Enterprise Systems: The Upgrade Process Model

SCHOLARONE™
Manuscripts

Enterprise Systems: The Upgrade Process Model

Abstract

Purpose – Enterprise Systems (ES) upgrade is a complex undertaking that recurs throughout the systems' life span, therefore, organisations need to adopt strategies and methodologies that can minimize disruptions and risks associated with upgrades. This paper sets-out to explore the processes undertaken during upgrading Enterprise Systems, in order to identify the upgrade project stages.

Design/methodology/approach – This research is grounded in a qualitative survey approach, and utilises a web-based survey questionnaire and semi-structured interviews as methods for data collection. The data was gathered from 41 respondents' and analysed using qualitatively inductive content analysis principles to derive meaning and to identify the trends about upgrade processes.

Findings – The study findings stress the importance of adopting a methodical approach to ES upgrades. In addition, it suggests that due consideration should be given to the impact of new version features and functionality, the risks and the effort required for supporting upgrade projects.

Research limitations/implications – The five-stage upgrade process model can be utilised as a strategy to minimize complexity and risks associated with upgrade projects. However, this study only proposes logical generalisations; therefore, future studies could explore these stages in depth to offer generalisable arguments applicable to ES upgrade phenomenon.

Originality/value – The study proposes a five-stage upgrade process model that offers a systematic approach to support upgrade projects. The proposed model extends previous models by proposing alternative strategies to support enterprise systems upgrade projects.

Keywords - Enterprise Systems, Systems Upgrade, Upgrade, Upgrade Process Model, Enterprise Systems Change, ERP upgrade

Paper type - Research paper

Introduction

As organisations continue to invest in value-added projects, it is reasonable to anticipate that they would upgrade their Enterprise Systems (ES) to take advantage of new processes, functionality and features, which support and streamline their operations (Beatty & Williams, 2006; Otieno, 2010). According to Zhao (2007) upgrade is a continuous process and will normally occur at least once every three years, hence, upgrading can lead to reduction in maintenance costs and re-examination of processes (Khoo & Robey, 2007; Zhao, 2007). Upgrading is defined as a process that aims to expand core system capabilities by improving functionality and taking advantage of new technology features (Vaucouleur, 2009). While Ng (2011) describes upgrading as replacing existing versions entirely or partly with a newer version. The common attribute between these definitions is that upgrade results in functionality and features improvement when compared to the current installed version. Thus, ES upgrade can be viewed as improving the existing system through implementing a newer version that offers additional features, functionality, and enhanced technological capabilities. However, few organisations choose to upgrade their systems despite the benefits of new features and additional functionality such as web-based services. The study by Hamerman *et al.* (2011) exemplifies this argument, as their study results reveal that more than 50% of the participants were utilising systems which are, at least two versions behind the latest release.

Opting not to upgrade has long-term implications such as a lack of continued technical support or obtaining support at a very high price or degradation of systems performance (Feldman *et al.*, 2015). Dempsey *et al.* (2013) attribute the reasons for not upgrading to risks of failure, disruptions, and high costs, for example, Bartels et al. (2014) cited in Morgan & Ngwenyama (2015) suggest that, in the United States alone, upgrading costs account for the major share of US$1.03 trillion spent on information technology (IT) in 2014. Additionally, Khoo & Robey (2007) advise that a new version's functionality and features would have an impact on the existing version, which in turn increases the possibility of disruptions. The impact can be due to either the compatibility of the changes in the new versions compared to the existing version or the new version's reliability and stability. According to Beatty & Williams (2006), these factors normally would consume the majority of the upgrade time, specifically during testing. Therefore, ES upgrade cannot be taken lightly, as these issues require significant attention as far as continuous improvement and benefit-realisation are concerned.

It can be argued that ES and Enterprise Resource Planning (ERP) Systems are different. Since ES are comprehensive, configurable, and integrated suite of systems that support organisation-wide operational and management processes (Davenport, 2000; Xu, 2011; Feldman *et al.*, 2015). Hence, as part of the ES landscape, many organisations implement other systems to either optimise the supply chain (e.g. Supply Chain Management systems), or automate the customer service management (e.g. Customer Relationship Management systems). These systems may be from the same vendor or different vendors, nevertheless these systems require being upgraded continuously (Voulgaris *et al.*, 2014). Although major ES vendors offer strategies, methodologies and best practices to manage and support upgrades, the focus of these strategies and methodologies are vendor-specific. Therefore, this results in upgrade needs not being supported, especially if the organisation has multiple systems from various vendors.

Recent studies have proposed practical guidance for managing upgrade projects. These studies focused on best practise and lessons learned (Beatty & Williams, 2006; Zarotsky *et al.*, 2006; Wenrich & Ahmad,

2009), success factors (Whang *et al.*, 2003; Nah & Delgado, 2006; Olson & Zhao, 2007; Shi & Zhao, 2009; Leyh & Muschick, 2013; Scheckenbach *et al.*, 2014) and business process changes alignment (Cao *et al.*, 2013; Paradonsaree *et al.*, 2014). Several other studies (Khoo, 2006; Claybaugh, 2010; Otieno, 2010; Ng, 2011; Dempsey *et al.*, 2013; Claybaugh *et al.*, 2015; Feldman *et al.*, 2015) have explored factors that influence the decision to upgrade. Although these studies offer valuable insights, there has been limited focus on upgrade processes, with only a few studies such as the maintenance and upgrade model proposed by Ng *et al.* (2003), the upgrade process model proposed by Zhao (2007) and the upgrade assimilation stages proposed by Claybaugh *et al.* (2015). These studies have focused mostly on ERP systems. However, Paradonsaree *et al.* (2014) and Scheckenbach *et al.* (2014) state that research on upgrades specifically on ERP systems is scarce, this argument supports Grabski *et al.* (2011) suggestions that further research is needed to explore the phase after systems 'go-live'. In addition, it is not clear whether these models are applicable to the whole ES landscape as many organisations employ their own methodologies when upgrading their systems to the latest version. Thus, this research aims to contribute towards ES upgrade literature through offering insights to the activities organisations undertake when upgrading their systems, along with offering strategies to support upgrade projects. In this context, this research addresses the following question, what is the approach organisations adopt when upgrading their enterprise systems?

This paper is organised as follows; the second section reviews existing literature on ES upgrade, and discusses existing upgrade process models. The third section explains the study approach. The fourth section presents the findings and proposes the upgrade process model and lastly we discuss the model implications.

