Critical Success Factors for the Migration of Legacy Information Systems to SOA

Master's Thesis, December 2011

Abdoel Gafoer Shaief

Critical Success Factors for

The Migration of Legacy Information Systems to SOA

THESIS

Submitted in partial fulfillment of
The requirements for the degree of

MASTER OF SCIENCE
In
COMPUTER SCIENCE
TRACK INFORMATION ARCHITECTURE

By

Abdoel Gafoer Shaief
Born in Paramaribo, Suriname

Web Information Systems Group
Department of Software Technology
Delft University of Technology
Delft, The Netherlands
http://eemcs.tudelft.nl

Logica
Working Tomorrow
George Hintzenweg 89
Rotterdam, The Netherlands
http://www.logica.nl

Critical Success Factors for The Migration of Legacy Information Systems to SOA

Author: Abdoel Gafoer Shaief
Student ID: 1197681
Email: m.s.abdoelgafoer@student.tudelft.nl

Abstract

Legacy Information Systems abbreviated as LIS are information systems that have been developed and deployed in the past and have been running critical business processes within an enterprise information system.

Legacy Information Systems are considered the backbone of the organizations business since these systems are robust and carry valuable information which is of significant business value to the company. However, due to the increasing demand of business and organizational agility these systems need to be modernized retaining their existing assets. The concept of a Service Oriented Architecture provides organizations the ability to reuse their existing assets and contributes to a higher level of organizational agility. Therefore, the possibility to modernize these systems with their existing assets toward a Service Oriented Architecture becomes very interesting for organizations coping with LIS's. However, migrating toward a SOA can be a very risky and complex undertaking. Few migration projects up until now have been completed successfully. Therefore, it is very interesting to perform research in this area. This research focuses on the critical success factors that contribute to successful LIS to SOA migration.

The main adopted research method in this study has been qualitative research and evaluation based on a multi case studies at 4 large organizations. Through extensive literature examination, a framework of critical success factors for the migration of legacy systems onto a SOA has been developed. The framework has been evaluated using the case studies as input. In this research, we concluded the success factors found in our case studies through cross case analysis and factor analysis.

In total 42 critical success factors were identified, of which 19 were most critical based on evaluation of these factors at 4 large globally known organizations where migration projects were carried out.

Key words: SOA, Legacy Information System, Migration, Successful SOA Migration, SOA Critical Success Factors.

Graduation Committee:
Prof. dr. ir. G.J. Houben, Faculty EEMCS, TU Delft
Dr. ir. A.J.H. Hidders, Faculty EEMCS, TU Delft
Dhr. B. de Neef, Manager, Logica
Dr. ir. J. van den Berg, Faculty TPM, TU Delft

Acknowledgements

This document has been produced in the master thesis project as part of the Master program Information Architecture at Delft University of Technology (TU Delft) from May 2010 until April 2011. The work on the project took place at the International Business Consulting Organization Logica located in Rotterdam.

This thesis could not have been completed without the guidance, assistance and support of a number of people that I would like to thank now.

First of all, my immediate supervisor at the TU Delft, Jan Hidders, who always made time to discuss the thesis project and was very helpful and patient in determining the theoretical and practical instructions. My professor, Geert-Jan Houben, who from a distance, was critical of my work when it was necessary and provided detailed comments during our biweekly colloquiums. The external committee member, Jan van den Berg, whose external view as an expert during our discussions was extremely helpful in allowing me to place my work in a much broader perspective.

My supervisors at Logica, Bert de Neef, Edwin Essenius and Eric van Ommen, who arranged all the resources I needed to be able to complete my work and stimulated me and kept me going especially during periods where the inspiration and motivation was lacking. I would also like to thank the consultants of the 4 organizations for answering countless of my questions during the evaluation process of this thesis which was of great value for my thesis.

And finally my direct colleagues at the Working Tomorrow department who made it easy to come into office every day for over 11 months without getting bored. The environment at Logica was stimulating to work in and I can recommend performing an internship at Logica to anyone that is interested.

Shaief Abdoel Gafoer
Delft, the Netherlands
August 10, 2011

Contents

List of Figures

List of Tables

Part I Preliminaries

1. Introduction

1.1 Background

Many organizations still rely on software systems that are 10 to almost 20 years old. These systems are slow, inextensible, running on old or obsolete hardware causing organizations to spend an excessive amount of money on maintenance. However, these systems are mission critical since they are embedded with and execute the core business processes that create profit for organizations. These old systems are called *legacy systems*.

The cost for maintaining these outdated systems keeps accumulating with higher margins every year causing organizations to spend more money just to keep their business running on a regular basis [1]. Therefore, there is a rising demand for using newer technology and concepts to modernize these legacy systems which in turn needs to result in reduced maintenance costs, improved flexibility and most importantly without losing business value. Even for successful companies this indicates utilizing and deploying state of the art technology in order to prevent coping with integration, collaboration and transformation concerns in the future. A service oriented approach is often applied to support and modernize organizations relying on legacy systems. SOA is an acronym for Service Oriented Architecture and strives for reusability and composability of workloads. This involves decomposing large processes and systems into more reusable and autonomous workloads, also known as services.

Service orientation plays a central role during legacy transformation projects, because architecture based on services is often characterized for its high level of business agility and provides reuse of existing assets. More on the concise distinction and definition of the different types of services follow in chapter 3.

1.2 Problem

Most of the Dutch Financial sector such as banks and insurance companies still utilizes a broad range of different types of legacy systems, which varies from redundant databases with unnecessary tightly coupled tables, inconsistent and non reusable coding in almost obsolete programming languages to inextensible mainframe and ERP systems [2].

Nowadays business is becoming very tight due to a high level of competition, in order to stay ahead of your competitors, businesses and organizations need to be focusing on agility. Businesses have made large investments in their supporting IT environments. However, these monolithic and silo based architectures and IT-landscapes are unable to provide the capabilities in order to stay competitive and act upon market or regulation changes. There is a real gap identified between the existing situation and the future or preferred situation.

Businesses face a real dilemma, since most of them are aware of the fact that modernization of their current IT is the way to go however, without throwing away their investments, by reusing most of the existing capabilities in a renewed or more stable environment.

There are several strategic approaches in order to transform or modernize a legacy environment. These alternatives are: migration, wrapping, redevelopment and maintenance. Based on the type of business contribution/value and system quality, one or more alternatives should be applied. These alternatives, their benefits and drawbacks have been introduced and described extensively during the literature study as part of the preliminary research for the Thesis project.

Several Legacy transformation projects have been executed in the last couple of years, mostly in terms of migration toward a Service Oriented environment as the preferred or expected future situation. During these projects several attempts have been made in order to analyze and obtain the hidden data in order to transform these entities into reusable functions and components. Each project is carried out by hours of brainstorm sessions in order to come up with a bulletproof migration strategy and SOA architecture. Over the past few years many SOA migration projects have been performed. However, there is not one methodology which focuses on both the business and the IT perspective. Moreover, the several methodologies that have been published are often not (completely) evaluated on large Legacy modernization projects. On top of that, organizations are reluctant to use a methodology they are not familiar with due to the unknown aspect and the potential high risk. The organization where this graduation project was carried out also developed a SOA migration methodology which however was almost never applied during legacy modernization projects. The reason behind this is the same as mentioned above, organizations are reluctant to use a complete new and unfamiliar methodology which will be costly and time consuming. There are more than 150 methodologies claiming successful LIS to SOA migration however most of these are seldom complete. There is no coverage of both the business and IT perspective, it is often very abstract or often too detailed and technical for the business owners and business stakeholders to understand how this can improve the business. There Is a much greater demand for a abstract set of success-factors based on best practices compared to a methodology which mostly focuses on part of the migration process namely the business process or the data migration. Therefore, there is a great need to abstract from all these methodologies and provide a set of consistent Critical Success Factors (CF's) which contributes to legacy systems migration success.

1.3 Focus & Demarcation

The main focus of my thesis project will be placed on the migration of legacy system towards a Service Oriented Enterprise. In this research, the aim is to identify a list of critical success factors for the migration of legacy systems onto a SOA. A Critical Success Factors (CSF) analysis tries to identify a small number of goals or factors that, if they

are reached or successfully completed, will tend to predict or indicate success [3]. While CSF analysis has often been used to elicit information requirements in the systems analysis and design phase of systems development, they could also be used to identify project success factors. In this case the elements which contribute to successful migration of legacy systems into SOA will be studied and collected by performing qualitative research in terms of conducting interviews at 4 organizations where legacy migration projects have been carried out.

1.4 Research Goal

The main research goal of this study is to identify, collect, analyze and evaluate the critical success factors or critical elements that contribute to successful SOA migration for organizations coping with legacy systems. In order to reach the main goal the qualitative research approach will be applied in terms of designing semi structured and closed questions after which interviews will be conducted in 4 different companies. These questionnaires will be designed by studying literature regarding 'Legacy to SOA' success factors and theory regarding effective qualitative research methods. The findings after the literature study and qualitative interview method will be collected and analyzed using the cross-case synthesis technique. This technique is extensively described in section 2.4.1.

1.4.1 Main Objective

In order to develop a framework consisting of Critical Success Factors (CSF's) based on theory and practice, the main research question which needs to be answered is:

How can we design a framework of CSF's that contributes to successful SOA migration to maximize the level of success in legacy systems migration projects?

In order to reach the main objective several research questions have been constructed:

1. What is IT project success and what are the critical success factors for the Migration of legacy information systems to SOA from scientific perspective?
   Strategy: analysis and study of literature regarding Legacy migration methodologies, best practices and elements that contribute to the successful migration of legacy systems. Also analyze and study frameworks and methodologies related to IT project success in general which will be applied to the critical success factors specifically for LIS to SOA migration projects.
   Objective: Build a generic theoretical model of IT project success which will be utilized to structure and formalize the critical success factors determined in the context of LIS to SOA in a scientific approved model.

2. What are the critical success factors for the migration of legacy information systems to SOA based on practical experience?
   Strategy: qualitative research, design a multi-case study supported by semi structured questionnaires and conduct interviews at 4 different organizations where legacy migration towards a SOA environment has been performed with both good and less success rates.
   Objective: The goal is to investigate, improve the theoretical framework of Success Factors for Legacy to SOA migration by using 4 independent organizations where migration projects have been carried out as input. The objective here is to fill in the missing factors (if there are any) which were not determined and accounted for within the scientific area of this topic.

3. Given the answers on question 1 and 2 how can we design a framework of critical success factors for the migration of Legacy Systems to a SOA?
   Strategy: generalization of success factors supported by using a cross-case analysis for empirical findings of the 4 independent organizations onto the findings from literature.
   Objective: The objective is to combine factors determined in RQ1 and RQ2. After these factors have been combined and are given a place within the Theoretical model of critical success factors the model can be evaluated in order to determine which dimensions and which factors were deemed to be important during the 4 independent migration projects. As a result of this part the most critical success factors of the Theoretical Model will be determined and a verdict about the generalizalibility based on these 4 cases will be provided.

1.5 Research Approach

In order to answer the main question an inductive research strategy was taken (see figure 1). During the process of research several phases are identified in order to answer every sub question.

In *phase I* research question 1 will be answered by performing extensive literature survey where relevant theory and frameworks regarding LIS to SOA transformations will be studied. As a result of the survey all the relevant and critical material determined will be used as input for the second phase.
The work performed within this phase consists of 2 independent parts which are analyzing and studying critical success factors for the migration of LIS to SOA and, secondly, understanding and studying theoretical frameworks and methodologies concerning LIS to SOA migrations. As a result of the first part an inventory of critical success factors will be developed. As for the second part of the research, a theoretical model will be developed which will be used to construct and formalize the critical success factors using a methodological foundation. This is supported by using the Seligman framework [4] and the IT Project Success Model [33] will be used as basis for the development of the theoretical framework

for critical success factors. The answer to question one will provide us with an analysis of critical elements on both the business perspective and IT perspective. As a result of the complete first phase a theoretical framework will be developed consisting of list of factors and 4 dimensions in which these factors are placed based on their contribution of reaching the objectives of these dimensions. The Theoretical Framework of Critical Success Factors will be used as input to initiate the second phase where case studies will be executed. The actual case study will be designed and executed by using questionnaires and conducting interviews where open questions and semi structured questions will be asked which are related to the elements and criteria from the success factor model.

In phase II several case studies will be performed where elements that contribute to legacy migration success will be identified by conducting interviews at 4 different large scale organizations where legacy migration projects have been performed in the past.
Research question 2 will be answered by using the qualitative research method in terms of open, semi-structured and closed questions. Moreover, a questionnaire will be handed out in order to evaluate the strength of each sub factor and criteria within the model according to the findings of practitioners. The objective of constructing open questions was to improve and complete the Theoretical framework of Critical Success Factors with additional factors which were not determined in phase 1(extensive literature study). The objective of the closed and semi structured questions was to evaluate, which was performed during phase III, the Theoretical framework by using only the 4 independent migration projects as input and the level of importance of each factor placed on a likert scale(1 to 7). The goal of the second phase is to identify the critical elements or success factors from the perspectives of the field experts in order to improve the Theoretical Framework of Success Factors.

In *phase III* the results of the second phase were investigated and the evaluation of the Theoretical model was performed. The level of generalizability based on independent projects carried out in 4 large organizations in the financial sector is described in this phase. In this phase research question 3 is accounted for by using the results of the evaluation performed in phase 2 and the cross-case analysis technique in order to categorize and find the most critical success factors for the successful migration from LIS to SOA. The cross-case analysis technique is explained in section 2.4.1.

The final phase, which is not a real phase during project execution, describes the concluding remarks and the future recommendations based on the findings in phase 3. In order to provide the reader with a complete overview of the steps which were carried out to complete the project, a detailed project approach is depicted in figure 2.
The high level project approach is also given in the lower left corner which will be used to provide a more detailed description of the steps taken and the supporting methods which were used within each (literature survey, case study design, case study execution and generalization of Success Factors) of these 4 steps.

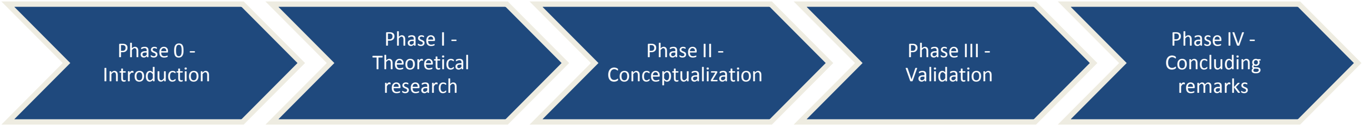

Phase 0 - Introduction
Phase I - Theoretical research
Phase II - Conceptualization
Phase III - Validation
Phase IV - Concluding remarks

Figure 1 – High level Project Approach

1.6 Structure of the thesis

The process, results and findings during the project will be expressed in the thesis document which will consist of at least 40 pages written text excluding appendices. The outline of the thesis is presented below.

Thesis Outline

Phase 0 - Introduction
Chapter 1 – Introduction
Chapter 2 – Research methodology
Chapter 3 – Project environment

Phase I – Theoretical Background
Chapter 4 - Legacy Information Systems & SOA

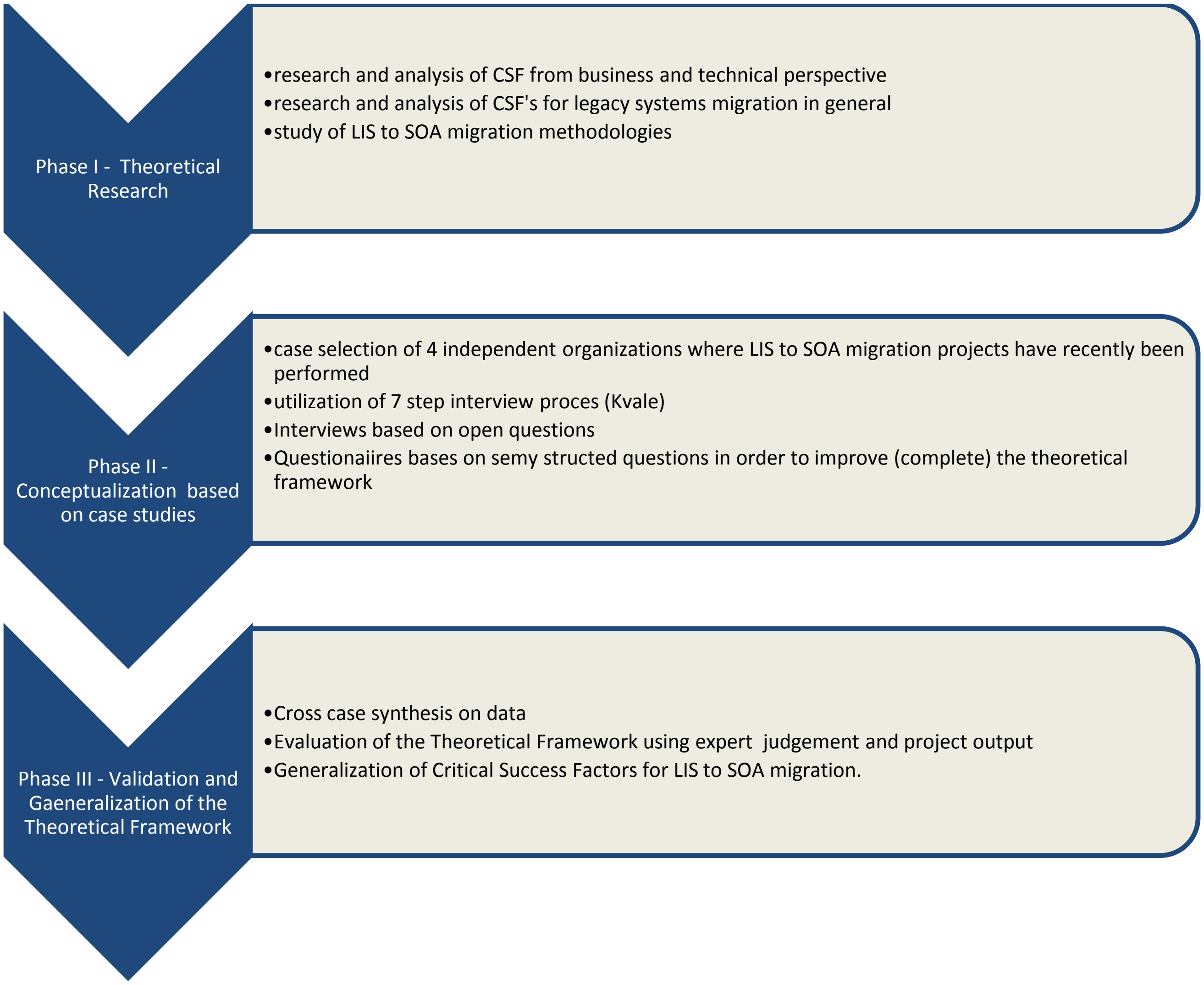
Phase II - Conceptualization
Chapter 5 – Theoretical Framework
Chapter 6 – Critical Success Factors

Phase III – Validation
Chapter 7 – Empirical Findings & Evaluation

Phase IV – Concluding remarks
Chapter 8 – Conclusions
Chapter 9 – Limitations & Future Work

2 Research Methodology

The research approach or methodology applied for the thesis is described in section 1.5. However, the methods and methodologies which were used to carry out tasks within each individual stage will be described in more detail in this chapter. The high level project approach consists of 5 phases, each contributing and adding more information compared to the previous stage. Out of these 5 phases, this chapter will focus on phase 1 up to 3 since phase 0 and 5, respectively introduction and conclusions, are not real phases during project execution. The 3 phases theoretical background, conceptualization and validation are depicted in the figure below. The complete process which was followed and the methods used supporting each stage to carry out the project starting from Literature survey up until the generalization of Success Factors is described in the following subsections.



Phase I - Theoretical Research

- research and analysis of CSF from business and technical perspective
- research and analysis of CSF's for legacy systems migration in general
- study of LIS to SOA migration methodologies

Phase II - Conceptualization based on case studies

- case selection of 4 independent organizations where LIS to SOA migration projects have recently been performed
- utilization of 7 step interview proces (Kvale)
- Interviews based on open questions
- Questionaiires bases on semy structed questions in order to improve (complete) the theoretical framework

Phase III - Validation and Gaeneralization of the Theoretical Framework

- Cross case synthesis on data
- Evaluation of the Theoretical Framework using expert judgement and project output
- Generalization of Critical Success Factors for LIS to SOA migration.

Figure 2 – Detailed level project approach

The underlying notion of this research is based on the inductive research strategy, where first observations and measures are made in order to detect and define patterns. Secondly a tentative hypothesis is designed which we can research in order to develop general conclusions and theories based on the observations, patterns and hypothesis that were drawn. The goal of the research is to develop a framework consisting of general critical success factors (CSF's) from specific observations within the scope of migration from legacy systems to a SOA. Based on the desired objectives and the provided approach the inductive research strategy suits best for this type of study [5]. Therefore the research approach taken to execute the project is adapted from the inductive research strategy. The process steps of inductive research in general and the project approach for this research are depicted in figure 2.

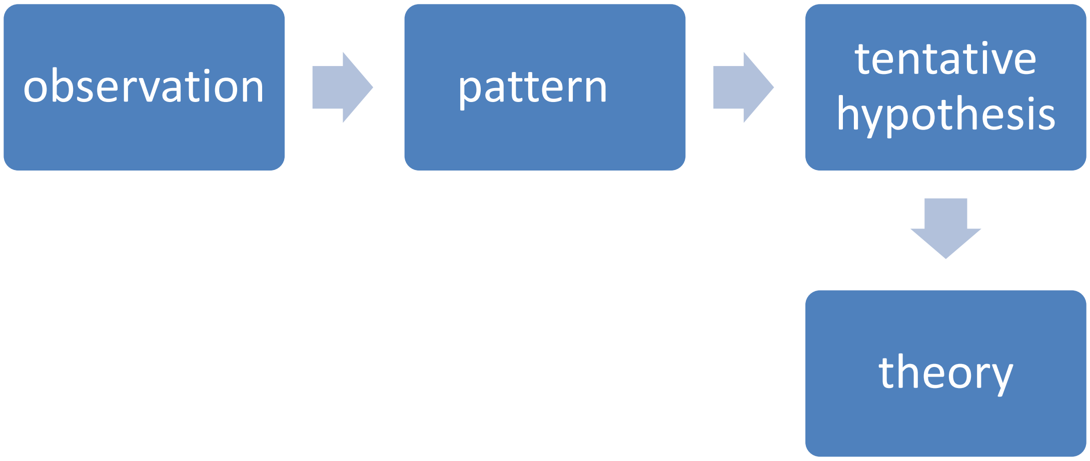


Figure 1 - Inductive research strategy

2.1 Theoretical Research

The research study which was performed here is described in chapter 5 and 6. The type of research which was executed falls in the category of descriptive research since the goal and the research question which needs to be answered is a "what" question which is descriptive one. The main research question that was mentioned in the previous chapter is:

"What elements contribute to successful SOA migration and how can these elements be generalized to maximize the level of success in legacy migration projects?"

Before the process of the literature survey is described in more detail, decomposition and definition of the terms used in the main research question is necessary to understand the approach taken during the literature survey.
The term *element* refers to a component or factor of a whole or one of the parts which is necessary to take into account when migrations of legacy information systems need to be carried out.
Several elements or factors contribute to migration success of legacy information systems.

SOA is an acronym which stands for Service Oriented Architecture which basically indicates that the World consists of consumers and providers of services which request upon one another using a services registry. In relation to the business perspective of an organization, a SOA regulates and controls the services provided and demanded by both the business and its environment. In a more technical sense, SOA is an approach to implement several design guidelines such as loose coupling, reusability, composability, service contracts and service discovery. More about design guidelines and SOA is extensively described in chapter 5 and 6.

Another important term is *project success* which is defined as the achievement of the organizations objectives, within a granted budget, within a specific time frame and most importantly, the quality of work delivered.

