Enterprise Information Systems Considerations in the context of Organizations

Author:

Malihe Tabatabaie

Supervisors:

Chris Kimble and Richard Paige

Qualifying Dissertation

University of York

Department of Computer Science

June 2008

Abstract

Around the globe companies are quietly and steadily becoming more connected. Therefore, there is an increasing demand for IT systems that can support large interconnected organizations. One of the main requirements of such a system is the ability to support these connections.

On the other hand, business changes due to the market events forces organizations to be flexible. Current solutions are mainly IT oriented and use middleware to add new required subsystems to the main system. Middleware is a useful solution; however, it causes it own complexity and cost to the final system. Hence, in the cases that solutions just rely on middleware, the gap between the expanding system and organization still exists.

There is a tension about how IT systems are currently built to support large interconnected organizations. In addition, the need for flexibility addresses new business requirements and opportunities. This dissertation will explore how we can address this tension and allow more flexible IT system to be built for this domain by understanding organization, goals, and modelling techniques.

Impact of the work

Revolutionary theories of how to design the structure of a system propose various solutions to the problems of IT-oriented systems. This type of system plays an essential role in supplying the required processes of an organization. Real world organizations use a variety of approaches for building ES and some do not seem to apply any organizational theory; for example in the interview that I had with IBM consultant, he described a number of projects that organizational theory played a very little role in development of system. Hence, the main focus of this work at the moment is understanding the domain and objectives for this type of system. By defining that, we aim to achieve a better design for the enterprise systems that can satisfy requirements of organizations.

List of Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Introduction

The role of information systems is undeniable in the context of modern industry and business. Many companies benefit from this technology in various ways. Some organizations use a simple version of this technology, such as email, for communicating; but others use this technology for complex processes such as banks. Furthermore, there are industries or businesses, such as Google, that bases their business processes on this technology.

The main focus of this study is about the type of information system that support large organizations and their requirements. Throughout this dissertation, these types of system will be called Enterprise Systems. Enterprise Systems could be a solution to some of the challenges of large-scale organizations and their processes; challenges such as communication between main organization and its partners or data integration within business processes during long life cycle. In this review of the literature, various case studies from industry were extracted that illustrates some the problems in IT-oriented systems that were mentioned earlier. The main reason for problems in IT-oriented systems, according to this survey, is the need to apply changes to IT-oriented system based on the changes in business processes, which is as the result of new requirements in a market or difficulties in developing hidden requirements. The following section presents an example of these cases, BMW, a large-scale automobile and motorcycle manufacturer. This case and similar cases highlight demand for more ideas about suitable systems to link IT and business.

1.2 Motivation

"The BMW Group is reputed to be the only manufacturer of automobiles and motorcycles worldwide that concentrates single-mindedly on premium manufacturing standards on outstanding quality for all its brands. Its brands include BMW, MINI, Rolls-Royce Motor Cars, and BMW Motorcycles. The BMW manufacturing plant in South Carolina is part of the BMW group's global manufacturing network and is exclusive manufacturing plant for the company's hot-selling Z3 Roadster and X5 sports Activity Vehicles" (Stair & Reynolds, 2006).

During the time that BMW was producing Z3 and X5, there was a great demand for these products. The amount of request was more than what BMW could respond to. As the result, BMW faced serious challenges. BMW's decision makers decided to increase the amount of their products to fulfil the market request. Although, the decision was applied without considering the abilities of the suppliers. Some suppliers were not able to produce the required parts in the time that was needed. Some parts took longer to be produced. All these reasons and other difficulties led to large amounts of excess parts in hand, and an unreliable delivery schedule for BMW, which was not good for their international reputation. In such a case, it was required to store the incoming parts and keep them until the rest of the parts arrived for the assembly line. BMW's IT system, at the time did not anticipate or handle such a problem. According to BMW's team(Pohle, 2005), "the old system was wasteful and worrisome" (Stair & Reynolds, 2006), therefore, the IT team at BMW sat down with the result of their investigation to establish objectives for a new and improved system.

The main feature of the ideal system was a system that included a tight supplier network and provided the ability to keep parts coming to the assembly line just in time. In addition, the main challenge that can be seen is dealing with hundreds of different suppliers, which are distributed in various geographical places; in particular, the aim was to provide a system that can tie the entire BMW organization and its partners together to respond to the market in the efficient manner.

After contacting a number of software vendors to find the suitable software system for their problem, SAP provided a solution. "Software giant SAP offered a custom

designed automotive procurement solution called mySAP Automotive. mySAP Automotive overcame the challenge of creating schedules for each of BMW's suppliers by automatically generating custom schedules for each part; the system generated schedules to match BMW's assembly line planning and sequencing directives" (Stair & Reynolds, 2006). According to this report, this new system not only solved the existing problem in BMW but also it gave other opportunities to the company that they had not previously wondered, and were helpful for their business.

"mySAP automotive helps us reduce order-to-delivery time, strengthens our supply chain activities in the areas of demand planning and tracking and tracing of material deliveries, and improves inventory accuracy across our Spartanburg plant-enabling us to significantly reduce time-to-customer for our popular X5 and Z3 models" (Stair & Reynolds, 2006).

The lesson learned from this case study declares that, in some organizations systems are not flexible enough to tolerate changes in the market. As the result, a company can face serious and risky challenges. Based on the requests of the market, new requirements for the system will be created. It is hard to have a system that can satisfy all the requirements and have a solution for new requirements too, therefore, the system should be develop in the manner that does not cause problem for the organization in case that it cannot satisfy new requirements, in addition to its ability to be improved during the time. In the BMW case study, the mySAP automotive system helped BMW to overcome its business challenges and provided them with other useful features. These types of cases, which occur very often in the business world, are the main motivation for this research. However, we want to understand how an organization works in order to help build and integrate IT systems in organizations.

1.3 Dissertation Structure

This dissertation is structured as follow: after presenting the motivation of this work in introduction chapter, chapter two, three, and four will continue with presenting the review on literature of Enterprise Information System (EIS). The review considers various aspects of EIS such as its history, definition, features, benefits, and challenges of

EIS. To clarify the arguments about the state of art definitions some side elements such as business model and organizations will be described too.

Chapter three will continue discussing another aspect of EIS which is its stakeholders and some of the well-known technologies that are currently used in this area. Also as the result of interviews and research in the literature two of the current approaches for developing EIS will be covered in this chapter.

To continue the path of introducing different aspects of EIS, chapter four will present the result of literature review on EIS architecture. Main quality attributes for EIS and the well-known approach for developing business applications, Service Oriented Architecture, will be discussed in this chapter. In additions, to cover one of the essences of developing any type of system, which is collecting its requirement, requirement of EIS will be covered in this chapter.

Chapter five will summarize the review on the literature and it will address the current gap which is understood from this review. The research question will be discussed in this chapter to open the gate for discussing the research proposal for addressing the research question.

Chapter six will present the research approach and methodologies that could be used to answer the question raised. Finally, chapter seven will contain the preliminary results that were achieved during literature review phase.

Chapter 2

Enterprise Information Systems Defined

2.1 Introduction

The aim of this chapter is to provide an overview of what Enterprise Information Systems (EIS) are and a working definition of an EIS that could be used throughout the rest of this dissertation. To achieve this goal, the history of enterprises is reviewed and subsequently in the state of the art definitions, various definitions are compared. Indeed the domain of each definition and its background is considered. Organizational and business models also play an essential role in defining EIS; therefore, this section covers a brief description of these terms too. After defining the working definition for this dissertation, the main challenges and benefits for these systems are addressed.

2.2 History

"The enterprise system appeared around the turn of the twentieth century when the factory system was effectively joined with a managerial hierarchy in production and distribution. It is the emerging coordination of previously independent organizations for production, management, and distribution-shop-floor, front office, and sales office- that generates the organizational innovation known as the Japanese enterprise system" (Fruin, 1992).

According to this book -(Fruin, 1992)- an Enterprise System (ES) was established after First World War, when new industries came to market and many industries joined together. Three types of enterprise were introduced: National, Urban and Rural, which have some common elements such as:

- Inter-firm relations

- Marketing
- Mode of competition
- Finance
- Ownership
- Management
- Administrative coordination
- Government relations

Mitsubishi is given as an example of an enterprise from 1926, considering all the affiliated companies such as Mitsubishi Heavy Industry, Mitsubishi Warehousing, Mitsubishi trading, Mitsubishi Mining, Mitsubishi Bank, Mitsubishi Electric, Mitsubishi trust , Mitsubishi property, Mitsubishi steel, Mitsubishi Oil, Nippon Industrial Chemicals, and Mitsubishi Insurance (Fruin, 1992). There are many other examples of enterprises all around the world. Some famous names in this area are BOEING, General Electric, Kodak, IBM, Norwich Union, Samsung, Philips etc. These companies are mainly famous for some particular products from average users' point of view such as car in Mitsubishi case; consequently, there is a belief that these companies just involve in one particular business, which is not correct. The legal name for these multi business organizations is 'Conglomerate'. Another example in this area is General Electric that involves in businesses such as healthcare, aviation, oil and gas, energy, electrical distribution, security, etc (GeneralElectric, 2008).

EIS have existed from the turn of the twentieth century; therefore, the literature contains various definitions for EIS. The next section will review the state of the art definitions.

2.3 State of the Art Definitions

Enterprise is "an organization created for business ventures [hence] growing enterprise must have a bold leader"(WordNet, 2007). The preceding quotation implies the complexity and a massive amount of work that is required for implementing an Enterprise Information System (EIS). Large companies continue to find that they need system that span their entire organization and tie everything together. As a result, an

understanding of enterprise systems is critical to succeed in today's competitive and ever changing world (Jessup & Valacich, 2006).

Another interesting definition in this area is as follows (Laudon & Laudon, 2007): "Enterprise systems integrate the key business processes of a firm into single software system so that information can flow seamlessly through the organization, improving coordination, efficiency, and decision making. Enterprise software is based on a suite of integrated software modules and a common central database. The database collects data from and feeds the data into numerous applications that can support nearly all of an organization's internal business activities. When new information is entered by one process, the information is made available immediately to other business processes. Organization implementing enterprise software would have to adopt the business processes embedded in the software and, if necessary, change their business processes to conform to those in the software. Enterprise systems support organizational centralization by enforcing uniform data standards and business processes throughout the company and a single unified technology platform". However, the definition that is considered for this dissertation is open to be improved when new suitable information is collected in future. For example, in the first case that, it mentions, "When new information is entered by one process, the information is made available immediately to other business processes" it can be argued that the information should be available to the other processes according to their access domain. By this I mean, the level of access to the information is different from process to process. It is not reasonable to expose information to the processes which do not required it. Therefore, based on the access level of processes, suitable information should be visible for them. This security policy does not have any contrast with the idea of enterprise processes, which their goal is to let the information flow seamlessly.

Also (Strong & Volkoff, 2004) defines an ES as "a system which its task is to support and integrate a full range of business processes, uniting functional islands and making their data visible across the organization in real time". This definition adds the fact to the previous definition that the data and information inside the system should be understandable for all the business processes.

Another interesting definition for enterprise systems is based on the legacy systems; a legacy system is an existing computer system or application program, which continues to be used because the company does not want to replace or redesign it (Robertson, 1997). Most established companies, who have been using a system for long time, are in this group and clearly, the number of these companies is not small. Legacy systems mainly suffer from deficiency of documentation, slow hardware and difficulties in improving, maintaining and expanding. However, there is evidence that overtime ES replaces the stand alone applications and the functionality of legacy systems (Strong & Volkoff, 2004). In contrast to enterprise systems, legacy systems are not designed to communicate with other applications beyond departmental boundaries (Jessup & Valacich, 2006). Middleware offer a potential solution to adapt the new part with the legacy system. Nevertheless, regarding price of developing a middleware, this question comes to mind: can middleware alone solve the problem of integrating new subsystems with a legacy system?

In short, the common idea in existing definitions illustrates that an EIS is about various businesses, business processes, organizations, Information systems, and information, which circulate across the enterprise. In the other word, it is about the businesses model in the organization. Review on the current definitions leads this research to go to the next section, which is defining the EIS's definition for this study.

2.3.1 Organization

Regarding to the review of the available definitions of enterprises and their history, it is understood that the definition of the enterprise has close relation with the definition of the organization. The Oxford dictionary defines an organization as "an organized body of people with a particular purpose, e.g. a business" (AskOxford, 2007). In a business dictionary, an organization is defined as: "Social unit of people systematically arranged and managed to meet a need or to pursue collective goals on a continuing basis. All organizations have a management structure that determines relationships between functions and positions, subdivides delegates roles, responsibilities, and authority to carry out defined tasks. Organizations are open systems in that they affect and are affected by the environment beyond their boundaries" (Dictionary, 2008).

People play a very important role in any organization; hence, EIS should consider them too. This dissertation will illustrate the importance of the people in ES by discussing the stakeholders in the future section. Human factors change during the life of a system; hence, predicting all their requirements is not possible, especially, for a large-scale enterprise system. Besides, as organizations are open systems (Dictionary, 2008), the environment is another changeable item that can affect an organization.