ES Upgrade Overview

The significant shift in operating conditions and dynamic market environments over the years has led many organisations adopting enterprise wide information systems as a mechanism to gain competitive advantage and improve performance. Usage of these systems, such as Material Requirements Planning (MRP) and Manufacturing Resources Planning (MRPII), tracks back to early 1960s, specifically in the accounting and manufacturing domains (Umble *et al.*, 2003). MRP and MRPII encouraged integration beyond the realm of manufacturing systems, resulting in cross-functional systems such as ERP systems (Sheilds, 2001). There are many instances where ERP systems and ES have been interchangeably used, implying that ERP systems and ES are synonymous. However, according to Davenport (2000) and Xu (2011), ES and ERP are not identical. In this paper, ES is referred to as a holistic system that incorporates numerous comprehensive systems, offering a range of capabilities to support end-to-end business processes, which enable integration, collaboration, and interactions between systems and departments.

Adoption of ES has facilitated streamlining and automating numerous operations, improving productivity, and facilitating real-time decision-making. However, Cao *et al.* (2013) and Voulgaris *et al.* (2014) suggests that the systems' real value and potential is realised after the 'go-live' stage, normally referred to as post-implementation phase. Upgrading is one of the essential components of the post-implementation phase and ensures the system is stable and operates efficiently (Hecht *et al.*, 2011). However, there is a significant difference between initial implementation and upgrade, as initial implementation will only happen once, whereas upgrades are recurring throughout the system's lifespan.

Secondly, the effort and activities are significantly different, as an upgrade involves extending an existing system and would be completed within six to eight months on average, with the majority of the time spent on planning the upgrade and testing the implemented changes (Zhao, 2007). In addition, changes in the new version standard code necessitate rigorous testing routines to make sure it is compatible with the existing version, a process that requires significant amount of resources and time. Despite these differences, Beatty & Williams (2006) advise treating upgrades as new implementation project. In addition Paradonsaree *et al.* (2014) discuss the importance of considering all the elements from strategy, technology, people and business process change to support upgrade projects. As a result, the same level of preparation and planning as in the initial implementation is required when upgrading, otherwise there is high risk of not achieving the desired outcome and introducing disruptions. Thus, an upgrade project depends on the adoption of efficient strategies to assist in gaining a detailed understanding of the new version's features and their dependencies.

Upgrade Process

In respect of ES upgrade, Beatty & Williams (2006) suggests that the major intention of upgrading is to take advantage of several benefits introduced through a new version such as new functionality. The upgrade process is normally a series of stages and activities undertaken to achieve the successful completion of set project tasks and objectives. Zhao (2007) suggests that adopting a practical process when upgrading increases the chances of achieving success, as it becomes easier to separate different activities into set objectives and offers a better overview of the general upgrade landscape. Such an argument is supported by Nicolaou & Bhattacharya (2006) when emphasising the importance of modular additions (i.e. upgrades and add-ons) which, in-turn, supports performance improvement, continuous growth and competiveness. As a result, Zhao (2007) offers a methodical approach by adopting Carr *et al.'s* (1996) change process model to explain upgrade processes as illustrated in Figure 1.

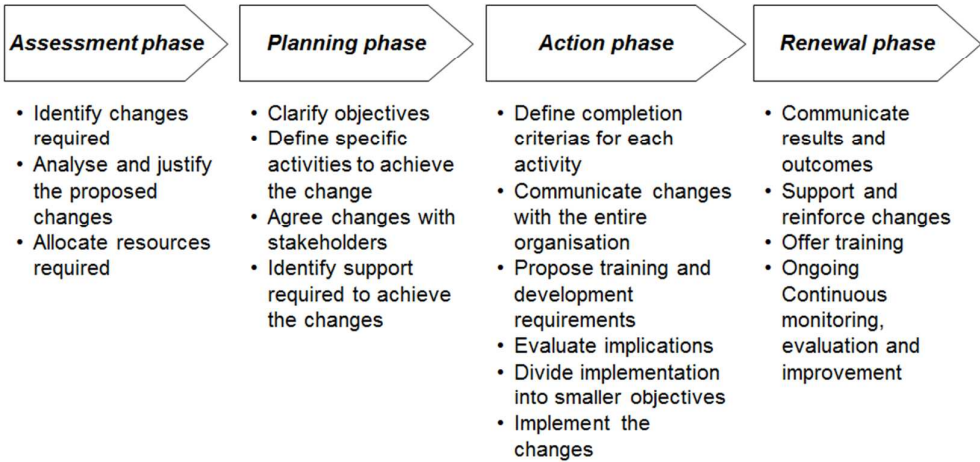| **Assessment phase**	**Planning phase**	**Action phase**	**Renewal phase**
• Identify changes required • Analyse and justify the proposed changes • Allocate resources required	• Clarify objectives • Define specific activities to achieve the change • Agree changes with stakeholders • Identify support required to achieve the changes	• Define completion criterias for each activity • Communicate changes with the entire organisation • Propose training and development requirements • Evaluate implications • Divide implementation into smaller objectives • Implement the changes	• Communicate results and outcomes • Support and reinforce changes • Offer training • Ongoing Continuous monitoring, evaluation and improvement

Figure 1: Upgrade process model (adapted from Zhao (2007))

Ng *et al.* (2003) and Ng & Gable (2009) propose 11 stages for upgrading as part of the maintenance model (Table 1). Stage 1 and 2 are concerned with utilising known facts and strategies in order to select a best option and reduce risks involved with upgrade projects, while stage 3 facilitates building an upgrade case. Stages 4 to 10 ensure there will be less downtime and the system performs as expected when upgraded.

Table 1: Upgrade stages (adapted from Ng *et al.* (2003))

	Stage name	Description
1	Design an upgrade project methodology	Collect information about the new version, tools and services to help define an appropriate upgrade method.
2	Explore available upgrade options	Select an optimal upgrade option by evaluating its stability, reliability, and compatibility against existing versions and comparing organisation objectives.
3	Develop a business case	Identify and justify the need to upgrade based on the different factors influencing the upgrade decision. The business case will define the objectives and goals of the upgrade project.
4	Modification assessment	Examine the existing system modifications and determine modifications that are obsolete or need to be re-implemented.
5	Asses new functionality and technical requirements	Evaluate the organisational requirements against the new version features and determine the technical requirements for introducing such changes.
6	Conduct impact analysis	Analyse the impact of the proposed changes to the organisation, system, and people. This helps allocating appropriate effort and resources for the upgrade project, along with outlining the benefits for upgrading.
7	Implement new version in development environment	Ensure the existing and new systems are up-to-date through applying earlier patches, bug fixes, and enhancements. Then install the new version in a development environment.
8	Construct the new system	Re-apply all the previous modifications and customizations (such as users' interfaces and reporting capabilities).
9	Testing	Verify accuracy of the system functionality and confirm that the proposed changes fulfil user requirements and align to the business objectives.
10	Perform trial upgrades	Undertake a trial upgrade to identify potential errors and challenges during the upgrade process.
11	Go live	Release the new version to the production environment.