The term legacy system or LIS, which stands for Legacy information Systems, is the acronym used by most scientific writers. These types of applications are systems which still rely and run on mainframe systems. Most of these applications are maintained by programming in an obsolete programming language such as COBOL. Moreover, legacy information systems are not extensible or reusable, operate as silos within organizations and are extremely difficult and costly to maintain. Many of the Enterprise Resource Planning systems which were designed and built 20 to 30 years ago are defined as legacy information systems not because of their age but because these systems operate as silos and were written in obsolete programming languages but most important of all these type of systems are not capable of meeting changing business requirements in a cost effective and time efficient way. However, these systems are very robust and provide the business value that is actually needed by the business environment.

Finally, the term *migration* involves transferring the current system to a lower cost platform while keeping most of its business value and system functionality [2]. Migration allows companies to retain their existing functionality without having to completely redevelop them. This strategic and technical approach is often seen as the most effective long term solution when legacy systems need to be modernized toward a Service Oriented Architecture [6].
A SOA approach can be utilized to reuse the business logic which is contained and locked in the legacy systems and provide them as reusable services in order for other business units to effectively share and reuse information and knowledge. The benefits of SOA and the migration methods often applied on legacy systems will be described in more detail in chapter 4. Before continuing with the actual methodology applied during this research, the above mentioned terms needed to be described in more detail in order to understand the terms separately and how they stand in relation to each other and more importantly how they contribute to obtaining and the development of the framework consisting of Critical Success Factors (CSF's).

The first step in our research was surveying the subject in order to obtain a framework of CSF's. During the process of investigation several sources were used to gather information about this subject. Sources such as digital libraries, scientific papers and books were all analyzed. Before studying the factors related to SOA migration, a theoretical framework needed to be constructed in order to position the critical success factors. Therefore, an extensive study of different software development methods and evaluation frameworks led to the design of the theoretical framework independently of the core research. It was necessary to construct the theoretical framework since a firm scientific model was needed to separate the critical success factors from each other. The analysis and development of the theoretical framework is described in chapter 5.

The main terms or keywords which were used during the research are: SOA migration success, SOA success factors, legacy information system migration, legacy systems modernization success and project success. An extensive and broad collection of scientific research papers were studied which led to the identification of CSF's for the migration of LIS to SOA. After determining these CSF's, these factors were applied to the Theoretical framework which was independently developed. The theoretical framework consists of several dimensions which was used as a classification for a specific set of CSF's based on the higher objective that is reached when a certain group of factors are taken into account. The findings during the first step are described in extensive detail which can be found in chapter 6.

2.2 Conceptualization – Case Study Design

After having completed phase I where extensive literature study was performed, a case study design was the next step in the process. A case study is a method of research in social science which is based on the investigation of a single entity group or event. Case studies may be descriptive or explanatory and may consist of a single case or multiple cases. During the case study design the type of study that was used here was the descriptive one, since research was focused on finding and collecting critical success factors. Also, the choice was made to use a multiple case design approach instead of a single case design approach in order to be able to draw an analytic generalization of the critical success factors [7]. In qualitative research analytical generalization can be achieved by using a Multi-case design approach based on replication logic [8]. Replication logic indicates that cases should be selected based on the prediction of similar results or contradicting results for predictable reasons. The cases within this project were selected based on the prediction of similar results. Cases for which legacy to SOA migration has been successfully performed were analyzed. Moreover, a qualitative research approach was used to construct open, semi- structured, and closed questions in order to investigate in more detail whether and how the factors found during the literature survey contribute to the migration of legacy information systems to a SOA. The semi structured and closed questions will determine the strength of the relationship between factors and their contribution to the dimension they are placed in, thus how much these factors are critical to achieving successful LIS to SOA migration. This was achieved by performing quantitative research. For each dimension a set of 2 to 3 semi-structured questions were formulated. As for the open questions we want to grasp what the critical success factors are according to experts in order to validate the factors from the framework in stage 1, and to improve this framework with the addition of some critical factors which are very important in practice and were not directly found during the literature survey. The case design will consist of 5 open questions which need to be answered before answering the semi structured questions. This is necessary to make sure the experts are not biased towards some critical factors they recently came across. The cases for this project were selected by interviewing several experts whose experience closely relates to the research area of this project. The multiple case studies will be conducted at 4 large financial organizations where several employees will be interviewed as part of the research study. For each project experts who were closely involved and knew most regarding the project had been selected for the interview process to determine a high quality interview and improvement of the Theoretical Framework of CSF's. However, interviewing the appropriate persons who knew most about the project can be in some cases very difficult due to lack of free hours for these employees. This could cause serious project delay and a higher work pressure to execute the project such that deviation from the planning is minimal. In this study a design was needed to conduct the research, this included the procedure of collecting, analyzing and observation interpretation. The design provides support in terms of: the questions to study, the relevancy of data, what data to collect and how the results need to be analyzed.

2.3 Conceptualization - Case study Execution

After having designed the case study, the next step was to proceed with the interview process. For the interview process the 7 stages of Kvale were used as a methodological approach [9].

These 7 stages are: thematizing, designing, interviewing, transcribing, analyzing, verification and reporting. The figure below depicts the 7 stages of the interview process.

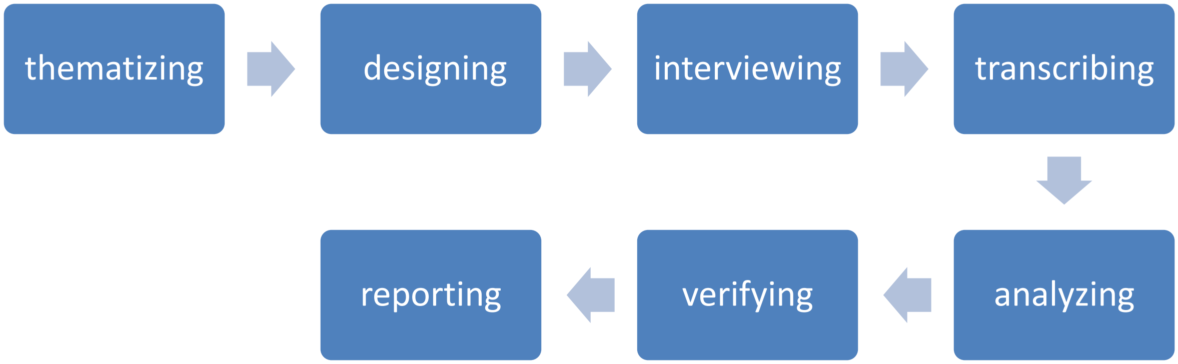


Figure 2 - interview process

Thematizing means here that the interview topic is based on the research area to find the critical success factors that could lead to migration of legacy information systems to a SOA within organizations. The second stage is focused on designing the interview which means taking all the 7 steps into consideration. During this step a clear planning should be made for the overall process and what products should be obtained within each stage. In stage 3 the actual interview takes place. An important part before starting the interview is briefing, where the interviewer provides the interviewee with the proper context en purpose of the interview. Debriefing also takes place right before ending the interview in to provide the interviewee with the ability to say whether there was anything else he or she wants to share [9]. The 4th stage is transcribing the result which means preparing the interview material for analysis. The use of a transcription from voice to text or a voice recorder will be put to use here.

During the 5th stage the obtained data will be analyzed by using the 6 steps of Kvale in which data analysis could be executed. Step 1: interpretation during the interview, 2 discovery of new relations during the interview, 3 interpret meaning and send back until one meaning is formed, 4 analyze the meaning after interpretation of interview, 5 re-interview opportunity to elaborate on initial statements and interpretations, 6 final conclusion and potential action which indicates that the final conclusions of the interview are drawn here (Fig 4).

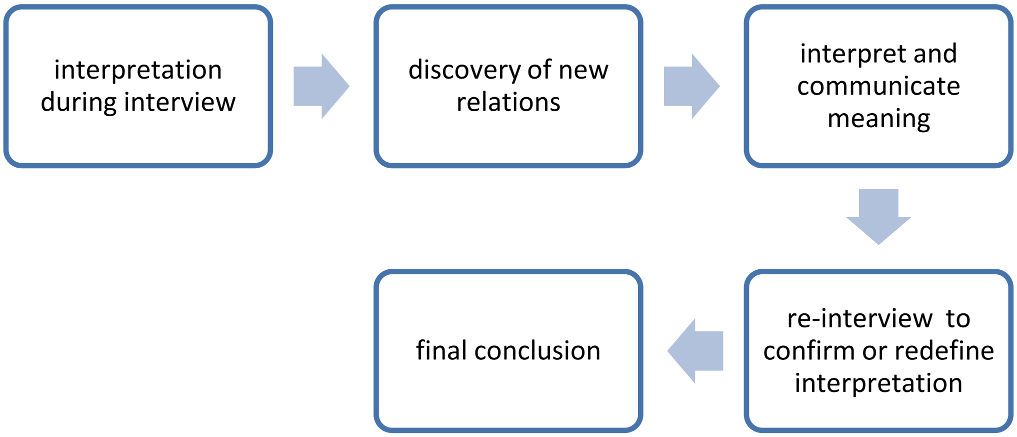

interpretation during interview
discovery of new relations
interpret and communicate meaning
re-interview to confirm or redefine interpretation
final conclusion

Figure 3 - Data analysis during interview process

In the 6th stage of the interview process the obtained data need to be verified. The interview findings need to be evaluated against criteria such as generalizability, reliability and validity. Reliability refers to the level of consistency of the obtained results which means that the study can be repeated with the same results. Internal and external validity is required to make sure the study of the investigation is still within scope of that which was to be studied. Generalizability will be described in more detail in the following section. Finally, in stage 7 the findings of the study or reported into a structural and readable product.

2.4 Validation & Generalization

Generalization is defined as “the applicability of evidence to situations and populations other than those in which the evidence was produced.” There are 2 types of generalizability which can be achieved as a result of conducting case studies which are statistical generalizability and theoretical generalizalibility. According to [10] statistical sampling and statistical generalizability is not necessary or even justifiable in qualitative research. In qualitative research theoretical generalizability should be the aim and is achieved by generalizing results from the multi cases to theory [7]. This means that generalizability in qualitative research refers to the extent to which theory developed within one or multiple studies may be exported to provide analytical and explanatory theory for the experiences of others who are in comparable situations. The theory that is developed becomes the foundation for generalization to other cases that have not been studied which are in scope of the research area and theory of study. In order to achieve theoretical generalization, replication logic should be used combined with a multi case design study treating each case as an individual study. Replication logic means that cases should be selected based on the prediction of similar results or contradicting results for predictable reasons. The cases within this project were selected based on the prediction of similar results. Cases for which legacy to SOA migration have been successfully performed were analyzed. Selecting 2 to 10 cases strengthens the empirical evaluation so more general conclusions could be drawn compared to having just a single case[7]. Apart from performing qualitative research also quantitative research is performed to determine how much a factor is critical. This part is described, during the evaluation of the Theoretical Framework of Success factors, in chapter 7. Generalization from case study to theory will be achieved by performing a cross case analysis on a multi-case study scenario by using the developed framework based on theory as template. In order to judge generalization external validity needs to be achieved by using replication logic for a multi case design setting and treating each case as an independent and separate study.

2.4.1 Cross Case Synthesis

In order to draw conclusions whether theoretical generalization is achieved, a cross case analysis should be performed where all separate cases are evaluated against each other. In order to perform this approach at least 2 cases should be available. By using each individual case study as a separate study for which an independent descriptive data model can be created. These data models are visualized by creating factor tables which display both the factors and their importance level of each individual case. These factor tables are then examined for cross-case patterns which strongly rely on argumentative and descriptive data rather than numeric values.

2.4.2 Research & Case Study Evaluation

In order to evaluate the generalizability of the study the multi case designs should be validated first. The 4 cases under study will be combined and the factors found in the studies are compared with respect to the theoretical framework of success factors.

The quality of research design is another important aspect of research which must be considered.

There are 4 criteria to validate the case study which are construct validity, internal validity, external validity, and reliability. The construct validity aims to check whether the obtained results from the different cases provided are sufficient in order to match with the findings from literature. Internal validity refers to pattern matching within each individual study whereas external validity is achieved by using the multi case studies with replication logic to establish generalizability for each individual case.

Reliability is used to measure how consistent the results were when a study is repeated.

3 Project Environment

This chapter describes the organizational and environmental aspects where the thesis project has been carried out for the past 9 months. Also the specific resources which contributed to the success of the project will be described below in more detail.

3.1 Logica

The thesis project was performed at a large ICT consulting organization named Logica which provides business and technology services across 36 countries employing over almost 40.000 people. Logica CMG was founded on 30 December 2002 which arose from the former Logica PLC and CMG. In 2008 the name of this organization was changed from Logica CMG to simply Logica.

Logica provides services that are classified in 3 domains which are business consulting, systems integration and outsourcing of business processes. The organization provides these services to the 4 main Industry Sectors which are Energy, Utilities and Telecom, Finance, Public Sector and Industry Distribution and Transport. By focusing on these main sectors the company is able to provide customized solutions based on the client's current and future needs. In the Netherlands departments within Logica divided among 3 main domains which are set up to focus on one of the above mentioned Industry Sectors. These 3 domains are formally called Service Lines (SL). The 3 SL's are Consulting, Professional Skills and Systems Integration Projects. Each Service Line is then further decomposed into several practices which can be perceived as a specific department aimed at a certain skill set for example Project Management, Java &Oracle, SOA or Working Tomorrow (WT). The organization chart is depicted in the figure below.

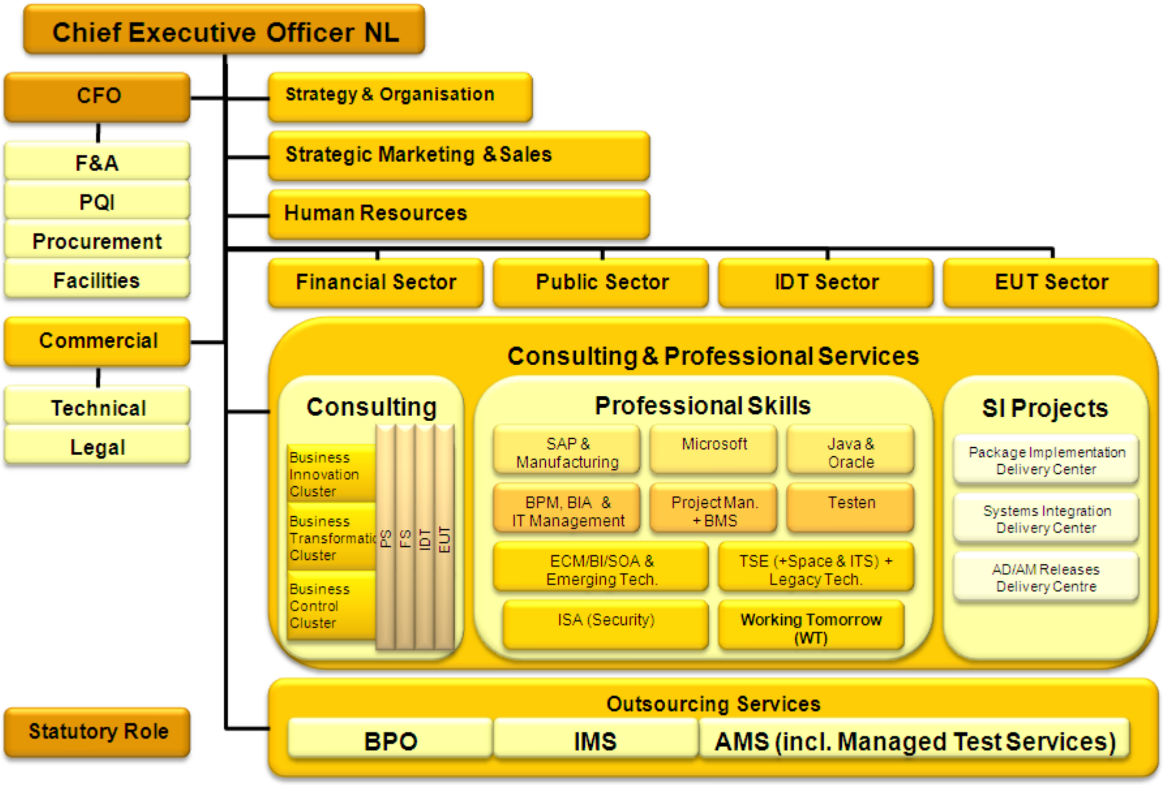


Figure 4 - Organization Chart Logica

3.2 Thesis Project

The thesis project was implemented within the Working Tomorrow practice. Logica started this program in order to provide students the possibility to execute their graduation projects on innovative subjects regarding technology, concept or methodology. The subject of research was formalized in partnership with the SOA practice and the Business Transformation Cluster namely the group that focuses on business migration in the Industry Sector Finance. This group has performed a few SOA migration projects and submitted several migration propositions that ultimately leads to obtaining the project or contract.

This group has gained a lot of experience regarding the process, deliverables and approach to tackle a SOA migration project, yet still there is no standardized way of working or approaching SOA migration projects. The lessons learned have not been documented and thus there is no shared intelligence or shared wiki regarding this type of research within Logica. The project teams who have worked on these type of projects are the only ones that benefit from the lessons learned. So in order to share this gained knowledge on enterprise level there is a need to define and document what the main elements(success factors) in order to successfully implement(migrate) a Legacy environment to a Service Oriented environment in the future. A framework of success factors or goals that need to be reached during the whole project life cycle had to be constructed. The benefits of the this framework will lead to shorter and thus cheaper time to propositions and act as a guide during migration projects which prescribes the set of activities in terms of success factors which need to be carried out. Using the designed framework of success factors should be used as a guide during complex Legacy system migration efforts.

The Working Tomorrow program has 4 main goals which are:

- Providing a central place where students have the ability to carry out innovative graduation projects,
- Recruitment of potential employee,
- Increase the reputation level of Logica in terms of innovation and
- Using demo's and other deliverables of graduation projects in order to gather paid assignments and acquire knowledge

Working Tomorrow is driven by innovative IT graduation projects, some examples of these innovative subjects are agent technology, multi touch technology, augmented reality and smart grids combined with smart metering. Working Tomorrow program is supported by 6 locations in Logica Netherlands. This program provides approximately 120 students a place to execute their graduation project annually.

3.3 Resources

During the internship Logica provides each graduate or trainee the opportunity to use several physical and non physical resources. A laptop and a deck phone is provided on the start of the very first day. There is also a large collection of books and thesis documents which can be used as reference any time. Also several internal knowledge engines such as their own wiki pages on specific subjects can be used to gather knowledge on the particular subject of research for example intranet and architecture pages regarding SOA, Legacy architecture, Business Migration etc. There are also mailing lists based on common competences which came out very useful during the questionnaire and interview process of the project. Last but not least, the focus on tutoring was tremendous since each graduate is provided at least 2 project mentors in order to successfully carry out the project.

Part II Theoretical Background

4 Legacy Systems & SOA

This chapter describes the theoretical background of the study regarding Legacy Systems and SOA before embarking on the main part of the research which is a study concerning the identification and evaluation of critical success factors for the migration of Legacy Systems to SOA. The main characteristics of both legacy systems and SOA will be described in the following sections.

4.1 History and Scope

Many organizations still rely on software systems that are 10 to almost 20 years old[11]. These systems are slow, inextensible, running on old or obsolete hardware causing organizations to spend an excessive amount of money on maintenance[12, 13]. These systems are difficult to replace since they have been embedded with and still carry out the most critical business processes for organizations since the day of their existence, such systems are called *legacy systems[14]*.

There are many perceptions of what legacy systems are but not many concrete definitions formed since it is very difficult to define a term that is used for every system that is outdated in some technical or technological sense. However some papers do provide an exact definition for legacy systems.

According to Bisbal [13] a legacy system is *”any information system that significantly resists modification and evolution”.* This indicates that legacy systems are not just old systems but also can’t easily be extended or modified which may cause organizations to spend more on maintenance and anticipation on changing regulations, organizational change, national and international economies may become very difficult due to inflexibility of their systems.

Another definition of legacy systems was given in [15] [1] ; “*Legacy systems are old socio-technical systems including software, hardware, data, and business processes and rules where failure of these systems would have a serious consequence on the day to day running of the business. These particular systems are mission critical”.* In other words they are essential for the normal functioning of the business.

The definition used in [13] implies that even newer systems could be categorized as being a legacy system. For example systems that are not web or service oriented, may also be categorized as legacy. According to [1] legacy systems can also be seen as enterprise systems consisting of a social context, business context, supporting data, applications and a hardware environment. These applications are critical since they provide support to the business by automating the business processes. Such systems were essentially built to support the core of business processes and therefore can’t even be (temporarily) shut down because execution of these processes is fundamental to the operation of the business. This study defines a legacy system as an association of both views given by the authors described above which is:

”Any information system that resists modification and evolution where failure or inconsequent modification would have serious consequences on the normal functioning of the business”

Both views were used to make a critical evaluation of legacy systems, their common structure and design. Most of the legacy systems have been written in older programming language such as COBOL, PASCAL, Python and more as illustrated in the figure below. Some of these programming languages are still widely used and maintained by many organizations. Business organizations have done large investments in applications written in COBOL. According to [16], every year approximately 5 billion lines of new COBOL code is written. It is estimated that 70 % of the worlds data is processed by COBOL and that nine out of 10 ATM transactions are performed by COBOL[17].

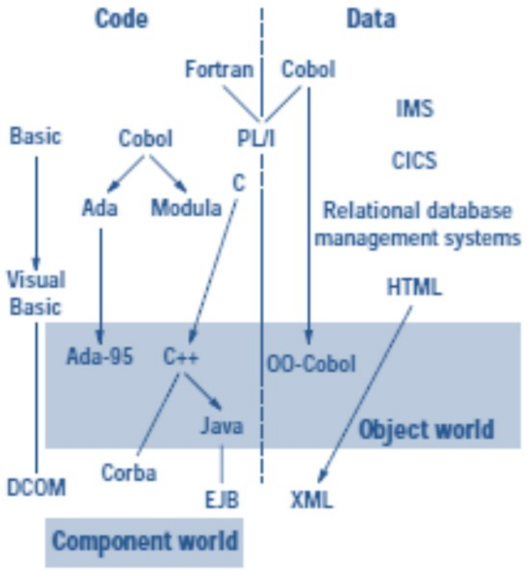


Figure 5 - Programming language evolution [16]

These systems based on the COBOL language are still used because of the large investments made by organizations in the past and they expect the system to be operational for the coming years in order to earn their investment back[16]. However maintaining these systems have been proven to be very costly and difficult due to a number of factors such as: lack of understanding of the system or lacking documentation which is caused by retirement of software architects and programmers who designed and implemented the legacy system in the past. Replacing these systems is rather complex and even more difficult for organizations, not only because of the direct financial cost of software development, but also the effort of reallocating the business logic (rules and processes) out of the code. These rules and processes are mostly intertwined in the code and cannot be easily extracted without the availability of proper software documentation and knowledge concerning the system[18]. So if the system is replaced these processes are replaced as well with unpredictable costs.

Also retirement of software architects and engineers as well as the fading interest in old programming languages such as COBOL contributes to systems becoming legacy or obsolete. For example, when programming and maintenance was done for several years by the same engineers, documentation was not often kept up to date since they understood the system very well. Retirement of these engineers induces problems when features are added or changed by other engineers who are not familiar with the system. This causes current programmers need to analyze the code in order to gain more knowledge about the function of the system as a whole which is very time consuming, costly and difficult, let alone making changes to the system. Often the business rules that are embedded in the code might not be documented elsewhere than written as code within some function. This is a serious problem since these rules were set up as a fundament to reduce risk. If the system was to be rewritten without inclusion of these rules, a company may accept high risk without even being aware of it, resulting in profit loss or in worst case a company could even go bankrupt.

For example, an insurance company may have embedded its rules for assessing the risk of a policy application in its software. If these rules are not maintained, the company may accept high-risk policies which will result in expensive future claims.