2.3.2 Business Model

Another main factor that influences an ES is the business model (Figure 1). Supporting the strong relationship between business processes is the aim of ES. In fact, defining various business processes in enterprise systems distinguishes them from normal systems for a company, for example in the case of BMW, producing car, engines for other car brands such as Rolls-Royce, and bicycle are examples of various businesses that they involve in. A normal system in a company contains components and subsystems that belong to one specific business and satisfy its requirements. Normal company may need to contact other companies to continue its business but involving partners or suppliers is not their main concern. In contrast to normal company where the focus is on one particular business, an enterprise focus on a collection of business processes which could be relevant to each other or not but all of them is under the arch of the main principals of the enterprise. Indeed, making profit is not one of the essences of business model. Although, there are non-profit governmental or non-governmental organizations such as healthcare organizations that have their own business model which deals with the process of treating patients.

What is a business? To (Clifton, Ince, & Sutcliffe, 2000) "business involve a complex mix of people, policy and technology, and exist within the constraints of economics and society".

An Enterprise System, and later in this chapter, enterprise architecture, are not about detailed implementation of business functions; their focus is mainly about a very top-level view of the whole business model of an organization. Figure 3 illustrates that the environment, which can be the market, or any other source that can influence the organization's business triggers the need for business process(s). This trigger can be a

change in the market or receiving new requirements. For example in the case of BMW that was discussed earlier in introduction chapter, changes in the market triggers the need for more product. A business process can be one that was in the organization from its beginning, or a new one that is needed as the result of the market changes; for example in the case of BMW, one of the possible business processes could be increasing the special product .Each business process contains various business functions, which are the elements that implement the business process. There is no limitation for sharing the business functions unless the exceptional cases; also business functions can be breakdown to the other business functions with more detailed level. In the case of BMW, for example, the business function for increasing the specific product can be request more parts from partner suppliers, or slow down a parallel product in the factory to make the resources available for increase in the product line for X5 and Z3.

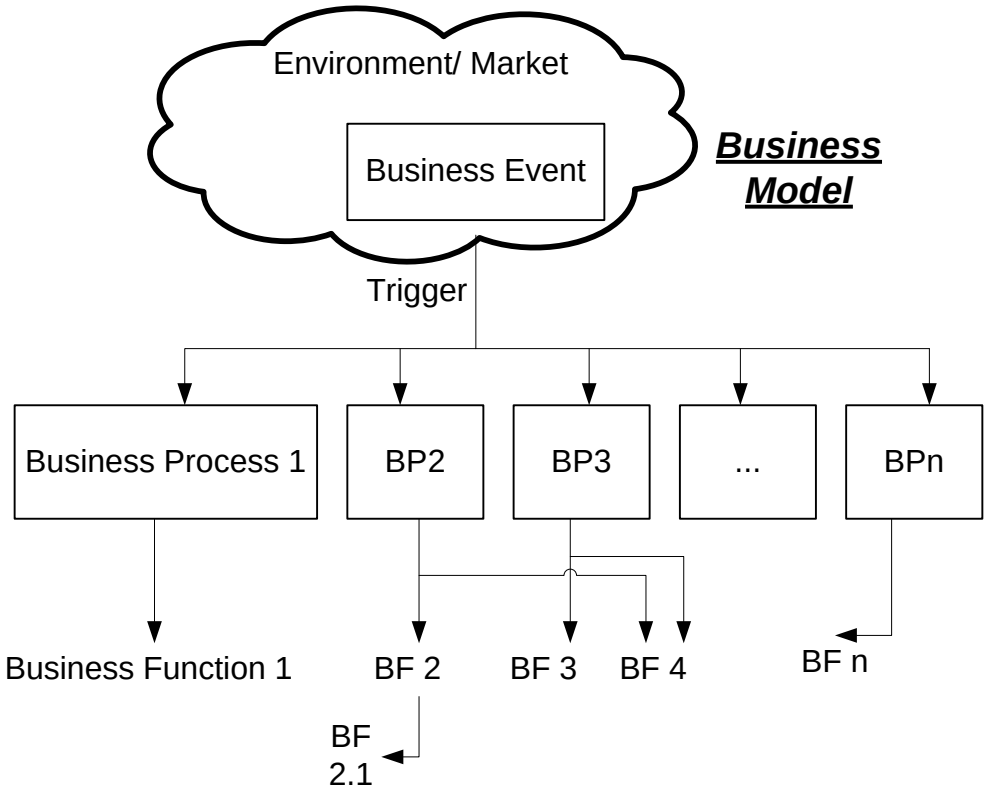

Figure 1: Business Model [based on (Kaisler, Armoiur, & Valivullah, 2005)]

At this level, it would be better to think about the characteristics of such a system too. Defining the characteristics of EIS can provide a better understanding for EIS that can be seen in the next section. Next section will also propose the working definition for this study using the knowledge that was achieved from reviewing the literature.

2.4 Working Definition

Aside from organization, human, and business model factors that were given earlier in this chapter, a company could be called enterprise according to the current literature which their domain at least contains the two following characteristics:

- Involve more than one business.
- At least one main business and other side businesses.

All of these businesses could lead to satisfy the defined main goal but containing various business processes is one characteristic that separate enterprise from normal company. Include more than one company, at least one main company and other side companies (for example BMW and its partners/suppliers), or more than one main company such as Mitsubishi or General Electric.

Considering the earlier discussions about available definitions for EIS and the characteristics of enterprises, following is the working definition of EIS.

"Enterprise systems integrate the key business processes of a firm into single software system so that information can flow seamlessly through the organization, improving coordination, efficiency, and decision making" (Laudon & Laudon, 2007). EIS is based on a set of integrated software modules and a common database or databases that act similar to a central database. The database collects data and feeds the data into numerous applications that can support nearly all organizations internal and external business activities. When new information is entered by one process, the information is made available immediately to other business processes according to their accessibility level. The organizational business processes should be adaptive with software system applications. It is possible to change the business processes in a manner so that it define to the software applications. "Enterprise systems support organizational centralization by enforcing uniform data standards and business processes throughout the company and a single unified technology platform" (Laudon & Laudon, 2007) however, the ideal system is independent form the specific technology or platform.

The above definition is in the early stage of this study, hence, it will be considered as the working definition. Throughout this study, I will revise the definition based on the

new requirements. To expand the knowledge about EISs, the next section will explain the benefits and challenges of having these types of systems for enterprises.

2.5 Benefits and Challenges

Based on the working definition that was introduced in the previous section some of the benefits and challenges of developing an EIS are now extracted. The first benefit of having an EIS is achieving an integrated key business processes that wrap whole organization and partners together. As the result, the information can flow seamlessly in EIS, and providing on demand access for the processes or authorized stakeholders that need the information.

To support the seamless information flow this definition includes the central database. However, this can be distributed database or clusters of databases, depending on the amount of data and available resources.

Another important benefit of EIS is being able to have a platform independent environment. The possible approach to achieve this goal is not in the domain of this dissertation. Therefore, the aim is having an EIS independent from particular technology or environment; because developing EIS is mostly very expensive and consuming large amount of resources. Hence, it is expected to be flexible enough to tolerate the changes in the environment during long-term period such as upgrading the technology.

The preceding paragraphs focused on some of the main benefits of having an EIS. However, every system and technology has drawbacks too. The following paragraphs will introduce some of the main challenges in developing EIS that are mentioned in the literature.

Considering the examples such as Mitsubishi and hundreds of other enterprises all around the world, it is clear that an enterprise exists; however, they suffer from lack of definiteness. Therefore, the main difficulty in this context is the EIS loose definition, which causes confusions. To the best of my knowledge, there is no official defined term for enterprise.

The second main challenge in this area is defining the flexible domain for EIS. Increasing the power of people makes a system more flexible and more innovative, but it also brings the challenges of its own. For example, the interview with IBM consultant

Patrick, which can be found in more detail in final chapter, showed that insurance industry face with people innovation challenge. There is a process in calculating the amount of insurance for each item that is calculated by people in detail in some cases, item by item. The changes in this part are too much that they decided to separate it from the rest of the software system; this small group has their own IT- oriented system, and a special IT team that deals with their changeable requirements.

"When technology has been applied, it is mainly for speed up the isolated components of an existing process". The processes are not developed based on IT ideas. Now some of the functions in the process are automated, there is a need for communication between them. "This creates communication problems within process and impediments to process redesign and enhancement" (Kurosu & Yamada, 2002).

The next challenge refers to the gap between business and IT. "Most business processes were developed before modern computers and communications" (Kurosu & Yamada, 2002). In an example, which is given in this article, "each organizational subunit within the process had optimized its own IT application, but no single subunit had looked at the entire process. We believe the problem this firm experienced is very common" (Kurosu & Yamada, 2002).

To sum up, (Davenport, 2000) mentioned four main criticisms for enterprise systems in his book that should be considered by decision makers who want to develop and adjust such a system. Criticisms such as:

- Inflexibility,
- Long implementation periods,
- Overly hierarchical organizations,
- Antiquated technology.

2.6 Conclusion

In summary, one of the essences of using a concept is defining relevant terms. In the case of EIS this chapter provides a working definition that can be used throughout the rest of this dissertation. The next chapter uses this working definition to describe technologies and concepts related to EIS.

Chapter 3

Enterprise Information System Described

3.1 Introduction

Having dealt with the basics, this chapter looks at certain areas in greater detail, focusing in particular on technologies that could be used to support EIS and current approaches for developing this type of systems. However, this chapter starts by discussing stakeholders of EIS before reviewing technologies that could be used in this area. Stakeholders are one of the pieces that should be considered in understanding any kind of systems. Stakeholders, their requirements, their abilities, and their goals are important for developing EIS.

3.2 Stakeholders

Stakeholders are factors that affect the system or benefit from the system. Their requirements will influence the system, hence in designing the architecture of the system they should be considered. Stakeholders mainly refers to the human factors in the software architecture approach: "Each Stakeholder of a software system- customer, user, project manager, coder, tester, and etc- is concerned with different system characteristics that are affected by the architecture. For example, the user is concerned that the system is reliable and available when needed; […] the manager is worried (as well as about cost and schedule) that the architecture will allow teams to work largely independent" (Bass, Clements, & Kazman, 2003). However, in this dissertation, organization considers as a stakeholder too. Goals of an organization, which in some cases are more that goals of individuals or groups, consider in the requirement elicitation and design stages.

For this study, to find the stakeholders in enterprise system we can group them in general stakeholders that most of the ES have them and specific stakeholders which are particularly for special organizations.

One of the main general stakeholders is a manager. Literature declares the role of the manager/leader is essential for organization hence they should be considered as important stakeholders for ES too. In the case of Mitsubishi, for instance, all parts of the enterprise were finally managed by a team of thirty managers. This structure is not against freedom and flexibility in management. In fact, enterprise managers are flexible to choose their management policy but three principles are honoured by all of them (Mitsubishi, 2007). Subsequently to the preceding argument, considering the role of managers as one of the main group of stakeholders is essential in designing an ES.

However, as it can be seen in figure 2, stakeholders can be grouped as follow too:

First group are the stakeholders that insert the data into the system; these stakeholders could be operators who insert command, and information, or manager who insert the rules or management decisions for instance.

The second group are the stakeholders that uses the outputs of the system such as final customer for instance or manager, which can make the final decisions with the help of the information, which can be generated by the system.

The third group are the stakeholders that are inside enterprise systems such as designer, developer, admin, and maintenance group as the example of technical stakeholders; also the non technical stakeholders such as internal users.

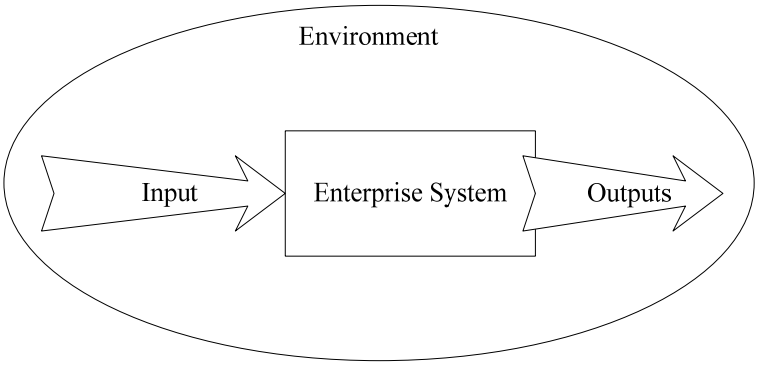

Figure 2: Stakeholders' model

After defining main stakeholders for the architecture of enterprise system, another important topic that should be discussed for developing ES is their relevant technologies.

3.3 Review of the related technologies

This section aims to introduce a group of technologies, which can be used in developing EISs: Information Systems (IS), and Distributed Systems (DS); or those used with EISs such as Virtual Organization (VO); or those that have very close characteristics to EISs such as System of Systems (SOS) or Adaptive Systems (AS). This group of technologies, which will be discussed briefly in the following sections, are examples of well known technologies that can be used in this area. Indeed, there is no claim that these are all the possible technologies. Therefore, this list could be extended during this research in the future work.

