Architecting the Future Enterprise: A Framework for Supporting Decision Making in the Selection of Future States

By

Matias A. Raby
B.S., Electronic Naval Engineering
Naval Polytechnic Academy, Chile

Submitted to:

the Engineering Systems Division and
the System Design and Management Program
in partial fulfillment of the requirements for the degrees of

Master of Science in Technology and Policy
and
Master of Science in Engineering and Management

at the
Massachusetts Institute of Technology
June 2012

Signature of Author ______________________________
Technology and Policy Program
System Design and Management Program
May 11, 2012

Certified By ______________________________
Prof. Deborah J. Nightingale
Professor of the Practice, Aeronautics & Astronautics and Engineering Systems Division
Thesis Supervisor

Certified By ______________________________
Prof. Ricardo Valerdi
Associate Professor of Systems and Industrial Engineering at The University of Arizona
Research Affiliate, MIT Sociotechnical Systems Research Center
Thesis Supervisor

Accepted By ______________________________
Prof. Joel Clark
Prof. of Materials Systems and Engineering Systems
Acting Director, Technology and Policy Program

Accepted By ______________________________
Patrick Hale
Senior Lecturer in Engineering Systems
Director, System Design and Management Program

This thesis is dedicated to my family –

Catalina, Arturo and Catita.

Architecting the Future Enterprise: A Framework for Supporting Decision Making in the Selection of Future States

By

Matias A. Raby

Submitted to the Engineering Systems Division and the System Design and Management Program on May 11, 2012, in partial fulfillment of the requirements for the degrees of Master of Science in Technology and Policy, and Master of Science in Engineering and Management.

Abstract

Enterprises that are successful over the long term are compelled to continuously transform in order to adapt to new contexts or economic environments. However, many of these transformation efforts fail to achieve their desired objectives. MIT Professors Nightingale and Rhodes have been developing an integrative approach that uses Enterprise Architecting as an instrument to support the planning of successful transformation. Although the approach has shown to be useful to guide transformations in various domains, feedback from previous users indicated the need for a more prescriptive and quantitative guidance in the process of moving an enterprise from the 'As-Is' to the desired 'To-Be' state.

This thesis introduces a framework that provides a structure of reasoning about the process of architecting the future state of an enterprise in the context of a transformation. The 'Architecting the Future Enterprise' (AFE) Framework is an iterative method that incorporates a systems thinking approach to design future states and a multidimensional evaluation process that compares competing architectures in terms of effectiveness, effort and risk. It enables the generation of an output in the form of an Architecture Tradeoffs Matrix, a quantitative visual representation to assess tradeoffs among competing architectures. A case study is included to illustrate a real application of the AFE Framework.

The implications of this research span across two areas. First, it aims to formalize enterprise transformation planning policies by providing practitioners a structure for reasoning that can help to minimize decision making errors. Second, by introducing quantification approaches to effectiveness, effort and risk, it improves the decision making process normally followed by enterprise leaders and architects to select the future architecture of their organizations. Finally, the framework leverages the use of simple engineering and management tools that lead to more informed decisions and to practical contributions to the practice of enterprise architecting and management decision making.

Thesis Supervisor: Dr. Deborah Nightingale
Professor of the Practice, Aeronautics & Astronautics and Engineering Systems Division

Thesis Supervisor: Dr. Ricardo Valerdi
Associate Professor at the University of Arizona
Research Affiliate, MIT Sociotechnical Systems Research Center

Acknowledgments

I am greatly indebted to a number of people for their assistance, guidance and support to complete this work, as I certainly could have never done this alone.

First, I am forever grateful to Dr. Ricardo Valerdi and Professor Deborah Nightingale for all their support during these years. Dr. Valerdi, thanks for being not only an inspiring advisor, but also an excellent mentor and friend. I will always remember that this research was inspired by a conversation we had on a taxi cab while travelling across Brazil. Professor Nightingale, I am forever in debt to yourself for making me part of the Lean Advancement Initiative. Thanks for given me the opportunity to work with extremely talented people and for all your professional advice and assistance during these years. I would also like to extend a special thank you to Dr. Donna Rhodes, for contributing her insights and expertise in the enterprise architecting practice. I greatly appreciate her support of this research.