4.2 Socio-Technical Systems

As mentioned before legacy systems are Socio-Technical Systems abbreviated as STS. A Socio-Technical System can be perceived as a system where social aspects, a social system depending on the social context, are combined with the use of multiple technologies[19]. The socio-technical system is mostly shaped by the social and ethical aspects of use supported by one or more technologies. It is a composition of the way technology, people, procedures and physical surroundings come together.

Socio-Technical Systems consist of hardware, software, physical surroundings, people, procedures, laws and regulations and data and data structures (see figure 8). The people and physical surroundings form part of the social and ethical aspect of the socio-technical system which are dependent of hardware and software and need to abide and work according to given procedures, rules and regulations. [1, 19]

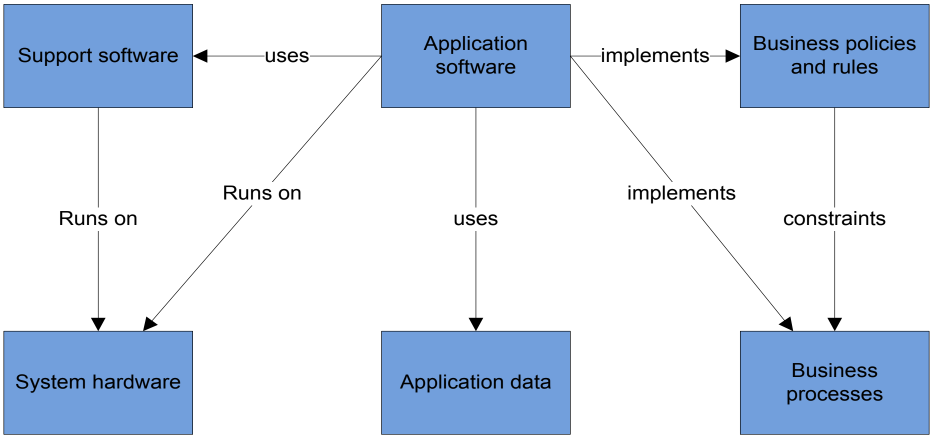


Figure 6 - Socio-technical system

These systems are very complex due to the fact of constant change, this could be in terms of people working within the company of position or a change of regulations, hardware and software support etc. This is a very complex situation and if the changes and modifications don't follow a structured and logical flow the complexity increases even more. This is the main reason why most of the legacy systems and their operating environment are both obsolete and extremely important to the business at the same time. Due to a combination of social, governmental and technical factors legacy systems continue to work effectively however the operating environment and the source code gets redundantly intertwined and complex even for the programmers who have programmed these systems for several years. The application software running of the mainframe providing the core business service may have been developed separately without using guidelines such as interoperability, reusability or standardization. Additionally, the supporting software and tools such as operating systems, modeling tools and compilers may also be outdated and no longer be supported by their manufacturer. The main purpose of the software applications is to provide support to the business. The business is often divided in business processes and business policies and rules. The processes are the core tasks that need to be carried out for the daily business of an organization, whereas the policies and rules are constraints on how the business should be carried out. These rules were set up by the business to reduce risk and were most of the time directly implemented in the programming code. This indicates that changing rules as a result of changing laws and regulations becomes a very complex task. Legislative rules and regulations are part of the policies and rules which are often changed during the course of time, resulting in multiple changes within the code without keeping or updating the proper documentation.

4.3 Consequences from the IT Perspective

Based on the above, we see that there is a large risk involved for businesses when making changes to legacy systems. However maintaining a system becomes more expensive as the system gets older. This indicates that businesses often need to deal with this dilemma of changing the system which carries high risk if there are a lot of unknowns or continue maintenance at accumulating cost. There are several reasons why changes made to legacy systems can be costly such as:

- Different parts of the system have been developed by different programmers or teams which didn't use a consistent programming style or standard for developing the whole system.
- Components of the system or even the whole system may be implemented using an obsolete programming language. It may be very difficult to find a specialist with proper knowledge about the programming language and outsourcing of system may be required which is again very expensive.
- The lack of adequate system documentation or coping with documentation that may be out- of -date. Sometimes the source code is the only documentation available and in worst case when the source code gets lost only the executable version can be used as reference.
- Multiple programs may have been added and interfaces in an unordered and unstructured way (ad hoc) may cause the system to go corrupt making it very complex and difficult to understand.
- Data duplication and the use of out-of- date, incomplete and inaccurate data.

4.4 Consequences from the Business Perspective

There are several consequences for the business of organizations that are highly dependent on the daily operation of their legacy systems. In the near future their systems will be needed to retire or changed due to changing business needs or a change of laws and regulations for example. This will be costly and times consuming each time changes are to be made to the system. The main business goals in general will not be reached for example:

- Business expansion will be difficult since the systems operate as silo's and integration or extension is almost impossible.
- Lower return on investments since maintenance cost increase every year
- Change of business mission and vision will be difficult because of the level of agility of their information systems.
- There will be more competition visible on the market due to utilization of state of the art technologies
- Higher maintenance cost for Information systems as a result of using obsolete technologies and main frame hosting services.
- Mergers or splitting of companies falls in the same category as business expansion where there is no standardization or extensibility possible due to brittleness of the systems.

4.5 Organizational Dilemma

Businesses with a large number of legacy systems are faced with an unpleasant dilemma. If they tend to keep using the current systems the cost for maintenance will rise with increasing margins each year, also there is no guarantee of how long the business needs will be fulfilled adequately. At the other end, replacing legacy systems with new systems will be costly and there is a high risk involved, also there is no guarantee that the system provides support for execution of all the business rules and processes. Consequently, many businesses are looking at modernization techniques which extend the lifetime of legacy systems and which reduce the costs of keeping these systems in use. Additionally, modernizing current legacy applications depend on the way it supports the daily business of an organization. When business strongly relies on legacy systems, due to automation and execution of the core processes, existing business rules and regulations concerning these systems must be extracted and reused within the new or modernized environment. Reusing existing assets is thus of fundamental importance for businesses that strongly rely on legacy systems. The reuse of existing assets involves reusing business logic and rules which constraints the business in order to mitigate risk, and in order for organizations to stay within certain boundaries, such as legislation and other important regulations. There are several modernization techniques for organizations coping with legacy systems. These techniques are discussed further in this chapter.

4.6 Modernizing legacy systems

Organizations relying on legacy systems which have decided to modernize their applications are in need of guidance in order to balance certain factors such as risk, cost and efficiency. According to [11-13], there are 4 strategic options to modernize legacy systems, which are application redevelopment, migration, wrapping and scrapping of the system. Additionally continuing maintenance (AS-IS) of the system may also be applicable, since changing the systems may simply be too expensive and have a high risk rate, therefore applications are left as they are. Systems that are low on business utilization as well as information quality are usually candidates for scrapping, because the cost to maintain these systems is substantially higher than their contribution to the business. There are several alternatives to modernize information systems, however deciding which alternative(s) are most adequate is determined by assessing the level of business support provided by the applications in use.

According to [1, 20], assessment should be based on system quality and the business value they provide. Business value is assessed by evaluating the level of system contribution to the business, thus how business relies on a certain system or part of it. Additionally, system quality also needs to be assessed. This involves validating system performance, fault tolerance, robustness, etc. According to [21], cost pressure is a factor which is equally important. The budget available for IT and the cost to maintain the current IT systems should be taken into account as well when assessing legacy systems. Another important factor that is not explicitly mentioned in [20] and [21], is the risk involved when modernizing legacy systems. For example, the risk of an incomplete or failed migration project might induce higher cost compared to current system, or may even be more expensive when the project finally finishes. Some migration project take more than 5 years, the cost involved here may be much higher than simply continuing maintenance for 5 years. Of course organizations benefit when migration is finished, yet still there is a tight budget pressure from the business, which could also lead IT systems to become legacy systems. Therefore, assessment should also include the cost for continuing maintenance against the investment needed for system modernization, as well as the risk involved here. A common matrix or even framework is used to provide guidance to organizations in need of a legacy assessment strategy. However, this matrix is yet incomplete since there is less attention given to the risk factor which also needs a place in the matrix. Based on literature a decision model was designed which is depicted in the figure below.

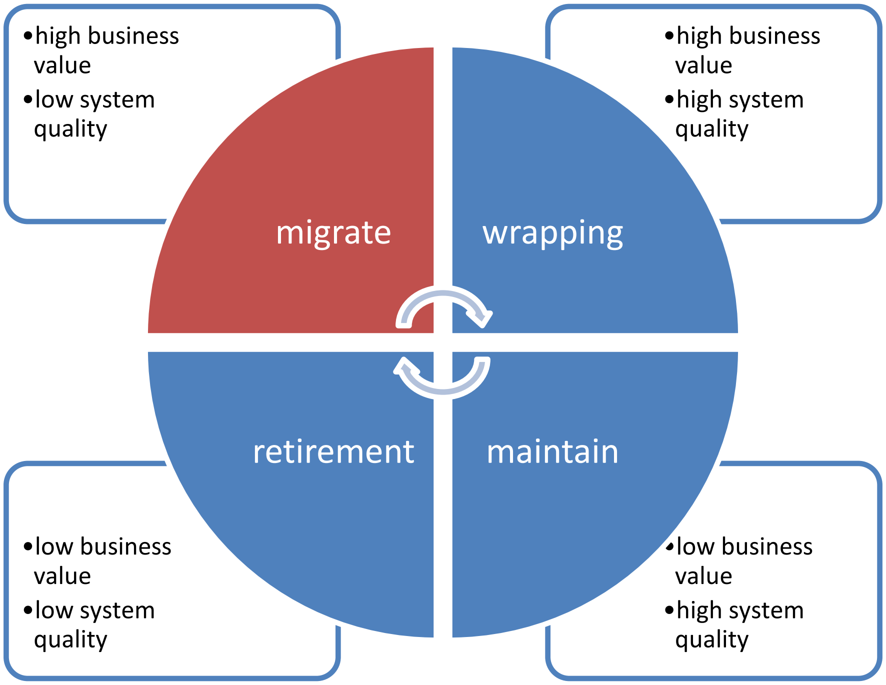


Figure 7 – Legacy Systems Assessment Framework

After legacy systems have been assessed, each application is positioned in 1 of the 4 clusters represented in the figure above. These alternatives are derived by assessing the current system quality against the business value they provide. The best practices and the available methodologies and tools supporting this transformation, are described in the following sub sections.

4.6.1 Maintenance

Continue maintenance or leave AS-IS- Keep maintaining the current system when it has a high business value and considerably low cost to maintain. The system should also have a low maintenance cost in the future otherwise the system may still need to be migrated or even redeveloped

4.6.2 Application Retirement

According to[13], application retirement or redevelopment is often applied within organizations when the business dependency and the business value obtained by the supporting applications are incredibly low.

4.6.3 Application Redevelopment

Application redevelopment involves (re)development from scratch. This alternative is often used when substituting low quality systems with an equivalent off - the - shelf software package, which possesses the same functionality at feasible cost with a reasonably low business risk. This option should be considered when the quality of the system is combined with a low business value while the cost for maintaining the current system keeps increasing.

.

4.6.4 Non- Invasive: Wrapping of Legacy Systems

Wrapping is considered to be one of the fastest and cheapest ways to modernize legacy systems. This alternative is preferred when organizations need to integrate some part of their legacy systems with other information systems or even other organizations. The wrapping process in finished with the development of a wrapper, which converts data elements from or to legacy systems into more workable and accessible elements. However, wrapping still remains a short term solution, because of the minor contribution to the current legacy architecture itself. Therefore, changing or adding new functionalities to legacy systems still remain nearly possible, since the construction of the data elements such as functions, components and their relations have been left unchanged. Wrapping is thus very practical when this involves improving data or system representation, and integration in terms of converting and communicating data elements in some other data structure. However, when legacy systems are in need of extension and evolution in order to be able to support the agile business environment, other transformation methods need to be addressed. One alternative that could be used is migration.

4.6.5 Invasive: Migration

Transforming an application involves reusing current information assets and reforming them to some extent in order to obtain a higher level of usability, communication and integration. This solution is highly advised when systems are of high business value, and the quality of the systems supporting the business varies from high to extremely low. High quality systems require less maintenance, while low quality systems often a higher level of maintenance which increases cost. Therefore low quality systems have a higher transformation priority then high quality systems. These systems are candidates for migration, while wrapping is applied to systems in need of fewer modifications. Both modernization alternatives were practically applied which resulted in development of best practices and methodologies to support and guide future transformations. Also during migration of one or more legacy systems other strategies could be combined since some parts or modules within the large legacy system require redevelopment, others will be seized to use while some parts that rarely changed could be wrapped. This study focuses on the migration of legacy systems only namely on identifying and generalizing the critical success factors for the migration of legacy systems onto a Service Oriented Architecture. The concept of a Service Oriented Architecture and its contribution and benefits for the migration of legacy systems will be described in the following section.

4.7 Service Oriented Architecture

A Service Oriented Architecture is not a technology itself; it is a paradigm that can be implemented using different technologies such as web services technology.

A service is defined as something that gets done based on request of another [22]. In a more technical sense services are individual autonomous units of logic which are independent and can be specified as a task, a business entity, or some logical grouping. The size and scope of the logic represented by a service may vary from small to extremely large, thus there is no limit of logic that may be presented. Furthermore a service could be a part of a larger service or could be formed by a collection of services [23]. This indicates that services can be used to automate a whole process, a process step or even sub processes (Fig. 9). Simply put, SOA presents a well defined business function as reusable services which are made available through standard protocols.

According to [23] Service Oriented Architecture is a term that represents a model in which automation logic is decomposed into smaller, distinct units of logic. These units can be distributed individually and collectively add value to a larger piece of business automation logic. Placed in terminology of services, a distinction is made between business services and technical services. The business services are the actual services provided to the customer (external business services) or services provided to employees within departments (internal business services). As a result a business services is implemented using a business function. This business function is made operational by executing one or more business processes. A business process is automated using technical services. So, in order to provide business services a decomposition of several technical services is required.

A Service may be designed using a bottom- up strategy which indicates that technical services are independently designed without any reference to the actual business function or provided business service. These types of services are known as fine grained services. On the other end when a top-down design strategy is taken, which is most favorable when engaging a SOA, business services are designed based on their internal and external business contribution. These services are called business services or coarse grained services [24]. In the figure below a process-step is an example of 3 fine grained services and the output of the whole process could be perceived as a coarse grained service. A coarse grained service seems to offer more functionality since it consists of a complete process (including several process steps) or even multiple processes. After execution of these lower level services, the coarse grained service replies with an answer.

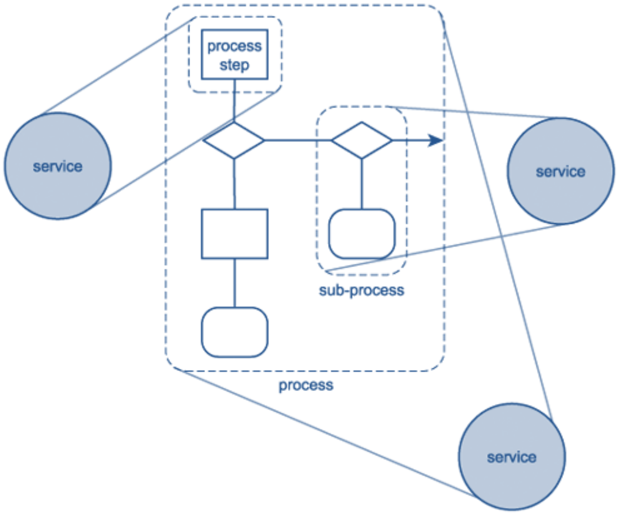


Figure 8 - Service decomposition

4.7.1 SOA Principles of Design

Just like Object-oriented design, service oriented design utilizes several design principles and guidelines. The difference between a principle and a guideline is that guidelines are best practices and have a more advising role without any obligations of essentially applying these rules. There is no official list of design principles when constructing Service Oriented applications, however, there are common design-principles which tend to support and provide guidance when service orientated environments are to be designed and build [23, 25] These common principles and their justifications are listed below:

Reusability, Services are to be designed to support reuse in order to decrease cost and improve flexibility. As long as a service has been deployed it can be used my multiple requesters. This avoids redundancy within software systems and databases. Development cost is also reduced since multiple parties can make use of the same functionality which was designed based on reuse capabilities.

Service contract, a service contract defines the specifications of the semantics of a service and its input and output. The requester of a service needs to comply with the pre conditions of the service upon request and the post conditions of a service defines the output that can be expected after the service has been executed [23].

Loose coupling reduces the interdependencies in a system between services. It allows software on each side of the conversation to change without impacting the other. Loose coupling is achieved by implementing things such as asynchronous communication style between services, weak type system and a distributed control of process logic.

***Abstraction**,* Provides transparency to service requesters by hiding irrelevant underlying logic and simply providing the logic encapsulated within the service provided or requested. This means that a service only provides its contract to the outside world and all other implementation details are kept hidden. Abstraction increases the reusability aspect of services.

***Composability**,* Services compose other services which allow logic to be represented at different granularity levels. In other words, services are designed in a certain way in order for them to be invoked by other services. This enables a chain of services also knows as an orchestration of services to create a composition of services. A composition of services is created in order to implement a higher level business process or part of the business process. More detail concerning the proper level of granularity follows in the following section.

Autonomy, Services do not rely on other services for them to be executed within their own boundaries. In other words, a service should be able to execute without a high level of dependency on other services. For business services this is very complex, however for technical services or fine grained services this requirement should be met since these are the services located on the lowest level of the architecture.

Statelessness, Services are to be designed without having to manage their state information, this can cause conflicts in their ability to stay loosely coupled. Stateful services cause an increase in the level of dependencies, and cannot be replaced when maintaining a state. Therefore services should be stateless. The state of a service should not be stored on the service but at the backend system or frond end system itself.

Discovery, Services need to be published and discoverable via appropriate discovery registers or mechanisms. In order to call a service it should be discoverable, services are made public by using a service registry.

4.7.2 SOA Granularity

As mentioned before, the SOA paradigm consists of 2 perspectives, namely the business and IT perspective. From the business perspective, a service is defined as a coarse grained service or business services, which provide the actual result of an executed business process. A business process can be defined as a complete end-to-end set of activities that together create value for the customer [26]. These coarse grained services are decomposed into several smaller technical services also known as fine grained services. The fine grained services are performing data requests or technical functions in order to provide the business service the information it requested. When SOA is deployed throughout the whole enterprise, this indicates that both perspectives think and design services based on the SOA paradigm. An enterprise which consists of 3 layers which are the business layer also referred to as business process layer where the business services are in fact defined. The second layer is the application layer which consists of the technical services that implement (automate) the business services based on its request. In order to execute a business services an orchestration of composed technical services would be created in design time in order to perform the requested action. Erl places the orchestration of services In a separate layer, however, this does not have to be the case since orchestration could be defined and coupled to business services on the business layer itself (figure below). An important aspect which is often discussed during SOA projects is the level of granularity when several types of services are defined. SOA adopts a top- down design strategy which indicates that business initiates the requirements all together with its users and stakeholders. As an output the business services are determined. After the business services are defined, the fine grained services can be coupled or implemented and composed in the orchestration to provide the business with its request. So, in other words, a business service is firstly defined and then decomposed into more technical services. It is never the other way around [23, 25].

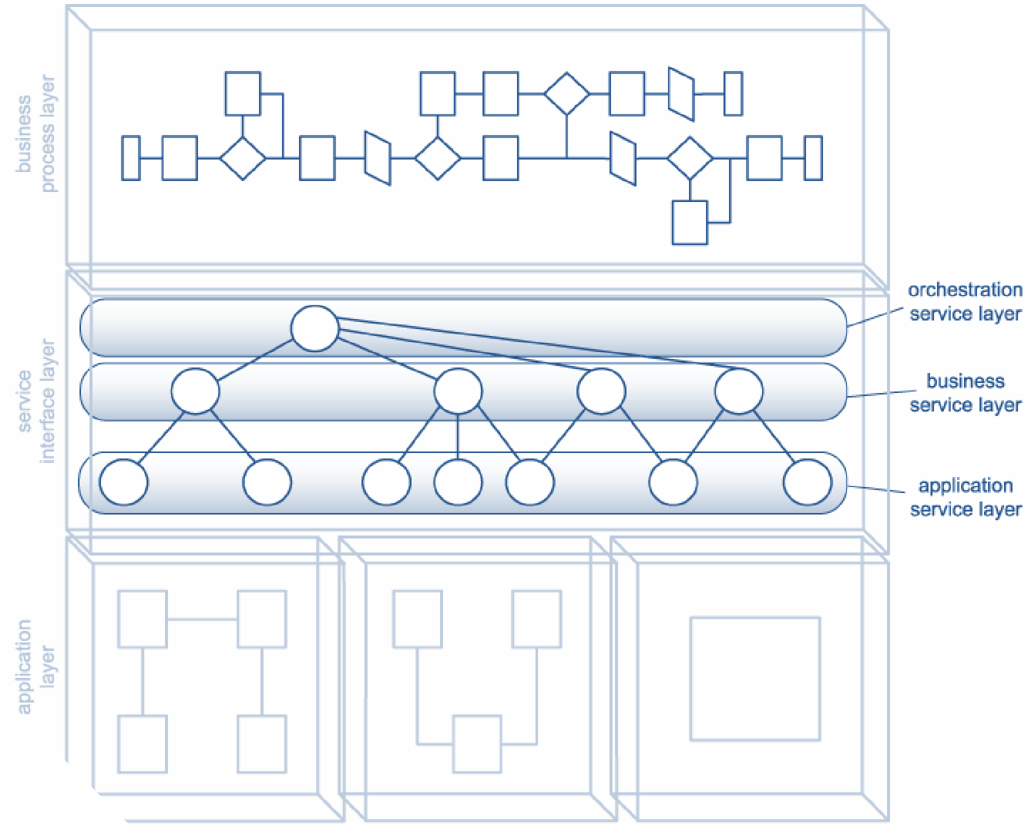


Figure 9 - Services layers: source [27]

4.7.3 SOA Governance

Implementing SOA throughout an organization is an iterative and incremental process which has to be overseen in order to ensure that IT performance meets the business objectives. SOA governance focuses on managing organizational dependencies in order to achieve the main objective.

SOA governance can be divided in technical and non-technical tasks [28].

Non-technical tasks include maintaining a reference-architecture or a high level blueprint of organizations SOA architecture including the technologies used on each layer. Managing roles and responsibilities within the new architecture is just as important. It is important to define the owner and responsible actors for a service and define the type of authorities and rights including its boundaries. In times of issues in terms of performance or other quality relates aspects the responsible actors should be approachable.

Technical tasks strongly enforce the use and maintenance of documentation. All non technical issues of governance such as processes, responsibilities and policies should be clearly defined and documented. Also management of services and their service contracts is very important, this involves managing the published and allocated services by providers and consumers of services and keeping the list of services up to date as possible. In order to ensure that services live up to their SLA's monitoring is required. Also monitoring a service quickly indentifies services that stop working or change in terms of quality attributes. Configuration and change management is also strongly advices since changing or updating a service might cause other services in the composition to stop working effectively.

4.7.4 Implementing SOA

SOA was designed by using 3 basic concepts namely: a service consumer, a service provider and a service registry. A service provider publishes its services onto the service registry. When a service consumer is interested in a particular service, the service requester needs to discover how and through which channel he can make use of the service. This information is provided by the service registry. These concepts and their relationships are depicted in the figure below.