3.3.1 Information Systems

Information technology is "the capabilities offered by computers, software applications, and telecommunications" (Davenport, 1990). It is clear that the definition of Information System (IS) is very wide and general. It includes a large range of different type of systems. According to (Stair & Reynolds, 2006), an Information System is a set of elements or components that collect input, process and store it, generate output; and provide a feedback mechanism to meet an objective of the owner. The input in these systems is the activity of gathering and capturing raw data, the process elements produce useful information from input data; and the output involves transferring or converting this information to the suitable output format for different users, usually in the form of document and reports. The feedback in information systems is the output that is used to make changes to input or process activities. The following diagram illustrates the definition of information systems, which was described earlier:

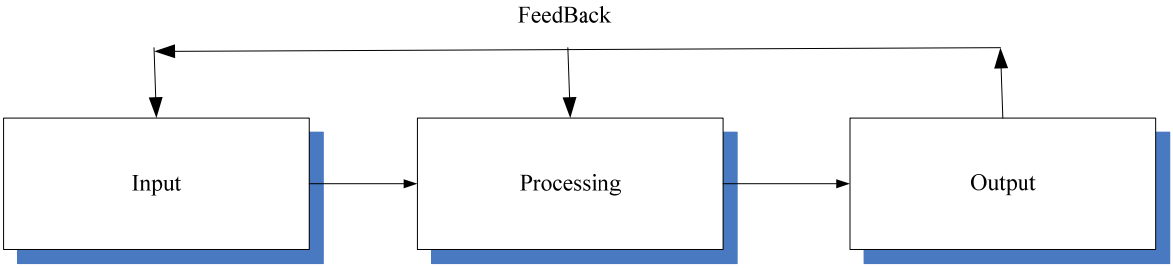

Figure 3: Information System

Feedback is important for managers and decision maker; it is also useful for forecasting future events. In the example which is given in (Stair & Reynolds, 2006), a bidding system used a computerized feedback system to link its supplier and plants. The output result from information system might indicate that inventory levels for a few items are getting low, which is a potential problem. At this point, the manager could use this feedback to order more inventories from a supplier. The new orders are the input to the system. It is clear in this example, that the feedback in the information system can help the managers and decision makers to make the right decision in the right time and before any major problem happen. Moreover, it can help to prevent form future problems by forecasting the future.

Analysing the diagram that can be seen in figure 3 illustrates that various type of system and not only IT-oriented system can be compatible with this diagram. For example, in the case of human body, for writing a text, a person can hear something from outside which is the input, process this input, and may store it for a while in the brain, and then write it down on the paper which is the output. In addition, he or she can use the result of this process to give some feedback to other or to him or herself. This examples illustrates the fact that the diagram in figure three is very general diagram that can be used to define various things such as EIS or IS; therefore, EISs could be a subtype of the IS. The main goal of EIS is to satisfy requirements of organizations; hence, it will receive some input from the environment of organization that could be operators who insert data or changes in the market that send some data from other partner systems. The input data will be sent to the processing components in EIS and the result will be sent to output components. At this stage it depends on the design of EIS, the result can be used in different manner to satisfy the goals. The feedback is also the result of output that can be

used to benefit the organization. This description of EIS is completely match with the definition of IS in figure 3. The aim of this argument was to assert that according to the definition of IS which is given earlier and the definition and description of EIS, it is most likely to introduce an EIS one type or subgroup of IS. The theory that EIS is one subgroup of information system does not make any difference in the existence of EIS. Because information system is a very general definition that contains various IT oriented systems.

Another subgroup of IS that is Distributed systems, which is mainly useful in the large scale organization. Therefore, the next section will discuss DS in relation to EIS.

3.3.2 Distributed Systems

"Evidence suggests that there is a large-scale move away from centralized mainframe systems towards distributed systems which is perceived as the key approach to infrastructure design for the foreseeable future" (Simon, 1996). Some of the requirements of distributed System (DS) are computer network and communication, distribute operating systems, database and file systems. One of the characteristics of EIS is including multi business and multi partners. Today, based on the available resources the organization are distributed around the globe hence the suitable system for them which in this study is introduced as EIS should support the distributed requirements. The earlier discussion justifies the need for distributed systems in most of the EIS. Therefore they should be considered in the design level. One of the technologies which is useful to support the distribution of the resources –mainly human resources- time and culture differences is virtual organization (VO). The next section will discuss the VO in the context of EIS. This technology is one of the possible technologies that could be considered in the design and development of EISs.

3.3.3 Virtual Organization

The aim of this section is to discuss some aspect of VO such as its definition, to help this research to evaluate VO as a potential component in developing EIS.

According to (Travica, 2005), "'virtual' or 'virtually' or 'virtuality' mean that something exists in potentiality, effect, essence, although not formally and tangibly". Virtual organization is a term that could be defined in various contexts such as commerce.

Also, the boundaries of this type of organization are flexible and fuzzy; it depends on the boundaries of the constituent members (Travica, 2005). Both of the above reasons lead to the idea of defining the new nature for the office work. (Husein, Moreton, Sloane, & Knoll, 1999) argues that "Globalisation of business practices, increasing use of computers and communication technologies are beginning to redefine the nature of the office work".

The virtual environment can be defined from various perspectives. One perspective, introduces it as the environment that is used for simulation of the real environment. The example of this case is the Virtual Enterprise Network (Rudolph F. Crew, Ed. D., & Chancellor, 2008) which is a virtual environment for the student to learn about enterprise businesses; or Second Life (Life, 2008), which is a simulation environment.

A Virtual Environment does not exist in the physical sense but only as an electronic network representing a partnership of businesses. "A virtual organization structure employs individuals, groups, or business units in geographically dispersed area. These people may be in different countries, operating in different time zones. They may never meet face to face in the same room, which explains the use of the word virtual" (Stair & Reynolds, 2006).

Despite this separation, there is at least one common rule for all projects, which is delivering within the deadline. Regarding to the lifetime of the VO, it depends on the strategy of the organization. "In some cases a virtual organization is temporary, lasting only a few weeks or month. In others, it can last for years or years" (Stair & Reynolds, 2006).

In the case of EIS, which are massive systems, distributed around various geographical places, a virtual organization plays a main role in enhancing an enterprise in addition to helping it towards its goals. With the help of VO, there is no limitation for employees location and their time differences or even the time, when they want to the work. As it was mentioned, the final aim is to hand in the suitable result within the deadline. The definition of the workplace according to Procter & Gamble is "anywhere someone is trying to be productive" (Stair & Reynolds, 2006). Another benefit of VO that can be beneficial for EIS and organizations is the extra level of security that can provide; "Many companies are now dispersing employees and using a virtual structure in case of

terrorist attack and disaster. If a disaster strikes at the primary location, the company has still sufficient employees at other locations" (Stair & Reynolds, 2006).

Information systems, distributed systems, and virtual organizations are examples of potential technologies that can be used in modelling and developing EIS. However, there are technologies, which have same characteristics or similar goals. One famous example of these technologies is System of Systems or SoS.

3.3.4 System of Systems

System of Systems (SOS) is a type of system that share similar interests and high level aims with EIS. There are various definitions in IEEE SMC 2005 (Jamshidi, 2005) for SoSs; the two of these definitions which are related to the EIS are as follow:

- Carlock and Fenton: "Enterprise Systems of Systems Engineering is focused on coupling traditional systems engineering activities with enterprise activities of strategic planning and investment analysis."
- Lukasik (Lukasik, 1998): "SoSE involves the integration of systems into systems of systems that ultimately contribute to evolution of the social infrastructure."

The idea of the system that can manage and control other subsystems in the organization or enterprise is similar to the aim of EIS. The main characteristics of SoS according to (Jackson & Fritzsche, 2007) are: "Systems of systems can be briefly characterized by the following: operational independence of the elements, managerial independence of the elements, evolutionary development, and high dynamicity of architectures, emergent behaviour, and geographic distribution. For such systems, interoperability, security and dependability are key concerns". Comparing the expected behaviour of SoS and EIS illustrates many similarities such as evolutionary development, high dynamic architecture, geographic distribution, etc.

However, there are studies in the area of SoSs to use this technology for military and air traffic control purposes. The (Jackson & Fritzsche, 2007) document is a witness for this comment: "the concept of system of systems (SoS) is spreading both in the military and civil domains". The improvement in THALES Information Systems declares the importance of testing this technology in a very sensitive and real time environment (Jackson & Fritzsche, 2007).

In the previous paragraph, independency of the elements and independency in management introduced as two of the many characteristics of SoS. Another technology that can be presented in the relation to EIS which has similar characteristics as SoS is Adaptive System (AS). The aim of presenting AS and some special characteristics of SoS that presented earlier is to illustrate the fact that even though these characteristics seem very optimistic and not realistic, still they worse to be discussed and considered; the same case for EISs. New technologies and ideas could propose approaches for satisfying new or optimistic requirements.

3.3.5 Adaptive Systems

In order to introduce adaptive systems a clear definition in (Brien, 2002) can be a good start. "A system that has the ability to change itself or its environment in order to survive is an adaptive system". Flexibility of the system is one of the desires in the EIS. The developers prefer to develop a system that can handle the changes in the environment such as market in the business-based organizations. In addition, in some organizations such as insurance the innovation of some group of employees are valuable for the success of the enterprise, hence the desirable system needs to have the ability to consider and handle people's innovations. Overall, the idea of developing an adoptive system provides benefits for EIS too.

(Brien, 2002) argue that organizations are adaptive systems by themselves. "Organizations are examples of open systems because they interface and interact with other systems in their environment. Finally organizations are example of adaptive system, since they can modify themselves to meet the demands of a changing environment" (Brien, 2002). This comment declares the fact that, organizations follow the demands of changes in environment; and this clarifies the fact that, if there is a system designed to respond to the requirements of organization, this system should be adaptive.

As figure 4 shows, (Brien, 2002) link the adaptive systems and EIS by defining virtual roles of IS for business enterprise. "Three vital roles that information systems can perform for a business enterprise are:

- Support of its business processes and operations
- Support of decision making by its employees and managers

- Support of its strategies for competitive advantage" (Brien, 2002)

Figure 4: Three vital roles [based on (Brien, 2002)]

These three roles illustrate that the important facts in the design and development of systems for companies and organizations are based on non-IT oriented roles. These roles are based on the priority of the processes in the organizations rather than IT oriented technologies. (Brien, 2002) explains this point: "A lot of people initially thought that the E in the E-business was more important than the business part".

The next section will discuss two of the current approaches in the market for developing EIS in practice.

3.4 Current Approaches to Develop EIS

There are various approaches that could be discussed in this context, but at this stage, the two of these approaches will be discussed. The first approach –CBM- is a practical approach that is currently used in IBM and the second approach is interesting because of its relation to object oriented theory.

3.4.1 Component Business Model

Component Business Model –CBM- is an approach that has been proposed by IBM. This section will explain the motivations for implementing CBM; moreover, it will explain the components of this solution. The result of this introduction will illustrate that ideas in CBM could be considered as a potential solution for developing EIS.

According to (Pohle, 2005) the aim of today's business is to create a global connectivity platform that removes the traditional boundaries and allows the customer to switch between providers easier. Reaching this aim is not easy, because the market is so changeable and satisfying all the requirements are challenging and in some cases impossible. Therefore, the suggestion for the industries could be specialized in the main and essential points. (Pohle, 2005) introduces CBM as an approach to help to improve this specialized focus, both internally and externally.

The research on developing suitable IT-oriented systems for organizations and industries mainly focused on the popular topics such as 'customer relationship management' and 'enterprise resource planning' (Davenport, 1990). However, none of these studies could suggest a complete solution for bridging the gap between the IT oriented systems and business. CBM aims to fill this gap by driving the specialized focus to the right place. The supporters of this technology want to help to satisfy the requirements of the market.

Two types of focus are discussed in IBM report (Pohle, 2005), external focus and internal focus. The internal focus rethinks and reconsiders the leverage -strategic advantages- that company can achieve using their assets and capabilities. On the other hand, the external focus help to find and use the capabilities that the organization or industrial group cannot create themselves; benefiting from others specialization and resources. Hence, combining internal and external specialization allows organizations to redefine their competitive positions in the face of sweeping changes in their industry.

Figure 5 describes the CBM model in the simple manner. The boxes are the symbol of business components. The stakeholders in this model are categorized in three main groups. The people who make the decisions in direct group, the people who manage and control the business component in control group, and the people who execute the business components in execute group.

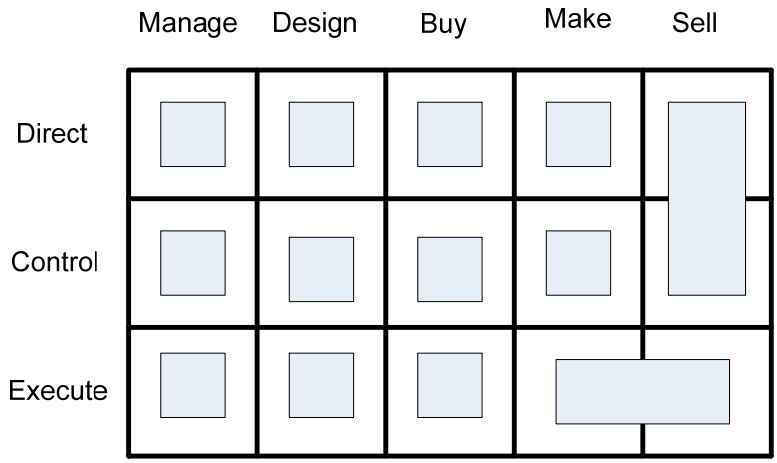

Figure 5: Component Business Model (Pohle, 2005)

The horizon line of this image shows the high-level description of the activity conducted. The boxes or business components "are the building blocks that make up the specialized enterprise" (Pohle, 2005). Figure 6 present more information on the business components in this model. The five dimensions of this model are also shown in figure 6. According to (Pohle, 2005), the business purpose is the logical reason for the existence of the component. Activities are mutually exclusive in each component to achieve its business purpose. The activities require resources. Based on the governance model each component is managed as an independent entity. Business services are the services that each business provides and receives.