For the Lean Advancement Initiative, I extend my gratitude to all students and staff for always making the office a great place to work while having inspirational conversations. Special thanks to Tom, JK, Bob, Joseph, John, Nicolene, Sarah and to all my LAI friends and colleges for making me feel part of a family.

I would also like to recognize the members of the TPP and SDM staff for the work that you do and for the great support you provide to students. I would be forever grateful to Mr. Pat Hale for giving me the opportunity to become part of the SDM community and making it possible for me to complete both programs on time.

I am also especially grateful to my SDM classmate and life friend Francisco Zini for being a great research partner. Our conversations not only enriched and shaped this research, but also make it a much more enjoyable task to accomplish.

Finally, to my family, I cannot thank you enough for the constant encouragement and support during these years. To my parents, for teaching me values such as perseverance and hard work, while encouraging me to follow my dreams. To my wife Catalina, there are no words to convey the sense of gratitude I feel. Thanks for your unconditional support, unending patient and for taking care of our most beautiful creations: our little kids Arturo and Catita. My love and this thesis are for you three!

This page has been intentionally left blank.

Contents

Table of Figures

List of Tables

1. Introduction

1.1. Context

"Businesses that survive and succeed over the long term must redefine or even reinvent themselves not just once, but repeatedly" (Birdsye & White, 1994). Changes in context and environment threatens survival, causing that most enterprises are compelled to continuously transform the way they perform in order to achieve their business strategic objectives (Nightingale, 2009). In most of the cases, the basic goal of these transformations effort is the same: to increase the value delivered to stakeholders and adjust to a new, more challenging market environment.

However, only a few of these enterprise transformation efforts are truly successful. Many of them are utter failures and "most of them fall somewhere in between, with a distinct toward the lower end of the scale" (Kotter, 1995). Although there is no right answer to the question of why that happens, one of the main issues is that managers tend to jump in solutions to fast, without following a series of phases that allow them to understand the complex interactions among several elements of the enterprise. A transformation process requires time, and skipping steps creates only the illusion of speed, but with the risk of producing devastating impact once implemented.

When undergoing enterprise transformation in the face of changing environments, there are many questions that enterprise leaders may ask (Glazner, 2011):

- What are the key phases that need to be followed to architect the future enterprise and minimize errors?
- Will a new business model requires changes to the enterprise's strategy, organization, processes, knowledge or information requirements?
- How can be alternative proposed architectures of the enterprise be generated and compared?
- How could trade-offs be assessed?

1.2. Research Motivation

Enterprise leaders have few tools available that can help them to answer these questions at the concept level. Although many conceptual frameworks have been developed by practitioners in the areas of enterprise transformation and architecting, most of them do not emphasize the consideration of different alternatives when deciding a future state for an enterprise. Too many decisions, as a result, are made from an overly narrow point of view, without considering that the generation of different alternatives leads normally to more optimal solutions.

At the same time, many of these conceptual frameworks are very specific to one aspect of the enterprise, but offer limited guidance for the design, alternative generation and trade-offs assessment of the future enterprise as a whole. For this reason, many major decisions concerning a future enterprise's architecture are made based on heuristics and one-dimensional evaluations.

This scenario leads to the principal research questions addressed in this thesis:

- ***How can the process of architecting the future enterprise increase its repeatability using simple tools?***
- ***How can decision makers make more informed decisions when evaluating and selecting among different enterprise architecting options?***

The 'Architecting the Future Enterprise Framework' (AFE Framework) presented in this thesis helps to address these issues. The central proposition at the core of this research is that:

"Simple engineering and management tools can be leveraged to generate a framework that will drive the process of architecting the future enterprise".

1.3. Thesis Scope and Objectives

Scope:

This research aims to complement particular aspects of the Enterprise Architecture approach that Dr. Nightingale and Dr. Rhodes had been developing since 2004 at MIT. The approach is one of the few that ensures time is spent developing and evaluating "could be states" given a set of desired criteria. In this thesis, we aim to complement Nightingale & Rhodes approach by providing a more explicit and quantitative process for generating and evaluating the future state of an enterprise.