Claybaugh *et al.* (2015) propose four stages as part of the upgrade assimilation model Figure 2. The first stage awareness or interest outlines when the organisation becomes aware of the availability of a new version, mostly through formal investigation. The second stage evaluation or commitment highlights that the organisation has accepted that there is a need to upgrade and either conducts an evaluation of the new version or decides not to upgrade their existing version. The third stage indicates that the new version is implemented and is being used as the production system. The fourth stage suggests that the new version has been in use for some time and the organisation is in a state in which it can assess the benefits of the new version.
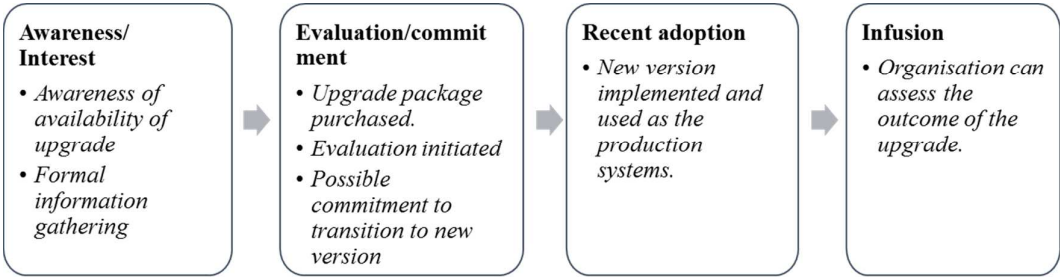
Figure 2: Enterprise software upgrade assimilation stages (adapted from Claybaugh *et al.* (2015))

Synthesis of the literature

The maintenance model and upgrade process model show commonality in their activities, such that some of the stages can be combined, as illustrated in Figure 3. These stages divide an upgrade project into phases and outline useful activities that need to be undertaken to ensure a successful outcome. Although these studies provide useful explanations on the activities involved in upgrade projects, their focus has been on ERP systems only, for example Ng *et al.* (2003) studied an upgrade of SAP R/3 system from version 3.1H to 4.6C. Hence, it is not clear whether similar processes are followed when upgrading other systems within the ES landscape apart from ERP.
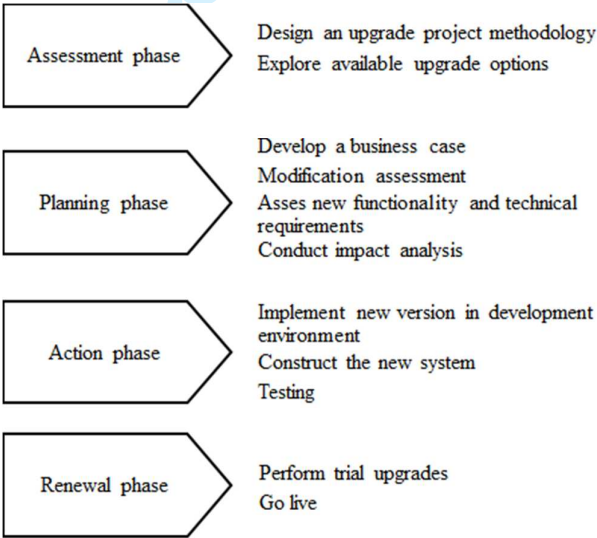

Figure 3: Relationship between Ng *et al.* (2003) and Zhao (2007) upgrade models

On the other hand, the main intention of the assimilation model is to define a mechanism to categorise organisations upgrading their systems in the different stages, therefore, this model does not represent the upgrade process. However, it provides a high-level view of the different states in which an organisation can be with respect to the upgrade project, thus, it can be positioned that such a model could enhance the thinking of the different stages and activities undertaken during upgrade projects. Based on the Enterprise software upgrade assimilation stages, it can be argued that it is important to explore whether there is new version available and to determine if an upgrade is required. While the models proposed by Ng *et al.* (2003) and Zhao (2007) highlight the necessity of assessing the upgrade options and plan for the upgrade, but generally bundle the awareness of the new version availability as part of other stages. Therefore, it can be assumed, the awareness stage will be a critical stage during the upgrade process and should be treated

independently. As organisations become aware of the different versions and understand the need to upgrade during the awareness stage. Thus, based on these three different models, it is conceptually recognised that a typical ES upgrade process would include the following stages as shown in Table 2.

Table 2: Conceptual upgrade stages

Stages	Descriptions
Awareness	- Identify the need to upgrade. - Formally and informally, explore available versions of current system landscape.
Assessment	- Assess the new version features
Planning	- Outline detailed requirements and challenges of the project.
Project definition	- Define the project scope, objectives and deliverables - Select the project and project team
Implementation	- Establish the existing 'as-is' stage - Test the system - Sign-off and migrate to production system
Go-live	- Assess if project objectives are fulfilled. - Provide continuous support and training. - suggest improvement and changes

Additionally, it is reasonable to anticipate that some organisations would develop their own unique approach when upgrading. Thus, this paper investigates the inter-related aspects of ES upgrade in order to understand the stages and activities followed when organisations are upgrading their systems. This will allow identifying whether the ERP upgrade stages proposed in the literature bear any similarity to what organisations adopt when upgrading their entire ES landscape.

Methodology

As the aim of the study is to understand the different upgrade processes, a qualitative approach was considered appropriate as such, an approach provided an opportunity to collect rich descriptive data from the respondents, and establish the meaning and relationship to provide a detailed understanding of the processes. In addition, it allowed the opportunity to explore the dynamics of the processes and gain an in-depth understanding. However, since these views and experiences were associated and subjective to the people involved in upgrade projects, a qualitative approach provide a platform to derive meaning and relationships in order to formulate a better understanding of the processes (Denzin & Lincoln, 2011).