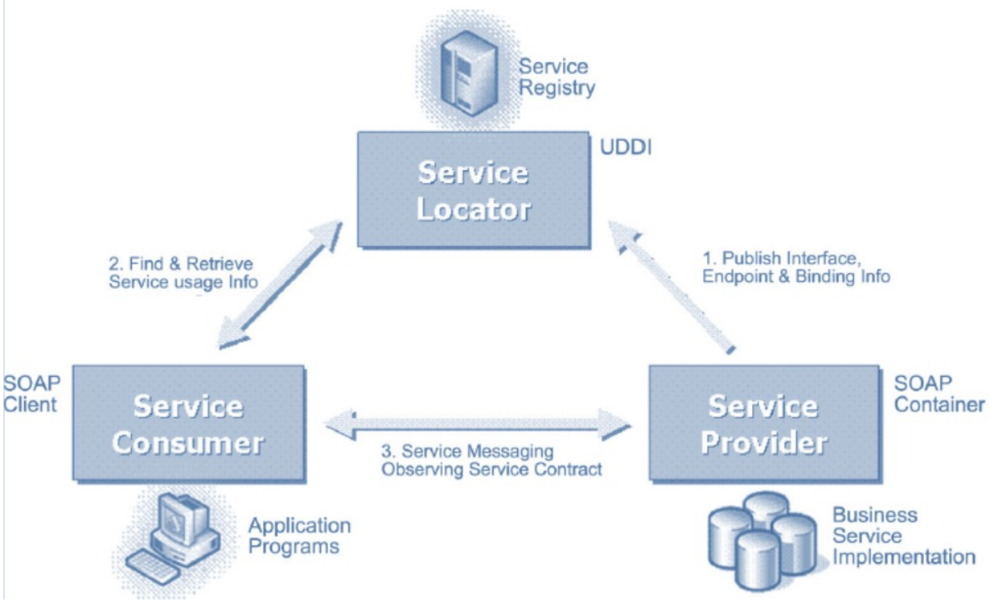

Service Registry
UDDI
Service Locator
2. Find & Retrieve Service usage Info
1. Publish Interface, Endpoint & Binding Info
SOAP Client
Service Consumer
3. Service Messaging Observing Service Contract
Service Provider
SOAP Container
Application Programs
Business Service Implementation

Figure 10 - Service implementation: [23]

SOA implementations follow the iterative SOA adoption process of 6 stages which are planning, analysis, design, implementation, testing and deployment [25]. These 6 stages are depicted in the figure below. Services are implemented using the supported technologies. Web services technology can be used for the implementation of a SOA. According to[29], a web service is a software application that supports web standards, such as XML, WSDL, SOAP and UDDI. This software application is identified by a URI (Uniform Resource Identifier) whose interfaces and bindings can be defined, described and discovered by XML artifacts[30]. By utilizing XML messages and network protocols web services may directly interact with other applications. A web service's interface is described in a standard format using WSDL, which is an XML-based document format for describing web services as a set of functionalities on messages regardless of messaging formats or network protocols. Applications interact with the web service in a manner prescribed by its description using SOAP messages [31]. By exchanging SOAP messages written in XML, services and applications describe their functionalities and publish them on a service registry. The web standards used to implement SOA are aggregated in a framework which is called the Web service Architecture Framework [23].

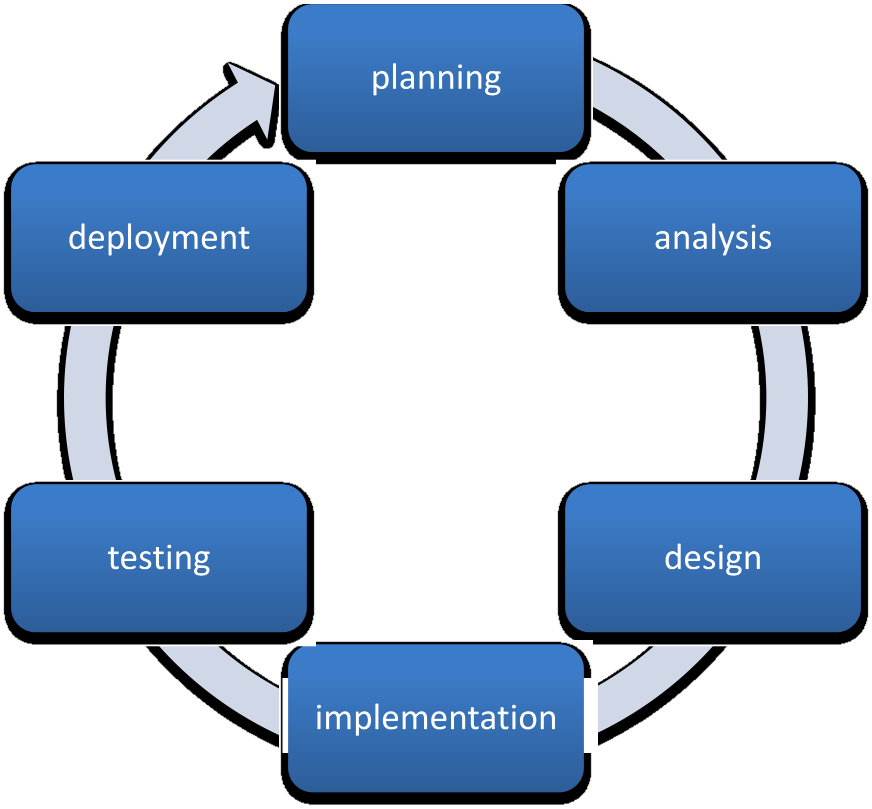


Figure 11 - SOA Adoption Process

4.7.5 Enterprise SOA

Service Oriented Enterprise or Enterprise SOA deals with applying the SOA principles and SOA adoption process on enterprise scale. The goal of Enterprise SOA is to achieve and develop enterprise scale applications out of composable services in order to support the business goal and business process. SOA can be applied on different levels within an organization and contributes to both the business and technical perspective. From technical perspective, services are designed as pure technical functionalities in order to automate some part of the business process or sub process which is also known as a fine grained service. Conversely, from business perspective services are defined as pure business functionalities, clearly related to business objects, supporting the achievement of the business goal also known as coarse grained services. Enterprise wide SOA supports process change or the development of new processes by simply composing already existing services as reusable building blocks to support the new business requirements. Enterprise SOA adopts the layered approach of Enterprise Architecture practice, which consists of a business layer, an application layer and infrastructural layer. The difference with the layered approach of Erl (2008) is the number of layers and their specifications. The business layer of Enterprise SOA consists of the business process, the required business services and the orchestration of services. On application level the technical services are defined and composed based. There is no layer for the infrastructural support defined in the SOA layered approach of Erl. The layers of an Enterprise SOA are depicted in the figure below.

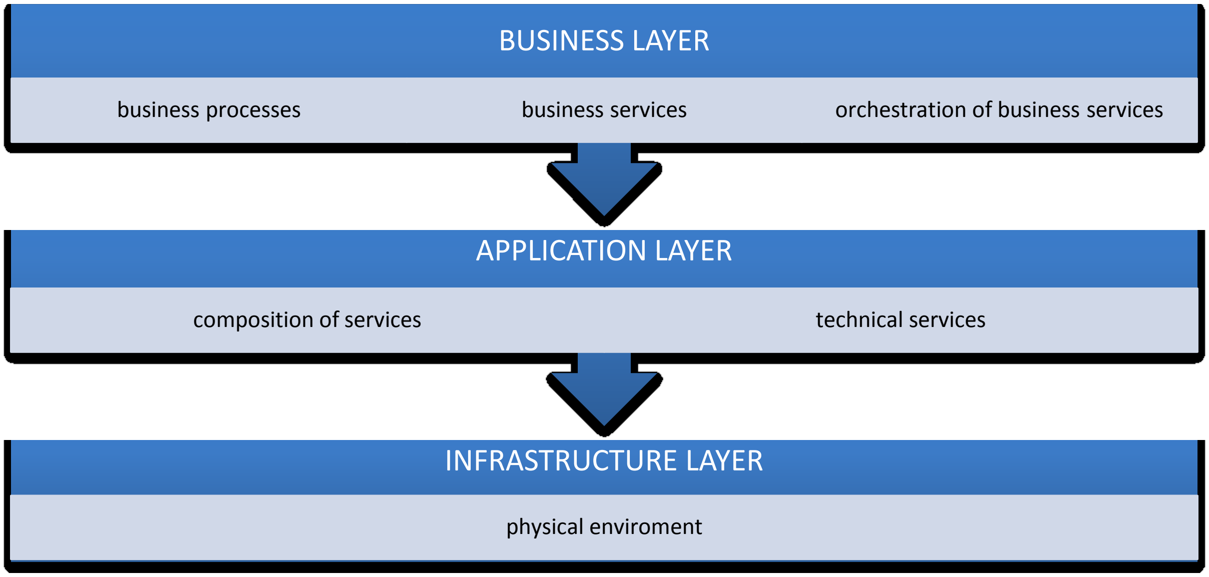


Figure 12 - Enterprise SOA layers

Part III - Conceptualization

5. Theoretical Framework

After having described the concepts of LIS and SOA extensively in the previous chapters, this chapter continues with the main research of this study, which is focusing on identifying and generalizing critical success factors from theory and practice in order to successfully carry out LIS to SOA migration projects. The goal of this study is to construct a framework of critical success factors for LIS to SOA migration. Before performing an extensive literature study on critical success factors for successful LIS to SOA migration, a theoretical framework was constructed to be able to classify and organize these success factors. This theoretical framework was developed independently from the determined critical success factors. The construction and the process of development of this framework is described in the following sections.

5.1 Theoretical framework dimensions

In order to organize and classify the critical success factors in the theoretical framework, there is a need to define multiple dimensions with separate properties. The critical success factors will be classified according to the Seligmann framework[4] and the dimensions of project success[32]. The ***what, where, who, how*** and ***with what*** were used to develop the dimensions of the theoretical framework.

What, refers to the objective which is to be covered by defining success in an IT Project and the elements that contribute to the success of an IT project.

Who defines the perspectives to whom the CSF's are relevant for within the scope of an IT project.

Where, refers to the methodological coverage of these elements within the Seligmann Framework. After the elements that contribute to IT project success have been defined, they will be placed in one or more dimensions of the Seligmann framework based on the level of relevance.

How, defines the work procedure of the activities and tasks to be performed in order to reach the objectives.

With what, defines the Critical Success Factors (CSF's) that contribute to achieving the goals perceived from different perspectives (who), which are also methodologically accounted for (where). The CSF's that were determined by extensive theoretical study will be described in chapter 6.

5.1.1 Project Success Framework

In order to be able to make judgments about potential factors that contribute to successful LIS to SOA migration, one needs to understand what is actually meant by project success and how this is measured for projects, and in particular for IT projects and how this relates to the generic software development process of Pressman [33]. The generic steps of the development process of Pressman (fig 14) consist of 5 phases, which are:

Communication – the communication purposes in order to define requirements and all aspects of communication with stakeholders, users and project team during the development process.

Planning – activities regarding the input and output of the project in terms roles, budget, cost, time and deliverables.

Modeling – the modeling activities in order to model and communicate the problem and solution space for the appropriate stakeholders and project team.

Construction – the actual construction of the software which includes generating code and testing.

Deployment – the actual deployment of the software to the client. Support to the different users and administrators of the software in terms of training for usage and management is part of this phase as well.

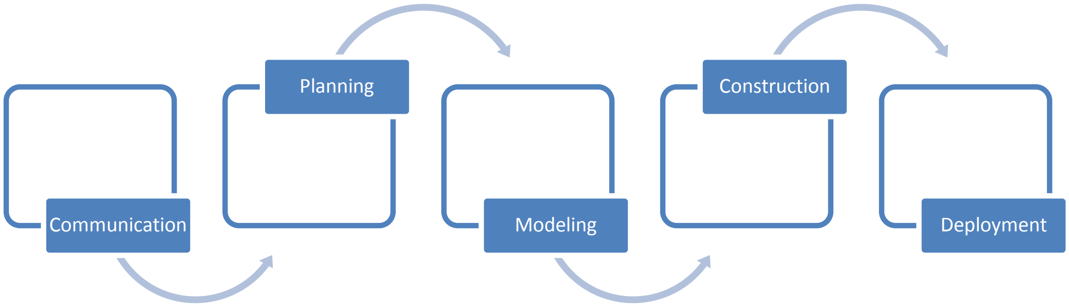


Figure 13 - Generic Software Development Process

Measuring time, cost and quality against the overall objectives of the project contribute to successful project management [34]. However, project management activities are subordinate to project success and are merely one criterion to measure project success. IT project success can be separated into 4 levels: business success, organizational success, technical success and project management success [32]. Business success depends on the business benefits that are obtained, also defining a clear and shared business statement of what will be regarded as success contributes to business success. Technical success depends on the technical development process and the quality of the delivered end product or system [35] . Organizational success depends on the use process and the impact on the organization in terms of user satisfaction and user support. The level of use is also considered a condition for success from organizational perspective [36]. Also the level of knowledge acquired and support provided to the users during and after the project is mentioned as a critical success factor for project success [37]. The figure below provides an overview of the 4 dimensions and views of project success. Also, the 5 steps of the generic development process were mapped onto the project success framework based on the extent to which each stage of the generic development process contributed to achieving the objective on operational level of the project success dimensions. Based on the activities on operational level, it seems logical to couple modeling & construction to technical success, planning to project management success, communication & deployment to organizational success.

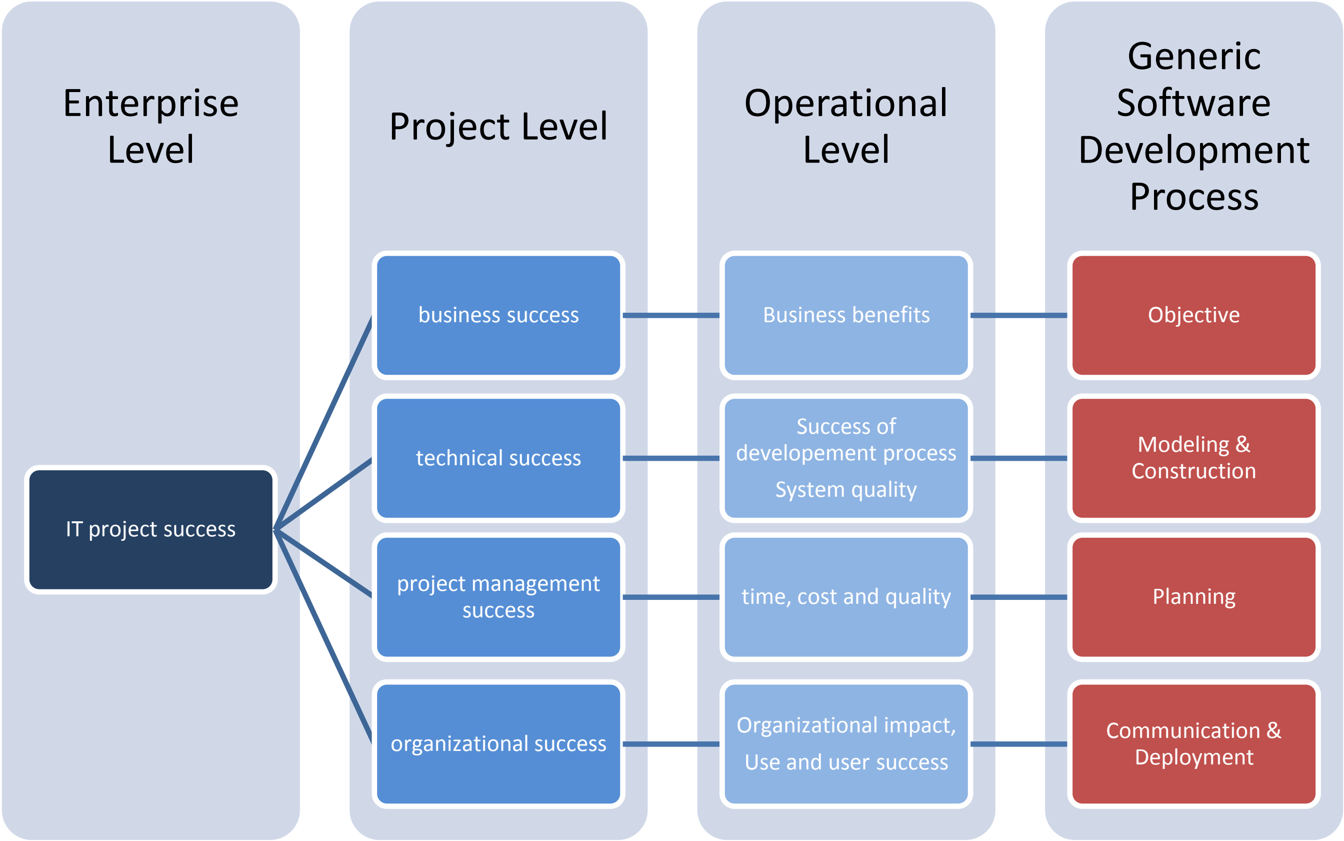


Figure 14 - Project Success Framework

5.1.2 Seligmann Framework

In the context of LIS to SOA migration where each of the 4 dimensions of CSF's determined by utilizing the IT project success Framework, the 6 ways of Seligmann will be used to determine a scientific foundation for these success factors which in the near future could be developed further into a formal methodology. The focus here is to couple the 6 ways with the 4 dimensions of the IT project success framework yet still in the context of LIS to SOA migration. The latter will be described throughout this chapter. Firstly the 6 ways of Seligmann are described as follows[4]:

The way of thinking forms the philosophy behind the methodology of how the world in context of the problem, the solution domain and different perspectives is actually perceived. Also the assumptions and viewpoints of the methodology are described as the way of thinking.

The way of modeling involves the availability of the conceptual modeling methods with its supporting notations and modeling language from different perspectives, altogether with the relationships among the different models.

The way of working structures the way of how information systems are to be developed in terms of operational tasks. It defines the possible strategies, processes and activities which need to be carried out by whom in order to deliver the product. Furthermore, guidelines and principles are provided to follow a structured and uniform way of working.

The way of controlling deals with the management and managerial aspects during information system development which include general project management, human resource management, evaluation and progress control and monitoring of performance and risk.

The way of supporting involves using tools (CASE tools) to support the activities in the way of working and modeling. Apart from technical studies, the way of supporting can also be perceived as the extent to which the way of working and modeling are being supported by stakeholders and users of the developing system. The level of commitment and support during the project is also an example.

The way of communicating defines how the abstract models designed are visualized and communicated among different layers within the organization. Often a graphical notation is used such as Object Role Modeling (ORM). In the following figure the 6 ways and their interrelations are depicted.

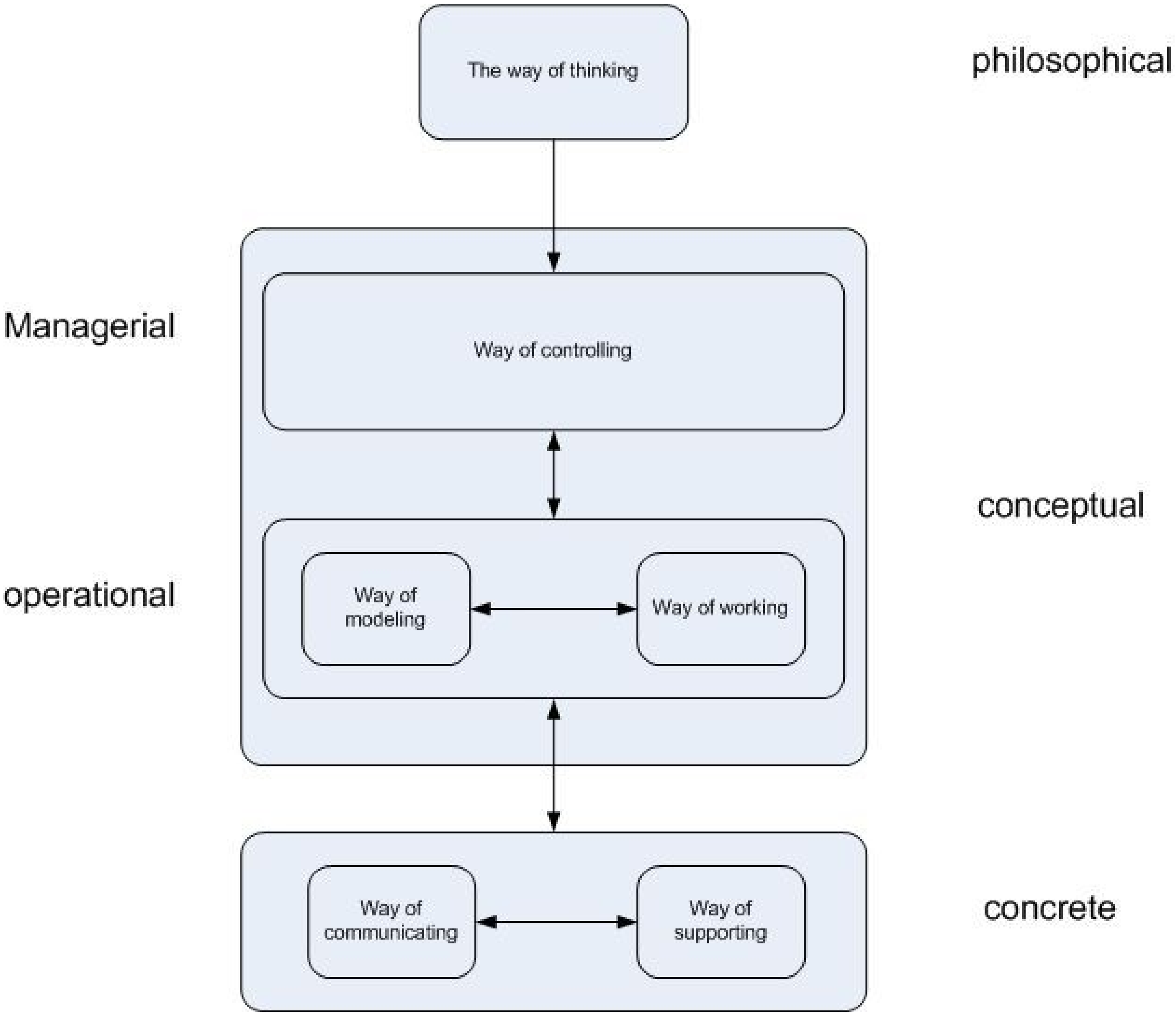


Figure 15 - Framework for Methodologies

5.2 Seligmann and Project Success Dimensions

In order to complete the where question, the 6 ways of Seligmann [4] will be coupled to the elements of IT project success [34]. These ways will be placed in the context of IT project success in order to obtain a consistent framework where the CSF's for successful migration of LIS to SOA can be placed in.

5.2.1 The Way of Thinking

The way of thinking describes the rationale or the philosophical assumptions behind the methodological approach and provides different views on the problem, solution and control domain. Rationale is defined as the fundamental reason or the basis and principles for a course of action or belief.

Placed in the context of IT Project Success we come across several similarities. The rationale in IT projects strongly is initiated or driven by one or more business reasons or business factors. The existence, co-existence and initialization of projects depend on the level of business success and business performance based on the business vision and strategies. IT projects will not be initialized if business doesn't benefit is not noticeable after the project has been executed. Whether, the benefits are perceived in financial or non financial terms there should be a benefit observed by the business. The drivers of the projects are always based on business objectives, business demand and business performance that can be achieved or improved with the execution of an IT project. The financial benefits according to project success could be expressed in Return on Investment, achievement of the goal, and project deliverable within budget. So the rationale and assumptions of executing an IT project is most of the time based on the business benefits that an organization wishes to achieve. These business benefits are closely related to the dimension of business success within the Project Success Framework. Therefore the objective **Business Success** has been defined, as methodological coverage within the Seligmann Framework, as **the Way of Thinking**. This dimension has 2 properties which are the **rationale** and the **objective**.

5.2.2 The Way of Working

The way of working is about the development process and it's supporting activities on operational level. It structures the way of how information systems are to be developed in terms of operational tasks. It defines the possible strategies, processes and activities which need to be carried out by whom in order to deliver the product. In the context of IT projects the way of working corresponds closely with the dimension technical success, since technical success is achieved by deploying the proper development or work process on operational level. Therefore **Technical Success** is coupled to the **Way of Working** in The Theoretical Framework. The properties defined are **work procedure** and **the level of prescription** of the work procedure**.** The level of prescription involves the concrete and explicitly defined tasks and deliverables of each phase of the work procedure.

