Enterprise Resource Planning Systems
Streamlining Upgrade Decisions

Gerald Feldman, Hanifa Shah, Craig Chapman and Ardavan Amini
Birmingham City University, Faculty of Technology, Engineering and the Environment, Millennium Point, Curzon Street, Birmingham, B4 7XG, U.K.

Keywords: Enterprise Resource Planning, Upgrading, Upgrade Decisions, Decision Support.

Abstract: Few organizations choose to upgrade their systems despite the benefits of new features and additional functionality such as web based services offered by upgrading Enterprise Resource Planning (ERP) systems. The reason for this is upgrading an ERP system remains a complex undertaking which requires strategies to minimise disruption to business operations. High costs and risks associated with upgrading imply that the decision to upgrade is not trivial and should be undertaken for the right reasons that make business sense and have clear benefits. Conversely opting not to upgrade has long term implications such as using outdated ERP systems which lack continued technical support or obtaining support at a very steep price. The paper will explore factors and challenges that influence ERP upgrade decisions and identify key features to streamline ERP upgrade. The outcome identifies that decision support methodologies and techniques could play a significant role in streamlining ERP upgrading decision.

1 INTRODUCTION

The need to remain competitive, streamline operations and improve collaboration encouraged the adoption of computer systems that facilitated cross-functional integration of the business process such as Enterprise Resource Planning (ERP) systems. ERP systems are classified as large complex systems, offering a range of capabilities which simplify cross-functional integration of data and processes to support information processing needs of the entire organisation (Esteves and Pastor, 1999; Nah et al., 2001; Dittrich et al., 2009). The implementation of ERP systems tracks back to the 1990s, with the purpose of addressing maintenance issues of legacy systems and reduce development risk. It was anticipated that the adoption of ERP systems would provide reliable and timely access to information and improve business efficiency (Grabski et al., 2011; Davenport et al., 2004). These efforts resulted in a situation whereby organisations with ERP systems were able to achieve better consistency of business processes and capability to automate business processes.

ERP systems have matured over the years and the dependency on these systems warrants organisations to adopt efficient strategies for upgrading their systems. ERP upgrade is a continuous process recurring throughout the system's lifespan at least once every three years (Olson and Zhao, 2007). Vaucouleur (2009) explains that upgrading is a process that aims to expand the core system capabilities by improving functionality and taking advantage of new technology features. While Ng (2011), defines the upgrade decision as "decision made which results in the installed old ERP version (partly or as a whole) being replaced by a newer version either for the same or different vendor's product". Therefore, ERP upgrade can be defined as an improvement to the existing systems, and involves changing an aspect of that system or implementing a newer version depending on the business requirements. Although, the decision to upgrade is not a trivial one, as a right balance between minimising the business disruption and leveraging latest technologies such as service-oriented architecture needs to be justified.

The timing of when to upgrade is important in establishing the balance and there are numerous internal and external factors to be considered (for example business needs, vendor support) prior to an ERP upgrade (Claybaugh, 2010). As justified by Ng and Gable (2009) " … upgrades require more thorough planning, business justification, money,

Feldman G., Shah H., Chapman C. and Amini A..
Enterprise Resource Planning Systems - Streamlining Upgrade Decisions.
DOI: 10.5220/0004420501280135
In *Proceedings of the 15th International Conference on Enterprise Information Systems* (ICEIS-2013), pages 128-135
ISBN: 978-989-8565-59-4

resources to implement, serious consideration of potential system downtime, effort for impact analysis and re-application of previous modifications or user-enhancements (if the new version has not incorporated the required functionality), and a longer time to complete".

Understanding when to upgrade combined with strategies which ensure upgrades are supported by a justifiable business case and are undertaken for the right reasons play a key role in supporting upgrade decisions. Recent research on ERP upgrade have extensively covered best practise (see Beatty and Williams, 2006) and success factors (see Nah and Delgado, 2006; Olson and Zhao, 2007) and decision models (see Ng, 2011; Otieno, 2010; Khoo, 2006). Yet, these studies have not adequately addressed methods and tools which can streamline upgrade decision making.

This paper is organised as follows, the first section provides an overview of ERP upgrade discussing the life span of an ERP system and briefly introducing the type of upgrades which can be undertaken and highlighting the challenges associated with upgrading. The second section focuses on the factors involved in the upgrading decisions made within the organisations and associates the decision to upgrade with the type of upgrade to be selected. The last section sets out to identify key features and methods to streamline the decision to upgrade by exploring the literature.