This research adopted a qualitative survey design, outlined in Figure 4. From a methodological perspective, a qualitative survey provides a multidimensional, varied outlook of the subject under investigation (Jansen, 2010; Lindgren & Münch, 2015). According to Fink (2003), qualitative survey design facilitates gathering realistic information, thus utilising such an approach would facilitate gathering respondents' knowledge and experience to describe, compare, and explain ES upgrade. It allows the correlation of information obtained from respondents' in order to establish common or different patterns in ES upgrade processes, thus offering insights into complex issues through examining the current state of affairs associated with ES upgrades projects.
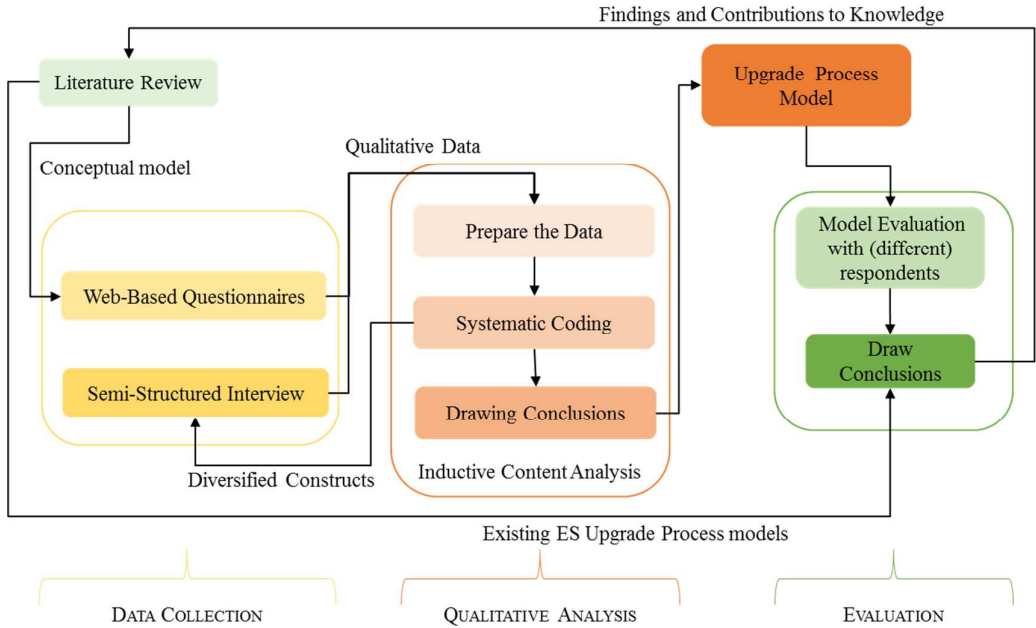
Figure 4: Research Design

Respondent Selection

A typical ES upgrade project team consists of personnel that represent the functional and technical aspects of the system, along with database and systems administrators to offer the diversified expertise needed to support upgrade projects. In addition, the team would include top management and user representatives in order to attain better support for the upgrade from management and end-users. Due to this, the research targeted respondents with these backgrounds in order to capture the different stakeholders' views and get a diverse pool of respondents who have knowledge and experience of undertaking ES upgrades. As part of the purposeful sampling strategy, the respondents targeted for the semi-structured interviews had at least 4 years of experience in managing ES and had been involved in at least one upgrade project to provide insights into the upgrade decision-making process. The cut-off point was based on the explanation by Zhao (2007) who suggests that upgrades are recurring at least once every three years, thus it can be argued that respondents with 4 or more years would theoretically have at least one upgrade project experience.

The respondents were enlisted from SAP and Oracle UK user groups; as both these user groups represented organisations, which either use systems from SAP or Oracle, JD Edwards, PeopleSoft, and Primavera. According to a report from Panorama Consulting Solutions (2013) the following vendors, SAP and Oracle, are the top two vendors whose ES (specifically ERP) are frequently adopted by large organisations, thus they hold the largest market share. Based on the above report, it was understood that members from SAP and Oracle user groups could provide a rich diversified pool of respondents. Secondly, a snowballing technique was used to search for respondents who may not be part of these user groups, as they could offer a different experience of upgrading other systems. The approach involved searching LinkedIn® professional networking services for respondents based on the description provided in their profiles, to ensure they meet the criteria for inclusion. An email was sent out inviting these respondents to

participate in the research. In addition, we politely requested them to forward the message to their contacts with similar experience.

Data Collection

The data collection phase included self-administered web-based questionnaires and semi-structured interviews. The questionnaires' main purpose was to establish experts' attitudes and experiences along with identifying the upgrade processes practiced in their organisations. In order to capture the respondents understanding of the ES upgrade process, the questionnaire logic was adopted from previous studies (Zhao, 2007; Claybaugh, 2010); however, suitable modifications were introduced to make it appropriate for a qualitative survey research approach. The instrument included both open-ended and close-ended questions; the closed-ended questions asked the respondent to indicate their level of agreement or disagreement based on a five point Likert scale and 'yes' or 'no' answer options. The open-ended questions supplemented the closed-ended questions, allowing probing for more details by encouraging the participants to provide descriptive account of their experiences on ES upgrade. The output from this stage helped to identify the activities that were undertaken; thus enabling a high-level view of the upgrade processes to be obtained. Next, semi-structured interviews were conducted to supplement and obtain an in-depth understanding of some of the diversified patterns identified in the previous data collection stage. Semi-structured interviews were used because it offers a flexible approach to explore complex issues, and gain insights based on people's experiences, as concepts could be expressed freely. This normally results in rich detailed information that can provide a clear focus on ES upgrade processes. The interview guide (Table 3) was prepared based on constructs identified after analysing the questionnaire data. The interview guide focused on three main areas that are the decision-making processes and the upgrade process, as these were identified as important and needed further clarification. Based on the interviewee responses, further questions were asked in order to gain a better understanding or clarification to the concepts expressed.

Table 3: Semi-structured interview guide

Interview questions
Your organisation has recently upgraded or is currently upgrading its Enterprise System, could you please describe you involvement in the upgrade process?
How the decision to upgrade was reached (for example was any formal discussion or process followed)?
Was any specific set of activities agreed and followed as part of the upgrade?
What were the reasons for adopting those activities and following that specific order?
In your opinion, when do you think is the right time to evaluate the functionality of the new version and why?
In your opinion, when do you think impact assessment needs to be performed and why?

In order to ensure the findings' quality and rigor, two data collection techniques were used to overcome insufficiencies and prejudices that may arise when using a single method (Creswell, 2009). To supplement the data collection techniques, respondent validation was applied. This means interviewees were provided with an opportunity to verify the contents of their individual interviews for accuracy and if necessary amendments were made to the interview summaries. Additionally, contents that were not clear were posed as additional questions to the other interviewees, to gather their opinions and views.

Qualitative Analysis

Qualitative data analysis adopts a funnel approach, whereby the data is organised as an abstraction, then grouped in order to identify the relationships between the different themes (Creswell, 2009). In order to ensure a coherent interpretation of the concepts is attained, inductive content analysis principles guided the overall data analysis strategy. From a qualitative perspective, an inductive content analysis process helps to identify the thoughts and ideas presented in the data to derive and synthesise the relevant meaning in order to understand the upgrade processes. The advantage is the capability to handle a vast amount of data even from different sources in order to provide substantiating evidence to the research findings (Patton, 2002). As part of an inductive content analysis approach, the following three steps were followed: preparing the data, systematic coding, and drawing conclusions. Preparing the data involved studying the data as a whole to get a broader picture on how it reflected the research question. Systematic coding allowed tagging, separating and grouping the data into meaningful categorizes; thus, deriving the categories inductively from the data and gathering theoretical attributes. Drawing Conclusions involved drawing inferences from the data through exploring the identified themes and their properties in order to stimulate new understandings.