5.2.3 The Way of Modeling

This involves modeling the problem and solution domains from different user perspectives each with their own corresponding notations and mappings from an abstract to a more specific level and vice versa. The way of modeling supports the technical development procedure in terms of providing modeling principles and modeling strategies in order to document, communicate, analyze and develop the solution domain from the problem domain. The way of modeling is subordinate to the development procedure which contributes to Technical Success. So the **Way of Modeling** is also coupled to **Technical Success**. The properties here are **formal notation** and **modeling strategy.**

5.2.4 The Way of Controlling

This involves the activities that are necessary to plan, structure the phases of the project and development as well as the control of the project in all stages of the development process. The way of control deals with the management and control aspects during information system development which include general project management, human resource management, evaluation and progress control and monitoring of performance and risk. Within the IT project Success Framework, project management success also focuses on these activities and tends to achieve the same objectives. Therefore a coupling is made **Project Management Success** and the **Way of Controlling.** The property defined is a **Control procedure**.

5.2.5 The Way of Communicating

The way of communicating defines how the abstract models designed are visualized and communicated among different layers within the organization. This way deals with all the communication aspects within a development methodology, which despite the communication through models also includes communication between project members and board meetings between stakeholders of the projects and peer reviews. Also, the communication with the current and future users of the system is very important. The level of communication within an organization should be very effective on each layer, starting from the end-users up until the level of SOA buy in (CIO or CEO). The higher the intensity of communication the more support will be provided by the organization where a new or changed application is being developed and deployed. Also the higher the level of awareness and support of the business will be when the level of communication is maximized. Communication adds value to the overall level of organizational support and organizational awareness on different levels. Therefore **the way of communicating** is coupled to **organizational success** of the project success framework. The defined measure here is **organizational awareness.**

5.2.6 The Way of Supporting

This involves using tools (CASE tools) to support the activities in the way of working and modeling. Apart from technical support by utilizing tools, the way of supporting can also be perceived as the extent to which the way of working and modeling are being supported by stakeholders and users of the system being developed. The level of commitment and support during the project is also an example of support. The level of awareness and commitment of the users, staff, business initiators and the project members contribute to organization wide support and ultimately to a more effective running organization. This contributes to a higher level of organizational success. Also the technical support in terms of user assistance and knowledge sharing is also perceived as a way of support to the end users. Supporting this group should increase the level of efficiency of working with the system that could result in a higher workload, which ultimately leads to a higher level organizational effectiveness. Therefore **the way of supporting** is also coupled to **organizational success** of the project success framework. The property of this dimension is defined as **user support**.

5.3 The Theoretical Framework for LIS to SOA Migration

The theoretical framework was developed by using the ***5 W's*** which are *what, how, where, who* and *with what.*

What, refers to the main objective which is necessary for an organization to achieve IT project Success. 4 dimensions were defined which are business success, organizational success, project management success and technical success. This was described in section 5.1.

How, refers to how the ***What*** part is supported by the phases of the generic software Development Process. This process is based on an agile approach which indicates that the process is both iterative and incremental. This process consists of 5 stages which are communication, planning, modeling, construction and deployment. In section 5.1 describes how the ***What*** and ***How*** have been connected with each other.

Where refers to the methodological coverage of the several dimensions of Project Success within the Seligmann Framework. The exact mapping was described in section 5.2.

Who, describes the different perspectives which were defined for the different dimensions of the framework. These perspectives are based on the tasks that need to be carried out in order to obtain the main objective. These perspectives are displayed in the figure 18.

With what describes the Critical Success Factors (CSF’s) which were identified, analyzed and evaluated in practice within 4 different organizations. These CSF’s are extensively described in chapter 6. The Theoretical Framework which was placed in the context of LIS to SOA Migration is depicted in the figure below. Based on the Theoretical Framework we have identified 4 categories of Critical Success Factors that contribute to overall IT project success which can also be placed within the Seligmann Framework. Furthermore, we also derived how each category contributes to the realization of each phase of the generic software development process of Pressman[33]. These 4 categories are business factors, organizational factors, control & management factors and technical factors. These 4 categories will be referred to as Domains. This developed framework was utilized and placed in the context of LIS to SOA migration projects, where we identified critical success factors related to each domain. A thorough analysis of these factors can be found in chapter 6.

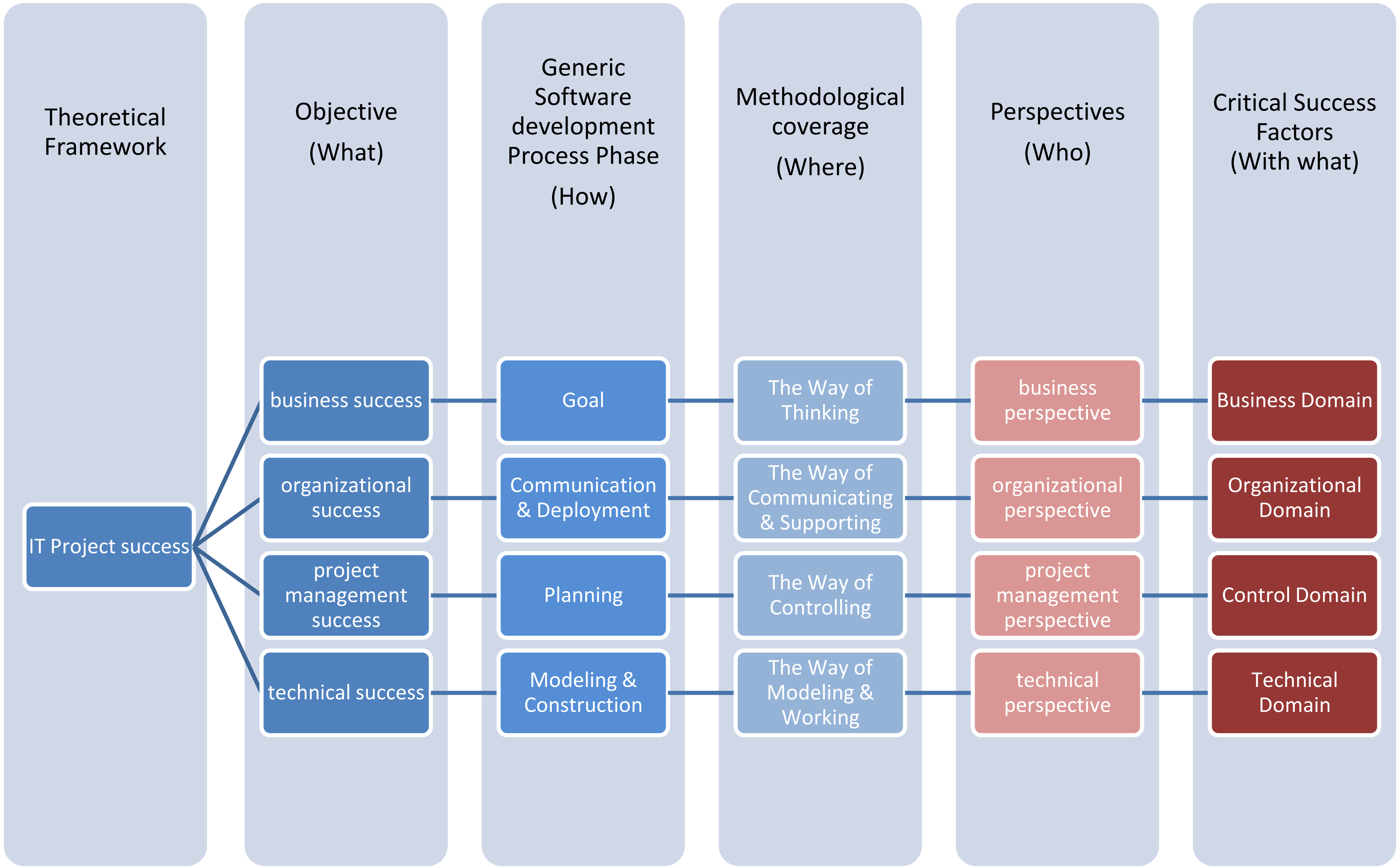


Figure 16 - Theoretical Framework for LIS to SOA Migration

6 Critical Success Factors

This chapter describes how the Theoretical Framework constructed in the previous chapter was used in the context of LIS to SOA migration in order to classify each critical success factor according to the appropriate dimension. the first two sections describe how the different success factors were placed within the proper classification. As a result, the framework of critical success factors in the context of LIS to SOA can be found in section 6.3.

6.1 Migration of Legacy Information Systems to SOA

Bisbal et.al. (1999) discussed 3 alternatives for the modernization of legacy systems toward a SOA which are redevelopment, wrapping and migration. As described before, migration involves transferring the current system to a lower cost platform while keeping most of its business value and system functionality. Since many of these organizations that rely on the daily operation of their legacy systems have most of the business logic intertwined within these silo-based applications. Most of these systems are robust and functions according to business needs. However, maintaining, extending and integrating these systems are where the complexity and cost burdens arise. Redeveloping these systems is costly and contains high risk since it is very difficult to completely understand the system, its source code and interrelations within these systems due to lack of documentation of the software. Apart from the high risk involved, the process of complete redevelopment can also be more time consuming and costly for organizations compared to maintenance. Therefore redevelopment is not viable, but also wrapping the whole system isn't an effective solution either, since wrapping is seen as a short term solution. A wrapped module still depends on the underlying infrastructure which is almost obsolete and could result in less or no support at all in the near future regarding the underlying hardware or source code. As a result these wrapped modules will be useless afterwards. Migration is often seen as the most effective long term solution when such systems need to be modernized toward a SOA. A SOA approach can be utilized to reuse the business logic which is contained and locked in the legacy systems and provide them as reusable services in order for other business units to effectively share and reuse information and knowledge. Not only will we able to reuse existing data and functionality by migrating to a SOA, but also create separation of concerns by defining business and technical services. The business services contain all business objects and business logic also known as coarse grained services, where the pure technical services also known as fine grained services contains the data objects and data functions that implement and automate the business services. Businesses will be able to configure new products or business services by simply changing the composition of technical services. This causes organizations to quickly adapt their IT environment to changing market conditions or changing rules and regulations whether these are based on legislations or mergers. Migrating legacy systems to a SOA could help organizations to lower their cost, since business services are reused by multiple departments. Also extending and integrating other applications can easily be performed by composition of other technical services, which could lead to additional or new business services. Migration of a legacy system to a SOA is not risk free and one still has to cope with several issues from different perspectives. Therefore, some factors should exist that affect the success of the migration process. In the following sections several factors that contribute to successful migration of legacy systems toward a Service Oriented Architecture are presented and positioned in the theoretical framework.

6.2 Literature Survey

The first step in our research was surveying on the subject in order to obtain a framework of Critical Success Factors (CSF's). A CSF analysis identifies a small number of goals or factors that, if they are reached or completed successfully, will tend to predict or indicate success [3]. A During the process of investigation several sources were used to gather information about this subject. Sources such as digital libraries, scientific papers and books were all analyzed.

The main terms or keywords which were used during the research are: SOA migration success, SOA success factors, legacy information system migration, legacy systems modernization success and project success. As a result several critical success factors were identified. After these findings, the relationships between the factors and the dimensions of the Theoretical Framework were analyzed in order to decide which category the critical success factor fits best. The identified critical success factors and the dimensions they were positioned in are described in the following sections.

6.2.1 Business Domain – Objectives & Benefits

By positioning the Theoretical Framework in the context of identifying critical success factors for the Migration of LIS to SOA we should be able to identify success factors such that each of these can be placed within one or more of the 4 domains.

The determined business factors contribute to achieving business success in the theoretical framework and were positioned in the way of thinking of Seligmann because the business should have a clear understanding of the SOA notion and the objectives SOA aims at. The fundamental objective of SOA should be in line with the benefits an organization wants to achieve in the long run, therefore it should be clear, from business perspective, what the benefits and possible drawbacks SOA could lead to before undertaking any SOA migration project.

6.2.1.1 Rationale & Objectives

There are several important factors that contribute to achieving business success in the context of LIS to SOA migration.

First of all, the hidden assumption or rationale from business perspective is that the business aims at a decrease of IT utilization cost, yet still extensively wants to maximize reuse of its vital legacy systems, since these systems provide high business value to their organization. Business *should define clear and measurable objectives* before embarking on a migration project. One of the main objectives is *retaining existing assets*, due to the high business value of the legacy systems. Also business should be aware of the benefits that SOA brings. The benefits of SOA include reduced IT-burden, increased Return on Investment and increased Organizational agility. Only if the *business objectives comply with the business benefits of SOA*, such migration or modernization efforts should be taken. The business benefits should be the most important driver for the migration project instead of pushing new technology standards of technological capabilities from IT perspective. Adapting to a SOA also means that the way of thinking should be adapted to *thinking in services and processes instead of functions*. A shift in mindset is necessary for the business to be successful. In a SOA the terminology shifts from processes and functions to business services and technical services. The world is perceived as container of business services which represent real business objects which are also known as coarse grained services. These services are executed by composing multiple technical services. These technical services are procedure calls and CRUD operations on data and data objects. Also a rearrangement of the organization and the way how a department performs its work and communicates could be affected since SOA focuses on reuse of services. Reusing services indicates that multiple departments should agree on the information to be consumed and provided. This also indicates that *business should be the initiator of potential migration projects*, instead of pushing state of the art technologies by the IT department, so there should be a *clear business demand* instead of an IT demand. Apart from setting a business objective before embarking on a migration project, the *cost and budget of migration* also plays a crucial role in terms of business success. If the cost is higher than its ROI and cannot be earned back in the coming years, the project will not be a success from business perspective. Another important business factor is *disruption of the business activities* during the migration project. Since the business systems strongly rely on the legacy systems for their day to day activities, there should be no disruption of the business activities during the migration project. Business continuity should be guaranteed at all times. The complete list of critical success factors that contributes to achieving business success is presented in table 1.

6.2.2 Technical Domain – Migration Strategy

In order to identify the critical success factors for the technical domain, a technical view on the migration of LIS to SOA is required. From technical perspective the theoretical framework consists of 2 criteria which need to be addressed in order to achieve success from technical perspective. The technical success factors identified were placed within one of these criteria.

6.2.2.1 Capabilities Legacy Environment

Strategy is important in migration of legacy systems to SOA. The strategies during migration projects could involve reusing most of the legacy code as components or web services, migration of specific modules or complete redevelopment. However, before organizations are able to define their migration strategy a clear overview of the current assets should be available. The capabilities of the running legacy environment and its contribution to business should be evaluated. An iterative and incremental top down research approach should be applied where the business processes that carry the highest value within the organization or mapped out onto the automated legacy systems. By evaluating the system quality against the business value, organizations have a clear overview of the most important business processes and the underlying legacy portfolio that executes these tasks. As a result one or more migration strategies could be applied. A clear understanding and inventory of both the business processes and its supporting technical environment in required before deciding upon what migration strategy to be used. Critical to the success of the migration strategy are *analysis of the reuse potential* by *understanding the business contribution of the legacy environment*, *assessment of business value against system quality*, *creating a legacy portfolio* in order to restructure or copy the code during the migration process.

6.2.2.2 Work Procedure

In the context of LIS to SOA migration using a work procedure is of fundamental importance. A work procedure or migration strategy helps the organizations coping with legacy systems to organize and structure the activities to be carried out and addresses the certain paths to follow. A migration strategy could include one or more modernization strategies such as partly wrapping, redevelopment and componentization of current assets. From business perspective, most organizations strive for a high level of reuse since redevelopment is costly and contains high risk. Redevelopment of such large legacy systems is often neglected since the risk of failure is high. This results from the lack of documentation and complete system understanding. Instead of redevelopment which is also known as the BIG BANG, organizations should consider an *agile process centric approach*. This migration strategy involves using a top *down approach iterative and incremental* starting with the *reusable processes*. Processes that are reused by multiple departments in another context are candidates for business services. After having identified the reusable business services out of the business processes, these services can be decomposed by identifying its corresponding application or the specific code that implements this service. From technical perspective, *code-refactoring* or application-reconstruction needs to be performed in order to create technical services that directly support part of the business service. Critical to the success from technical perspective is the use of an adequate migration strategy that is based on an *agile process centric approach* that also complies to the *principles of Service Design* and the proper *level of granularity* described in chapter 5. Another success factor is the use of supported *middleware technology standards and protocols* for the target operational environment. The critical success factors from technical perspective are listed in table 2.

6.2.2.3 Service Design & Modeling

Modeling is one of the most important tasks during software development projects. However, during Legacy system migration efforts the modeling tasks itself are rather straightforward. The underlying assumption of modeling services is that the design principles of SOA should set the basis for service design as described in section 4.7. Up until now there is no formal notation defined for the modeling of service oriented architectures. Service models are mostly generated using standard UML notations. However, UML doesn't support all levels of service modeling. By levels we refer to architectural layers such as the business layer, technical layer and the infrastructural layer. The most important aspects during the design phase vary from reaching the *proper level of service granularity* based on *reuse potential from business perspective*. This indicates that services should be designed based on its business objectives supported by the different *business objects*, instead of centralizing design around functional objects. Employing a *Top-down modeling strategy* starting from the business architecture and working down towards the infrastructural layers results in a Service oriented architecture of the organization. This obtained model which contains a set of business services and its decomposition to the lowest layer in the architecture is also referred to as the Enterprise Architecture based on the concept of SOA. Since SOA is closely related to Enterprise Architecture from modeling and design perspectives, Enterprise Architecture modeling notations and formalisms could be used in the modeling and design phase. A modeling notation that consists of modeling capabilities for all architectural layers as well as services modeling from different perspectives is the one managed by The Open Group, known as Archimate.

6.2.3 Control Domain – Governance & Management

There are several success factors that contribute to controlling the project from non-technical and technical perspective. From non-technical perspective *project management* principles should be applied for each deliverable in the iterated and *incremental migration process*. Since SOA is adopted incrementally within the organization, services and their capabilities have to grow step by step. Therefore *managing cost, budget and service quality* at each milestone is a critical success factor. Also the consumers and providers of the services should be defined in order to make a clear distinction between owners, users and the ones liable for the adequate operation of the service. *Governance plays a crucial role* when organizations start to steer in the direction of employing a SOA. The process of transitioning from a legacy environment to a SOA needs to be overseen, therefore, governance is critical to the success of SOA. *Documentation of all non technical* issues concerning governance should be considered, such as processes, responsibilities, policies etc. Also managing services and their service contracts in order for service consumers to locate and use the services registered by providers. Also maintaining and keeping the list of services and contracts up to date in order to keep the registries organized. Another success factor is *Monitoring*. Such that this contributes to ensuring that services live up to the service level agreements (SLA), and aid in noticing when services stop working or performing below expectations compared to its operation in the legacy environment. In order for organizations to stimulate reuse across multiple actors, legacy code needs to be regrouped or its architecture reconstructed. This ongoing process of incrementally adding more knowledge to a service according to its reuse potential needs to be managed, since changing and reconfiguring legacy code or legacy applications could have consequences for other departments who have not been considered yet but at the same time rely on some part of the application. So *proper change and configuration management principles should be in place and documented* in order to ensure business continuity and controllable change at all stages of the migration project. An overview of the critical success factors that contributes to controlling the legacy system to SOA transition can be found in table 3.

6.2.4 Organizational Domain – Awareness & Commitment

This domain accounts for the organizational implications and cultural aspects in terms of success factors that need to be accounted for in order to achieve LIS to SOA migration projects. The importance of organizational and cultural aspects is often underestimated for such migration projects. The section below describes the importance of addressing both characteristics for successful SOA migration.

6.2.4.1 Organizational Awareness

Apart from the challenges from technical perspective and the level of risk involved for the successful migration from LIS to SOA, cultural aspects are almost evenly important to reach an adequate level of success from organizational perspective. In each phase of the migration project communication and commitment of the business service owners and users is strongly encouraged, since reuse of services involves collaboration of independent departments who need to reach consensus about the potential service delivery. These departments have been operating independently as long as their existence and employing a SOA encourages them to collaborate and work together. This indicates that there is a *mind shift and culture change* required in order to think together in terms of services instead of functions bounded to internal departments. So there is a shift in culture from *individualism to collectivism*. The culture change management and the awareness of business owners to the project are critical to the success of SOA migration projects. Since the success of SOA is not perceived at the end of a migration project. It mostly takes years to see the benefits of deploying a SOA. Therefore, businesses might be very reluctant to invest and continue with the project if some achievement is not directly visible. Without the *commitment of business process owners*, it will be difficult to obtain the proper financial assets and inter-departmental support. *Communicating and involving the business system and process owners in every minor and major decision phase* within the SOA migration project is evident. Frequently communicating with different levels within the organization (director, manager, user, etc.) helps to over bridge the fear of the unknown when thinking in and using services other then they were acquainted with. The critical success factors that contribute to Organizational success are depicted in table 4.

6.2.4.2 User Support

Successful migration and deployment of a SOA also depends on the *users of the target system* and *developers and administrators of the current systems*. For the users of the target system, *the level and ease of use* have an influence on the effectiveness and efficiency of the organization as a whole. For the developers and administrators of the system an important success factor is the level *of knowledge about the system* in order to analyze the current architecture and legacy code before and during the migration projects. Organizations coping with legacy systems often do not have the knowledge to understand and analyze the legacy environment, it a critical to attract potential field experts with the proper knowledge to reverse engineer or re-document the current legacy environment. Exchanging and sharing this knowledge Is evenly important, since the organization should have engineers within the organization who can tackle problems in the future instead of strongly relying on external experts. In order to support the level of knowledge exchange and provide the proper education and training for users and administrators of the target system, a center of competency should be formed. This team or department should educate and improve the level of education concerning SOA and support the users on how to use the target systems in order to achieve a much higher level of organizational effectiveness. The critical success factors from organizational perspective are displayed in table 4.

6.3 Framework for Successful LIS to SOA Migration

The following 4 tables present the critical success factors identified by extensive literature which were placed in the context of the LIS t to SOA Theoretical Framework.