2 ERP UPGRADE OVERVIEW

The reliance and growing use of ERP systems to support and streamline business processes creates a necessity to explore the stages after 'go-live' as this is where the actual business value of the system becomes visible. Willis and Willis-Brown (2002) explain that organisations which only considers systems 'go-live' as the final stage fail to realise the full potential of the ERP system.

2.1 ERP System Life-cycle Model

To understand the stages after 'go-live' it is important to understand ERP system life-cycle in order to differentiate the time span the system undertakes. The ERP life cycle stages are organised systematically to represent the different activities undertaken from system adoption up to when the system is phased-out. Several authors have defined numerous stages (table 1), ranging from three (see Law et al., 2010) to a maximum of six stages (see Cooper and Zmud, 1990). Even though these stages are defined differently some commonality between the different researchers' view exists. For example, the following stages: chartering, agenda formation and project initiation have similar emphasis that is defining the business case and set of actions such as team formation and selecting the software. This

Table 1: ERP life-cycle stages summary.

References	Stages in ERP life-cycle					
	1	2	3	4	5	6
Cooper and Zmud (1990)	Initiation	Adoption	Adaptation	Acceptance	Routinize	Infusion
Esteves and Pastor (1999)	Adoption	Acquisition	Implementation	Use and maintenance	Evolution	Retirement
Markus and Tanis (2000)	Chartering	Project	Shakedown	Onward and Upward		
Bajwa et al. (2004)	Awareness	Selection	Preparation	Implementation	Operation	
Pan et al. (2007)	Agenda Formation	Design	Implementation	Appropriation		
Worrell (2008)	Project Initiation	Implementation	Stabilization	Post-implementation		
Law et al. (2010)	Initiation stage	Contagion stage	Integration			

paper does not address these stages in detail as it is not within the scope and previous literatures (for example, Markus and Tanis, 2000; Pan et al., 2007) have extensively addressed these stages. Willis and Willis-Brown (2002) groups the different stages into implementation and post-implementation phases.

Examining the different ERP life-cycle stages, some confusion arise as to which stage best represent the implementation and post-implementation phases, since some activities overlap between the two phases. Worrell (2008) modifies Markus and Tanis (2000) life-cycle model to provide a distinction between the implementation and post-implementation stage. Through this model, Worrell proposes that activities such as bug fixing and perfomance tuning are not part of the post-implementation phase and only introduced to support the system after 'go-live'. While, Nah et al. (2001); Ng et al. (2002); Hecht et al. (2011) consider the post-implementation phase to involve activities such as bug fixing, user training, performance tuning, enhancement and upgrades which are critical components of the ERP maintenance. Based on Willis and Willis-Brown (2002) explanation and considering Nah et al. (2001) categorisation of maintenance activities, it can be summarised that an ERP life-cycle which clearly offers a distiction between the implemention and post-implementation phases will in theory include activties that stabilise and extend the ERP system after 'go-live'.

2.2 ERP Systems Upgrading

As vendors continuously improve the underlying technology of ERP systems, it is reasonable to anticipate that organisations will take advantage of the new version functionality and features such as service oriented architectures. However, organisations maintain control of their systems irrespective of the ERP version release cycle and will only upgrade when the stability and reliability of the new version can be assured. In addition, most vendors support more than one version at a time rendering it not important to upgrade whenever a new version is available as stipulated by (Khoo and Robey, 2007). Resulting in an environment where organisations delay upgrading in order to establish a strong business case which defines the added value for upgrading. Hamerman et al. (2011) survey results supports this thinking as more than 50% of the survey participants are still utilising ERP versions which are at least two versions behind the current version. Yet, opting not to upgrade introduces an environment where organisations utilise outdated ERP systems and risks losing continued technical support (Ng, 2001).

ERP upgrade can be classified as technical, functional or strategic upgrade. Most of the time organisations will undertake technical upgrade, but it is not always the most feasible option, resulting in the need to incorporate technical and functional upgrade simultaneously. Yet, the cost and risks associated with integrating both upgrade strategies needs to be justified, according to Swanton (2004) and Otieno (2010) the upgrading cost ranges between 20% to 30% of the initial ERP implementation cost. Therefore, it is important to have strategies that streamline the complexity of the process and support undertaking both technical and functional upgrade.