In summary, 29 completed questionnaire responses were received and 12 (30 - 45 minutes) semi-structured interviews were conducted. The web-based questionnaire represented 18 organisations and additional 5 organisations from the semi-structured interviews. It should be noted that although there were 12 respondents for the semi-structured interviews, 6 respondents had taken part in the study through the web-based questionnaire. All the respondents represented 23 large organisations (with 250+ employees) which were currently upgrading, or in the process of upgrading in the next 6-12 months or upgraded their ES systems, in the previous 6 months. These systems are illustrated in Figure 5.

Systems	Count
Enterprise Resource Planning (ERP)	16
Customer Relationship Management (CRM)	6
Supplier Relationship Management (SRM)	5
Supply Chain Management (SCM)	4
Business Intelligence (BI)	9
Human Resources management (HRM)	10
Enterprise Collaboration System (ESC)	2
Integrated Service Management	2
Other	4

Figure 5: List of implemented and upgraded systems

Table 4 and Table 5 highlight the respondents' diverse level of expertise and experience, signifying the richness of the responses to address the research question.

Table 4: Respondents' Roles

Role	Count
Solution Architect	7
Project Manager	10
Systems Analyst	4
Functional Lead	9
Technical Lead	7
Database Administrator	4
Systems Administrator	2
Chief financial controller	1
Database Administrator	1
Information systems manager	1

Table 5: Respondents' experience

Experience	Count
Less than 1 year	0
1 to 2 years	1
2 to 4 years	5
4 to 6 years	4
6 to 8 years	14
More than 8 years	17

Evaluation

In this stage, the finding of the study was presented to another set of respondents that were not involved in the study before, in order to evaluate the interpretations and proposed stages, along with gauging the accuracy, applicability, and significance of the findings in respect to ES upgrades projects. These 10 respondents from 7 different organisations respondents had upgrade experience and knowledge and were involved in at least two ES upgrade projects. In addition, the respondents were asked to indicate if the proposed model was relevant and applicable to all systems within the ES landscape.

Findings and Discussion

Upgrading is a complex undertaking, which is dependent on multiple stakeholders' views and several factors aligning in order to achieve the expected outcome. Mostly upgrades will be undertaken to achieve targeted benefits that support medium to long-term goals, such as lower operational costs, improved performance, and security. However, there are several risks associated with upgrades specifically when different levels of modifications are applied, therefore, organisations attempt to apply the required changes in a measured way in order to reduce risks and disruptions. One of the approaches utilised to achieve a smooth transition from one system version to another is through defining a clear project scope. This includes identifying the workflow actions, deliverables, timing, budget, and resources, along with accounting for possible disruptions points, combined with steps to address these challenges when they occur.

The upgrade project scope is dependable on the upgrade approach that is either a 'big bang' or a 'modular' approach. The big bang approach involves applying all the significant changes at a single instance, resembling a new implementation. This implies that technical and functional aspects of the system will be upgraded at the same instance, which results in a large project scope. As a result, there are a significant number of achievable objectives, consequently increasing the upgrade risks, costs, and demand for resources. Adopting a 'big bang' approach is more viable when there is a need to be consistent with the latest version, especially if the installed version is several releases behind. On the other hand, a modular approach requires a dedicated team, planned budget, and flexible time to complete the project, since it is undertaken in batches. The modular approach generally implies either focusing purely on technology aspects of the system or extending the functionality and optimising existing processes. Alternatively, since most ES have the capability of modular configuration, the phases are defined depending on modules to be

upgraded. This suggests the modular approach would simply mean configuring different modules; however, this is not a common practice.

Another approach used to minimize disruptions and risks involves assessing and understanding how these changes affect existing processes and ways of operating. The findings highlight existing system evaluation as an important aspect that establishes the 'as-is' systems' operational state. This includes technically evaluating pre-existing functionality to confirm that it would not be disturbed or removed and establishes the level of modification required. Then the improvements offered in the new version would be evaluated to determine how it influences the current landscape by comparing and assessing the introduced changes against the requirements. In addition, the impact of these changes can be measured by comparing the 'as-is' stage against the proposed changes, to identify strategies which can address the challenges that may arise. All these activities are normally undertaken within a specific timeline during the upgrade projects, as shown in the proposed upgrade process model.

Proposed Upgrade Process Model

To achieve such detailed and accurate information, it is important to adopt a systematic approach that provides guidance to assess and evaluate their systems landscape. This paper propose a five stage upgrade process model (Figure 6), which offers a systematic approach to upgrade and provides useful critical outputs to support upgrade projects.
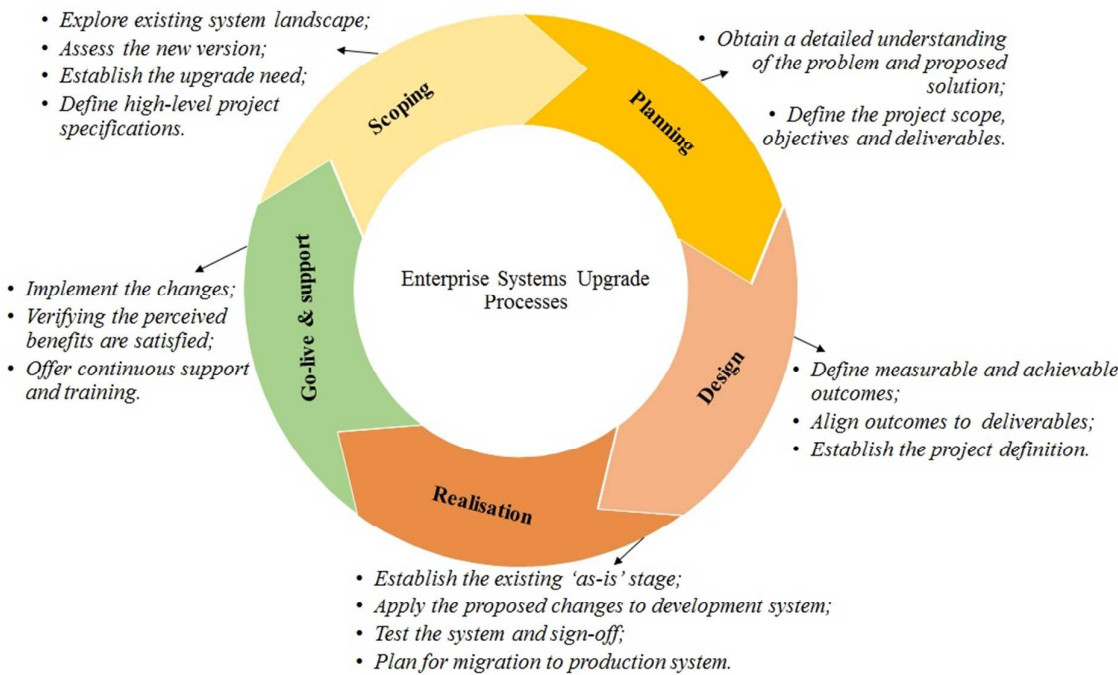