The research aims also to leverage 'simple' tools used in other engineering and management fields. By 'simple', we refer to understandable tools that can be easily applied by managers and architects without previous training.

Objectives:

Our main objective is ***"to generate a systematic technique that will provide architects with a structure for reasoning for the process of architecting the future enterprise by using simple engineering and management tools".***

The specific objectives include:

- To determine improvement opportunities of the Enterprise Architecting approach developed by Dr. Nightingale and Dr. Rhodes at MIT.
- To define a framework to support the generation and selection of future enterprise states.
- To incorporate simple engineering and management tools to each of the steps defined in the framework.
- To test the results in a real case study.
- To discuss potential policy implications of the findings and identify areas of future research.

1.4. Thesis Roadmap

The thesis is divided into 6 chapters. Each of them attempts to meet one of the specific objectives detailed in the previous section. The aggregation of all of them attempt to respond to the main objective and research questions of the research.

- **Chapter 1** addresses the description of the context, the research motivation, the thesis scope, objectives and organization, and the research approach used to answer the research questions.

- **Chapter 2** provides background information about terminologies, the Enterprise Architecting approach developed by Nightingale & Rhodes, and the opportunities of improvement that that approach offers.

- **Chapter 3** describes the methodology used to develop the AFE Framework and provides an overview of its major phases.

- **Chapter 4** details each of the AFE framework's steps, providing prescriptive information to the reader for the generation and evaluation processes of future enterprise states.

- **Chapter 5** provides an illustrative application of the AFE framework in a real case study.

- **Chapter 6** synthesizes the policy implications, conclusions, limitations and areas of future work of the research.

1.5. Research Approach

We divided the research in three different stages: (1) identification of improvement opportunities in the Nightingale & Rhodes approach, (2) the theoretical development of a framework to architecting the future enterprise, and (3) the application and testing of the framework in a case study.

1. Identification of improvement opportunities in the Nightingale & Rhodes approach

   The first stage of the research involved a literature review in the areas of enterprise transformation and enterprise architecting (EA), informal interviews with Dr. Nightingale and Dr. Rhodes, and the analysis of feedback papers written by 56 previous users of their enterprise architecting approach. The interviews and the previous users' feedback revealed improvement opportunities that focused the research in the generation of a systematic technique to assist architects in the process of generating and evaluating the future enterprise architecture.

2. Theoretical development of a framework to architecting the future enterprise

   The second stage of the research involved several tasks:

   a. Literature review in the areas of decision making, organizational change, evaluation methods, cost estimation and risk management.
   b. Analysis of heuristics principles developed from the application of the Nightingale & Rhodes approach to several case studies.
   c. Formal interviews with industry expert Dr. Jorge Sanz, of IBM Research.
   d. Formal collection of data from 18 industry and academic experts with practical experience in the field of enterprise transformations.

   The completion of these tasks enabled us to develop a theoretical framework to guide architects in the process of generating and evaluating future enterprise architecture.

3. <u>Application and testing of the framework in a case study</u>

In the third phase of the research, a case study was conducted to test the framework with a real case scenario. The process helped us adjust certain aspects of the framework, as well as to illustrate its applicability for supporting decision making in the selection of future states.

As a complement to the case study developed in this research, the framework will also be tested in other projects by MIT graduate students taking the Enterprise Architecting course during the spring term of 2012.

2. Background Information

2.1. Common Definitions

Many of the terms used in this thesis have a broad applicability across different domains. As a consequence, experts have provided different definitions for them. In the next lines, we provide common definitions for key terms, according to how they were used in this research.

- **Enterprise:** "One or more persons or organizations that have related activities, unified operation or common control, and common business purpose" (Garner, 2009). The term enterprise can be applied to a single integrated company or to collections of inter-organizational partners. Furthermore, enterprises can also be made up of the activities of sub-parts of companies (Purchase, Parry, Valerdi, Nightingale, & Mills, 2011).

- **Architecture:** "The fundamental design of the enterprise's strategy, organization, processes, and systems" (Glazner, 2011).