2.2.1 Technical Upgrade

Technical upgrade entails changing the existing ERP version to a newer version from the technology perspective, it does not involve adopting new functionality or modifying the core ERP's system architecture to incorporate the organisations business processes. Generally technical upgrade is commissioned to sync the ERP systems to the version that is supported by the vendors, thus ensuring continuous technical support however, it is not a straight forward swap of the systems. Beatty and Williams (2006) suggest in order to take full advantage of ERP upgrade, organisations may require to assess their information technology infrastructure and have mechanisms that ensure the systems will perform smoothly after the upgrade. Despite the fact no modifications are introduced the underlying code and standard of the new ERP version may be different, requiring previous modifications to be converted for smooth transition into the new system.

The different changes imposed on the systems require rigorous testing to guarantee the systems work with minimum interruption and its performance is not affected. The testing process and strategies results into the majority of the developers' workload to be associated with testing of the modification introduced to the system (Beatty and Williams, 2006). Traditionally manual test approaches were utilised, whereby test cases were identified in the development environment, potentially preventing code reusability and prone to errors and dependent on individual knowledge (Dittrich et al., 2009). Several automation testing tools are available to assist in detecting and identifying probable problematic areas introduced by

modifications, for example Dor et al. (2008) proposes a program slicing algorithm which simulates the impact of previous system modifications. The output from the automated algorithms offer detailed information on the complexity of the upgrade task and provide the impact of upgrading.

2.2.2 Functional Upgrade

Functional upgrade involves implementing the generic functionality offered in the newer version to replace existing modification and reduce system complexity by automating existing business process to simplify future upgrades (Riedel, 2009). New functionality can be added through modifying the existing system architecture, although this is considered to be a major technical challenge which can result in bugs and performance degradation of the underlying system (Beatty and Williams, 2006). In this paper modifications will be categorised as either user or vendor modifications, in order to differentiate and establish the impact of modifications on the upgrade decision. Vendor modifications are changes introduced to the system by vendors in collaboration with the organisations' functional and technical staff. These changes are normally required because the underlying code of the system needs extensive modifications to include features and functionalities which may alter the core system. Once these changes are integrated for that specific client, normally the next step would be to incorporate this new functionality in the standard future versions to benefit other clients in that industrial sector.

User modification refers to the changes introduced by the organisation to meet desired functionality in accordance to their business process. This requires an extensive understanding and knowledge of the underlying system and business processes, as changes applied in one business module may affect other modules of the associated system (Rothenberger and Srite, 2009). Additionally, ERP vendors do not support extensive user modifications and several documentations have reported on the potential threats introduced by these modifications (see Brehm et al., 2001; Vaucouleur, 2009). Hence, as mechanism to add new functionality and increase flexibility some organisations decide to upgrade their systems thus gaining the additional capabilities and features introduced in the new version.

2.2.3 Strategic Upgrade

Strategic upgrade entails consolidation of different systems, through implementing a technological platform that provides better agility and flexibility to support system integration. The main focus is on functionality extension and optimising business process based on the core new functionality. This involves significant business process re-engineering and implementation of new components to accommodate the business needs to enhance performance and competitiveness in the market (Worrell, 2008). The frequent change in business structures and process, dictate the need for newer functionality and better technology that can enable integration with other systems (Olson and Zhao, 2007; Khoo and Robey, 2007).

Likewise Davenport *et al.* (2004) suggest that the integration of different instances of the ERP systems is an on-going process due to mergers and acquisitions. This creates the need for organisations to consolidate and stabilise their process and systems across the different business units. Strategic upgrade offers a platform where organisation can merge their business process and simplify procedures in order to leverage the capabilities offered by the new ERP version (Olson and Zhao, 2007).

3 ERP UPGRADE DECISIONS

The literature (see Khoo and Robey, 2007; Ng, 2011; Otieno, 2010) outlines a number of reasons that influence the upgrade decision (table 2). However, business needs, improved usability and end of maintenance are found to be more critical reason and have a direct impact on the decision to upgrade. According to Beatty and Williams (2006) the reasons to upgrade can be classified as vendor pull and organisation push. The classification only includes vendors, while leaving Government and consultants aside, hence vendor pull needs redefining.

External pull is associated to all external influences on the decision to upgrade, for example the reliance on vendor for technical support stresses the need to upgrade when vendors withdraw support for the older versions. Organisation push is the internal upgrade drive which could be influenced by a variety of reasons, which also shows the association of the upgrade reasons to the upgrade strategy. From the classification, it is easier to deduce that organisation push is an important factor when making the decision to upgrade.

Table 2: ERP upgrade decisions classification.