6.3.1 The Way of Thinking – Business Factors

	Business Factors		
The Way of Thinking	**Critical Success Factor**	**Description**	**Source**
	Agility, efficiency and flexibility benefits	Extent to which benefits of adjusting to business environments drive the program	[2, 27, 38]
	Financial benefits (ROI)	Extent to which benefits of increased revenues and or decreased expenses drive the program	[27, 38]
	Degree of definition of objective	Extent to which the scope and the goal of the project has been set and operationalised in terms of measureable assets	[27, 39, 40]
	Competitive, market and regulatory differentials	Extent to which competitive, market and regulatory first mover	[2, 27, 38]
	Business demand	Extent to which business demand for enhanced service from technology drives the program	[15, 27, 38]
	budget and cost of migration	Extent to which the budget and cost of the migration drives the program	[15, 27, 38, 39]
	business processes	Extent to which the core business processes of the organizations rely on legacy systems	[13, 41]
	Business continuity	Extent to which the business activities are (partly) disrupted by migration projects	[42]
	Reusability of assets	Extent to which multiple services using software technologies is a goal of the program	[38]
	Service Orientation	Extent to which technical and business staff is receptive to principles	[36]

Table 1 - Business related Factors

6.3.2 The Way of Working & Modeling – Technical Factors

The way of modeling & Working	Technical Factors		
	Critical Success Factor	**Description**	**Source**
	Migration Strategy		
	Using an incremental business process centric approach for design and development	Extent to which the business process and business objects is used as the main driver for design and development of reusable services	[2, 14, 24, 27, 41]
	Incremental approach for interface migration	Extent to which several interfaces are migrated based on the reusability by several processes and users	[14, 41, 43]
	Incremental approach for application migration	Extent to which an application is migrated in multiple steps based on reusable functionalities and reusable business processes	[14, 43]
	Incremental approach for database migration	Extent to which the target database is being fed with data from the source legacy database	[14, 41, 43]
	Incremental approach for testing functional and non functional requirements	Extent to which the functional and quality requirements comply with the service level agreements or service contracts to produce one or more business services	[14, 41, 43]
	SOA principles of design during design and implementation	Extent to which the main SOA principles are complied with during the migration project(such as reusability, composability, abstraction, loose coupling)	[27, 43-46]
	SOA implementation standards and protocols for target operational environment	Extent to which implementation standards, platform and middleware will be used(esb, wrappers, adapters, orchestration engine, platforms, datatools. ,xml)	[27, 44, 46, 47]
	Security Standards	Extent to which security supporting standards is included on the program	[38, 39]
	Capabilities Legacy Environment		

	(High) Business value of the systems	The extent to which the system contributes to achieving the business objective(s)	[1, 24]
	Interrelations of elements	Interrelations between components	[39]
	Recreation of documentation using reverse engineering practices	Using reverse engineering as a technique to recreate requirements and appropriate documentation	[15, 24, 43, 46]
	The number of reusable business processes and services in the domain	The focus should be placed on identifying high level business services rather than low level technical services	Lit 13,33,12, [47]
	System quality Did it increase or decrease the level of understanding the systems and its functionalities?	The quality of the systems in terms of maintainability, interoperability and extensibility, reliability, efficiency, functionality and user- friendliness.	[15, 46]
	Number of different platforms, norms and standards	The dependability on different vendors and use of different norms and standards	[39]
	Level of Dependency business process on legacy systems (business objective and business performance)	The extent and ease to which a business process change has influence on the system and vice versa	[24, 27, 39, 46, 47]
	Analysis of the semantics(relationships and data definitions within a business domain or business process)	The extent to which the relationship between the process, its supporting applications and data definitions are clearly understood and evaluated for reuse	[27, 46]
	Use a process centric approach(top down)	Using a top down approach in order to understand the business process and the functionalities responsible for implementation of the process	[24, 27, 44, 46]

Table 2 - Technical Factors

6.3.3 The Way of Communicating & Supporting – Organizational Factors

	Organizational Factors		
	Critical Success Factor	**Description**	**Source**
The Way of Communicating & Supporting	center of competency	Extent to which a centralized team is evident for furnishing SOA expertise help and educate the program staff	[38, 39]
	Responsibilities and roles	Extent to which responsibilities and roles of staff on the program are clearly defined for completing project tasks	[38, 39]
	Education and training	Extent to which formal skill training on services and SOA is evident for program staff	[38, 39]
	Knowledge exchange	Extent to which processes and procedures are evident for informing business and technical staff of progress of the program	[38, 39]
	Active participation and commitment of business process owners	Extent to which participation and commitment of the process owners are actively visible	[39]
	Business client participation	Extent to which business departments consent, contribute and furnish content and guidance to the program	[38]
	Organizational change management	Extent to which cultural change management is evident in helping business and technical staff embrace the program	[38]

Table 3 - Organizational Factors

6.3.4 The Way of Controlling – Control & Management Factors

The Way of Controlling	Control & Management Factors		
	Critical Success Factor	**Description**	**source**
	Management		
	Change management	Extent to which change of services is monitored and managed in order to assure efficiency and effectiveness of the services in use	[38], [25, 39]
	Configuration management	Extent to which version management is utilized and the consistency of the functional and non functional aspects of services is monitored in order to assure efficiency and effectiveness of the services in use	[25, 38, 39]
	Risk management	Extent to which procedures are evident for mitigating failure or loss caused by SOA	[25, 38]
	Information management	Extent to which procedures are evident for ensuring data integrity and quality for technical and business functions	[25, 38, 39]
	Project management	Control and monitoring of project progress in terms of cost, duration, scope and budget	[25]
	Control		
	Monitor roles and responsibilities	Roles and responsibilities of external and internal employees should be monitored	[25, 38, 39]
	Monitoring of services	Extent to which services are constantly monitored on the quality aspects defined in the SLA's	[25, 38, 39]
	SOA governance	Extent to which the functional and non functional deliverables comply with the agreed SLA's	[25, 47]

Table 4 - Control & Management Factors

Part IV Validation & Generalization

7. Empirical Evaluation

This chapter describes the evaluation of the critical success factors positioned in the theoretical framework and the procedure which was used to obtain the results. Furthermore the case study protocol and interview methodology which were used to execute the case studies and interviews is briefly presented. Also the results of the different case studies are presented and discussed within this chapter. Finally, remarks will be made and conclusions will be drawn based on the evaluation of the critical success factors.

7.1 Case Study Protocol

The case study used for the research followed the methodology of Yin [7] using replication logic for case selection and evaluation. Replication logic means that cases should be selected based on the prediction of similar results or contradicting results for predictable reasons. The cases within this project were selected based on the prediction of similar results. Cases for which legacy to SOA migration have been successfully performed were analyzed. Selecting 2 to 10 cases strengthens the empirical evaluation so more general conclusions could be drawn compared to having just a single case[7].
In order to find out how important and relevant the identified critical success factors are according to field experts, a multi case study was executed. The theoretical framework consisting of 4 dimensions of CSF's were evaluated by testing the level of importance at 4 different organizations where legacy systems were successfully migrated to a SOA. These 4 organizations were chosen based on their characteristics which are the size (>2000 employees), complexity of the legacy environment and the recent execution of the project (>2006). The cases only needed to cover successful projects in terms of technical implementation and increased business benefits. The case study was designed by constructing both semi-structured and closed questions which were meant to measure the level of importance of the CSF's positioned in the Theoretical framework.

7.2 Interview Method

The multi case study consisted of open and closed questions which needed to be answered during the interview sessions by the relevant members that contributed to the technical and business success of the project. A selection was made of 12 interviewees, 3 per project. These members had to have a different perspective on the project and were also expected to have the most knowledge regarding the project and its ongoing. Therefore business architects, information architects and software developers were asked to cooperate and provide their view concerning one similar project. As mentioned in section 2.4 the interview method of Kvale was used to construct questions to increase the reliability and validity. Kvale strongly recommends the use of themes during interview sessions. Therefore, 4 themes were developed which are business related, technical related, communication and awareness and the control and monitoring theme. These 4 themes are directly related to the 4 domains in which all the critical success factors were positioned in. Apart from open questions, closed questions were constructed to evaluate the importance of the theoretical framework of CSF's based on their experience from the project they carried out.

The interview consisted of 19 open questions and 5 closed questions which were answered during the interview sessions. By means of the open questions interviewees were asked what elements or factors contribute to LIS to SOA migration success without explicitly mentioning or referring to the theoretical framework of CSF's. For the closed questions, interviewees were asked to provide a number for the level of importance of each critical success factor of the theoretical model based on the 7-point Likert scale depicted in table 5. Each interviewee was to rank each factor from a scale from 1 to 7, where 1 stands for not important and 7 for very important. Every factor with an average score of 5.75 and above was considered important enough to be mentioned as a Critical Success Factor that contributes to successful LIS to SOA migration (CSF= $\text{Factor}_{avg} \geq 5.75$), also the lowest value rated had to have been at least a 5.

Each interview session was planned in for 90 minutes, were the first 15 minutes were used to introduce the study and provide sufficient information concerning the objective of the study and the purpose of the interview. The interview setup which consists of the introduction, open and closed questions are in Appendix A. During each interview sessions notes were taken and transcriptions were made for each organization. All the interview transcriptions are in Appendix B.

Level of Importance	Measurement
Very High	7
High	6
Somewhat High	5
Low	4
Somewhat Low	3
Very Low	2
Not Important	1

Table 5 - CSF Evaluation Scale

7.3 Results

In this section the results of the interviews are presented, introduced by describing how the results were processed. After that the results of the domain importance per organization is given followed by the factor importance results. Finally the results of the factor importance are discussed and remarks are made regarding the interesting findings about factors which did not score at the level which was expected according to literature references.

7.3.1 Processing

For all the employees which were approached and interviewed transcriptions were created and the results of the closed questions were taken. After having conducted the interviews and processing of the results, some interviewees were asked to motivate their answer, since some factors were mentioned to be of high importance by 90% of the interviewees while the same factors were noted as to have little or no importance by the other 10% and vice versa. These factors will be elaborately discussed in subsection 7.4. Each interviewee was asked to define their notion of a successful migration project from LIS to a SOA. Also the critical success factors that contributed to the level of success were asked for. After having noted all the answers transcriptions were made and factors mentioned which were logically related to one or more dimensions were added to the framework. After all transcriptions were translated to the framework, conclusions were made regarding missing factors from literature. No new factors were mentioned during the interviews or follow-up questions which were not already positioned in the framework. Before going into much detail regarding the scores and averages scores of the sub-factors within the framework, an overview is presented of the dimension importance concerning each of the 4 projects.

7.3.2 Dimension Importance

In the figure below the importance of every dimension for each organization is presented. The organizational factors have been given the highest importance level overall. Also for almost every organization the business and technical factors are dominant along with the organizational factors. The lowest level of importance has been given to the control factors according to the interviewees of the “Large Social Insurance Company” and the “Large International Road Transport Company”. When organizations are treated separately, as depicted in the figure below, Interviewees believe that control factors are least important and the organizational, business and technical factors are the ones most important. The graph in the figure below provides an overview of the average scores given by interviewees within the 4 different organizations. The underlying figure does not state anything about the level of criticality of the determined success factor itself for the migration of LIS to SOA, it simply provides an overview of which dimensions are perceived as being the most important in which organization.

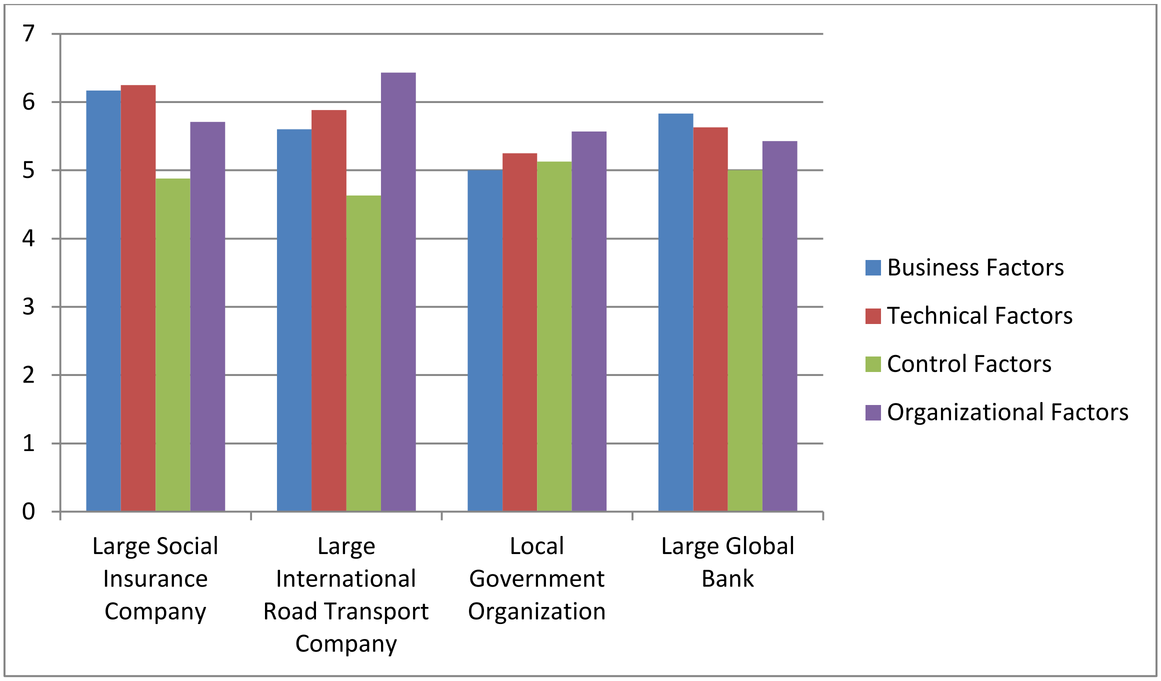


Figure 17 - Dimension Importance

7.3.3 Factor Importance

Out of the 42 factors contributing to successful LIS to SOA migration, the factors with an average score of at least 5.7 on a scale of 1 to 7 were considered as the critical success factors. The results are presented in the table below. The interviewees believe that the most critical success factors are related to the “The way of Communicating & Supporting” & “The way of Modeling & Working”. This indicates that the level of IT project success strongly depends on organizational and technical factors. The interviewees also stated that the business factors also contribute to migration success since the budget/cost of the migration and the reliance of the business process on the legacy environment has a strong influence on outcome of a migration project.

Seligmann	Dimension	Critical Success Factor	Average score
The way of Communicating & Supporting	Organizational	Active participation and commitment of business process owners	6.75
The way of Communicating & Supporting	Organizational	Business client participation	6.25
The way of Working & Modeling	Technical	SOA implementation standards and protocols for target operational	6.25

		environment	
The way of Working & Modeling	Technical	Business value of the systems	6.25
The way of Working & Modeling	Technical	Incremental process centric approach for design and development	6
The way of Working & Modeling	Technical	Incremental approach for interface migration	6
The way of Working & Modeling	Technical	SOA principles of design during design and implementation	6
The way of Working & Modeling	Technical	Level of Dependency business process on legacy systems	6
The way of Communicating & Supporting	Organizational	Organizational change management	6
The way of Communicating & Supporting	Organizational	Center of competency	5.75
The way of Thinking	Business	Degree of definition of objective	5.75
The way of Thinking	Business	Business processes	5.75
The way of Thinking	Business	Business continuity	5.75
The way of Thinking	Business	Business demand	5.75
The way of Thinking	Business	budget and cost of migration	5.75
The way of Working & Modeling	Technical	Incremental approach for testing functional and non functional requirements	5.75
The way of Working & Modeling	Technical	Incremental approach for application migration	5.75
The way of Thinking	Business	Service Orientation	5.75
The way of Thinking	Business	Agility, efficiency and flexibility benefits	5.75
The way of Working & Modeling	Technical	Process centric approach	5.75

Figure 18 - Critical Success Factors LIS to SOA Migration

7.4 Discussion & Remarks

The following sections discuss and elaborate on the 19 critical success factors depicted in the figure above. Furthermore the factors with the lowest average score will also be critically discussed. Furthermore the level of generalization based on the study findings will be discussed as concluding remark.

7.4.1 Business Factors – The Way of Thinking

The theoretical framework of success factors consisted of 10 business factors which were evaluated using 4 different cases. According to the constraints and evaluation measures, the results prove that 7 out of 10 factors are critical success factors which are “Degree of definition of objective”, “Business processes”, “Business continuity”, “Business demand”, “Budget and cost of migration”, “Agility, efficiency and flexibility benefits” and “Service Orientation”.

Interviewees believe that business process owners and executives should think more in terms of processes and services instead of functions. Moreover, they also believe that a clear definition and objective should be set at the start of the project since business owners tend to change and are unable to define their exact needs and expected benefits beforehand. An important success factor is the level of business continuity during the migration project. Business architects and information architects believe that this is crucial to the success of the ongoing projects and has high influence on the perception of business owners and their commitment for future SOA projects or broader or larger scale. Since migration of LIS to SOA is a rather complex intensive task and the few success stories available, most architects use the “learn by doing” principle. Due to this matter a fixed budget may have a large impact on the level of success of the migration project. Therefore, many interviewees mentioned that the budget and cost is crucial since there are no fixed guidelines to tackle all the problems beforehand. Some interviewees also mentioned that the duration of a migration project will take over 3 years since the project continues when the budget becomes available again.

According to Erl [23], SOA provides 3 strategic benefits which are higher ROI, Reduced IT Burden and Organizational agility, however, according to the interviewees the “financial benefits” factor was less important compared to organizational agility which was measured by the factor “agility, efficiency and flexibility benefits”. Another remarkable finding from business perspective was that “reusability of assets” was also less important and was not seen as a driver to obtain success from business perspective, which is in contrast with most of the scientific studies regarding the SOA concept. The drivers from business perspective in order to achieve business success are mostly centered around business continuity, agility and thinking in services rather than functions. Thinking in services would also indicate to think in terms of reuse; however this was not mentioned as critical by the interviewees from business point of view.

7.4.2 Technical Factors – The Way of Working & Modeling

Out of the 17 technical factors, 9 were mentioned to be the most critical success factors from technical perspective.

The scores given within this domain varies from 5.75 up to 6.25. The factor considered most important was “SOA implementation standards and protocols for target operational environment” and “business value of the system”. Interviewees strongly depend on vendor products to analyze, integrate and deploy services from the old into the new environment. In order to take the appropriate measures in terms of migration strategies and reverse engineering practices, it is considers important to analyze how the system contributes to which part of the business and business process. What also became clear was that an agile development approach was favored most. This is in line with the literature findings concerned with the type of which technical migration strategy should be adopted for such complex undertakings [2, 12, 14, 48].

There are several striking conclusions which could be made regarding factors that were mentioned as less important by average caused by the broad variation of opinions. These factors include “reusable business processes and services, Number of different platforms, norms and standards in the domain”, “System quality” and “Re-documentation using reverse engineering practices”. According to [1, 11, 40, 42], the above mentioned factors can reduce the level of complexity during the analysis and understanding phase of the migration project and gives insight into the type of migration strategy to take (wrapping, redevelopment etc.). Fortunately, some of technical experts believe that these factors are surely important for the migration of LIS to SOA migration and others believe that these factors are least important and thus on average do not contribute to the success of the migration project from technical perspective. Most of the interviewees noticed that these factors may contribute to understanding the way the system contributes to the business and what the interrelations are, however, these factors are still considered less important from technical perspective since understanding of the systems is not that complex using the proper engineers. In order to make a clear judgment about these factors more interviews need to be conducted at other organizations where migration projects at the same scale have been performed.

7.4.3 Organizational Factors – The Way of Communicating & Supporting

The factor that was mentioned as the most important critical success factor in general relates to the organizational dimension which is "Active participation and commitment of business process owners". Almost all interviewees mentioned the high importance of this factor during LIS to SOA migration projects. Participation and commitment is crucial to the success of SOA was mentioned, other interviewees noticed that SOA is shifting from the technology perspective to a culture perspective since technology is not the problem nowadays. The challenge lies in convincing and achieving the support of the business to invest in SOA solutions was said by one other interviewee. Since SOA forces organizations to restructure the way the work is performed because of using shared resources. The culture shift is a challenging aspect according to one interviewee: "where previous departments have been working independent for years are asked to change their way of working and thinking in order to achieve a higher level organizational agility". To obtain the level awareness and support needed by all the involved business departments is where the challenge lies according to the majority of interviewees.

Interviewees also believe that a center of competence should be created in order to educate the users, managers and owners on the benefits of SOA and providing support during the deployment of the newly developed SOA environment. Knowledge exchange in terms of progress reporting does not contribute to organizational success according to interviewees, also the education and skills of the program staff has scored below the limit of 5.75. The conclusions drawn here based on the interview findings are in line with the literature findings, therefore no striking conclusions were found for both the critical and not so critical success factors contributing to organizational success.

7.4.4 Control & Management Factors – The Way of Controlling

A rather striking finding was that interviewees did not believe that a single factor within the control and management factors was considered important enough to be perceived as a critical success factor. However, the author still believes that SOA governance, Monitoring of Services, risk management, and change management are evident in terms of controlling and monitoring the project progress. SOA governance in terms of SLA's and change management in terms of work processes and principles should be documented and followed as long as the services deployed are still being used. A clear documentation and taxonomy of services should be created in the registries with its providers and consumers along with the adequate SLA's. Even though the majority of interviewees believe that the control factors are less important, according to the author's findings and believe these factors should not be seen as less important. Additional studies of these factors should be performed by conducting interviews with field experts who also carried out SOA migration projects of the same scale.

Since the beliefs of the field experts concerning the importance of control & management factors contradict the findings of the literature study, more time was spent to find out whether the outcomes of this dimension can be judged or verified by performing additional research. The results found here show strong resemblance if we compare them with the Independent SOA Maturity Model(ISMM) found in [45, 49]. A brief description of the ISMM is given in Appendix D.

As mentioned earlier there are strong resemblances when the results are compared to the ISMM, and in particularly on level 1 of the maturity model. This level of maturity often involves independent SOA project. Within these projects enterprises find them self at the starting point and gain first experiences with services. During this phase of SOA adoption, there is often no common technology or standards used. Also, services are designed and deployed without reuse capabilities. The services architecture is formed by a list of unstructured and unsorted services without taxonomies and couplings of actual consumers and providers of services. This level is mostly regarded as the level where experience is gained and best practices are developed for the implementation of an enterprise scale SOA. Therefore there is no intention of monitoring services, creating a clear and unambiguous service repository with a separation of service consumers and providers and their binding SLA's, thus no Governance mechanisms yet. If the interviewees who defined the above mentioned factors as being less important were carrying out migration projects based on the characteristics above, we may conclude that the beliefs of these interviewees regarding SOA governance and other control mechanisms can be justified. Yet still, the control factors remain important and must not be discarded.

7.5 Generalization of Success Factors

Generalization is defined as the applicability of evidence to situations and populations other than those in which the evidence was produced [50]. Although the objective of qualitative research is not generalization but has a more exploratory and descriptive character, yet still an attempt was taken to generalize the findings based on the statistical evidence that was within reach for this research project. In total 42 factors were evaluated by using 4 different LIS to SOA migration project as case study input. Several interviewees were asked to evaluate the model in terms of importance based on their experience formed by their work performed on the migration projects and their general opinion as field experts. Out of 42 factors 19 of them were believed to critical according to the limit which was 5.7 by average. The theoretical framework covers 45% of critical success factors based on the results of the evaluation by multiple interviewees at 4 large independent organizations. The table below provides an overview of the total factors and factors for every dimension. The graph in the figure below also gives an overview of the critical and less critical success factors. These 19 critical success factors are generalized in the context of this research study, since generalization was based on common factors identified as a result of successful LIS to SOA migration by the interviewees who are field experts in the area of this research. However, evaluating these 19 factors at 4 independent organizations by their corresponding interviewees still provide insufficient evidence for statistically grounded conclusions. Therefore statistical generalization of the conclusions to other SOA migration projects other than the ones analyzed as part of this research study is not (yet) possible. However, as stated in section 2.4, statistical sampling and statistical generalizability is not necessary or even justifiable in qualitative research. According to [6], theoretical generalization should be the aim when performing quantitative research studies. In order to achieve theoretical generalization, replication logic should be used combined with multiple case studies varying between 2 and 10 cases. Since 4 cases were designed and evaluated using replication logic and common factors determined based on professional judgment of interviewees in general and based on successful implementation of LIS to SOA migration projects, one may conclude that theoretical generalization of this study is in fact achieved for LIS to SOA migration projects. The theoretical framework that has been developed can be used as a foundation for generalization to other cases that have not been studied which are in scope of the research area and theory of study.