- **Architect:** The person who leads the analysis, evaluation and design of the future architecture of an enterprise. He or she plays a facilitator role between the architecting team and the enterprise leaders in a transformation process.

- **Enterprise Transformation:** Refers to the end to end process of transforming an enterprise that often involves fundamental change to the architecture. Leading authors in the field described enterprise transformation as "a shift within a defined enterprise that is: (i) a response to radical changes in the economic, market or social environment; (ii) a fundamental alteration of context; and (iii) a step change in performance" (Purchase et. al., 2011).

  An Enterprise Transformation normally involves a *strategic*, a *planning* and an *execution* cycle to complete it (Nightingale, Principles of Enterprise Systems, 2009). Several tasks are normally defined within those cycles. In general terms, a transformation process goes from stimulating the need for change (at the beginning of the process), to institutionalize the change (at the end of it).

- **Enterprise Architecture:** Is an approach that can be seen as a mean to transformations. It enables more successful transformations by focusing in particular in the planning cycle. Enterprise architecture "provides strategies/ approaches to ensure time is spent developing and evaluating 'could be' states, and selecting the best alternative given a set of desired properties and criteria for the future enterprise" (Nightingale & Rhodes, 2012).

  It is important to note, that in the context of this research, Enterprise Architecture is applied following a holistic thinking to enterprise design, evaluation and selection. This approach differs from the traditional IT focus that Enterprise Architecture has had in the past and from more IT specific frameworks such as the Zachman Framework, the Open Group Framework (TOGAF) or the DoD Architecture Framework (DoDAF). Figure 2.1 illustrates the relationship of enterprise transformation and enterprise architecting.

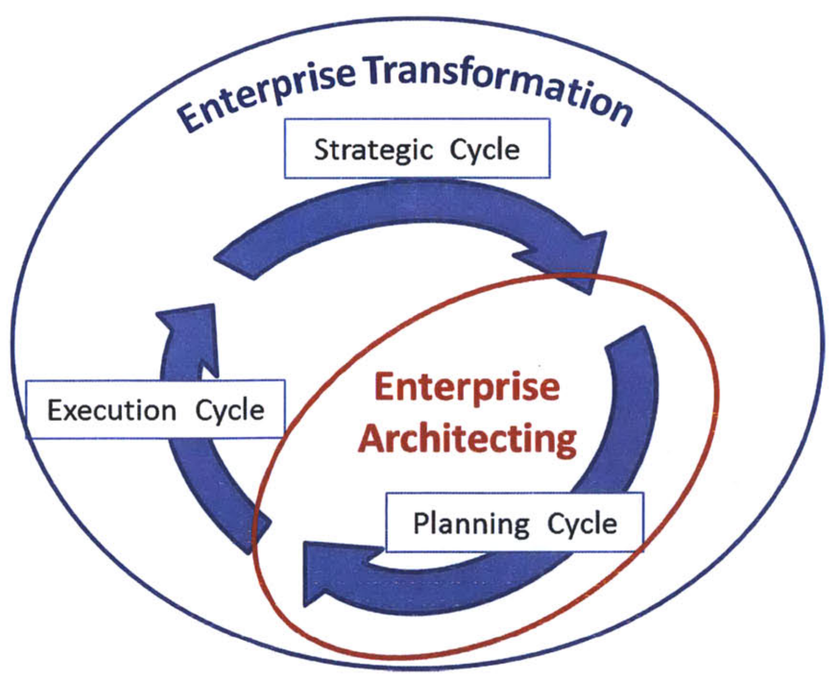


Figure 2-1: Relationship between Enterprise Transformation and Enterprise Architecting

2.2. Nightingale & Rhodes Enterprise Architecting Approach

The enterprise architecting approach developed by Dr. D. Nightingale and Dr. D. Rhodes at MIT is an integrative method that provides guidance to the planning cycle of a transformation. The approach has been evolving since 2004 and has shown to be a useful method to support transformation initiatives in various domains, including aerospace, healthcare, start-ups and high tech industries.