Classification	Categorisation	Reason to Upgrade	Upgrade Strategy
External Pull	Vendor dependency	End of maintenance	Technical Upgrade
		Technology enhancements	
	Risk mitigation and compliance	Legal compliance	
		Consultant dependency	
Organisation push (demand)	Business needs	New functionality	Functional Upgrade
		Standardize functionality	
		Improve usability	
	Business policy	Management philosophy	
		Resources availability	
	Strategic direction	System Integration	Strategic Upgrade
		Consolidation of business process	
		Improve communication between suppliers and customers	

3.1 ERP Decisions Support Approaches

Decision support tools have been used previously during ERP software selection to provide mechanisms and strategies to support the complex decision making process involved. The proposed methodologies and frameworks aimed to support the selection decisions incorporating numerous criteria such as business requirements, functionality. For example, Teltumbde (2000) proposes a methodology evaluating ERP selection utilising participatory learning based on nominal group technique and Analytical Hierarchy Process (AHP). Wei *et al.* (2005) proposes ERP system selection grounded on decision maker' tangible and intangible measures that are evaluated by using AHP-based approach. The measures are considered with respect to organisational requirements that are essential for the system selection. Liao *et al.* (2007) proposed a selection model based on 2-tuple linguistic information processing. In that study, the ERP systems information which was represented in different linguistic terms is grouped using a similarity degree algorithm. Karsak and Ozogul (2009) proposes a framework that incorporates quality function deployment, fuzzy linear regression and zero-one goal programming to facilitate ERP software selection decisions. The framework takes into consideration both the system characteristics and the company demands to evaluate their relationships and interactions.

However to date, research on ERP upgrade have not addressed the usefulness and importance of decision support approaches in regards to facilitating ERP upgrading decision. One reason could be that the use of decision support tools is unwarranted as Olson and Zhao (2007) explains that upgrading is regarded to have minimum complications in comparison to the original ERP implementation. Though, Beatty and Williams (2006) suggest that ERP upgrading should be regarded as a new implementation and therefore requires to be justified with a strong business case and supported by business requirements. Hence, there is need to explore methods and tools that can efficiently manage and establish the added-value for upgrading, in the process streamlining the upgrade decision making process.

3.2 Upgrade Decision Support Tool

At some point either due to technological changes or new functionality or withdrawal of vendor support for the older version, ERP clients will have to upgrade their systems. At this stage the decision becomes about the timing of the upgrade and the version to be adopted depending on the resources and support availability. Khoo and Robey (2007) suggest that the availability of resources has an impact on the decision to upgrade, as organisations will prioritise internal resources only when ERP upgrades can no longer be postponed. Therefore, the decision makers are tasked with identifying the best path to be undertaken when upgrading, one argument is that the process can be deduced and fulfilled by common sense. Yet, in order to appreciate the different versions and decide which one to adopt, it is vital to understand the functional enhancements in each new version. It is perceived that system documentation serves as groundwork for upgrades as it provides detailed explanation of the changes in the new systems (Ebersteins and Grabis, 2011). Yet, Zarotsky et al. (2006) points out that

vendor documentation does not extensively outline the enhancements of the new version which results into clients failing to see the added-value for undertaking upgrades.

The literature portrays that decision support approaches have been useful in supporting decision makers during ERP software selection, hence similar methodologies and frameworks have significant potential to streamline the ERP upgrading decisions and highlight the added value. However, for the decision support tool to be effective, it should not only support the management strategies but provide a best possible roadmap for undertaking the upgrade from the technical perspective. Achieving this feature will require the tool to take into account the factors that influence the upgrade decisions and assess the challenges involved during upgrading. The output from this evaluation should provide a detailed explanation of a suitable strategy to be adopted with a detailed process map using non-technical terminology. The decision support tool should incorporate functionality that can gauge the effort and resources required for the upgrade in order to determine the impact of the upgrade from a resource and cost perspective.

The decision support tool can be extended to facilitate objective evaluations of functionality in different versions to identify the functional enhancements and map out the functionality against the business requirement. These evaluations can be achieved either through conducting gap analysis or functional impact analysis of the new version. As a result, it will facilitate identifying the functionality gap and highlight the added value in order to support the upgrading decisions. For example Ng and Gable (2009) propose an upgrade assessment and recommendation report based on fit-gap analysis which evaluates the new functionalities with respect to organisation requirements. The report can be used to asses if previous modification should be applied or not, based on understanding the functionality of the system and in one way influence the decision to upgrade.