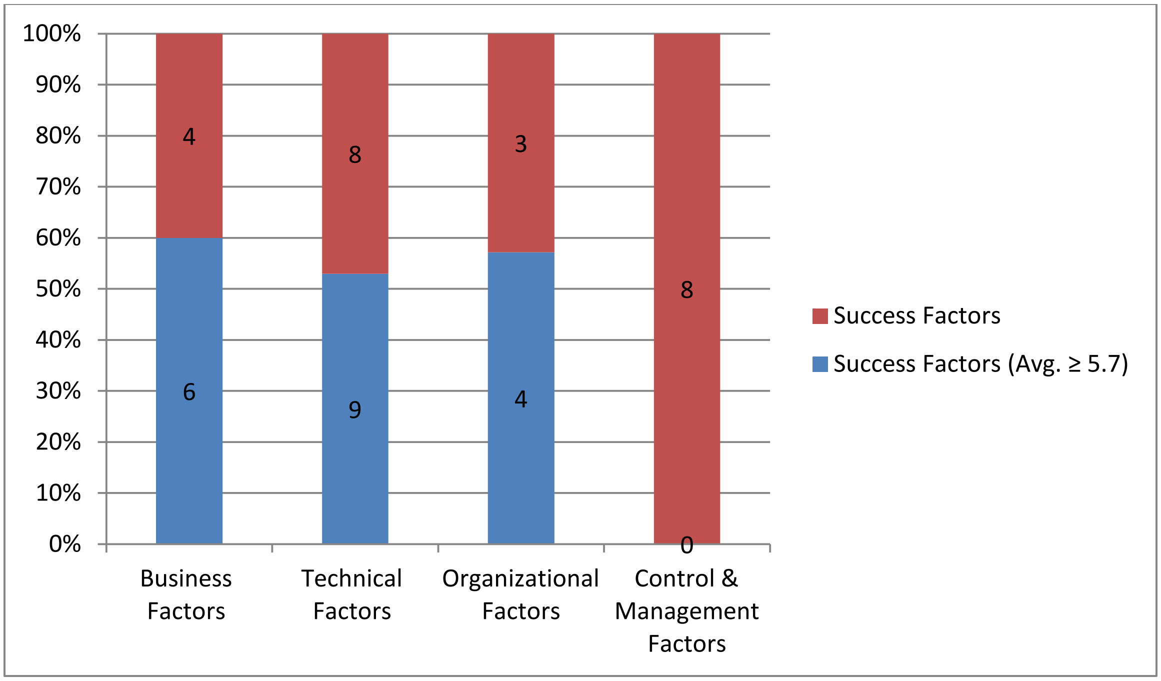


Figure 19 – Validated CSF's

Seligmann	Dimension	Nr. of CSF's (%)	Total Factors
The Way of Thinking	Business	6 (60%)	10
The Way of Working & Modeling	Technical	9 (53%)	17
The Way of Communicating & Supporting	Organization	4 (57%)	7
The Way of Controlling	Control & Management	0 (0%)	8
Total coverage of theoretical framework	**LIS to SOA migration**	**19 (45%)**	**42**

Table 6 - CSF Coverage of Theoretical Framework

Part V – Conclusions & Recommendations

8. Conclusions

This chapter describes the main contributions of the thesis, the process, the results, answers to the research questions and the main research goal.

8.1 Contributions

The main contribution of this thesis from a scientific perspective is the identification and overview of the critical success factors that potentially contribute to the successful migration from LIS to SOA. Also how these factors relate to the dimensions that contribute to achieving IT project Success in general was described. Furthermore an analysis was given of how the success factors relate to and could be positioned in the Framework of Seligmann. Apart from the identification of success factors that contribute to SOA migration success, a theoretical framework was developed independently of these success factors which can be utilized in some other context, for example to evaluate the success level of other IT related projects.

8.2 Process and Results

The main research goal of this thesis was:

"To identify, analyze, collect and evaluate the critical success factors that contribute to successful LIS to SOA migration projects"

The thesis project addressed the research goal by first developing a framework that contributes to IT project success in general. This was performed by analyzing and studying the elements and dimensions that contribute to IT project success in general. Furthermore, possible success factors that contribute to successful migration of LIS to SOA were analyzed and were each positioned in the theoretical framework.

The design and execution of the multi case study was performed to improve and evaluate the theoretical framework of success factors. The case study was performed at 4 large organizations where migration projects were carried out over within the last 6 years. The case study was executed by using interviews based on open and closed questions.
The result of the case studies was used to improve the framework by identifying the most critical success factors according to the interviewees. Out of 42 factors identified, 19 were mentioned as being the most critical success factors based on successful LIS to SOA implementations within 4 organizations. A striking result of the multi case study results were that no single control & management factor was rated above 5.7. According to the results none the "Control & Management" factors were found to be critical success factors. Approximately 50% of all the success factors determined using extensive literature study and literature analysis were believed to be critical success factors according to the field experts (interviewees).

The literature review consisted of an extensive examination of migration methods, possible success factors, theoretical models, evaluation models, descriptions, research papers, white papers and survey data concerning the migration of Legacy Information Systems onto A Service Oriented Architecture that led to the findings of possible critical success factors mentioned in this thesis.
Although not part of the thesis, validations of the interview results were carried out by performing a post- literature examination. In order to find out whether the results found in practice regarding the importance level of the factors could be confirmed by literature findings. The post-literature study which was done after the evaluation of the theoretical framework was found to be very practical since most of the results were in line with the results of the evaluation.

8.3 Research Questions

The main research question is:
"What elements contribute to successful SOA migration and how can these elements be generalized to maximize the level of success in legacy systems migration projects?"

The main research question was answered by answering each one of the 3 sub questions listed below.

1. *What is IT project success and what are the critical success factors for the Migration of legacy information systems to SOA from scientifical perspective?*
   The critical success factors were determined by extensive literature study and examination of legacy migration methods and best practices described in books and white papers. The list of critical success factors determined as a result of the literature examination is described extensively in chapter 6(section 6.2 and section 6.3). These factors were positioned in the theoretical framework which was developed independently. This is described in section 6.1 and 6.2.

2. *What are the critical success factors for the migration of legacy information systems to SOA based on practical experience?*
   The qualitative research method was chosen to gather empirical evidence in order to improve and evaluate the theoretical framework of success factors developed in sub question 1. The research process followed the interview method of Kvale[9] and the multi case study design of Yin[7]. The multi case study design and interview method is extensively described in chapter 2 (section 2.3). The interviews were carried out as part of the case study. A multi case design was created and executed at 4 large organizations using interviews which were based on open and closed questions. The case study design and interview questions can be found in Appendix A. the results of the case study are discussed in chapter 7 (section 7.3). Detailed result information about the interview transcriptions and the results of the closed questions can be found in Appendix B and Appendix C, respectively.

3. Given the answers on question 1 and 2 how can we design a framework of critical success factors for the migration of Legacy Systems to a SOA?
   After the evaluation and improvement of theoretical model which was done in sub question 2, all factors with a minimum average score of 5.7 was considered a critical success factor for the migration of LIS to SOA. Based on the the factors determined in literature, their evaluation based on expert opinions and the post-literature examination, we may conclude that the 19 factors determined in chapter 7 (section 7.5) may be considered as the general success factors in the context of LIS to SOA migration for these 4 organizations. However, the author still believes that Control & Management factors are critical as well to achieve successful LIS to SOA migration which is also extensively discussed in section 7.4.4. Theoretical generalizability has been fully achieved and therefore the theory developed as a result of this research study does provide strong analytical and explanatory theory for the experiences of others who are in comparable situations. The theory that is formulated must then become the vehicle for generalization to other cases that have not been studied.
   However, statistical generalization of the critical success factors to SOA projects other than those which were taken under study is partly possible due to the insufficient evidence for statistically grounding conclusions. In order to achieve complete statistical generalization of CSF's for LIS to SOA migration the developed framework should be utilized for a broader range of migration projects.

8.5 Main Research Goal

In order to achieve generalization in the context of successful LIS to SOA migration, both theoretical and statistical generalization should be reached. As mentioned earlier, the aim of this research was to develop a framework of CSF's for the migration of LIS to SOA and to achieve generalizability of critical success factors for LIS to SOA migration. Since this study was by and large based on quantitative research, therefore theoretical generalizability and partly statistical generalization has been achieved so far, which means that the theory developed can be used as a basis to study other cases which are in the area of research and have not been researched yet. The theoretical framework developed forms a solid foundation of success factors that contribute to the different dimensions that ultimately lead to successful migration of LIS to SOA. Since both theoretical and statistical generalizability is necessary in order to reach the objective clearly, further research is necessary. The type of research and course of action is described in chapter 9.

9. Limitations and Future Work

This chapter describes the limitations of the research study and interesting propositions for further research.

9.1 Limitations

The results of this thesis project are sufficient to draw conclusions regarding the theoretical generalization of critical success factors for the migration of LIS to SOA. The theory developed here can be used as a foundation to study other cases which are in scope of the research area and have not been studied yet. However, the conduction of interviews and studies at only 4 organizations aren’t sufficient to draw conclusions regarding statistical generalization.

In order to achieve both theoretical and statistical generalization further research is necessary. The extensive literature examination and the positioning of the factors in the framework was performed from a personal viewpoint, influenced by my personal skills and knowledge gathered about the critical success factors related to SOA and Legacy Information Systems. In order to obtain more reliable theoretical framework of critical success factors analysis of the material and positioning of these factors in the framework should be performed by different persons.

The literature review was performed using a large library of resources consisting of online scientific libraries as well as information made available by TU Delft and Logica.

9.2 Future Work

Since this research study is insignificantly based on quantitative research due to its undersized statistical evidence for statistically grounded conclusions, judgments concerning statistical generalization of critical success factors are not possible for organizations other than the ones in which the studies were performed for. It would be interesting to test the identified success factors by performing extensive quantitative research with a larger population with a higher sample size. This indicates including a larger number of people and organizations per dimensions of the theoretical framework in a quantitative research in order to judge the statistical generalizability of the success factors in migration of Legacy Information Systems to SOA, thus gathering of more empirical evidence for the theoretical framework is necessary.

As mentioned above other factors may exist that were unavailable during the period of this research project, therefore there is still room for further exploration of the findings of this thesis on how far they influence the migration process and its importance on which scale (small, mid-sized or large organizations). Such research can be performed by both qualitative and quantitative methods with a focus on exploration.

9.3 Final Remark

It is the author’s belief that this master thesis is a valuable addition to research in the field of SOA and LIS. With quite an extensive list of critical success factors based on thorough literature examination during the analysis and validation phase with a broad coverage of almost 50% of the most critical success factors according to 4 large migration projects performed within the last 6 years, presents other researchers with interesting material for further research.

Appendices

A. Case Study Interview

The following questionnaire was produced by conducting both extensive literature study regarding success factors that contribute to the main goal which is: successful migration of legacy systems to a SOA. Based on the factors found as a result of surveying literature, a framework was designed consisting of critical success factors for the migration of legacy information systems onto a SOA. In order to evaluate this model a multi case study design approach was followed using replication logic.

The goal of the study was to identify and evaluate the success factors found both in literature and in practice in order to generalize from practice to theory, in other words to achieve theoretical generalization. In order to generalize the factors identified in literature a practical evaluation is necessary. Therefore several cases were analyzed by using a uniform semi-structured questionnaire, where questions were derived based on the structure of the designed framework. A list of the case study questions, which was used for the evaluation of each of the 4 projects, is presented below.

Case study questions

Introduction:

1. Company name:
2. Working position:
3. Duration of employment:
4. Answers to be treated confidentially? Y / N
5. When did the company adopt SOA?
6. What was the reason to adopt SOA?
   a. Technical initiative
   b. Business initiative
   c. Other, ……
7. What were the promised benefits?
8. Can you provide me a short description of the organization:
   a. Business goals and services
   b. Number of departments and employees
   c. SOA environment
9. What was your role during this project?

The main success factors:

10. Has the organization conducting a migration from a legacy environment to a SOA succeeded? How did you measure this in terms of SOA promised benefits such as
    a. Reusability
    b. Agility
    c. Efficiency
    d. Cost reduction
    e. Faster time to market
    f. Increased return on investment
    g. Increased organizational agility
    h. Increased maintainability

11. What were according to you the main success factors which contributed to the migration of legacy systems to a SOA environment?
    a. From technical perspective(technical factors)
    b. Business perspective(business factors)
    c. Organizational perspective(cultural or organizational factors)
12. How did you analyze and assess the legacy system in order to decide which parts needed to be migrated and which parts did not? Why?
    a. Low level code
    b. Business process approach(top down)
    c. Or other, …..
13. How clear were the business objectives and business performance indicators set during the initiation of the project? And how were these measured from beginning to the end of the project?
14. To what extent was the cost and budget of the migration project an important success factor within this project?
15. Did you use a migration strategy during this project?
    a. What type of strategy did you use?
        i. Bing bang approach
        ii. Incremental approach top down(process oriented)
        iii. Incremental approach bottom up(code oriented)
16. To what extent did the use of SOA implementation standards and protocols such as middleware and web services contribute to the success of the migration project?
17. How did you achieve to bridge the culture change within the organization, thus create organizational effectiveness and commitment during this project?
18. How important was the commitment of the business and other organizational entities to the level of achieved success for this project?
19. How did you monitor and control the project in terms of:
    a. SOA governance(SLA's)
    b. business and application services
    c. roles and responsibilities
    d. Risk management
    e. Project management
    f. Quality of service

Closed questions:

In the following tables you will find a list of factors. On top of each table the main theme is given followed by the factors contributing to achieving this main theme. There are 4 tables with each main 4 themes and several sub factors per table.

To what extent are these 4 main themes and the contributing sub factors important for the migration of legacy information systems to a SOA?

To provide an answer please use number and draw this down in the Measurements column.

In order to provide your notion on the importance level please use the seven point scale described below to determine the importance level of each sub factor and main theme.

Level of Importance	Measurement
Very High	7
High	6
Somewhat High	5
Low	4
Somewhat Low	3
Very Low	2
Not Important	1

Table 7 - Possible Answers CSF's Evaluation

The 4 main themes are displayed in the following tables.

	Business Factors		
Business Objective & Performance	**Critical Success Factor**	**Description**	**Measurement**
	Agility, efficiency and flexibility benefits	Extent to which benefits of adjusting to business environments drive the program	
	Financial benefits (ROI)	Extent to which benefits of increased revenues and or decreased expenses drive the program	
	Degree of definition of objective	Extent to which the scope and the goal of the project has been set and operationalised in terms of measureable assets	
	Competitive, market and regulatory differentials	Extent to which competitive, market and regulatory first mover	
	Business demand	Extent to which business demand for enhanced service from technology drives the program	
	budget and cost of migration	Extent to which the budget and cost of the migration drives the program	

	business processes	Extent to which the core business processes of the organizations rely on legacy systems	
	Business continuity	Extent to which the business activities are (partly) disrupted by migration projects	
	Reusability of assets	Extent to which multiple services using software technologies is a goal of the program	
	Service Orientation	Extent to which technical and business staff is receptive to principles	

Table 8 - Business objective & Performance Theme

	Technical Factors		
Technical development Theme	Critical Success Factor	Description	Measurement
	Migration Strategy		
	Using an incremental process centric approach for design and development	Extent to which the process is used as the main driver for design and development of reusable services	
	Incremental approach for interface migration	Extent to which several interfaces are migrated based on the reusability by several processes and users	
	Incremental approach for application migration	Extent to which an application is migrated in multiple steps based on reusable functionalities and reusable business processes	
	Incremental approach for database migration	Extent to which the target database is being fed with data from the source legacy database	
	Incremental approach for testing functional and non functional requirements	Extent to which the functional and quality requirements comply with the service level agreements or service contracts to produce one or more business services	
	SOA principles of design during design and implementation	Extent to which the main SOA principles are complied with during the migration project(such as reusability, composability, abstraction, loose coupling)	
	SOA implementation standards and protocols for target operational environment	Extent to which implementation standards, platform and middleware will	

		be used(esb, wrappers, adapters, orchestration engine, platforms, datatools. ,xml)	
	Security Standards	Extent to which security supporting standards is included on the program	
	Capabilities Legacy Environment		
	(High) Business value of the systems	The extent to which the system contributes to achieving the business objective(s)	
	Interrelations of elements	Interrelations between components	
	Recreation of documentation using reverse engineering practices	Using reverse engineering as a technique to recreate requirements and appropriate documentation	
	The number of reusable business processes and services in the domain	The focus should be placed on identifying high level business services rather than low level technical services	
	System quality Did it increase or decrease the level of understanding the systems and its functionalities?	The quality of the systems in terms of maintainability, interoperability and extensibility, reliability, efficiency, functionality and user- friendliness.	
	Number of different platforms, norms and standards	The dependability on different vendors and use of different norms and standards	
	Level of Dependency business process on legacy systems (business objective and business performance)	The extent and ease to which a business process change has influence on the system and vice versa	
	Analysis of the semantics(relationships and data definitions within a business domain or business process)	The extent to which the relationship between the process, its supporting applications and data definitions are clearly understood and evaluated for reuse	
	Use a process centric approach(top down)	Using a top down approach in order to understand the business process and the functionalities responsible for implementation of the process	

Table 9 – Technical development Theme

Organizational Commitment & Support	Organizational Factors		
	Critical Success Factor	**Description**	**Measurement**
	center of competency	Extent to which a centralized team is evident for furnishing SOA expertise help and educate the program staff	
	Responsibilities and roles	Extent to which responsibilities and roles of staff on the program are clearly defined for completing project tasks	
	Education and training	Extent to which formal skill training on services and SOA is evident for program staff	
	Knowledge exchange	Extent to which processes and procedures are evident for informing business and technical staff of progress of the program	
	Active participation and commitment of business process owners	Extent to which participation and commitment of the process owners are actively visible	
	Business client participation	Extent to which business departments consent, contribute and furnish content and guidance to the program	
	Organizational change management	Extent to which cultural change management is evident in helping business and technical staff embrace the program	

Table 10 - Organizational Commitment & Support Theme

Monitoring & Control	Control & Management Factors		
	Critical Success Factor	**Description**	**Measurement**
	Management		
	Change management	Extent to which change of services is monitored and managed in order to assure efficiency and effectiveness of the services in use	
	Configuration management	Extent to which version management is utilized and the consistency of the functional and non functional aspects of services is monitored in order to assure efficiency and effectiveness of the services in use	
	Risk management	Extent to which procedures are evident for mitigating failure or loss caused by SOA	
	Information management	Extent to which procedures are evident for ensuring data integrity and quality for technical and business functions	
	Project management	Control and monitoring of project progress in terms of cost, duration, scope and budget	
	Control		
	Monitor roles and responsibilities	Roles and responsibilities of external and internal employees should be monitored	
	Monitoring of services	Extent to which services are constantly monitored on the quality aspects defined in the SLA's	
	SOA governance	Extent to which the functional and non functional deliverables comply with the agreed SLA's	

Table 11 - Monitoring & Control Theme

B. Interview Transcriptions

B1. Large Social Insurance Company

Introduction:

1. Company name: Logica
2. Working position: Enterprise Architect
3. Duration of employment: 15 years
4. Answers to be treated confidentially? N
5. When did the company adopt SOA? 1998 - 2000
6. What was the reason to adopt SOA?
   The reason to adopt SOA back then was Enterprise Application Integration, thus a technical initiative rather than a business initiative
7. What were the promised benefits? The promised benefits were a reduced level of complex interfaces
8. Can you provide me a short description of the organization:
   a. Business goals and services: Agility, quicky adopting legal changes, client centric
   b. Number of departments and employees 3.000
   c. SOA environment Oracle/proposed Tibco
9. What was your role during this project? Lead Architect

The main success factors:

10. Has the organization conducting a migration from a legacy environment to a SOA succeeded? How did you measure this in terms of SOA promised benefits such as
    a. Reusability - architecture consisted of limited number of generic services and was based on a Model Driven Architecture
    b. Agility – the organization wants to be customer centric and offer customized services, also for local instances – e.g. social security
    c. Efficiency - strong steering on efficiency by generalizing employee profiles and limiting number of specialists
    d. Cost reduction - less manual inference, but automated via chain (registries) for 96%. 2% via customer portal and 2% complex cases via employee.
    e. Faster time to market – increase and faster implementation for the new legislations
    f. Increased return on investment - scalable solution, minimize development effort must result in 20% higher productivity.
    g. Increased organizational agility - end of regional and functional boundaries in business. Help client anywhere, anytime
    h. Increased maintainability - 80% is model driven generated software components or services. Focus on maintaining business models in close cooperation was client business.

11. What were according to you the main success factors which contributed to the migration of legacy systems to a SOA environment?
    a. From technical perspective – Seamless bridge between legacy and service buss based on tibco connectivity software.
    b. Business perspective – Possibility of migrating legacy per settlement via a risk based approach. End-of-life when all settlements within specific legacy system are migrated.
    c. Organizational perspective - Migration from a vertical settlement oriented business towards an horizontal client focused approach.
12. How did you analyze and assessed the legacy system in order to decide which parts needed to be migrated and which parts did not? Why?

a. Low level code
b. Business process approach(top down) Yes, based on business process
c. Or other, system tiers in order to define the information streams with legacy within a specific business process.

13. How clear were the business objectives and business performance indicators set during the initiation of the project? And how were these measured from beginning to the end of the project? First part was a business case based on which the architecture effectiveness could be measured. A migration business-wise approach was defined at the beginning of the project.
14. To what extent was the cost and budget of the migration project an important success factor within this project? Since migration would take a couple of years, conversion costs/budget were critical.
    a. Did you use a migration strategy during this project?
    b. What type of strategy did you use?
        i. Bing bang approach
        ii. Incremental approach top down(process oriented)← based on risk assessment
        iii. Incremental approach bottom up(code oriented)
15. To what extent did the use of SOA implementation standards and protocols such as middleware and web services contribute to the success of the migration project? Process parts and knowledge domains where automatically translated into web services. Tibco allowed for smooth connecting different services in an automated way.
16. How did you achieve to bridge the culture change within the organization, thus create organizational effectiveness and commitment during this project? A specific governance deliverable was part of the project. Culture was part of this governance roadmap.
17. How important was the commitment of the business and other organizational entities to the level of achieved success for this project? Crucial, since it was model driven, main steering had to come from business. Approach was agile en iterative requiring high involvement of business.

B2. Large International Road transport Organization

Introduction:

1. Company name: Large International Road Transport Organization
2. Working position: information analyst, system designer & systems developer Oracle SOA Suite
3. Duration of employment: 10 months at this organization
4. Answers to be treated confidentially? Y ~~/ N~~
5. When did the company adopt SOA? 2007
6. What was the reason to adopt SOA?
    a. Technical initiative
    b. Business initiative: Strong business initiative
    c. Other, ……
7. What were the promised benefits? Business process automation, flexibility to quickly adopt business process changes, automated document creation, improved billing process, introduction of Digital Dossier
8. Can you provide me a short description of the organization:
    a. Business goals and services – This organization handles on behalf of the Dutch local government the applications of transport licensing regarding road transportations
    b. SOA environment - Oracle SOA Suite 10.1.3.1, Oracle 10g database, Oracle JHeadstart/ADF (front end app)
9. What was your role during this project? Process analyst, database designer, SOA designer and developer

The main success factors:

10. Has the organization conducting a migration from a legacy environment to a SOA succeeded? How did you measure this in terms of SOA promised benefits such as
    a. Reusability - generic services designed for the licensing process
    b. Agility – process change was easy implemented as a result of generic services
    c. Efficiency – higher level of service efficiency
    d. Cost reduction
    e. Faster time to market
    f. Increased return on investment
    g. Increased organizational agility – model driven software and reusable services. Ease of orchestration of newly services and compositions of components
    h. Increased maintainability – model driven software components which is easy manageable

11. What were according to you the main success factors which contributed to the migration of legacy systems to a SOA environment?
    a. From technical perspective - small team of experts, agile development method
    b. Business perspective - process orchestration, easy to adopt to changes, self documenting process implementations
    c. Organizational perspective - open minded key players who are willing to invest and cooperate in the SOA initiative. Also patient players since the soa benefits will be observed years after.
12. How did you analyze and assessed the legacy system in order to decide which parts needed to be migrated and which parts did not? Why?
    a. Low level code
    b. Business process approach(top down): Built from scratch in close cooperation with key users to meet actual needs
    c. Or other, …..
13. How clear were the business objectives and business performance indicators set during the initiation of the project? And how were these measured from beginning to the end of the project? Continuous testing/reviewing by key users

14. To what extent was the cost and budget of the migration project an important success factor within this project? Cost was an important factor since the budget was limited and based on the success of early small successful deliverables
15. Did you use a migration strategy during this project?
    a. What type of strategy did you use?
        i. **Bing bang approach based on process oriented analysis**
        ii. Incremental approach top down(process oriented)
        iii. Incremental approach bottom up(code oriented)
16. To what extent did the use of SOA implementation standards and protocols such as middleware and web services contribute to the success of the migration project? Very much, SOA Suite is based on open standards and Web services and has a very good design time UI.
17. How did you achieve to bridge the culture change within the organization, thus create organizational effectiveness and commitment during this project? By involving key user from the very start until the end of the project, communicating on a daily basis
18. How important was the commitment of the business and other organizational entities to the level of achieved success for this project? Very important, the whole business case for the project was business driven (not from technology).
19. How did you monitor and control the project in terms of:
    a. SOA governance(SLA's)
    b. business and application services
    c. roles and responsibilities
    d. Risk management
    e. Project management agile development
    f. Quality of service testing, open communication, agile development, early testing

B3. Large Local Government Organization

Introduction:

1. Company name: Large local Government Organization
2. Working position: Consultant & Software Developer
3. Duration of employment: *1 year for this project(3,5 years for this customer) & 10 years*
4. Answers to be treated confidentially? Yes absolutely, actually you have to have some manager clear my support to your research before you use the answers at all
5. When did the company adopt SOA? SOA platform was designed and developed in 2009, a year later this platform went live
6. What was the reason to adopt SOA?
    a. **Technical initiative**
    b. Business initiative
    c. Other, ……

7. What were the promised benefits? Flexibility, reuse and shorter development lifecycle, faster Time To Market, Robust, Performance, long term Re-usability

8. Can you provide me a short description of the organization:
    a. Business goals and services :reintegration of the unemployed
    b. Number of departments and employees: multiple departments consisting of a total 3000 employees
    c. SOA environment: Oracle SOA Suite
9. What was your role during this project? Java software architect/developer & information architect

The main success factors:

10. Has the organization conducting a migration from a legacy environment to a SOA succeeded? How did you measure this in terms of SOA promised benefits such as
    a. Reusability - multiple systems now use the delivered SOA landscape, components are ready to be reused, however, reuse was left out of the scope of this project

    b. Agility - less than expected, reuse means coupling
    c. Efficiency - better cost efficiency for enhancements, but lower machine efficiency. Extensive pre and post testing of services to achieve at least the same level of performance as the legacy applications.

    d. Cost reduction - In the beginning up until the end of the project implementation cost was high. Creating the soa landscape was much more effort then creating it as normal applications, but enhancements are cheaper. Implementing future change or modifications will be a lot cheaper.
    e. Faster time to market - for extensions, yes, but when service interface change, absolutely not
    f. Increased return on investment – not measured
    g. Increased organizational agility – not measured
    h. Increased maintainability for the main services and consumers absolutely, but the middleware layer has the risk to grow into a spaghetti

11. What were according to you the main success factors which contributed to the migration of legacy systems to a SOA environment?
    a. From technical perspective (technical factors) clear separation of service responsibilities. Stable interfaces. Also Involve product specialists, do POC, start small and work towards a small deliverable to obtain more trust and commitment of the business process owners and users.

b. Business perspective – Apart from defining clear business goals and business requirements, the customer needs to think in services for implementing business value, not applications

c. Organizational perspective - Re-organize first and building a SOA comes after. Customers need to realize that their services are now share among colleagues at other departments. They need to talk about using each other's services as well as. Therefore consensus is needed on organizational level before even starting to think in terms of building a SOA.