The approach differentiates from other transformation frameworks in that it considers the generation and evaluation of alternatives when selecting an enterprise future state. It also differentiates from other traditional enterprise architecting approaches in that it analyzes enterprises through multiple lenses (strategy, process, organization, knowledge, infrastructure, information, products and services), going beyond the traditional IT focus used by others in the field.

One of the key tools used in the approach, is a ten element framework that encourages architects to think holistically about the enterprise by analyzing it through different lenses. Figure 2.2 illustrates the major steps of the approach.

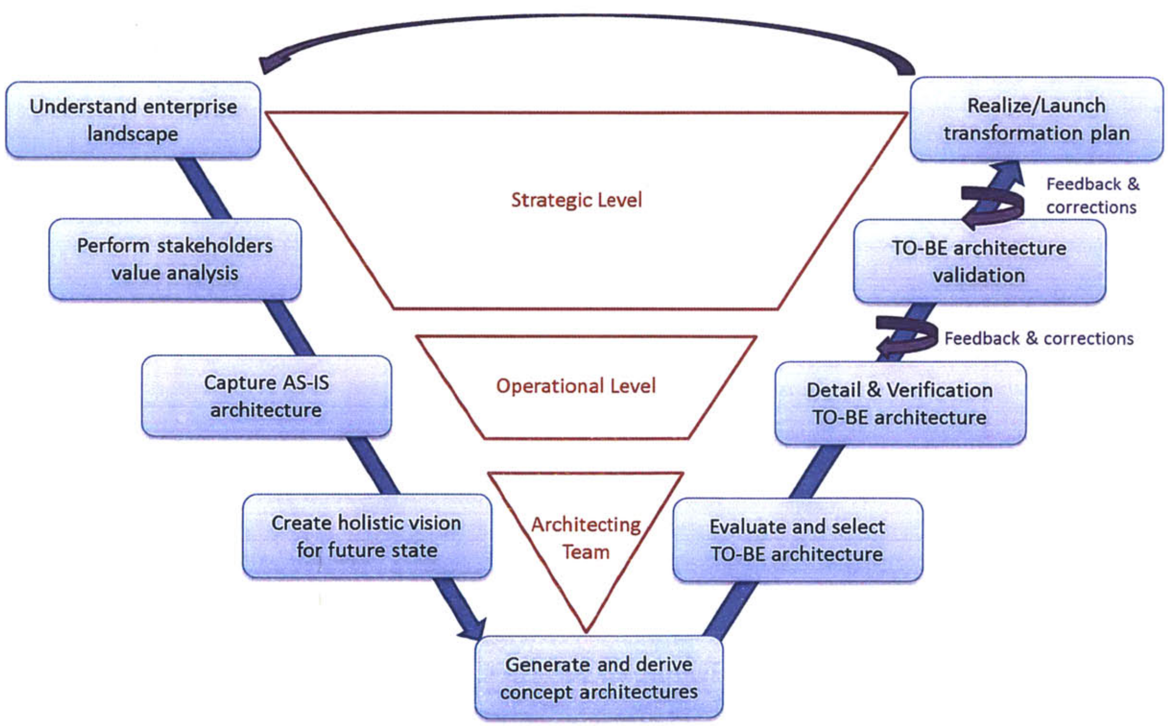


Figure 2-2: Nightingale and Rhodes EA approach

In its current level of maturity, the approach incorporates several tools and quantitative techniques for the analysis and understanding of the current state of an enterprise. On the other hand, the steps at the bottom and right side of the V are guided mainly by qualitative heuristic principles and lessons learned that have emerged from the application of the approach to previous case studies. The combination of all steps generates a value proposition that consider both 'art' and 'science' in the process of taking an enterprise from its current state to a future state.

2.3. Improvement Opportunities for the Nightingale and Rhodes Approach

As we mentioned before, the N&R approach has been evolving since 2004. Its application in several transformation projects enabled us to identify possible improvement opportunities and possibilities of enhancements to increase its usability as a guidance process.

At the beginning of this research, we conducted a field study where we analyzed the feed-back of 56 previous users of this approach who worked on 14 different enterprise architecting projects. We identified several favorable feedbacks to the approach, being the more important: its holistic approach, 10 elements framework and its method to perform a stakeholder analysis. On the other hand, we also identified some improvement opportunities. Based on the quantity of comments, table 2.1 lists the most important ones.