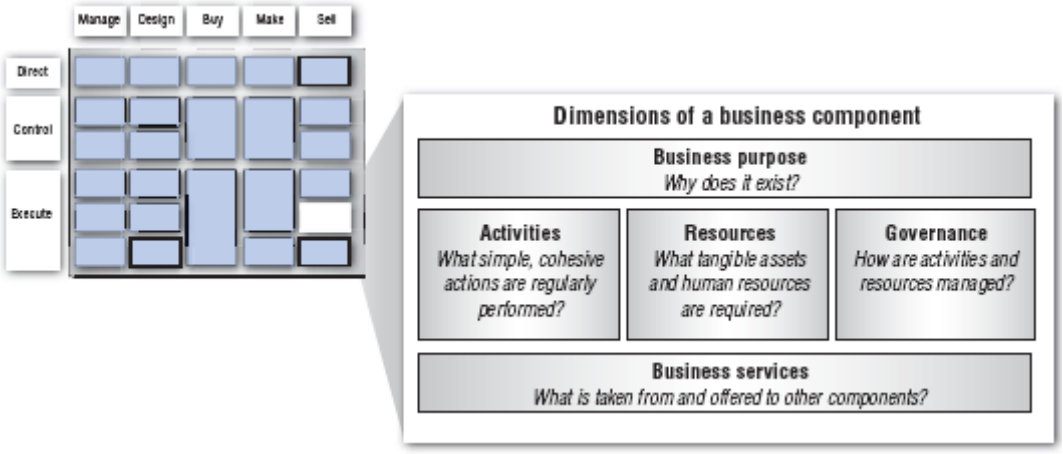

Figure 6: Components in Component Business Model (Pohle, 2005)

Earlier paragraphs introduce CBM briefly. The importance of the internal and external specialization was also discussed. Evidence declares that specialized enterprise is

made of business components; and according to (Pohle, 2005) the aim of CBM is to satisfy the internal and external specialization partially too.

A good feature that can be observed in CBM is the manner that this model classifies business components. The classification is based on grouping main stakeholders that influence the component and the activities. This high-level design view helps the non-IT technical to involve in the design process. Moreover, this model gives a high-level view on the components that are shared between different activities and the one that are isolated. It also participate activities from different partners of the main organization when require.

Two main advantages of this model is engaging the suitable stakeholders and the right activities; also the high level view that this model provides for the designer and other related stakeholders introduces a clear approach that can be used for brainstorming during the early steps of EIS design.

In short, the decision makers should rethink about the organization's approach according to the market changes and find out what they can do themselves and what they cannot do. The things that they cannot do should be given to other partners that have the capabilities of satisfying the requirement.

CBM is not the only practical solution in the market for developing EIS. The next section introduces Catalysis approach, which is based on object oriented ideas as another example of a potential approach for developing EIS.

3.4.2 Catalysis Approach

Evidence indicates that the concept of developing software in the Catalysis approach starts from describing the user's domain and the software domain (D'Souza & Wills, 1999). Catalysis approach is based of two basic concepts, Object and Action. Objects "representing cluster of information and functionality" and actions "representing anything that happens (event, task, job, message, change of state, interaction, and etc)" (D'Souza & Wills, 1999).

For the start, this approach replies to four main questions:

a. State what happens
b. Which object is responsible for doing it

c. Which one is responsible for initiating it

d. How it is done

To satisfy the object oriented ideas, this model tries to use case study, use case, sequence diagram, and other object oriented tools to describe the important aspects of a case. After collecting the required requirement and data, the information will be distributed to different groups from different views. The amount of information, given to each group, varies. The main idea is not to provide every stakeholder in a project with all the available data.

The above idea leads to main abstractions which seem to be the aim for this approach. Abstraction as a characteristic for Catalysis contains five following layers:

- business model
- requirement specification
- component design
- object design
- component kit architecture

Another characteristic of Catalysis is its three modelling levels:

a. Domain/business: cover all concepts of relevance to clients and their problems (the environment) of the software

b. Component specification

c. Internal design: how a component is assemble from smaller parts that interact, to provide the required overall behaviour

In short, this approach has three principals (D'Souza & Wills, 1999):

d. Abstraction: separate the most important aspects of a problem from the details; enable developer team to tackle first things first.

Precision:

a. Expose gaps and inconsistencies early by being precise
b. Trace requirements explicitly through models
c. Support tools at a semantic levels well beyond diagrams and databases

Pluggable:

a. Building adaptable software is about designing components and plugging them together
b. Get the most from each piece of design work

c. Gain fast, reliable development through reuse
d. Reuse not only classes but also frameworks, patterns and specifications

After reviewing the characteristics and principles of this approach, it appears there are few differences between this model and normal software architecture models and ideas. The CBM model is more close to practical project in industry than this approach. Catalysis approach is good for detailed development compare to CBM. CBM can be used for the earlier stages of design and development of system of systems. Indeed the idea of abstraction is interesting and it could be merged with other models such as CBM.

Acquiring information from real world and public authorities to provide a service is too complex and detail in most cases. "This narrows the range for an effective solution from the beginning of the project. Hence, there are methodologies to fix and sort this problem" (Regner, Wiesinger, Kung, & Wagner, 2004).

The research in this approach illustrates that developing software in this field can be based on the business models. Hence, the paper proposed the hierarchically structured business model; which is mainly based on the abstraction ideas. Abstraction from global view: 'value chain, business process map, process diagram, etc', and abstraction from business domain to technical representation.

The review on the two related technologies in the area of the EIS development demonstrates that each of these technologies has their own drawbacks and positive aspects. A new model for developing EIS would ideally use the positive aspects and avoid the current drawbacks.

3.5 Conclusion

In conclusion, this chapter covered some of the widely known technologies in this area that are presented in literature. However, by increasing system and software knowledge, new technologies will be offered to the market; therefore, EIS should be flexible to handle changes in the environment. This flexibility mainly should be considered in the architecture of each system. Indeed, each EIS based on the environment that surrounds it requires its own architecture but there are main components in developing system architecture that are common for most of system, in this case EIS.

Stakeholders are one of the basic components in software architecture that were discussed in this chapter. Next chapter will discuss the architecture of the EIS with more detail.

Chapter 4

Enterprise Information Systems Architecture

4.1 Introduction

In the conclusion of the preceding chapter EIS architecture or Enterprise Architecture (EA) were discussed briefly. The rest of this chapter will explain EA in more detail. Moreover, requirements of these systems will be discussed, to open the path for getting to suitable discussion in chapter 6.

4.2 Enterprise Architecture

Enterprise Architecture (EA) is 'blueprints' for systematically defining an organization's current (baseline) and/or desired (target) environment (Bellman & Rausch, 2004). Enterprise Information Technology Architecture is the blueprint for creating enterprise-wide information systems (Armour, Kaisler, & Liu, 1999). It is not about the detail of the design; it is high level thinking. Getting the enterprise pieces to seamlessly interact and evolve according to wide-range strategic business objectives may seem like mission impossible; for a large distributed organization, however, it is mission critical (Armour, et al., 1999).

There are various case studies that illustrate the role of EA in active organizations; the one, which is discussed in the next section, is a case about US federal enterprise architecture framework (Bellman & Rausch, 2004) from electronic government magazine. The aim of discussing this case is mainly to illustrate challenges of designing the architecture of large scale EIS in practice.

4.2.1 Case Study: Federal Enterprise Architecture

"In July 2002, US congress passed the e-Government Act of 2002 that provides for a comprehensive framework for information security standards and programs, and uniform safeguards to protect the confidentiality of information provided by the public for statistical purposes, and to expand the use of the Internet and computer resources in order to deliver Government services for a citizen-oriented, results-oriented, and market-based government". (Bellman & Rausch, 2004). Hence, a massive study in the area of enterprise architecture started. One of the first achievements of this study was defining the framework for enterprise. The federal enterprise architecture framework contains four models:

Business reference model describes a standard set of Federal lines of businesses in which it is a mission for all the agencies.

Performance reference model measure how well are these lines of business function.

Data and information reference model defines aggregate (formed of separate units gathered into a whole) data and information categories to support business line operations.

Service component reference model: IT service components, which support service components.

Technical reference model is about how technology supports delivery of service components.

The enterprise architecture can be general or specific for special purposes. For example, department of defence, in US has its own architecture framework, which contains three views – operational, system, and technical.

"**Operational view** visually presents and describes tasks and activities, operational elements, and information flows required to accomplish or support an operation". This view describes the information exchange in detail.

"**System view** describes systems and interconnections that support the activities of interest". It describes which system supports the requirements.

"**Technical view** is the minimal set of rules governing the arrangement, interaction, and interdependence of system parts or elements". It is about the rules about implementation of each required system capability.

One of the important points that should be considered in the architecture of enterprise is that the architecture should be good enough in the first step then it will incremental to get closer to the optimal architecture. In addition, it should be understandable for the different group of stakeholders, which are mainly from business background and IT background (Kaisler, et al., 2005); considering the fact that each of these groups expect to see their different requirements satisfied.

The recognition of lines of business could be interpreted as satisfying requirements; however, most of the effort is to categorizing technology and applications. In case of US defence architecture, "it was done by organizing large focus groups, bringing to Washington IT staff from the different US Forest service regions around the country" to sort the gathered requirements. After a huge expensive effort and creating the architecture and putting it into the model, finally, it was not able to address the business problems that were posed by business use cases. This common issue occurs when business and IT facts are not considered in parallel levels.

Cases in the literature declare that today's definition of enterprise adds the business value to the IT applications and IT theories. It has been a period that most of the businesses relying on technology blindly and without the concern that the business side is the most important one and the IT side are suppose to help it and not be visa versa. According to the case studies, the idea of IT-centric systems is reconsidered in the today's enterprise architecture. "The guidance offered by the Federal Enterprise Architecture Framework (FEAF) involves understanding the business and not just the technology and applications layers of an agency's architecture" (Bellman & Rausch, 2004). EA is a system architecture that can address the issues that comes from business use cases and just the technology is not the main concern for making decisions.

The next thing, which is coming to mind, is what are the characteristics of enterprise architecture? How to evaluate this architecture? According to (Kaisler, et al., 2005) the US Government Accounting Office (GAO 2004) has released a framework for

assessing and evaluating enterprise architecture. However, the main characteristics of EA according to Kaisler's (Kaisler, et al., 2005) study are as follows:

- The EA's boundaries are well-defined,
- EA elements have already assigned responsibilities,
- Interfaces, formats, and protocols between EA elements are well-defined; alternatively, every interface supports only approved formats and protocols,
- An external auditor adequately documents the EA to permit compliance review and assessment.

In short, the gap between IT and business from Architecture point of view defines as follows: "often after a grate deal of money is spent building the IT centric architecture, its business value cannot be demonstrated" (Kaisler, et al., 2005). Hence, enterprise architecture is an approach to help to solve this problem by designing goal oriented system. To satisfy goals of each enterprise, its main quality attributes should be defined. Moreover, defining systems' main quality attributes is another basic and common step for designing architecture of systems and in this case EIS. The next section will discuss EIS's main quality attributes in more detail.

4.3 Quality Attributes

According to (Bass, et al., 2003) quality attributes are the result of business considerations: "business considerations determine qualities that must be accommodated in a system's architecture". To define the main quality attributes of the enterprise systems, it would be useful to remind the aim of this type of systems. Around the globe, companies are getting more connected to each other. Hence, they require a system that not only can satisfy the regular requirements of the organization but also can support the changes in scale and business processes. Managers in this system are able to monitor the processes of the organization in near real time, without waiting for monthly reports that can be cross-referenced with other monthly reports. In addition, information in this system can flow seamlessly (Davenport, 2000). Considering the important points about enterprise systems that presented above, it seems that to define the qualities of the ES, the quality attributes should be studied from different perspectives which are relate to each other because of

enterprise systems; organizational view, and IT view are the two important points that are considered in this section. However, some argue that business quality attributes are also another view of the overall quality attributes of enterprise systems but at this stage the literature in this survey could not justify any particular business quality attributes for enterprise systems.

From the organizational point of view, uniformity of the whole organization with the help of IT-oriented system could be the first quality attribute. Another quality attribute in this area is organizational management, the manager is one of the essential stakeholders, but the organizational management is another way to say the uniformity of the organization and a way to handle the human brain motivations. By this, I mean, the organizational management should be in a manner that can handle the innovation in the organization.

The IT oriented quality attributes that are desired to be achieved in an enterprise system according to the literature are: integrity, platform independent, performance, and availability.