12. How did you analyze and assessed the legacy system in order to decide which parts needed to be migrated and which parts did not? Why?
    a. Low level code
    b. Business process approach- top down approach combined with SIG(Software Improvement Group)
    c. Or other, ….. SIG analyzed the code base and we knew from experience that the whole system had to be rebuilt. Business process approach to identify the main processes, their sub processes and the coupling with the legacy systems.
13. How clear were the business objectives and business performance indicators set during the initiation of the project? And how were these measured from beginning to the end of the project?
    Somewhat clear, measured by business review of functional designs
14. To what extent was the cost and budget of the migration project an important success factor within this project?
    Cost was not really a factor for this project, but we a had a more or less fixed scope and timeline
15. Did you use a migration strategy during this project?
    a. What type of strategy did you use?
        i. Bing bang approach <= more or less
        ii. Incremental approach top down(process oriented)
        iii. Incremental approach bottom up(code oriented)
16. To what extent did the use of SOA implementation standards and protocols such as middleware and web services contribute to the success of the migration project?
    Unknown
17. How did you achieve to bridge the culture change within the organization, thus create organizational effectiveness and commitment during this project?
    Lots of customer contact in writing the specifications on a service level. This was not achieved to the expected level, since business driver was performance and robustness. The project was technology driven.
18. How important was the commitment of the business and other organizational entities to the level of achieved success for this project?
    Since the project was mostly technology driven business commitment was necessary however not more or less compared with other projects.
19. How did you monitor and control the project in terms of:
    a. SOA governance(SLA's): Not monitored
    b. business and application services: functional service design review
    c. roles and responsibilities: not monitored
    d. Risk management: testing
    e. Project management: waterfall method
    f. Quality of service: testing

B4. Large Globally known Bank

Introduction:

1. Company name: Large Global Known Bank
2. Working position: Business Architect & IT Architect
3. Duration of employment: 10 years & 25 years
4. Answers to be treated confidentially? Y
5. When did the company adopt SOA? The project started in Jan 2010 and the planned finish in Dec 2012
6. What was the reason to adopt SOA?
   a. Technical initiative
   b. Business initiative – strong business initiative as a result of merging of business profiles and pruducts
   c. Other, ……

7. What were the promised benefits? cost reduction, business efficiency, higher level of customer satisfaction and faster time to market

8. Can you provide me a short description of the organization:
   a. Business goals and services – 3 separate business products operating in the same market segments
   b. Number of departments and employees – departments over 50 countries for which these account for 115.000 employees and 1.4 million clients
   c. SOA environment – tibco software package(business works and i-process) and oracle
9. What was your role during this project? Business architect and information architect

The main success factors:

10. Has the organization conducting a migration from a legacy environment to a SOA succeeded? How did you measure this in terms of SOA promised benefits such as
    a. Reusability – business services were developed by approaching separate departments in order to identify their needs, the reusable services were developed after all needs were analyzed and documented
    b. Agility – business process agility through tool support of tibco
    c. Efficiency – higher level of efficiency through the use of reusable services and clean registries
    d. Cost reduction – not measured since the first years cost increase in terms of development will be observed, however, when the proper level of reuse is achieved and most services are developed change or integration of new services would enormously reduce the level of complexity and configuration cost.
    e. Faster time to market – reuse of services and orchestration of services for the development of new products
    f. Increased return on investment – not measured, left out for the near future
    g. Increased organizational agility - model driven software and reusable services. Ease of orchestration of newly services and compositions of components
    h. Increased maintainability – model driven software components which is easy manageable

11. What were according to you the main success factors which contributed to the migration of legacy systems to a SOA environment?
    a. From technical perspective - clear separation of service responsibilities. Stable interfaces. Also Involve product specialists, do POC, start small and work towards a small deliverable to obtain more trust and commitment of the business process owners and users.

    b. Business perspective – collaboration of business process owners, users and architects in order to document the reusable business services
    c. Organizational perspective – iterative based approach to communicate with stakeholders and users of the services. Start early in the project to communicate with managers. Commitment and good teamwork between the business process owners and the development team is important.

12. How did you analyze and assess the legacy system in order to decide which parts needed to be migrated and which parts did not? Why?
    a. Middle out approach bottom up and top down simultaneously in order to decide which services needed to be rebuilding, reused or re-factored.

13. How clear were the business objectives and business performance indicators set during the initiation of the project? And how were these measured from beginning to the end of the project? Business objectives were clear from the very start, however the proper level of granularity and reuse of services was difficult to achieve to satisfy the departments using the service.
14. To what extent was the cost and budget of the migration project an important success factor within this project? During this project it was very important, since the budget was limited and the project success strongly depends on this financial factor. This particular project was characterized as “learning by doing” since the experts involved hadn’t executed a migration of this magnitude. So a clear prediction of the cost was difficult to give. If the budget runs out, the project will halt and continue when business has the budget available
15. Did you use a migration strategy during this project?
    a. What type of strategy did you use?
        i. Bing bang approach
        ii. Incremental approach top down(process oriented) – a combination of both based on an incremental and iterated approach for analysis and design
        iii. Incremental approach bottom up(code oriented)
16. To what extent did the use of SOA implementation standards and protocols such as middleware and web services contribute to the success of the migration project? The use of tibco software resulted in a much more efficient way of design, modeling and the development of the interfaces.
17. How did you achieve to bridge the culture change within the organization, thus create organizational effectiveness and commitment during this project? By insisting early participation of the managers, business users and owners. Also cooperating and collaborating with the business owners on a frequent basis and involving them in every stage of the project so they can provide their view and opinions on the project.
18. How important was the commitment of the business and other organizational entities to the level of achieved success for this project? Commitment was very important, since the budget comes from the business. Also the level of success of SOA depends on collaboration and cooperation among departments to reach consensus concerning the reusable services and also between the developers, departments and business owners.
19. How did you monitor and control the project in terms of:
    a. SOA governance (SLA’s) – not monitored, spaghetti of services could result in higher cost in the near future since no service repository is used so there is no taxonomy of services. Allocation of the proper services could be problematic resulting in the development of new services which adds more complexity, redundancy and cost to the initial problem of not using proper governance mechanisms
    b. Business and application services – testing of services and communication with service users to obtain the level of use and ease of use.
    c. roles and responsibilities – not monitored, also no monitoring of service users and consumers
    d. Risk management - testing
    e. Project management – managing cost, time and quality by the project manager using IBM project management methodology
    f. Quality of service – testing and (de)briefing to owners and users

C. Results Closed Questions

Critical Success Factors	Average Score	Standard deviation	Highest score	Lowest score
Business Factors				
Degree of definition of objective	5.75	0.43	6	5
business processes	5.75	0.43	6	5
Business continuity	5.75	0.43	6	5
Business demand	5.75	0.43	6	5
budget and cost of migration	5.75	0.43	6	5
Service Orientation	5.75	0.83	7	5
Agility, efficiency and flexibility benefits	5.75	0.83	7	5
Reusability of assets	5.50	0.50	6	5
Financial benefits (ROI)	5.50	0.87	7	5
Competitive, market and regulatory differentials	4.75	1.25	7	4
Technical Factors	**Average Score**	**Standard deviation**	**Highest score**	**Lowest score**
Migration Strategy				
SOA implementation standards and protocols for target operational environment	6.25	0.43	7	6
Incremental process centric approach for design and development	6	0.71	7	5
Incremental approach for interface migration	6	0.71	7	5
SOA principles of design during design and implementation	6	0.71	7	5
Incremental approach for testing functional and non functional requirements	5.75	0.56	6	5
Incremental approach for application migration	5.75	0.83	7	5
Security Standards	5.25	0.43	6	5

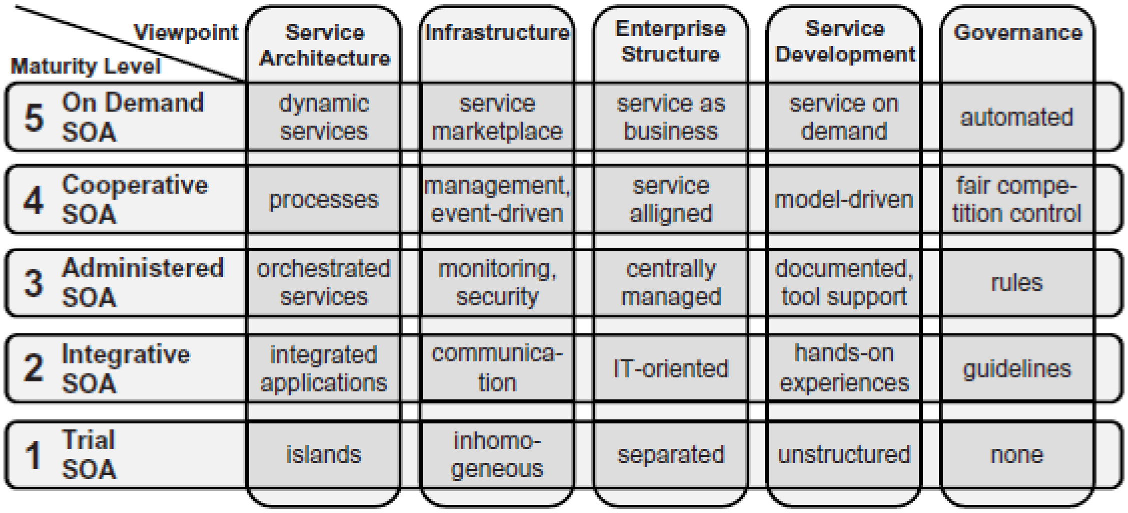
Incremental approach for database migration	5	0.71	6	4
Capabilities Legacy Environment				
Business value of the systems	6.25	0.83	7	5
Level of Dependency business process on legacy systems	6	0.71	7	5
Process centric approach	5.75	1.09	7	5
Interrelations of elements	5.25	0.83	6	4
reusable business processes and services in the domain	5	1.87	7	2
Number of different platforms, norms and standards	4.75	1.64	6	2
Analysis of the semantics	4.75	1.64	6	2
System quality	4.75	2.14	7	2
Re-documentation using reverse engineering practices	2.25	1.64	5	2
Organizational Factors	**Average Score**	**Standard deviation**	**Highest score**	**Lowest score**
Active participation and commitment of business process owners	6.75	0.43	7	6
Business client participation	6.25	0.43	7	6
Organizational change management	6	0.71	7	5
center of competency	5.75	0.43	6	5
Responsibilities and roles	5.50	0.87	7	5
Knowledge exchange	5.25	0.83	6	4
Education and training	5	0.71	6	4
Control & Management Factors	**Average Score**	**Standard deviation**	**Highest score**	**Lowest score**

Management				
Information management	5.5	0.50	6	5
Configuration management	5.25	0.43	6	5
Project management	5.25	0.83	6	4
Risk management	5	0.71	6	5
Change management	4.75	0.83	6	4
Control				
Monitor roles and responsibilities	5.25	0.83	6	4
SOA governance	4.25	0.83	5	3
Monitoring of services	4	0.71	5	3

Table 12 - Factor Analysis

D. Independent SOA Maturity Model

The Independent SOA Maturity Model is presented in the figure below. For this research an elaborate description will be given for level -1 which is Trial SOA. The levels 2 up to 5 will not be described, since it does not add more value to this research study.



Viewpoint / Maturity Level	Service Architecture	Infrastructure	Enterprise Structure	Service Development	Governance
5 On Demand SOA	dynamic services	service marketplace	service as business	service on demand	automated
4 Cooperative SOA	processes	management, event-driven	service alligned	model-driven	fair competition control
3 Administered SOA	orchestrated services	monitoring, security	centrally managed	documented, tool support	rules
2 Integrative SOA	integrated applications	communication	IT-oriented	hands-on experiences	guidelines
1 Trial SOA	islands	inhomogeneous	separated	unstructured	none

Figure 20 - SOA Maturity Model [45]

1 - Trial SOA

This level of maturity often involves independent SOA project. Within these projects enterprises find them self at the starting point and gain first experiences with services.

There is often no common technology or fixed set of standards that is used within all projects.

Service architecture is focused on removing the point to point coupling between legacy applications and wrapping them as services. Several services are designed and deployed without thinking in terms of reuse and standardization. The result of the service architecture ends up being more or less a set of unstructured and miscellaneous service island instead of a service architecture with clear taxonomies and reuse potential.

Infrastructure. Due to the fact of small independent SOA projects, incompatible standards and protocols are used which could lead to incompatibility for future reuse of services by other projects or departments.

The enterprise structure is characterized by separation into independent business departments. Every department has its own application landscape which is developed and maintained by separate IT personnel. Cooperation across business unit borders often happens.

The development of services is unstructured and done independently for each SOA project. In most cases, the purpose is to gain experience and develop best practices and develop guidelines for the implementation of an enterprise-scale SOA adoption.

Governance. Early SOA projects are mainly initiated by IT departments, which are responsible for the integration of diverse applications. SOA is therefore often regarded as a pure IT project, which only marginally affects other business units. This is usually accompanied by a lack of support of SOA projects by the top management of the enterprise[45].

References

1. Sommerville, I., *Software Engineering 6th edition*. 2000: CWS.
2. Bisbal, J., et al., *Legacy information systems: Issues and directions.* IEEE Software, 1999. **16**(5): p. 103-111.
3. Rockart, J., *Chief executives define their own information needs.* Harvard Business Review, 1979. **57**(2): p. 81-93.
4. Seligmann, P., G. Wijers, and H. Sol. *Analyzing the structure of IS methodologies, an alternative approach*. 1989.
5. Trochim, W.M.K. *Research Methods Knowledge base*. 2006 January 2011 [cited January 2011; Available from: http://www.socialresearchmethods.net/kb/.
6. Coyle, F., *Legacy Integration: Changing Perspectives.* IEEE Software, 2000. **17**(2): p. 37-41.
7. Yin, R., *Case study research: design and methods, Applied Social Research Methods Series, vol. 5.* Thousand Oaks: Sage, 1994. **1**(2): p. 3.
8. Tellis, W., *Introduction to case study.* The qualitative report, 1997. **3**(2): p. 1-11.
9. Kvale, S., *Interviews: An introduction to qualitative research interviewing*. 1996: Sage Publications, Inc.
10. Merriam, S., *Qualitative Research and Case Study Applications in Education. Revised and Expanded from" Case Study Research in Education."*. 1998: Jossey-Bass Publishers, 350 Sansome St, San Francisco, CA 94104; phone: 415-433-1740; fax: 800-605-2665; World Wide Web: www. josseybass. com ($21.95).
11. Coyle, F.P., *Legacy Integration Changing Perspectives*. 2000, IEEE Software: Southern Methodist University.
12. Jesus Bisbal, D.L., Bing Wu, Jane Grimson, *A Survey of Research into Legacy System Migration*. 1997, Citeseer: Dublin, Ireland.
13. Deidre Lawless, J.B., Bing Wu and Jane Grimson, *Legacy Information Systems: Issues and Directions*. 1999, IEEE Software: Trinity Dublin College.
14. Wu, B., et al. *Legacy system migration: A legacy data migration engine*. 1997: Citeseer.
15. Sommerville, I., *Software Engineering. 6th.* Harlow, UK: Addison-Wesley.
16. Townsend, C., *Future of COBOL*. 2003, LegacyJ corporations.
17. Haney, C.A.v.a.J.D., *Creating Web Services for Legacy COBOL*. 2009, EDSIG: Northern Arizona University.
18. Thierry Bodhuin, E.G.a.M.T., *Migrating COBOL Systems to the WEB by using the MVC Design Pattern*. 2002, IEEE Computer Society: Benevento, Italy.
19. Hoffman, P. (2006) *Socio-Technical Systems and Organizational Values*.
20. Ransom, J., I. Somerville, and I. Warren. *A method for assessing legacy systems for evolution*. 1998.
21. Microsoft, *The Business Value of Legacy Modernization*. 2007.
22. Block, J., *Service - Oriented Architecture* 2007.

23. Erl, T., *Service-oriented architecture: concepts, technology, and design*. 2005: Prentice Hall PTR Upper Saddle River, NJ, USA.

24. Channabasavaiah, K., K. Holley, and E. Tuggle, *Migrating to a service-oriented architecture.* IBM DeveloperWorks, 2003. **16**.

25. Papazoglou, M.P. and W.J. Van Den Heuvel, *Service-oriented design and development methodology.* International Journal of Web Engineering and Technology, 2006. **2**(4): p. 412-442.

26. van der Aalst, W., A. Hofstede, and M. Weske, *Business process management: A survey.* Lecture Notes in Computer Science, 2003: p. 1-12.

27. Erl, T., *Service Oriented Architecture - Concepts, Technology, and Design*. August 2005: Prentice Hall PTR.

28. Woolf, B., *Introduction to SOA governance.* IBM Developer Works, 2006.

29. Natis, Y.V., *Service-Oriented Architecture Scenario*. 16 April 2003, Gartner.

30. K.J., K.J.W.a.L., *An approach to service-oriented architecture using web service and BPM in the telecom-OSS Domain*. 2007, Emerald Group Publishing Limited.

31. Saboudh Marwan, S.J., Dock Allen, Paul Silvey, Denning Paul, *Workshop on Web services*. 2001, World Wide Web Consortium: San Jose, CA USA.

32. Ballantine, J., et al., *The 3-D model of information systems success: the search for the dependent variable continues.* Information Resources Management Journal, 1996. **9**: p. 5-15.

33. Pressman, R.S. and D. Ince, *Software engineering: a practitioner's approach*. 1982: McGraw-Hill New York, NY.

34. Cooke-Davies, T., *The “real” success factors on projects.* International journal of project management, 2002. **20**(3): p. 185-190.

35. DeLone, W.H. and E.R. McLean, *Information systems success: the quest for the dependent variable.* Information systems research, 1992. **3**(1): p. 60-95.

36. Saarinen, T., *An expanded instrument for evaluating information system success.* Information & Management, 1996. **31**(2): p. 103-118.

37. Wateridge, J., *How can IS/IT projects be measured for success?* International journal of project management, 1998. **16**(1): p. 59-63.

38. Lawler, J., et al., *Critical Success Factors in the Planning of a Service-Oriented Architecture (SOA) Strategy for Educators and Managers.* Information Systems Education Journal, 2009. **7**: p. 94.

39. Lewis, G., E. Morris, and D. Smith. *Service-oriented migration and reuse technique (smart)*: IEEE.

40. Geraldi, J.G. and G. Adlbrecht, *On faith, fact, and interaction in projects.* Project Management Journal, 2007. **38**(1): p. 32-43.

41. Bisbal, J., et al., *A Survey of Research into Legacy System Migration*. 1997, Citeseer.

42. Thomas, G. and W. Fernández, *Success in IT projects: A matter of definition?* International journal of project management, 2008. **26**(7): p. 733-742.

43. Chikofsky, E.J. and J.H. Cross II, *Reverse engineering and design recovery: A taxonomy.* IEEE Software, 1990: p. 13-17.

44. Al Belushi, W. and Y. Baghdadi. *An approach to wrap legacy applications into web services*. 2007: IEEE.

45. Rathfelder, C. and H. Groenda. *iSOAMM: an independent SOA maturity model*. 2008: Springer.

46. Li, S.H., et al., *Migrating legacy information systems to web services architecture.* Journal of Database Management, 2007. **18**(4): p. 1.

47. Oracle, *Unlocking the Mainframe: Modernizing Legacy Systems to a Service-Oriented Architecture.* 2008: p. 14.
48. Beck, K., et al., *Manifesto for agile software development.* The Agile Alliance, 2001: p. 2002–04.
49. Inaganti, S. and S. Aravamudan, *Soa maturity model.* BP-Trends, APR, 2007.
50. Green, L.W. and R.E. Glasgow, *Evaluating the relevance, generalization, and applicability of research.* Evaluation & the Health Professions, 2006. **29**(1): p. 126.
51. Gopala, I.S.a.B.k., *BPM and SOA Handshake*. 2007, BPMTrends.
52. Sharpe, B.M.a.J., *Chapter 7: Introduction to BPEL*, in *SOA for the Business Developer, Concepts, and SCA*. 2007.
53. Laury, V., *BPM the promise and the challenge*. 2004, QUEUE.
54. Stephen, W.A., *Using BPMN to Model a BPEL Process*. 2005, BPTrends.
55. Booth David, H.H., McCabe Francis, *Web Services Architecture*. 2004, W3C.
56. Paul Fremantle, S.W.a.R.K., *Enterprise Services*. 2002, ACM.
57. M., H., *Beyond Reengineering*. 1996, Harper Business: New York.
58. Janelle B. Hill, J.S., Davind Flint, Michael James Melenovsky, *Gartner's position on Business Process Management*. 2006, Gartner.
59. Leyman F., R.D., and Schmidt M., *Web services and business process management*. 2002, IBM Systems Journal.
60. Stephane, W.T.a.G., *BPM and SOA: Synergies and Challenges*. 2005, Springer - Berlin.
61. Jasmine, N., *BPM and SOA: Better Together*. 2005, IBM.
62. Newcomer, E., G. Lomow, and S.T.B. Online, *Understanding SOA with Web services*. Vol. 444. 2005: Addison-Wesley Boston;.
63. Daclin, N., D. Chen, and B. Vallespir, *Enterprise interoperability measurement-Basic concepts.* Enterprise Modeling and Ontologies for Interoperability, Luxemburg, 2006.
64. Smaczny, T., *Is an alignment between business and information technology the appropriate paradigm to manage IT in today's organisations?* Management Decision, 2001. **39**(10): p. 797-802.