Improvement Opportunities	Comments	Ranking
More complete candidate evaluation methods	14	**1**
Improvement in the toolkit and quantitative methods used	12	**2**
Need of a more detailed roadmap for enterprise architecting	9	**3**
Need for more guidance in the candidate generation process	6	**4**
Consideration of new lenses (e.g. culture, external factors)	2	**5**

Table 2-1: Main improvement opportunities for the N&R approach (N=56)

The results of this field study are an indication that although heuristic and lessons learned are valuable starting points for generating and evaluating architectures, they have limitations in being prescriptive because they don't tell architects what specific steps should be taken to accomplish the end result.

Two of the issues identified in table 2.1 were addressed by Dr. Nightingale and Dr. Rhodes during the fall of 2011. The development of an Enterprise Architecting roadmap, as well as the development of the 10 elements framework tackled the issues ranked #3 and #5 in table 2.1.

As a consequence, we decided to focus this research and the three remaining opportunities of improvement:

1. The need for more guidance in the candidate generation process
2. The need for more complete candidate evaluation methods
3. The need for improvements in the toolkit and quantitative methods used

3. Development of the AFE Framework

In order to address the improvement opportunities found in the previous section, we developed a framework to guide the architecting, evaluation and selection processes for the future enterprise. We named it "Architecting the Future Enterprise Framework" (AFE Framework). It is a spiral model that was shaped by two major sources:

1. Heuristics principles developed by the application of the N&R approach in different case studies.
2. Decision making theory, in particular the PrOACT approach, a proven method developed by experts from the Harvard Business School.

3.1. Heuristics Principles from the N&R Approach

One of the starting points in knowledge generation is the consideration of heuristics. As described by Maier and Rechtin, heuristics "are guidelines, abstractions, and pragmatics generated by lessons learned from experience" (Maier & Rechtin, 2009). During the years, users of the N&R approach have created a spectrum of over 1000 heuristics that provide insights on different steps of this enterprise architecting approach.

The development of the AFE Framework was shaped by a number of these heuristics, in particular those related to generating, evaluating or selecting future state architectures. We selected which heuristics to use based on four criteria (Varledi, 2011): First, there had to be a relationship between the heuristic and the design of the model for generation and evaluation we are creating. Second, there had to be agreement among experts that the heuristic was useful and correct. Third, the applicability of the heuristic had to be apparent over time. Fourth, the heuristic had to be resilient across different scenarios, beyond the one under which was created.

Some of these heuristics are listed in table 3.1. Information about its general context and the relationship with the model is also provided. Some heuristics have been reworded from their original form in order to make them applicable for different cases.

Table 3-1: Influence of heuristic principles to the AFE Framework

Enterprise Architecting Heuristic (Nightingale & Rhodes, 2011)	Influence to AFE Framework
"Don't jump ahead in a transformation without a holistic understanding of the enterprise"	Generate architectures based on inputs from enterprise's current state and context.
"No ideas should be excluded in the early concept generation"	First create alternatives, then evaluate them.
"Expand the boundaries to foster ingenuity and creativity to widen the range of possibilities"	Incentive fresh thinking and generation of alternatives from different perspectives.
"The same evaluation criteria might have a different answer when looked through different point of views".	Evaluate alternatives considering different stakeholders.
People don't like change. And you must change wisely, or else your EA initiative will succumb to "flavor of the day" syndrome.	When evaluating, consider also the implementation effort associated with each alternative.
"No organizational change is independent - there are always unforeseen implications"	Look for potential side effects of each alternative.
"Enterprise Architecture evaluation is about balancing tensions; an architecture cannot be perfect on all measurable performance dimensions"	Evaluation should consider different aspects and tradeoffs among them.