4.3.1 Integrity

According to the Oxford dictionary, the term Integrity means "the quality of being honest and morally upright" (AskOxford, 2007). In case of quality attribute in this study, integrity of the data, information, and knowledge illustrates the reliability of the system; not only the correctness of the data inside the system but also the correctness of access to the suitable data which is called 'user authentication'. In some systems, the normal user shall not have access to the information or components, which the administrator has; or the manager has different level of accessibility to the information/components inside the system. In short, integrity is a confusing term, and in this dissertation, it considers two parallel issues:

1. Data validation
2. Correct access to data helps to insure data integrity

4.3.2 Platform Independent

The system is distributed around the world and it contains various businesses, which can add or remove to the EIS during its lifetime, hence it should not be dependent on the platform, which can be changed during the time or change according to place and in some cases perhaps change based on the businesses requirements.

In addition, longevity is a clear result of platform undependability. Longevity is a business quality attribute, which states the lifetime of the system. Evidence implies that longevity in an EIS, which is expensive and complex to design and develop, is essential. The time life of EIS is based on the design and implementation decisions and based on the defined protocols, which the subsystems and components will communicate with respect to them.

4.3.3 Performance

EIS are large and complicated systems, hence, performance plays a key role in design and implementation of these systems. Performance considers two perspectives; the first one is resource considerations and the second one is how fast the system can reply to requests.

From the resource perspective, the idea of allocating the resources to the processes and components plays the main role. If the resources cannot be allocated properly, the price of theses systems will increase and it will be more than what is required with sufficient resource allocation. This study focuses on two types of resources: Human resources and Computer resources.

Performance from the second perspective, deals with the response time of the system to the actions such as inserting commands. The response to the commands and processes shall be fast enough to keep the system up to date; otherwise, there is the risk of loosing the real value of the essential information inside the system. In emergencies, for instance, the decision of the managers should be broadcast immediately through the system otherwise the latency in receiving the decision could cause damages for the whole enterprise.

4.3.4 Availability

"Availability is concerned with systems failure and its associated consequences. A system failure occurs when the system no longer delivers a service consistent with its specification" (Bass, et al., 2003).

Like any other big system, ES contains different subsystems and components. The role of the ES is to keep the enterprise's parts all together. If a failure occurs in some part it is the responsibility of the system to prevent from propagating the failure through the system, otherwise the result of the failure in such a sensitive system which the essential decisions of the managers and the reaction of other stakeholder is depend on the availability of the system could follow serious consequences.

After describing two main elements of designing system architecture, which are defining stakeholders and main quality attributes, the next section will discuss one of the well-known approaches for developing business oriented systems which is called Service Oriented Architecture (SOA). It seems that ideas in SOA can be useful in developing EIS too.

4.4 Service Oriented Architecture

There are many definitions for Service Oriented Architecture (SOA) declares that there are a large number of definitions available for this concept. This fact illustrates that SOA is a popular and interesting concept to be discussed. The reason for the popularity of SOA according to (Carter, 2007) is its successes in the area of system development for companies. "Behind every successful service oriented architecture (SOA) is the Business. With its promise of using existing technology to more closely align Information Technology (IT) with business goals, we have seen that SOAs have proven to help companies realize greater efficiencies, cost savings, and productivity" (Carter, 2007).

One of the clear and practical definitions of the SOA in literature is as follow: "Service oriented architecture is an architectural concept in software design that emphasizes the use of combined services to support business requirements" (Graham, 2006). This definition emphasizes the role of services as the key approach for supporting business requirements. Note that satisfying business requirements is the aim of enterprise

systems too; hence, this is the key link between SOA and EIS. It seems that understanding SOA could help to propose better approaches for designing and developing EISs; hence, the main question that comes to mind as the result of introducing SOA is the definition of services in this context. (Sprott, 2004) defines services as "a Component capable of performing a task". The important fact about services is their loosely coupled characteristic. As the result of loosely coupled characteristics, their interfaces are independent of their implementations. This helps to increase the flexibility of the final system. Increasing the flexibility is one of the aims of EISs too.

Breaking the enterprise system into smaller parts is an approach to attack a potentially huge development process. The aim of defining agents, components, objects, and services seems to follow the approach of breaking the system in smaller pieces to make the development and maintenance processes easier. "In SOA, resources are made available to serve consumers or a network as independent artifacts that are accessed in a standardized way" (Graham, 2006). Some of these items are loosely coupled to show that each part can be design and develop separately. As it was mentioned earlier, this fact makes the system more flexible because maintenance of each bit would be easier than changing or upgrading the whole system at once. Moreover, the flexibility should not be only inside pieces but also in the links between pieces. In some cases, to restrict the amount of flexibility, designer considers the changes within layers.

Another point related to using SOA to develop EISs is the separation of data and business rules. Data and business rules should be separate from each other; therefore, they will be more flexible when different parts merge as the result of companies merge. On the other hand, problems in this area are more dangerous. Because problems in the system will be more general, subsequently the reputation of the organization will suffer. This threat also exists in development of EIS.

Another fact related EIS and SOA is the users that benefit from system and their goals. "the route to SOA involves supporting business goals by supplying services to the users of the system so that business can be conducted more easily" (Graham, 2006). It is possible that goal oriented techniques would be suitable approach for attacking the problem in development of EIS. Hence, the real users and their goals are important in SOA and EIS. (Graham, 2006) explains the real users, their goals, and the link with IT

oriented systems as follow: “To supply a set of services it is necessary to understand the business and find the real users of the system and these are not usually the people sitting in front of the keyboard and screen. The real users of the system want to achieve a goal, and a computer system is just a tool to help to achieve that goal. SOA is about designing and developing systems that supply services fit for the purpose of helping users attain their goals” (Graham, 2006).

In summary, this section aimed to introduce some of the techniques used for developing IT-oriented systems. Experiences in Service Oriented Architecture can be helpful in the context of EIS development. However, literature declares that EIS could contain various other IT oriented technologies too. Choosing the right technology or group of technologies relies on requirements of the system. Therefore as it is stated in software development models, collecting requirements is one of the early stages. Hence, next section provides the argument, which is the result of reviewing the literature.

4.5 Requirements

According to (Vickers & Brooks, 1998) requirements in software development is “one of the most crucial stages of a successful software development”. Therefore, this section will give a review on the requirements of this type of system throughout the literature. Regarding to the definition which is quoted in (Christel & Kang, 1992) from (STEP, 1991). Requirements are “function or characteristic of a system that is necessary”. Requirements are “the quantifiable and verifiable behaviours that a system must possess and constraints that a system must work within to satisfy an organization’s objectives and solve a set of problems” (Christel & Kang, 1992). To collect the requirements of such a system, this study focuses on high-level requirements’ elicitation techniques to collect the requirements of organizations in relation to IT system. The suitable technique considered the compatibility with the definition of EIS that discussed in the earlier sections. Figure 7 shows more details of the requirement elicitation process. However, this section will give a very high-level discussion on the requirements gathering techniques, which are suitable for EISs.

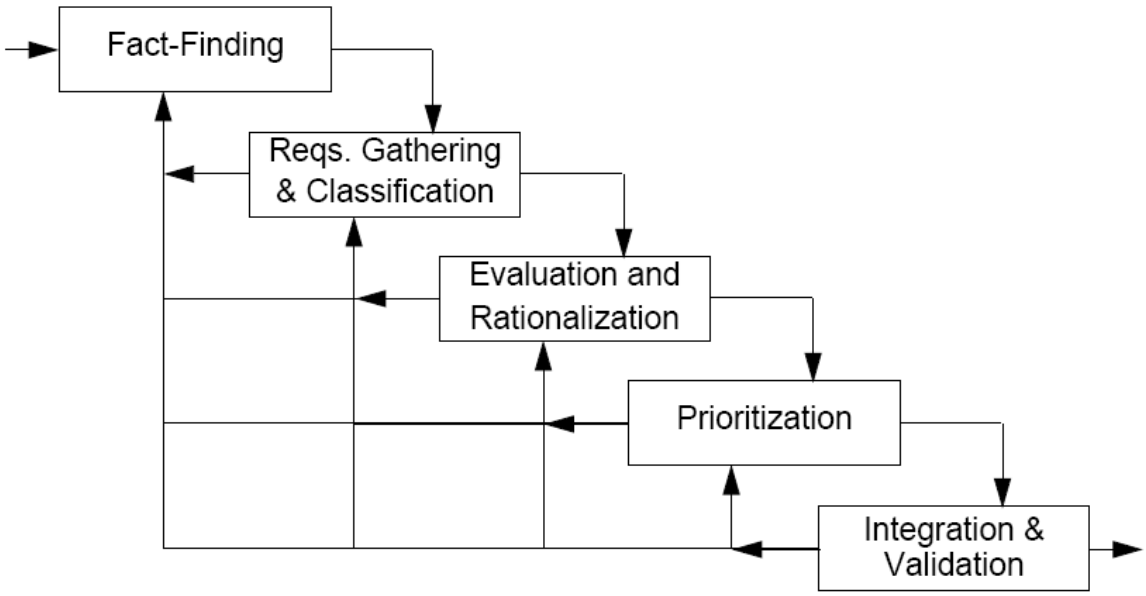

Figure 7: Requirements elicitation process model (Christel & Kang, 1992)

As it was discussed earlier in the SOA section, based on the goals of the system, the requirements of the human and the organization should be collected. As can be seen in figure 10, later in this chapter, collecting the correct requirements is one of the early stages in developing software systems. Indeed, this step is very important, because the rest of the software development will be built over requirements. In the organizational view, the business requirement plays the effective role too. In addition, "people organize because they believe that this is the most efficient way to reach their goals", hence the user requirements could be considered.

Literature introduced some requirements, which are useful for EISs (Kurosu & Yamada, 2002) such as Developability, Felxibility, Re-Configurability, Improvability, Maintenability, Accessability, and Comfortability.

Survey on the literature declares some point that seems interesting to be considered for the requirement elicitation of EISs. The first point about requirements of EISs is keeping the top down view. To collect the requirements of EIS, the collectors should consider the system as one system. Therefore, they should have a top down view overall system. However, a bottom up view is suitable for the later phases, when the details of the requirements are required, also for the review of the whole requirements; it would be good to have a bottom up view too. "Business process contains business

functions, to improve the quality of business operations; the decision makers should consider the whole process rather than functions. Therefore, it could be understood that the enterprise system should consider the whole business rather than just particular business functions“ (Davenport, 1990).

The next point in requirement elicitation considers the redesign process. Process redesign occurs when a company understand that their processes are not efficient. Five steps in process redesign are as follow (Kurosu & Yamada, 2002) (Davenport, 1990):

- “Develop business vision and process objectives”; “prioritize objectives and set stretch targets”
- “Identify processes to be redesigned”; “identify critical or bottleneck processes”
- “understand and measure existing processes”; “identify current problems and set baseline”
- “identify IT levers”; “brainstorm new process approaches”
- “Design and build a prototype of the process”; “implement organizational and technical aspects”

The third point in the context of the suitable requirement elicitation technique for EISs is goal oriented requirement engineering. Goal oriented theory could provide suitable approach for developing EISs because it can focus on the goals of enterprise. Goal oriented requirement engineering is concerned with the use of goals of the enterprise in different phases of requirement engineering (A. V. Lamsweerde, 2004). Axel van Lamsweerde defines goals in the same paper as follow: “Goal may refer to functional concerns or quality attributes”. Goal oriented techniques are generally a mixture of bottom up and top down techniques which is compatible with the first point that was mentioned in this section. There are various techniques in the area of goal oriented or goal driven requirement engineering, and it shows the interest of the industry and academics in this concepts. Goal oriented techniques have been used in various industries such as health care and air traffic control (A. V. Lamsweerde, 2004). All this facts conclude to the success of this theory in system development. However, I do not want to focus on any particular technique at this pint, but there are various techniques in literature, for example Goal Structuring Notation (GSN) (T, J, J, & S, 1998). The main point at this stage is to focus on goal-oriented theory for developing EIS. However, same as any other techniques there

are challenges for goal-oriented techniques too. For example, conflict between goals or their requirements could be a clear challenge in this method. Keeping the known benefits and challenges of this technique in mind, the research proposal in chapter 6 discusses this technique with more detail.

4.6 Conclusion

In summary, this chapter presented more information about another aspect of EIS, which is its architecture. As the result of reviewing the architecture, I went through a well-known technology that supports business-based systems. SOA, which were discussed in this chapter, also can be used for various purposes but considering business-based systems creates a common environment with EIS. However, SOA is a new and improving technology, therefore more changes and improvements will be expected in this area.

Besides, in the context of requirements in this chapter, flexibility of the system proposed as a desirable and challenging point. One approach to implement a flexible system is to follow the incremental model in software development. This chapter will conclude by providing a brief discussion about the recommended approach by expert system developer, for creating flexible and more realistic system.