- *Explore existing system landscape;*
- *Assess the new version;*
- *Establish the upgrade need;*
- *Define high-level project specifications.*

Scoping

Planning
- *Obtain a detailed understanding of the problem and proposed solution;*
- *Define the project scope, objectives and deliverables.*

Go-live & support
- *Implement the changes;*
- *Verifying the perceived benefits are satisfied;*
- *Offer continuous support and training.*

Enterprise Systems Upgrade Processes

Design
- *Define measurable and achievable outcomes;*
- *Align outcomes to deliverables;*
- *Establish the project definition.*

Realisation
- *Establish the existing 'as-is' stage;*
- *Apply the proposed changes to development system;*
- *Test the system and sign-off;*
- *Plan for migration to production system.*

Figure 6: ES upgrade process model

Scoping

The scoping stage is concerned with finding possible problems and identifying what will potentially provide better outcomes to improve existing systems. This requires collaborating and communicating with external and internal stakeholders to gain insights of what is required. Typically scoping happens prior to upgrade project commencement, as it is vital to determine if there is a need to upgrade. It involves

exploring the existing system landscape to establish new requirements and understand any challenges within the system landscape such as licensing issues, in order to justify the need to upgrade. Next, the functional landscape is assessed, to identify any necessary improvements required for supporting the business needs. As a result, the scoping stage formulates the foundation for exploring the new version functionalities, in order to identify if it can support the organization's need. In many cases, the new version incorporates functionality surplus to requirements, therefore, reviewing the enhancements offered, providing an understanding of the significant features, and prioritises functionality for implementation. Usually, a two-step process is followed when assessing the new version functionality. First, a formal approach that involves searching vendors' website to get a broad understanding of the proposed changes that will be introduced in the new version. This is followed by scheduling meetings with vendor's representatives to get information of the new version enhancements. Secondly, informal approaches, through gathering information from colleagues and other companies with respect to the new version features and stability. The output from the scoping stage is a relatively high-level project specification incorporating changes that are to be introduced as part of organisations' continuity strategy.

Planning

Planning involves obtaining a detailed understanding of the problem and proposed solution, including its value proposition. Generally, the planning stage inherits information from the scoping stage, to facilitate identifying detailed requirements and project challenges. Planning begins by selecting a project coordinator, who will be responsible for assembling the project team and initiating existing system assessment. Next, the project scope will be defined, providing the project breadth, based on reviewing existing processes. As a result, a better understanding of existing processes usage is obtained, along with identifying redundant and obsolete processes. In return, it provides a baseline for establishing detailed requirements to be compared against new versions functionalities. Although one of the methods used to get new version functionality information is through system documentation, it does not effectively outline the additional features and it is subjective and limited; thus, supporting Zarotsky *et al.'s* (2006) view that vendor documentation is not an effective mechanism for gaining in depth information about the new version. To supplement such limitations, consultants are invited to provide objective evaluations and in-depth explanation of additional features available in the new version. The planning stage is not only about reviewing functionality, as the main aim is to establish the scope of the project. The output from this stage is a project initiation document (PID) containing various information about the project, such as team compositions, objectives and deliverables of the project.

Design

The concern in the design stage is to define measurable and achievable outcomes and align these outcomes to specific objectives, therefore, it involves interrogating the information presented in the PID to ask questions about the problem and options presented in order to identify current and future value propositions offered by upgrading. In addition, it involves evaluating and mapping new version functionality against requirements identified in order to establish a clear vision of how well the new version satisfies the required features. Further, it establishes outdated modifications and determines what alterations are required for existing modifications and functionality, along with determining the effort required to support testing and

re-application of these modifications. In situations when a new version does not support the 'must have' requirements, then there is a possibility to consider introducing new modifications as part of the design phase, thus, supporting Beatty & Williams' (2006) suggestion that, in order to take full advantage of upgrades and achieve the project scope and objectives, it is vital to assess the complete IS infrastructure.

In addition, upgrade impact assessment is undertaken, by performing volume testing and sizing of the existing version. These two tests are fundamental to determining the impact on hardware and supporting technologies (such as database and operating system). It is also important to measure the impact of upgrade on the process and rules, in order to prepare measures for overcoming any risks and assuring no hidden issues arise that can force the project to roll back. Considering these implications allows for allocation of sufficient funding and resources to support the upgrade project. This stage provides a clear and detailed project definition, including upgrade approaches and testing strategies. Additionally it will also define criteria used to determine cut-off points that will indicate when the replication of the changes to the production environment can begin. The cornerstone of upgrade projects is a project blueprint that contains details about the project, to ensure no questions are left unanswered. Additionally it acts as a manual, which the upgrade team can constantly refer too, ensuring all agreed plans and outlined strategies are followed.

Realisation

This stage resembles an 'implementation rehearsal', which determines when the systems 'go-live' based on the cut-off point. The sole purpose is to ensure that everything is functioning as required and minimises the possibility of errors before the changes are applied to the production environment. Realisation is considered the most important phase in upgrade projects and consumes the bulk of the upgrade project time. Such requirements are very noticeable in large projects that deliver across diverse scopes, thus, becoming dependable on resource availability. This argument is supported by Khoo & Robey (2007) who argue that resources are critical when planning upgrades, especially when the projects are deemed critical and unavoidable.

In general, when upgrading, three environments (system boxes) are utilised, these are development, quality assurance and production to minimise risks of failure and increase chances of success. Realisation begins by replicating the existing landscape into the development box, followed by executing different testing scenarios defined in the blue print, to establish the 'as-is' stage. Next, the proposed changes, any new or existing modifications, and user screens are applied to the development box, followed by execution of testing scenarios again to establish the 'to-be' outputs. Similarly, Beatty & Williams (2006) mention that adopting different testing approaches as one of the critical success factors when performing upgrades. Not surprisingly, testing is a main activity and several different testing strategies are utilised to ensure the changes are practically correct, in order to support smooth system operation. When the system is stable, changes are replicated to the quality assurance box, which contains meaningful data. This provides an opportunity for performing 'sanity' tests aiming at evaluating that everything is in line with expectations and validates integration and information flow between existing and new processes. Next 'safe check' tests will be undertaken, to determine if the upgrade fulfils the 'go or no go' criterias and cut-off points, to determine if the changes could be migrated to the production environment.