3.2. Proven Decision Making Approach

In addition to consideration of heuristics principles, we researched also for insights from proven decision making models to avoid being constrained only to the enterprise architecting field. We were especially interested in ensuring we were covering all the critical aspects about how good decisions should be made.As a consequence, the development of the AFE Framework was also influenced by PrOACT, a proven decision making approach developed by decision making experts from HBS (Hammond, Keeney, & Raiffa, 1999). As in our case, their approach does not tell users what to decide, but rather show them how to frame the process of thinking. In their research, they found that even the most complex decision can be analyzed considering a set of five core elements and three additional elements. These eight elements and their influence to the AFE Framework are listed in table 3.2.

Table 3-2: Influence of decision making theory to the AFE Framework

Eight Elements of PrOACT	Influence to AFE Framework
"Work on the right decision problem"	Consider the input of the 'motivation for change'.
"Specify your objectives"	Consider inputs from 'future vision' and 'stakeholder value analysis'
"Create imaginative alternatives"	Incentive fresh thinking and generation of alternatives from different perspectives.
"Understand the consequences".	Evaluate alternatives both in terms of effectiveness and effort.
"Grapple with your tradeoffs"	Evaluation should consider different aspects and tradeoffs among them.
"Think hard about your risk tolerance"	Consider evaluation of risk of each alternative, as well as the risk tolerance of the enterprise leaders.
"Clarify your uncertainties"	Before consideration of risk, identify both internal and external uncertainties.
"Consider linked decisions"	In the selection process, think in future consequences of that decision.

3.3. Introduction to the AFE Framework

Bringing together the influence of the heuristic principles, as well as the inputs from decision making theory, we developed a spiral framework that considers six major steps, as shown in figure 3.1.

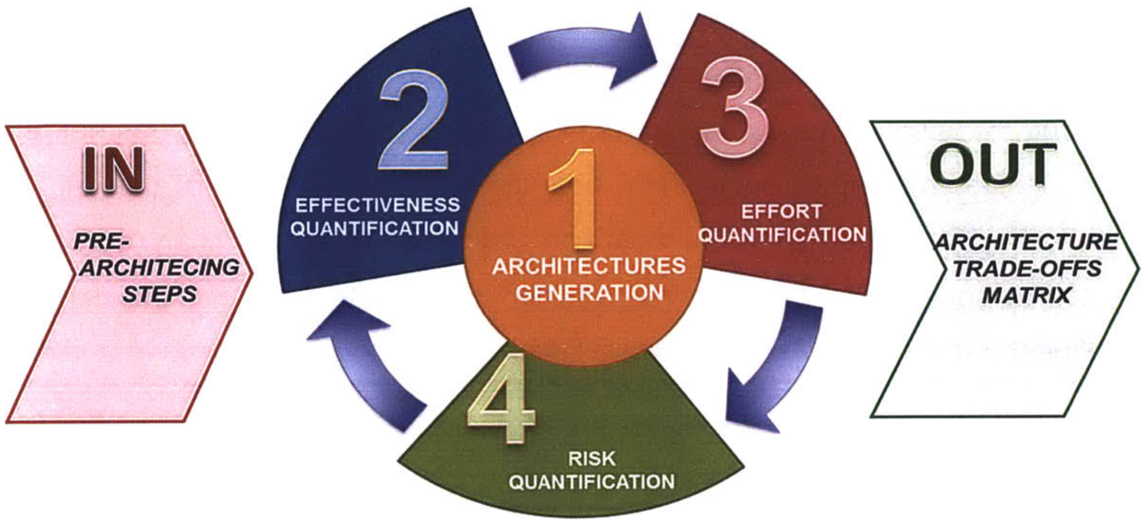


Figure 3-1: AFE Framework

- **Pre-architecting Steps:** Clarify the problem, the current state of the enterprise and define the specific objectives for the future. It includes the understanding of: the motivation for change, the enterprise landscape, the stakeholder's values, the 'As-Is' enterprise and de future holistic vision.

- **Step 1, Architectures Generation:** Provides guidance to develop several candidate architectures using a systems thinking perspective. This step focuses only in the generation of alternatives, not in their evaluation.

- **Step 2, Effectiveness Quantification:** Estimates how close each proposed architecture is with respect what is desired to achieve or obtain by the future enterprise. In its evaluation considers both: future strategic competencies needed and the future values of multiple stakeholders.