4.6.1 Incremental Models

Two main software development process models are sequential models such as waterfall and revolutionary models such as incremental model (Pressman, 2000). As it can be seen in Figure 8 and 9, both models have four similar components such as analysis, Design, code, and test. These four components are the skeleton of the software development processes. The main difference between these two models is the amount of flexibility that they give to the model. The sequential model is famous for its static strategy, which is suitable for systems with static procedures that have no or little amount of changes during their lifecycle. On the other hand, revolutionary models are interested in dynamic systems which flexibility is one of their characteristics during their lifecycle. Considering the fact that flexibility is one of the desired requirements in EISs, evidence

declares that revolutionary models are more suitable for EISs rather than sequential models. However, it is clear that each of these models have their own pros and cons that should be dealt with during the development phases. In conclusion, this study will consider incremental model as the suitable model for developing EIS at this stage. Figure 10 illustrates the draft revolutionary model for EISs, which is the mixture of incremental and spiral model.

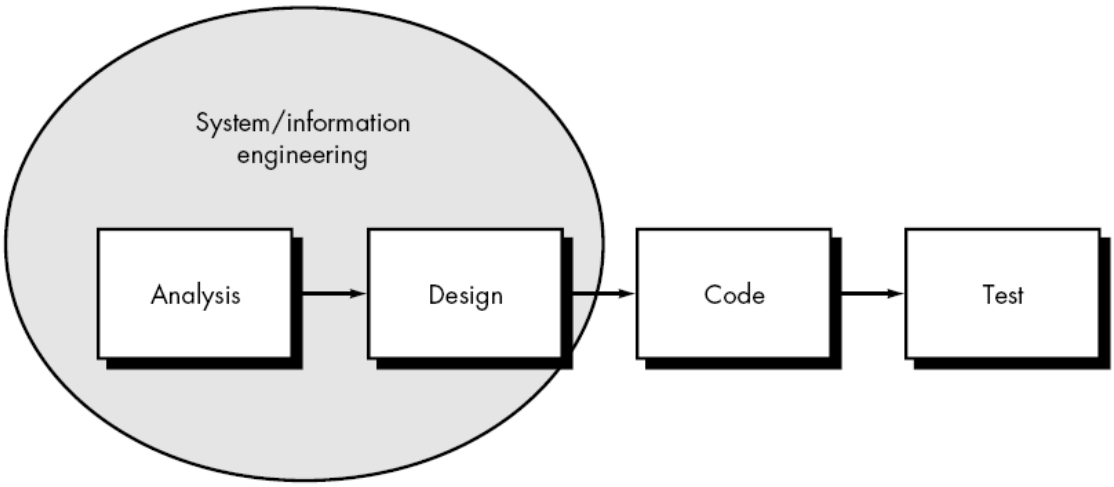

Figure 8: Sequential (Pressman, 2000)

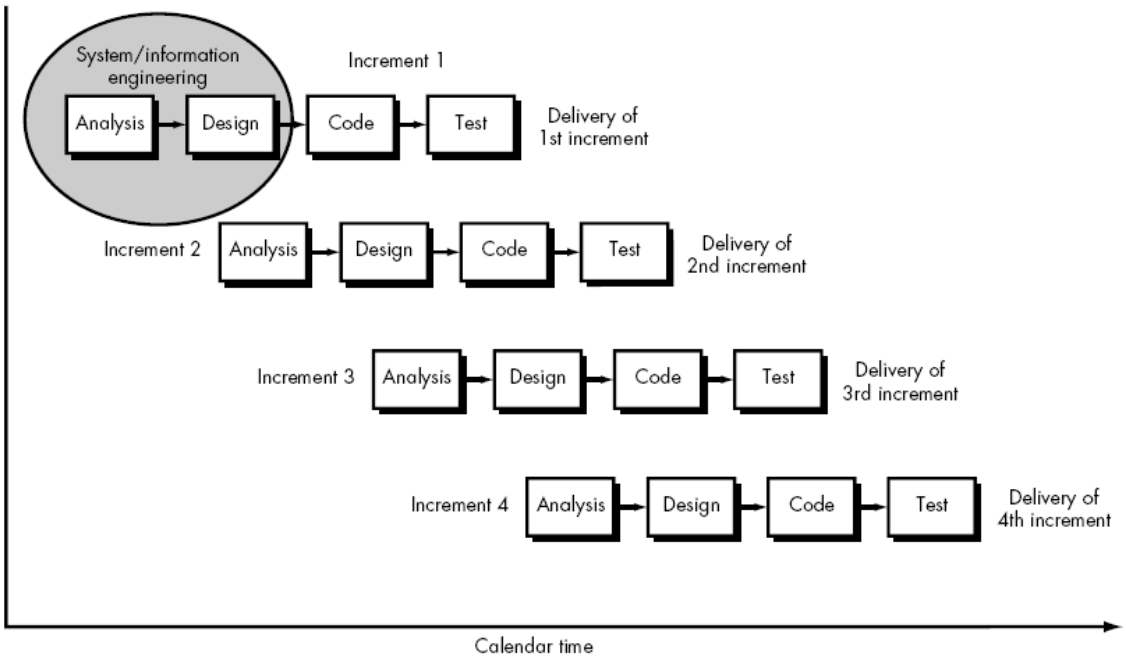

Figure 9: The incremental model (Pressman, 2000)

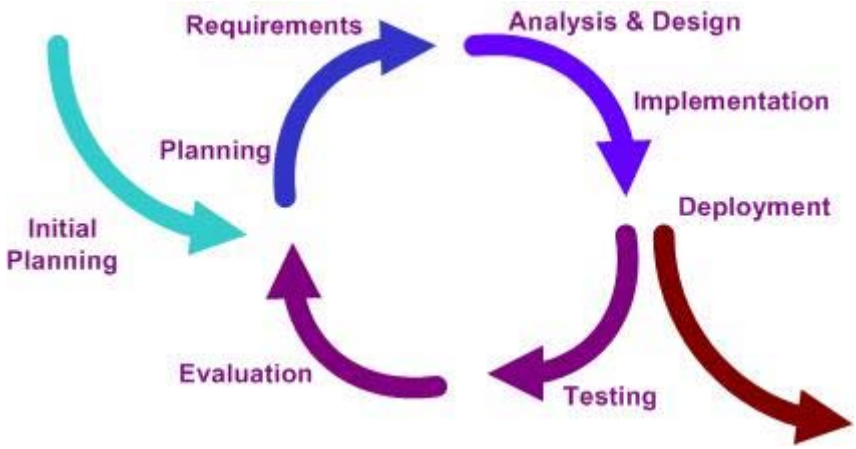

Figure 10: Draft revolutionary model for EISs (Wikipedia, 2008)

The effect of the revolutionary model will be seen in proposal chapter later in this dissertation. The next chapter will provide the summary of literature review that was presented in chapter two to four and the result of this review, which is discovering the gap, and main research questions that can trigger the future discussions and research.

Chapter 5

Research Question

5.1 Introduction

This chapter aims to summarize the literature review from the previous chapters. We also identify open research questions based on this summary, which lead to presenting a plan for future work in the following chapter.

5.2 Summary

We started by discussing the history of enterprises. After explaining that enterprises have existed since the First World War. We turned to identify the state of the art definitions in the literature. Several definitions were discussed and as the result, a working definition for EIS was created. Subsequently the benefits and challenges associated with this type of system were discussed and in the following sections the architecture of EIS were considered to show the different aspects of this type of system. Throughout the review on the architecture of these systems, the relevant quality attributes and stakeholders were discussed briefly. Moreover, service oriented architecture as an available solution for building business systems were studied. Other sections covered the main technologies that could contribute to EIS development. In addition, two available approaches for developing EIS were reviewed. The first one was based on industrial research from IBM, and the second one was based on object-oriented theories. The final section in this review focused on requirement engineering theories; because requirement elicitation is one of the early stages in developing systems. Specific research on goal-oriented or goal-driven requirement engineering was considered; this may be useful for

understanding business goals, which in turn may help improve how we develop EIS. This approach will be discussed in the next chapter.

5.3 Gap

Shrinking the gap between processes in IT-oriented systems and the equivalent business process in organizations have been a challenge for most systems, mainly for large organizations. Indeed, changes in environment that causes changes in a system is one of the main challenges in this area. Systems that can be flexible enough to deal with changes in environment and new innovations in functionality are desirable for enterprises; however, dealing with such flexibility adds a significant challenge in developing EIS.

The main questions arise when users observe difficulties in software systems that support large-scale organizations; questions such as the following:

- Why does an EIS not improve the functionality of business processes in organization?
- Why do certain systems not support organizational goals?
- Are organizations supposed to match their processes with the system or should the system be made compatible with the processes of organizations?

An important solution that is suggested in this area is to isolate each part of the system and make it stand-alone. As the result, changes in the environment or changes in technology will not affect the whole system. The gap between the IT Oriented system and organization is a challenging area that involve large amount of researchers. This study tries to restrict the domain of the challenge by looking at not all type of systems but only EIS. However, because of the loose definition of EIS and vague domain border, developing EIS become a challenging task.

However, industry and researchers are still interested in EIS because of their desirable benefits such as (Davenport, 2000):

- Cycle time reduction
- Faster information transactions
- Better financial management
- Laying the groundwork for electronic commerce

- Making tacit process knowledge explicit

Therefore, dealing with challenges of this type of system can provide benefits for industries and organizations.

5.4 Deriving the Research Question

After illustrating the motivation for this research, the next step is to clarify the research question. The following figure illustrates a mind map for a systematic questioning approach to help to extract a research question.

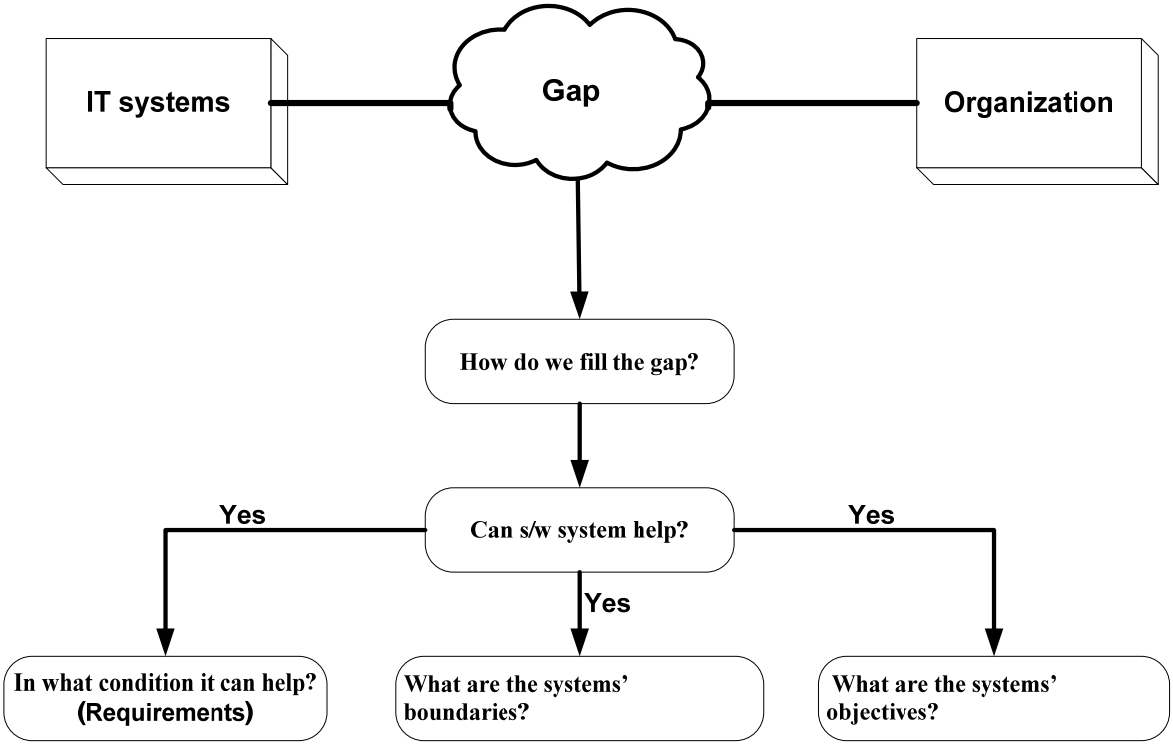

Figure 11: Approach to Research

Surveying industrial examples indicates the difficulties that exist in modern industry of the modern age (Laudon & Laudon, 2007; Stair & Reynolds, 2006). IT-oriented systems are an approach to face difficulties in modern industry by speed up business processes, make them more accurate, and provide access to distributed resources. Nevertheless, the role of the human in IT-oriented systems is still valuable because they provide innovative ideas; they supervise functionalities of the system to prevent faults

from propagating in some cases, and they can present solutions in cases where the system is faced with situations that are not programmed.

Indeed, IT-oriented systems improve business processes in some cases such as BMW that was mentioned earlier in introduction chapter. However, as it was mentioned in section 5.3 there is a gap between functionality of processes that are developed in IT-oriented systems and business processes in organization. This gap could cause unsatisfied result for business processes. An example of this problem was mentioned in the case of BMW in introduction chapter. Hence, studying the above argument illustrates the existence of a gap between functionality of business processes in IT-oriented systems, and business processes' functionality in active organizations.

Defining the gap that could be the hypothesis for this study leads this dissertation to discuss the research question. Figure 11 illustrates the approach to research question. This figure shows the process of systematically raising the research question by questioning the gap; and figure 12 illustrates the motivating questions in star shapes that rise during discussing the main questions in the rectangular boxes.

