- [6] Jones, M.C., M. Cline, and S. Ryan. "Exploring Knowledge Sharing in ERP Implementation: An Organizational Culture Framework." *Decision Support Systems*, 2006: 41(2), 411-434.
- [7] Ke, W., and K.K. Wei. "Organizational Culture and Leadership in ERP Implementation." *Decision Support Systems*, 2008: 45, 208-218.
- [8] Zviran, M., N. Pliskin, and R. Levin. "Measuring User Satisfaction and Perceived Usefulness in the ERP Context." *Journal of Computer Systems*, 2005: Spring, 43-52.
- [9] Davenport, T.H. "Putting the enterprise into the enterprise system." *Harvard Business Review*, 1998: 76 (July/August), 121-131.
- [10] Kumar, K., and J. Van Hillegersberg. "ERP: Experiences and evolution." *Communications of the ACM*, 2000: 43 (April), 23-26.
- [11] Nicolaou, Andreas I., and Bajor. "ERP Systems Implementation And Firm Performance." *The Review Of Business Information Systems*, 2004: 8(1), 53-60.
- [12] Moon, Young B. "Enterprise Resource Planning (ERP): a review of the literature ." *Int. J. Management and Enterprise Development*, 2007: 4(3), 235-264.
- [13] Hitt, L. M., D.J. Wu, and X. Zhou. "Investment in Enterprise Resource Planning: Business Impact and Productivity Measures. ." *Journal of Management Information Systems*, 2002: 19(1): 71-98.
- [14] Poston, R., and S. Grabski. "Financial Impacts of Enterprise Resource Planning Implementations." *International Journal of Accounting Information Systems*, 2001: 2: 271-294.
- [15] Jones, M.C., and R.L. Price. "Organizational knowledge sharing in ERP implementation: Lessons for industry." *Journal of End User Computing*, 2004: 16(1), 21-40.
- [16] Markus, L.M., and C. Tanis. "The Enterprise Systems Experience - From Adoption to Success." In *Framing the Domains of IT Research: Glimpsing the Future Through the Past*, by R.W. Zmud, 173-207. Cincinnati,



OH.; Pinnaflex Educational Resources, Inc., 2000.

- [17] Nicolaou., Andreas I. "Quality of Post implementation Review for Enterprise Resource Planning Systems." *International Journal of Accounting Information Systems*, 2004: 5: 25-49.
- [18] Ross, J. "Surprising Facts about Implementing ERP." *IT Pro*, 1999: July/August: 65-68.
- [19] Robey, D., J. W. Ross, and M. Boudreau. "Learning to Implement Enterprise Systems: An Exploratory Study of the Dialectics of Change ." *Journal of Management Information Systems*, 2002: 19(1), 17-46.
- [20] Hustad, Eli, and Aurilla A. Bechina. "A study of the ERP Project Life Cycles in Small-and-Medium-Sized Enterprises: Critical Issues and Lessons Learned ." *World Academy of Science, Engineering and Technology* 60. 2011. 110-116.
- [21] Esteves, J.M., and J.A. Pastor. "An ERP Life-cycle-based Research Agenda." *Enterprise Management and Resource Planning: Methods, Tools and Architectures - EMRPS-99; in First International workshop . Venice, Italy*, 1999. 359-371.
- [22] Aloini, D., R. Dulmin, and V. Mininno. "Risk management in ERP project introduction: Review of the literature." *Information and Management*, 2007: 44(6), 547-567.
- [23] Esteves., J., and J. Pastor. "Towards the unification of critical success factors for ERP Implementations." *10th Annual BIT conference.*, Manchester, UK, November (2000), 2000. 1-9.
- [24] Gelinas., U., and S. Sutton. *Accounting information systems*. Boston (MA): South Western, 2002.
- [25] Nazemi, Eslam, Mohammad Jafar Tarokh, and & G. Reza Djavanshir. "ERP: a literature survey." *Int J Adv Manuf Technol*, 2012.
- [26] Yin, R.K. *Case Study Research: Design and Methods*. Thousand Oaks: Sage, 1994.
- [27] Chan, Xin, and Lau Yui-yip. "Evaluation Of Erp Implementation On Firm Performance: A Case Study Of A&T."
- [28] Rajagopal, Palaniswamy, and Frank Tyler. "Enhancing Manufacturing Performance with Erp Systems." *Information Systems Management*, 2000: 17:3, 1-13.
- [29] Nah, Fiona Fui-Hoon, Silvana Faja, and Teuta Cata. "Characteristics of ERP software maintenance: a multiple case study ." *Journal Of Software Maintenance And Evolution: Research And Practice*, 2001: (13) 399-414.
- [30] Rosemann, Michael, and Jens Wiese. "Measuring the Performance of ERP Software – a Balanced Scorecard Approach." *10th Australasian Conference on Information Systems*. 1999. 773-784.
- [31] Chen, Zhuang, and Taige Wang. "An IT governance framework of ERP system implementation for Chinese enterprises." *7th World Congress on Intelligent Control and Automation*. 25-27 June, WCICA 2008, 2008. 4929-4932.
- [32] Ruivo, P., T. Oliveira, and M., Neto. "ERP post-adoption: Value impact on firm performance: A survey study on Portuguese SMEs." *Information Systems and Technologies (CISTI), 7th Iberian Conference*. 20-23 June, 2012. 1-6.
- [33] Tsai, W.-H., T.-S. Tsaur, Y.-W. Chou, J.-Y. Liu, and J.-L. Hsu. "Evaluating the information systems success of ERP implementation in Taiwan's industries." *IEEE International Conference on Industrial Engineering and Engineering Management*. 8-11 Dec, IEEM2009, 2009. 1815-1819.
- [34] Parthasarathy, S, and N Anbazhagan. "Significance of Software Metrics in ERP Projects." *India Conference*, . Sept., 2006 Annual IEEE, 2006. 1-4.
- [35] Wei, Chun-Chin. "Evaluating the performance of an ERP system based on the knowledge of ERP implementation objectives." *International Journal of Advanced Manufacturing Technology*, 2008: 39, 168-181.
- [36] Botta-Genoulaz, Vale'rie, and Pierre-Alain Millet. "A classification for better use of ERP systems." *Computers in Industry* , 2005: 56, 573-587.
- [37] Musaji, Yusuf. "ERP Postimplementation Problems ." *Information Systems Control Journal*, 2005: 4, 1-9.
- [38] Saunders, Trish. *Measuring the Value of Your ERP System*. December 1, 2004.
- [39] Davenport., T.H. *Living with ERP*. December 1, 1998. http://www.cio.com.au/article/108202/managing_-_living_erp/.
- [40] Eriksen, L.B., S. Axline, and M.L. Markus. "What happens after 'going live' with ERP systems? Competence Centers can support effective institutionalization." *Proceedings of the Fifth Americas Conference on Information Systems*. Association for Information Systems: Atl. Atlanta GA, 1999. 776-778.
- [41] Benchmarking_Partners. *ERP's Second Wave: Maximizing the Value of ERP-Enabled Processes*. Cambridge, MA, 1998.
- [42] META_Group. *Enterprise Resource Management (ERM) Solutions and Their Value*. . Stamford, CT., 1999.
- [43] Peterson, W.J., L. Gelman, and D.P. Cooke. *ERP Trends*. New York, NY: The Conference Board; Report 1292-01-RR, 2001.