4 CONCLUSIONS

In general, a typical ERP life cycle offers little distinction between implementation and post-implementation phases, though, a clear breakdown of these phases can provide opportunity for proposing improved mechanisms that support ERP systems after 'go-live'. ERP upgrade is considered as an important aspect in the system life-cycle and requires significant effort and attention to facilitate continuous business improvement. This paper has given an account of ERP systems upgrading, outlining the upgrade strategies and suggesting possible challenges associated with each upgrade strategy.

Understandably, the decision to upgrade is a complex undertaking as it involves multiple influencing factors and upgrade strategies, which when not planned well may result in disruption to the business. Therefore, despite limited literature from an ERP upgrade perspective on the usefulness of decision support tools, we propose the use of decision support methodologies to help streamline the ERP upgrade decision making process. The upgrade decision support tool will provide a clear strategy which takes into account the project plans, timing, resources and training needs to help the organisation cope with the upgrading process. Further, it will establish the functional enhancements of the new version and the added value for upgrading. Through the use of process maps, it should support the managerial aspects during the decision making as well as provide a roadmap for the upgrade process, highlighting the impact of the upgrade. This outlined features will form the basis of future work through expanding the decision support approach to maximise its usefulness in streamlining the upgrade decision making process.

REFERENCES

Bajwa, D. S., 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*, 44(3), pp.81-90.

Beatty, R. C. and Williams, C. D. (2006) ERP II: Best practices for successfully implementing an ERP upgrade. *Communications of the ACM*, 49(3), pp.105-109.

Brehm, L., Heinzl, A. and Markus, M. L. (2001). Tailoring ERP systems: a spectrum of choices and their implications. *In proceedings of the 34th Hawaii International Conference on System Sciences*. Hawaii, USA: IEEE Computer Society, pp.8017 -8025.

Claybaugh, C. C. (2010) *Timing of Technology Upgrades: A Case of Enterprise Systems.* PhD Thesis, University of Wisconsin - Milwaukee.

Cooper, R. B. and Zmud, R. W. (1990) Information Technology Implementation Research - a Technological Diffusion Approach. *Management Science*, 36(2), pp.123-139.

Davenport, T. H., Harris, J. G. and Cantrell, S. (2004) Enterprise systems and ongoing process change.

Business Process Management Journal, 10, pp.16-26.

Dittrich, Y., Vaucouleur, S. and Giff, S. (2009) ERP Customization as Software Engineering: Knowledge Sharing and Cooperation. *IEEE Software*, 26(6), pp.41-47.

Dor, N., Lev-Ami, T., Litvak, S., Sagiv, M. and Weiss, D. (2008). Customization change impact analysis for ERP professionals via program slicing. *In proceedings of the 2008 international symposium on Software testing and analysis*. Seattle, Washington: ACM Press, pp.97-108.

Ebersteins, M. and Grabis, J. (2011) A Method for Documenting Modifications in ERP Systems. *Scientific Journal of Riga Technical University. Computer Sciences*, 43, pp.57-64.

Esteves, J. and Pastor, J. (1999). An ERP lifecycle-based research agenda. *First International workshop in Enterprise Management and Resource Planning: Methods, Tools and Architectures (EMRPS)*. Venice, Italy: 99, pp.359-371.

Grabski, S. V., Leech, S. A. and Schmidt, P. J. (2011) A Review of ERP Research: A Future Agenda for Accounting Information Systems. *Journal of Information Systems*, 25, pp.37-37.

Hamerman, P. D., Moore, C. and Magarie, A. (2011) ERP Customers Demand Better Flexibility, Cost Transparency, And Mobility [online] Forrester Research. [Accessed 25 November 2012]. Available at: <http://www.forrester.co.uk/rb/Research/trends_2011_erp_customers_demand_better_flexibility%2C/q/id/58390/t/2>.

Hecht, S., Wittges, H. and Krcmar, H. (2011). IT capabilities in ERP maintenance - A review of the ERP post-implementation literature. *In proceedings of the 19th European Conference on Information Systems*. Helsinki, Finland: AIS Electronic Library.

Karsak, E. E. and Ozogul, C. O. (2009) An integrated decision making approach for ERP system selection. *Expert Systems with Applications*, 36(1), pp.660-667.

Khoo, H. M. (2006) *Upgrading Packaged Software: An Exploratory Study of Decisions, Impacts, and Coping Strategies from the Perspectives of Stakeholders.* PhD Thesis, Georgia State University.