This paragraph explains the questions in figure 11 and 12. The main and high-level question is how to fill this gap? There is another question, which is why this gap should be filled (figure 12)? In the next level of the research question's hierarchy, the question is if software systems or IT-oriented systems can help to fill the gap. The parallel question is why software systems/ IT-oriented systems could help to fill the gap. This question leads to a series of other questions such as, is there any other solution for filling the gap? This dissertation tries to have a research on possible solutions and approaches. The last level of the research question's hierarchy consists of three main questions. The first one is, in which condition an approach can help to fill the gap. This question focuses on the requirements of EIS. The next question considers system's boundaries. However, the side question at this step is why boundaries of these particular systems matter? What are limitations of these types of systems? The last question in this series is about the objectives of these types of systems. What is the aim of these types of systems and what characteristics they should have?

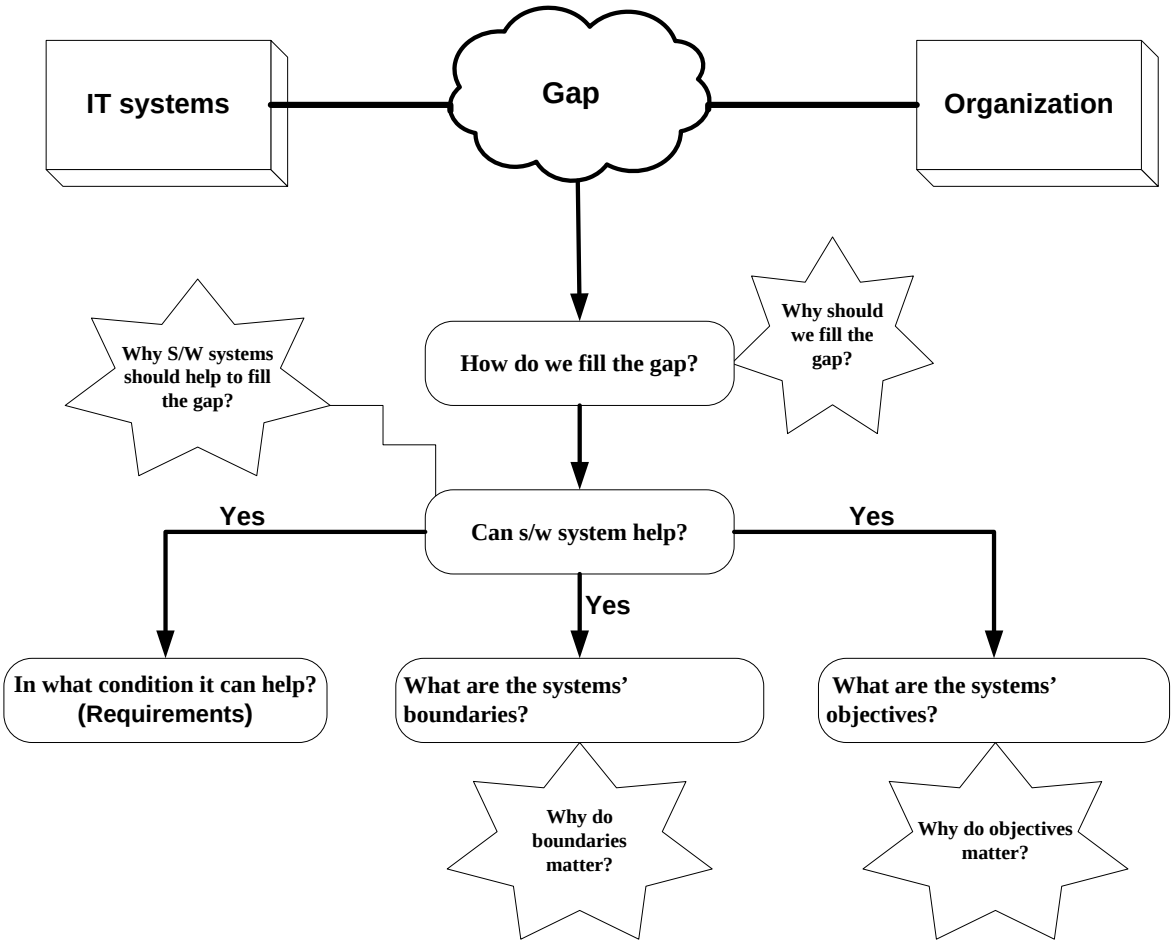

Figure 12: Motivating Questions

This literature review establishes the research question based on the arguments in this chapter and throughout this dissertation.

5.4.1 Research Question

Based on the collected information, there is a gap between functionality of business processes in IT-oriented systems, and their functionality in organizations. Subsequently the research question, which is derived from this survey, is as follow:

Can we build a software development methodology that will blend considerations of evolving business goals and the functionality of EIS?

Outcome: The outcome of this research question will be a development methodology.

Gap/Problem: The Gap, which this research question is referring to, is between evolving business goals and the functionality of EIS.

The result of literature survey is leading us to think about a development process that considers simultaneously business organizational goals, and IT-oriented systems to build more flexible EIS. The domain of flexibility should be defined to restrict the domain of the research. Therefore, the main two flexible categories that were emphasizes in literature review and interviews with industrial stakeholders are as follow:

- Flexible to accept changes in system components such as accepting new subsystems during the time
- Deal with people innovations and stakeholders new requirements that could be as the result of market changes, changes in goals.

5.5 Conclusion

Besides the main challenge that this survey deals with, which is the limitation of submitted materials, particularly about enterprise systems; we summarized our literature review findings, which in many cases based on company's technical reports and consultants opinions. In summary of the literature review, the gap, which is the were illustrated. Subsequently the line of the argument lead to present relevant question to bridge this gap and in the end the research question were discussed. The next step is to propose the approach for this research in the following months. Next chapter will cover this point.

Chapter 6

Proposal

6.1 Introduction

The aim of this chapter is to discuss a possible approach for addressing the challenging gap that was mentioned in the preceding chapter. As a result of the survey of the literature, it is clear that there are issues integrating IT-oriented systems with organizations. However, the aim of this dissertation is to focus on EIS rather than any other type of IT oriented system. EISs are large systems that connect and pervade the whole organization. Subsequently, developing such a large-scale system is very complex. Goal-oriented techniques, which are currently used in requirements elicitation, could be a helpful approach for developing better EISs.

6.2 Proposed Research

This section will discuss the aim of this research proposal and how we plan to achieve it. In addition, the next section will present some of the initial ideas that will be studied in more detail in future work.

6.2.1 Aim and path

As was mentioned earlier, the aim of this proposed research is to improve the process of developing EISs. We consider that a system desired by an organization should be developed by considering the structure of the organization; the system should be flexible enough to accept changes in the organization, and should satisfy the overall goals of the organization.

The path to do this research is firstly to increase my knowledge about goal-oriented techniques such as goal-oriented architecture. As was mentioned in the previous chapter, goal oriented techniques are mainly used in requirement phase and there are small signs of goal oriented technique in software assessment too. i* or i star framework (Axel van Lamsweerde, 2001), KAOS method (A. V. Lamsweerde, 2004), Goal Structuring Notation (GSN) (T, et al., 1998), etc. are techniques, languages, or methods that created in this area.

The second step is to analyze ideas of goal-oriented techniques in the context of iterative software engineering model. It was stated in chapter two that an iterative style of development is more suitable for EISs in compare to sequential model. The result of this analysis could lead to extending the idea of goal-oriented methodology to all software development phases.

6.2.2 Initial ideas

As was mentioned in the requirement section in chapter four, the main fact in goal-oriented or goal-driven methods aims to discover goals of the system's stakeholders. These goals can be a good start for building the rest of the system on top of them. One way of collecting goals of the system is using use cases. However, there are challenges in defining goals. One of the most familiar challenges is the conflict between goals and subsequent conflicts between requirements. In some critical situations, goals can be against each other; hence, the team of designers should choose one at a time.

The solution for dealing with conflicts of goals, however, can be hierarchy tree of goals. The basic idea at this stage is to create a tree of goals for the system, and break each goal into sub goals. The defined goals can be given priority, and the priority can be from right to left in tree's branches. The left node has higher priority than the right node for instance.

Hence, the first step of applying goal-oriented technique for EIS is to define enterprise's goals. The next step is to decompose the goals and break them in sub goals within the domain of the system. The decomposition will continue until all the possible goals and sub goals of the system are defined for the development team. The next step is

to define the requirements for satisfying these sub goals and continue it to the higher levels in the goal tree.

By using this goal-oriented technique, requirements of the system could be collected; therefore, when goals of the system change according to the environment changes, tracing and editing requirements is possible. This techniques hope to bring flexibility to the system development. Nevertheless, changing the requirements based on the changes in the goals cannot be done in one-step, whereas it needs to be done via various iterations. This will trigger the discussion in section 4.6.1 about iterative approach in software development. To have a flexible system that can tolerate changes in its goals and consequently changes in the requirements this dissertation considers iterative software engineering model. The six steps of software development process are defined as follow, considering goals of systems and goal-oriented techniques:

Planning: Defining goals of the system. Goals of the system can be collected by methods such as brainstorming with high-level stakeholders and technical users. Any method that can help the developer team to collect most accurate and complete list of goals is useful in this step; methods can benefit from textual report or graphical notations to collect and present goals that are more accurate from different stakeholders' point of view.

methods such as using graphical notations or textual reports, it mainly depends on the preference and level of stakeholders; also on the complexity of the system, moreover, it is possible to mix different methods to achieve the better result.

Requirements: Goal-oriented requirement elicitation techniques can be used to collect requirements of the system. Techniques such as GSN (Goal Structuring Notation) uses graphical notation which are supported by textual comments to illustrate requirements. As it was mentioned in preceding section, other examples of goal oriented techniques are i* and KAOS method.

Analysis and Design: System can be designed using collected requirements and goals of the system. Similar to planning phase, different methods can be used for analysing and designing the system. Various criteria can be used to measure the success of this step. Any methodology that is used for achieving a better design such as UML or other graphical or textual methodologies should hand in the practical result that can

satisfy requirements and open a clear path for the next step, which is implementing based on the design.

Implementation: programmer can use any suitable programming language and technology to implement the system based on the design.

Testing: Other than normal tests such as black box or white box, and any suitable test for coding and used technologies, there could be a test for checking goals of the system. The aim of this test is to check if goals of the system are still valid or they changes. Moreover, this test can check if defined goals are satisfied or not.

Evaluation: Evaluate if the system achieves its goals and how much resources were used to achieve defined goals. Moreover, system's goals could be evaluated for finding better goals.

6.3 Criteria of evaluating success

The main objective of this study is to suggest a framework for developing EIS based on goal-oriented theories. For evaluating this work, following criteria are considered:

Case study: The main result of this research will be applied to a case study. Using a case study will help to evaluate the results and methodologies more realistic and justifiable. In addition, it can help to make suitable decisions for next steps of the research. The case study should be limited enough so that it can be completed within the PhD registration period.

to be doable for the domain of this project within PhD period.

Publication: Throughout my research, I have planned to publish the result of each step; present frequent talks about the progress of the work and achieved results. Subsequently, I can collect feedback from other researchers in this area and this can help to improve the work and its evaluation in each step.

Compare with relevant technology (ies): Current technologies such as SOA for developing EIS and similar systems could be used as a criterion to evaluate the success of this research. It is likely to compare the positive and negative points of this work with related successful software engineering methodologies that mainly consider business goals.

Checklist: I also expect to create a checklist to evaluate the success of results within the case study. Checklist is one of techniques for evaluating case studies (Robson, 2002).

Questionnaire and interviews: These two methods are other possible approaches to collect stakeholder's ideas about the framework. Users who are involve in the domain of the work and other researchers in this area can give their feedbacks in the form of questioners and interviews.

6.4 Progress

I have already started interviewing people in this area who can give me realistic ideas about approaches used for developing large scale and complex systems. Moreover, I am investigating case studies; there is a potential source in the NHS connection for Health IT infrastructure. I am also attending newsgroups to start networking with people in this area.

6.5 Methodologies

The high-level research method which is considered for this study is qualitative. This method consists of mixture of methods such as 'Grounded theory', 'case study' and 'iterative and incremental strategy'.

The three traditional flexible design research strategies based on (Robson, 2002) are: case study, ethnographic study, and Grounded theory study.

A case study is about "development of detailed, intensive knowledge about a single case or of a small number of related cases" (Robson, 2002).

According to (Robson, 2002), typical features of this strategy are:

- Selection of a single case or cases of a situation, individual or a group of interest or concern"
- Study of the case in its context
- Collection of the information via a range of data collection techniques including observation, interview and documentary analysis.

Regarding to the definition and features of the case study strategy and the aim of this research, the future work will focus on the enterprise systems cases. However, some brief cases such as BMW, Mitsubishi, Amazon were reviewed during this period. They have been studied from their enterprise system point of view and their underlying organizational theories. The main data collection technique until this level was based on interview and reviewing their documents.

Another research methodology that may be suitable for this research is Grounded theory study. "The central aim is to generate theory from data collected during the study" (Robson, 2002). As enterprise systems research area is a new and popular topic in computer and business science, it motivates us to consider Grounded theory to the list of research methodologies. Grounded theory is "particularly in new, applied areas where there is a lack of theory and concepts to describe and explain what is going on".