Go Live and Support

In this stage the focus is about migrating and implementing proposed changes to the production system box, these occur after all system checks are successful and the project is signed-off. The following instructions should be given a due consideration when migrating. First, migration should be planned when there is sufficient time and the production system is least utilised, in order to minimise risks of disruptions and reduce conflicts occurring in the underlying system mechanics. Secondly, a hold should be placed on all changes to data, including execution of all previous version transactions. Not adhering to these instructions when upgrading may increase the time required to apply the changes and result in system inconsistency. The next component is continuous improvement that involves assessing that all the tasks defined in the blueprint are achieved, the perceived benefits are verified, and objectives are fulfilled. Additionally, it includes addressing known issues and bugs, along with acclimatizing users through continuous training, which supports identification of new requirements, thus creating the groundwork for the next upgrade cycle.

Synthesis of the Findings

The study findings support our assumption that it is important to explore the need for upgrade and establish if there are new versions. Similar to the assimilation stages proposed by Claybaugh *et al.* (2015), the exploration of this information is done through formal approaches. In addition, the findings indicated that organisations also utilise informal methods to gather information about the new versions especially in terms of its stability and reliability. Therefore, it can be suggested that the proposed model extends the models proposed by Ng *et al.* (2003) and Zhao (2007), by introducing the scoping stage as part of the upgrade process model. In addition, the proposed model introduces alternative activities within each stage that, when adopted by organisations, would help to reduce the risks and minimize disruptions associated with the upgrade projects. Moreover, the proposed upgrade process model signifies the importance of exploring, collaborating, and communicating with all the stakeholders involved with the upgrade projects in order to gain insights into the requirements that will support the organisation's operating needs. This is an important aspect in any upgrade project, as it will allow appropriate planning and defining the objectives of the project.

Based on the findings from the evaluation of the model with respondents, it can be suggested that the proposed model offers stages and activities that can be utilised within the entire ES landscape. Thus, it can be argued that the proposed model is flexible and adaptable to accommodate the different systems that organisations plan to upgrade. Additionally, it is necessary to understand how the new version affects existing modifications and functionality, since it provides complete details of the upgrade tasks. This concurs with arguments raised by Whang *et al.* (2003) that upgrades require a well thought plan with clear objectives and goals that need to be achieved when upgrading the systems. In addition, it includes researching the proposed new version's capabilities either through using external consultants and vendors' information outlets for example their website or information from companies that have recently upgraded. Thus the findings presented through the five-stages of ES upgrade model proposed in this research corresponds to the suggestions by Zhao (2007) that it is important to understand when and why organisations need to upgrade their systems. In addition it supports the arguments raised by Paradonsaree *et*

al. (2014) that upgrade projects need to be supported with strong strategies and different stakeholders. Similar to the earlier upgrade process stages, the proposed model places emphasis on approaching an upgrade project through clear defined stages, that can guide the project step by step to ensure the risks are minimised and project objectives are achieved. Thus, the combination of these findings provides support for a theoretical proposition that upgrade projects would benefit from adopting a coherent and methodical approach.

   Additionally, our findings suggested that most organisations prefer to undertake modular upgrade as it allows them to define realistic and achievable objectives. These align to the suggestion by Nicolaou & Bhattacharya (2006) that modular changes could result in improvements to the installed systems, this could be functional or performance, therefore, it is important to understand the upgrade approach, as this has an impact on the project scope, resources, and objectives. In addition, defining the upgrade approach ensures that the upgrade makes business sense for top management to buy-in. In this paper, maintenance and upgrade are regarded to be uniquely distinct activities and the proposed model only concentrates on upgrade activities, as this was the view that most organisations considered most feasible. Such a view supports the arguments raised Botta-Genoulaz *et al.* (2005) and Motiwalla & Thompson (2009) that upgrade and maintenance being different activities. In addition such an argument differs from the views that ES maintenance and upgrades should be referred to as upgrades (Paradonsaree *et al.*, 2014) or upgrades should be treated as part of the maintenance process (Hecht *et al.*, 2011).

Conclusions

This paper has given an account of the different activities and processes performed during an ES upgrade. The paper draws from upgrade experiences, challenges, and approach of 23 organisations and proposes a five-stage upgrade process model. The proposed model underscores the importance of evaluating the impact of the proposed changes and estimating the effort required to support upgrade projects, in addition to comparing and establishing the new version features and functionality.

This study is significant for several reasons. First from a theoretical standpoint, it supplements and extends previous research on ES upgrades by demonstrating the applicability of the earlier proposed upgrade process models to all systems within the ES landscape. The study provides insights into how organisations approach upgrade projects when upgrading their systems. In addition, it highlights the importance of adopting a systematic approach when upgrading, which is significant, given the frequency of vendors' new version release cycle. Second from the organisation perspective, this study provides an in-depth understanding on how other organisations approach upgrade projects and provides a detailed account of upgrade experiences, understandings, and perspectives from various respondents. One key observation is that upgrades have to be supported by the different stakeholders involved in the project, as it is the combined expertise and effort that would make the project a success. Therefore, following the steps outlined in the proposed process model would increase the possibility of achieving the required deliverables and objectives for undertaking an upgrade, along with ensuring the systems operate as expected after upgrading.

Despite the relatively small group of respondents involved in this research, it was still possible to discover details relating to different aspects of upgrade projects. However, the findings reflect views from mostly

large organisations and only logical generalisations are proposed, therefore, the proposed model could be regarded as context sensitive. While this study is one of the few studies that have explored ES upgrade processes, similar to any qualitative research, further efforts to expand and extend these findings are required. Future research could adopt an alternative methodological approach to offer a broader understanding of the upgrade process and the applicability of the proposed upgrade model to a wide-range of ES upgrade phenomenon and provide probabilistic generalisation. Additionally, the manner in which organisations approach ES upgrade could possibly evolve over time. Thus, in order to extend the proposed model, future research could opt to undertake a longitudinal study to provide an extensive perspective of upgrade processes. Alternatively, the proposed model can be utilised in upgrade projects to confirm and establish any similar or conflicting arguments that can be used to produce outcomes that are more generalisable. In addition, upgrade projects are a continuous process, which involves different stakeholders that are driven by different agendas, thus future studies could apply change management concepts to explore the full upgrade cycle in order to provide a detailed understanding of the dynamic nature of ES upgrade and its interactions, from people, process, and technology aspects. As part of our findings there have been suggestions that upgrading could lead to competitive advantage, though there were no indication on how organisation actually achieve this competitive advantage. Therefore, it would be interesting to understand how organisations gain competitive advantage when opting to upgrade their systems through exploring the relationship between upgrading and competitive advantage. The outcome of such study could help organisations recognise the advantage of upgrading their systems.

References

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.

Botta-Genoulaz, V., Millet, R.A. and Grabot, B. (2005) A survey on the recent research literature on ERP systems. *Computers in Industry*, **56**(6), pp.510-522.

Cao, J., Nicolaou, A.I. and Bhattacharya, S. (2013) A Longitudinal Examination of Enterprise Resource Planning System Post-Implementation Enhancements. *Journal of Information Systems*, **27**(1), pp.13-39.