Khoo, H. M. and Robey, D. (2007) Deciding to upgrade packaged software: a comparative case study of motives, contingencies and dependencies. *European Journal of Information Systems*, 16(5), pp.555-567.

Law, C. C. H., Chen, C. C. and Wu, B. J. P. (2010) Managing the full ERP life-cycle: Considerations of maintenance and support requirements and IT governance practice as integral elements of the formula for successful ERP adoption. *Computers in Industry*, 61(3), pp.297-308.

Liao, X., Li, Y. and Lu, B. (2007) A model for selecting an ERP system based on linguistic information processing. *Information Systems*, 32(7), pp.1005-1017.

Markus, M. L. and Tanis, C. (2000) The enterprise systems experience-from adoption to success. *in* Zmud, R. W. (ed.) *Framing the domains of IT research: Glimpsing the future through the past.* Cincinnati, OH: Pinnaflex Education Resources, pp.207-173.

Nah, F. F.-H. and Delgado, S. (2006) Critical success factors for enterprise resource planning implementation and upgrade. *Journal of Computer Information Systems*, 46(5), pp.99-113.

Nah, F. F.-H., Faja, S. and Cata, T. (2001) Characteristics of ERP software maintenance: a multiple case study. *Journal of Software Maintenance and Evolution: Research and Practice*, 13(6), pp.399-414.

Ng, C. (2011) Enterprise Resource Planning (ERP) Upgrade Decision: Toward A Unified View. *In proceeding of Pacific Asia Conference on Information Systems, PACIS 2011: Quality Research in Pacific Asia*. Brisbane, Queensland, Australia AIS Electronic library.

Ng, C. S.-P. and Gable, G. G. (2009) Maintaining ERP packaged software – A revelatory case study. *Journal of Information Technology*, 25(1), pp.65-90.

Ng, C. S. P. (2001) A decision framework for enterprise resource planning maintenance and upgrade: A client perspective. *Journal of Software Maintenance and Evolution: Research and Practice*, 13(6), pp.431-468.

Ng, C. S. P., Gable, G. G. and Chan, T. Z. (2002) An ERP-client benefit-oriented maintenance taxonomy. *Journal of Systems and Software*, 64(2), pp.87-109.

Olson, D. L. and Zhao, F. (2007) CIOs' perspectives of critical success factors in ERP upgrade projects. *Enterprise Information Systems*, 1, pp.129-138.

Otieno, J. O. (2010) *Enterprise Resource Planning Systems Implementation and Upgrade.* PhD Thesis, Middlesex University

Pan, S. L., Newell, S., Huang, J. and Galliers, R. D. (2007) Overcoming knowledge management challenges during ERP implementation: The need to integrate and share different types of knowledge: Research Articles. *J. Am. Soc. Inf. Sci. Technol.*, 58(3), pp.404-419.

Riedel, M. (2009) *Managing SAP ERP 6.0 upgrade projects.* Bonn: Galileo Press.

Rothenberger, M. A. and Srite, M. (2009) An Investigation of Customization in ERP System Implementations. *IEEE Transactions on Engineering Management*, 56(4), pp.663-676.

Swanton, B. (2004) *Build ERP upgrade costs into the business change program – not the IT budget* [online]. [Accessed 24 November 2012]. Available at: <http://www.computerweekly.com/opinion/Build-ERP-upgrade-costs-into-the-business-change-programme-not-the-IT-budget >.

Teltumbde, A. (2000) A framework for evaluating ERP projects. *International Journal of Production Research*, 38(17), pp.4507-4520.

Vaucouleur, S. (2009). Customizable and Upgradable Enterprise Systems without the Crystal Ball Assumption. *In proceedings of Enterprise Distributed Object Computing Conference* Auckland, New Zealand: IEEE computer society, pp.203-212.

Wei, C. C., Chien, C. F. and Wang, M. J .J. (2005) An AHP-based approach to ERP system selection. *International Journal of Production Economics*, 96(1),

pp.47-62.
Willis, T. H. and Willis-Brown, A. H. (2002) Extending the value of ERP. *Industrial Management & Data Systems*, 102(1-2), pp.35-38.
Worrell, J. L. (2008) *Running the ERP marathon: ehancing ERP- business fit in the post-implementation phase.* PhD Thesis, Florida State University
Zarotsky, M., Pliskin, N. and Heart, T. (2006) The First ERP Upgrade Project at DSW. *Journal of Cases on Information Technology*, 8, pp.13-23.