According to (Robson, 2002) the typical features of this strategy are:

- "Applicable to a wide variety of phenomena;
- Commonly interview-based;
- A systematic but flexible research strategy which provides detailed prescriptions for data analysis and theory generation"

The last methodology that this dissertation includes to this list of possible methodologies is Evolutionary Software Process Model. This model will apply when "a set of core product or system requirements is well defined but details of product or system extension has yet to be defined" or because of the business and product changes or tight market deadlines a complete system cannot be developed. As evolutionary models are iterative, they enable the system to be improved during the time. According to the characteristics of enterprise systems, and their architecture, this type of development could be suitable (Kaisler, et al., 2005; Pressman, 2000). In addition, Evolutionary Software Process Model is a well known software development model that have been used in various projects already, therefore considering this model as the methodology could be software development approach and indeed more understandable for users with software engineering background and software developers.

6.6 Conclusion

Based on the collected information, in this chapter I proposed the approach with initial ideas and plan of attack that would be considered for this research during PhD period. Besides, the criteria to evaluate the result were one of topics that were discussed in this chapter briefly. Moreover, some of methodologies that could be used throughout this research were justified. To achieve goals of this research that were discussed earlier, I scheduled the following plan. This is an initial plan that could be changed throughout this research.

6.6.1 Schedule

The aim of this schedule is to declare the plan for this research in future. Tasks, which are in bold in table 1, are defined deadlines from department. Other tasks are mainly based on iterative model in software engineering. Developing initial ideas based on the information that are collected from literature is the first step in the first iteration. Literature review is a continuing task until final stage of this research. After collecting enough data that makes me confident to start proposing the first draft of framework (refer to task ID 1 to 5 in table 1). The next step analyses the given framework (refer to task ID 6 in table 1) and apply the result of this analysis to the framework in the next iteration (refer to task ID 8 and 10 in table 1). At this stage, two main iterations (refer to task ID 5,6,10, and 11 in table 1) are considered in the schedule. At the end of these iterations, I will evaluate results of this research in general (refer to task ID 13 in table 1).

ID	Task Name	Start	Finish	Duration	Q4 07			Q1 08
					Oct	Nov	Dec	Jan
1	**Literature review**	10/10/2007	01/09/2010	151w 1d				
2	Developing Idea 1	05/06/2008	05/06/2008	1d				
3	Collect data	05/06/2008	05/06/2008	1d				
4	**Qualifying Dissertation**	25/03/2008	30/06/2008	14w				
5	1st iteration: Propose the first iteration	01/08/2008	12/01/2009	23w 2d				
6	1st iteration: Analyse result of first iteration	01/08/2008	12/01/2009	23w 2d				
7	**Progress Report**	25/11/2008	30/01/2009	9w 4d				
8	2nd Iteration: Propose the second iteration	12/01/2009	10/07/2009	26w				
9	**Thesis Outline**	08/06/2009	30/06/2009	3w 2d				
10	2st iteration: Analyse result	10/07/2009	10/09/2009	9w				
11	1st step of evaluation and fixing problems	10/09/2009	08/01/2010	17w 2d				
12	**Thesis Audit**	18/01/2010	29/01/2010	2w				
13	2nd step of evaluation and fixing problems	11/01/2010	09/04/2010	13w				
14	**Thesis Seminar**	28/06/2010	30/07/2010	5w				
15	**Thesis Submission**	04/06/2010	30/09/2010	17w				
16	**Viva**	04/10/2010	03/12/2010	9w				

Table 1: Project plan

In conclusion, the aim of this chapter was to illustrate the overview on the initial plan for the future work and options that I am considering for achieving my goals, which is proposing a framework for developing EIS. This approach could be extended in most of other software system; therefore, it could be an approach for improving software system development. This is a dynamic plan that could change during the time according to the new information that I will collect and the type of case that I will continue working on it. Next chapter will present the preliminary result that I done to satisfy the initial proposed approach.

Chapter 7

Preliminary result

7.1 Introduction

The first step in iterations is to collect suitable and useful data that makes us confident to continue with the next iteration. For the first iteration, not only reading literature review but also interview with the people in the area of software development and management were carried out. This chapter will discuss the result of the three interviews that I had regarding EIS.

7.2 Interview Background

The first interview was done with Dr.Kiran Jude Fernandes from University of York Management School. His area of research includes IT system, therefore he has been doing research about management and organizational theories; he also studied various software systems that helped me to interview him for his point of view about EIS.

The second interview was with Patrick Willis. He is a senior IT architect in IBM. His position as a consultant from IBM gave him the opportunity to involve in various IT-oriented systems including EIS. Therefore, his wide view about various IT-oriented systems, their challenges and difficulties, and current approaches were interesting for keeping this research close to practical approaches.

The third and the last interview at this stage were with Keith Mantell. He is a Software architect and project manager at the Solution Architecture Team, IBM Rational. Throughout his interview, I collected his ideas about EIS and Service Oriented Architecture (SOA). His experience in developing large scale and complex systems and his research about SOA was interesting for this research proposal.

Following section is the summary and my discussion about the result of these interviews.

7.3 Result discussion

The result of these interviews illustrates that EIS is not a legal term. As the result, it is not easy to define EIS. All the interviewees were agree that EIS is mainly a big company but during the discussion they could not define 'big'. This fact declares the challenge that was mentioned earlier about ill definition for EIS. However, when they wanted to define the characteristics of EIS, some attributes such as data federation and involvement in business were distinguishing points. Keith Mantell believes that having an IT department in the company that had the ability to develop their software system makes a big company and enterprise. At this stage I argued that many big companies that have the potential to be enterprise purchases their system from professional software vendors rather than their IT department, because some companies such as IBM or SAP have many years of experience in developing software. Therefore, in the case of BMW, SAP could provide the suitable system to wrap BMW and its partners all together. It is reasonable that even big companies or enterprises prefer to provide their systems from IT professional companies.

Keith Mantell suggested SOA for developing EIS and Patrick Willis suggested Component Business Model for this purpose. However, based on the discussion that I had mainly with Keith Mantell, I concluded that finding the goals of the system's stakeholder and making use cases are practical idea for developing system. On the other hand, both individuals rejected the idea of developing system based on organizational theories, because it seems that in real world most enterprises do not follow organizational theories, hence it would not be practical. Dr Fernandes however had studied on organizational theories and he explained that there are hundreds of these theories at the basis of organizations. In conclusion, Patrick Willis suggests focusing the research to mapping to organizational boundaries. This is similar to the aim of this project, which is defining the objective, and boundaries of EIS to define the framework.

7.4 Writing Chapter

Writing a book chapter for 'Enterprise Information Systems for Business Integration in SMEs: Technological, Organizational and Social Dimensions' is one step toward developing my ideas about EIS and their boundaries. This book will be published in 2010 and its abstract is already accepted.

7.5 Conclusion

In summary, this chapter presents part of the work, which is done during literature review period. Besides, it presents preliminary results that were achieved from theses interviews. Limitation of the research and the potential area for future research were better understood with the help of these interviews.

In conclusion, this dissertation presents part of the review on the literature about EIS from software engineering point of view. As the result of the research, which is done throughout literature review that includes, brainstorming with my supervisors and interviewing expert people in industry and academy some decisions about future work are made.

Future work of this research will include study on goal oriented techniques and uses cases to evaluate their effect on developing EIS. Considering the research proposal in chapter six, this research will continue on collecting suitable data for creating a development process for developing EIS.

References

1. Armour, F. J., Kaisler, S. H., & Liu, S. Y. (1999). A Big-Picture Look at Enterprise Architectures. *IT Pro, IEEE*.
2. AskOxford (2007). Oxford Dictionaries. http://www.askoxford.com
3. Bass, L., Clements, P., & Kazman, R. (2003). *Software Architecture in Practice*: Addison-Wesley, Pearson Education,Inc.
4. Bellman, B., & Rausch, F. (2004). Enterprise architecture for e-Government. *Electronic Government*.
5. Brien, J. (2002). *Management Information Systems* (5th ed.): George Werthman.
6. Carter, S. (2007). *The New Language of Business: SOA & Web 2.0*: IBM Press.
7. Christel, M., & Kang, K. (1992). *Issues in Requirements Elicitation*: CMU/SEI-92-TR-12.
8. Clifton, H., Ince, D. C., & Sutcliffe, A. G. (2000). *Business information Systems* (Sixth ed.): Pearson Education.
9. D'Souza, D., & Wills, A. (1999). *Objects, Components, And Frameworks with UML. The Catalysis Approach*: Addison Wesley.
10. Davenport, T. H. (1990). The new industrial engineering: Information technology and business process redesign. *Sloan management review*.
11. Davenport, T. H. (2000). *Mission Critical: Realizing the Promise of Enterprise Systems*. Boston, MA, USA: Harvard Business School Press.
12. Dictionary, B. (2008). http://www.businessdictionary.com/definition/organization.html.
13. Fruin, M. W. (1992). *The Japanese enterprise system*: Oxford university Press.
14. GeneralElectric (2008). Product and services, http://www.ge.com/products_services/index.html.
15. Graham, I. (2006). *Business Rules Management and Service Oriented Architecture: A Pattern Language*: John Wiley & Sons.
16. Husein, T., Moreton, R., Sloane, A., & Knoll, H. (1999). A process improvement framework for small to medium sized enterprises taking into consideration electronic communication technologies. *Proceeding of the forth UKAIS conference University of York*.
17. Jackson, A., & Fritzsche, M. (2007). *MODELling solution for comPLEX software systems*.
18. Jamshidi, M. (2005). System-of-Systems Engineering - a Definition.
19. Jessup, L., & Valacich, J. (2006). *Information Systems Today, Why Is Matters* (2nd ed.).
20. Kaisler, S., Armoiur, F., & Valivullah, M. (2005). Enterprise Architecting: Critical problems. *Hawaii International Conference/ IEEE*.
21. Kurosu, N., & Yamada, S. (2002). Living Manufacturing Systems with Living Organizations. *Conceptual Modeling for New Information SystemsTechnologies*.
22. Lamsweerde, A. v. (2001). Goal-Oriented Requirements Engineering: A Guided Tour. *Proceedings RE'01, 5th IEEE International Symposium on Requirements Engineering, Toronto*.
23. Lamsweerde, A. V. (2004). Goal-Oriented Requirements Engineering: A Roundtrip from Research to Practice. *Proceedings of RE'04, 12th IEEE Joint International Requirements Engineering Conference, Kyoto, Sept. 2004, 4-8*.

24. Laudon, J. P., & Laudon, K. C. (2007). *Management Information Systems* (10th Edition ed.): Prentice Hall.
25. Life, S. (2008). *Your World, Your imagination,* http://secondlife.com/.
26. Lukasik, S. J. (1998). Systems, systems of systems, and the education of engineers. *Artif. Intell. Eng. Des. Anal. Manuf., 12*, 55--60.
27. Mitsubishi (2007). About Mitsubishi. http://www.mitsubishi.com/e/group/about.html.
28. Pohle, G. (2005). *Component Business Models*: IBM.
29. Pressman, R. (2000). *Software Engineering, A Practitioner's Approach* (fifth ed.): McGraw-Hill.
30. Regner, P., Wiesinger, T., Kung, J., & Wagner, R. (2004). Software Acquisition Based on Business Models. *Electronic Government*.
31. Robertson, P. (1997). Integrating legacy systems with modern corporate applications. *Commun. ACM, 40*(5), 39--46.
32. Robson, C. (2002). *Real World Research* (2nd ed.): Blackwell.
33. Rudolph F. Crew, Ed. D., & Chancellor (2008). Virtual Enterprise, http://www.tnellen.com/school/ve.html.
34. Simon, E. (1996). *Distributed information systems: from client/server to distributed multimedia*: McGraw-Hill.
35. Sprott, D. a. W., L. (2004). Understanding Service-Oriented Architecture. *Microsoft Architect Journal*.
36. Stair, M. R., & Reynolds, W. G. (2006). *Principles of Information Systems*: Thomson.
37. STEP (1991). *Software Test & Evaluation Panel (STEP), Requirements Definition Implementation Team. Operational Requirements for Automated Capabilities, Draft Pamphlet (Draft PAM), April 23, 1991.* http://www.sei.cmu.edu/pub/documents/92.reports/pdf/tr12.92.pdf: CMU SEI.
38. Strong, D. M., & Volkoff, O. (2004). A roadmap for Enterprise system implementation. *IEEE Computer Society*.
39. T, K., J, M., J, M., & S, W. (1998). The goal structuring notation: a means for capturing requirements, rationale and evidence. In A. J. Vickers & L. S. Brooks (Eds.), *Requirements Engineering at the University of York*: University of York.
40. Travica, B. (2005). Virtual Organization and Electronic Commerce. *36, No. 3*.
41. Vickers, A. J., & Brooks, L. S. (1998). *Requirements Engineering at the University of York*: University of York.
42. Wikipedia (2008). *Iterative and incremental development*.
43. WordNet (2007). A lexical database for the English language. http://wordnet.princeton.edu/perl/webwn?s=enterprise

(This report is submitted in the University of York for the Literature Review dissertation in the first year of P.h.D)

The End

June 2008