Carr, D.K., Hard, K.J. and Trahant, W.J. (1996) *Managing the change process: a field book for change agents, consultants, team leaders, and reengineering managers.* New York: McGraw-Hill, p.248.

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

Claybaugh, C.C., Ramamurthy, K. and Haseman, W.D. (2015) Assimilation of enterprise technology upgrades: a factor-based study. *Enterprise Information Systems*, pp.1-34.

Creswell, J. (2009) *Research design : qualitative, quantitative, and mixed methods approaches.* 3rd ed. Los Angeles: SAGE Publications.

Davenport, T.H. (2000) *Mission critical: realizing the promise of enterprise systems.* Boston, MA: Harvard Business School Press.

Dempsey, S., Vance, R. and Sheehan, L. (2013) Justification of An Upgrade of An Enterprise Resource Planning (ERP) System – The Accountant's Role *Global Journal of Human Social Science*, **13**(1).

Denzin, N.K. and Lincoln, Y.S. (2011) Introduction: The Discipline and Practise of Qualitative Research. *in* Denzin, N. K. and Lincoln, Y. S. (eds.) *The Sage handbook of qualitative research.* 4th ed. Thousand Oaks: SAGE Publications.

Feldman, G., Shah, H., Chapman, C. and Amini, A. (2015). Enterprise Systems Upgrade Drivers: A Technological, Organisational and Environmental Perspective. *In the Proceedings of the 19th Pacific Asia Conference on Information Systems.* Singapore: paper 83: AIS Electronic Library (AISeL).

Fink, A. (2003) *The survey kit.* 2nd ed. Thousand Oaks, California: SAGE Publications.

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* Forrester Research.

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

Jansen, H. (2010) The Logic of Qualitative Survey Research and its Position in the Field of Social Research Methods. *Forum Qualitative Sozialforschung / Forum: Qualitative Social Research*, **11**(2).

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.

Leyh, C. and Muschick, P. (2013). Critical Success Factors for ERP system upgrades – The Case of a German large-scale Enterprise. *In the proceedings of the 19th Americas Conference on Information Systems*. Chicago, USA: AIS Electronic Library.

Lindgren, E. and Münch, J. (2015) Software Development as an Experiment System: A Qualitative Survey on the State of the Practice. *in Agile Processes, in Software Engineering, and Extreme Programming*. Springer, pp.117-128.

Morgan, H.M. and Ngwenyama, O. (2015) Real options, learning cost and timing software upgrades: Towards an integrative model for enterprise software upgrade decision analysis. *International Journal of Production Economics*, **168**, pp.211-223.

Motiwalla, L.F. and Thompson, J. (2009) *Enterprise systems for management.* NJ: Pearson Prentice Hall, p.332.

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.

Ng, C.S.-P. (2011). Enterprise Resource Planning (ERP) Upgrade Decision: Toward A Unified View. *In the proceedings of the 7th Pacific Asia Conference on Information Systems*. Adelaide, Australia: AIS Electronic library, pp.1039–1054.

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., Gable, G.G. and Chan, T. (2003). An ERP maintenance model. *In the proceedings of the 36th Annual Hawaii International Conference on System Sciences*. Hawaii, USA: 8: IEEE Computer Society, p.234.

Nicolaou, A.I. and Bhattacharya, S. (2006) Organizational performance effects of ERP systems usage: The impact of post-implementation changes. *International Journal of Accounting Information Systems*, **7**(1), pp.18-35.

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

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

Panorama Consulting Solutions (2013) *2013 ERP REPORT- A Panorama Consulting Solutions Research Report*.

Paradonsaree, R., Singh, M. and Gekara, V. (2014). Business Process Changes for ERP Upgrades: Impact on organizational capabilities and improvements *In the proceedings of the 35th International Conference on Information Systems*. Auckland, New Zeland: http://aisel.aisnet.org/.

Patton, M.Q. (2002) *Qualitative research and evaluation methods.* 3rd ed. Thousand Oaks, Calif: SAGE Publications.

Scheckenbach, T., Zhao, F., Allard, E., Burke, J., Chiwaki, K. and Marlow, S. (2014) Issues of ERP Upgrade in Public Sectors: A Case Study. *in* Kurosu, M. (ed.) *Human-Computer Interaction. Applications and Services.* Springer International Publishing, pp.754-763.

Sheilds, M.G. (2001) *E–Business and ERP: Rapid Implementation and Project Planning.* NY: John Wiley & Sons Inc., p.9.

Shi, H. and Zhao, F. (2009) Exploring Critical Factors (CFs) of ERP Post-Implementations. *Pacific Science Review*, **11**(2), pp.203-209.

Umble, E.J., Haft, R.R. and Umble, M.M. (2003) Enterprise resource planning: Implementation procedures and critical success factors. *European Journal of Operational Research*, **146**(2), pp.241-257.

Vaucouleur, S. (2009). Customizable and Upgradable Enterprise Systems without the Crystal Ball Assumption. *In the proceedings of the 13 International EDOC Enterprise Computing Conference*. Auckland, New Zealand: IEEE computer society, pp.203-212.

Voulgaris, F., Lemonakis, C. and Papoutsakis, M. (2014) The impact of ERP systems on firm performance: the case of Greek enterprises. *Global Business and Economics Review*, **17**(1), pp.112-129.

Wenrich, K. and Ahmad, N. (2009) Lessons Learned During a Decade of ERP Experience: A Case Study. *International Journal of Enterprise Information Systems*, **5**(1), pp.55-73.

Whang, J., Lee, M. and Kim, K. (2003). A Case Study on the Successful Upgrade of ERP System. *In the proceedings of 7th Pacific Asia Conference on Information Systems*. Adelaide, Australia: AIS Electronic library, pp.1030-1038.

Xu, L.D. (2011) Enterprise Systems: State-of-the-Art and Future Trends. *IEEE Transactions on Industrial Informatics*, **7**(4), pp.630-640.

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.

Zhao, F. (2007) *An empirical study of enterprise system upgrades.* PhD Thesis, University of Nebraska.

(Figure 6: ES upgrade process model)

Enterprise Systems Upgrade Processes

Planning
- *Obtain a detailed understanding of the problem and proposed solution;*
- *Define the project scope, objectives and deliverables.*

Design
- *Define measurable and achievable outcomes;*
- *Align outcomes to deliverables;*
- *Establish the project definition.*

Realisation
- *Establish the existing 'as-is' stage;*
- *Apply the proposed changes to development system;*
- *Test the system and sign-off;*
- *Plan for migration to production system.*

Go-live & support
- *Implement the changes;*
- *Verifying the perceived benefits are satisfied;*
- *Offer continuous support and training.*

Scoping
- *Explore existing system landscape;*
- *Assess the new version;*
- *Establish the upgrade need;*
- *Define high-level project specifications.*