- **Step 3, Effort Quantification:** Compares the level of effort required to implement any of the proposed candidate architectures. It allows the consideration of trade-offs that exist among different options.

- **Step 4, Risk Quantification:** Assess the level of risk associated with each of the candidate architectures. It identifies uncertainties, likelihoods and the consequences that unexpected/ unforeseen events might have on different architectures. It provides important complementary information to decision makers.

- **Output, Architecture Trade-offs Matrix:** Allows an easy visualization of the strengths and weaknesses of the different alternatives. It helps decision makers and architects reason about architectural decision by showing informed trade-offs caused by the interaction of multiple elements.

We considered these steps are sufficient because they incorporate all the inputs discussed in the previous section. The order is also relevant, because each step provides inputs for the next one. The model also acknowledges the importance of having an iterative process, where the evaluation of architectures using different dimensions lead inherently to improvements on the initial designs and to a better final solution.

4. Steps of the AFE Framework

This chapter describes the major steps of the 'Architecting the Future Enterprise Framework' (AFE Framework), a systematic approach to support enterprise leaders and architects when deciding a future architecture along a transformation process. As introduced in the previous chapter, the framework follows a spiral model of design that leads to more refined architectures and to the understanding of the tradeoffs that exists among them. Through the consideration of multiple evaluation criteria, its final goal is to assist decision makers in making more informed decisions when evaluating and selecting a future architecture.

One important aspect of the framework is that it not only guides the process of generating and evaluating candidate architectures, but also is being developed as a complement to critical aspects of the Enterprise Architecture approach developed by Nightingale & Rhodes at MIT (Nightingale & Rhodes, 2004).

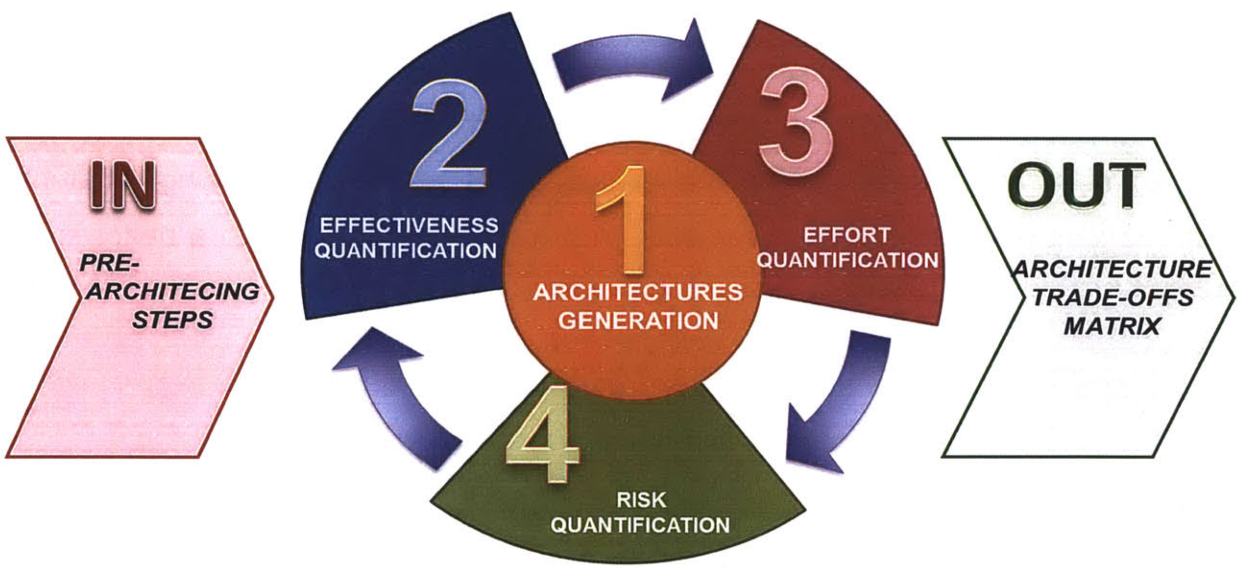


Figure 4-1: AFE Framework Overview

4.1. Inputs: Pre-Architecting Steps

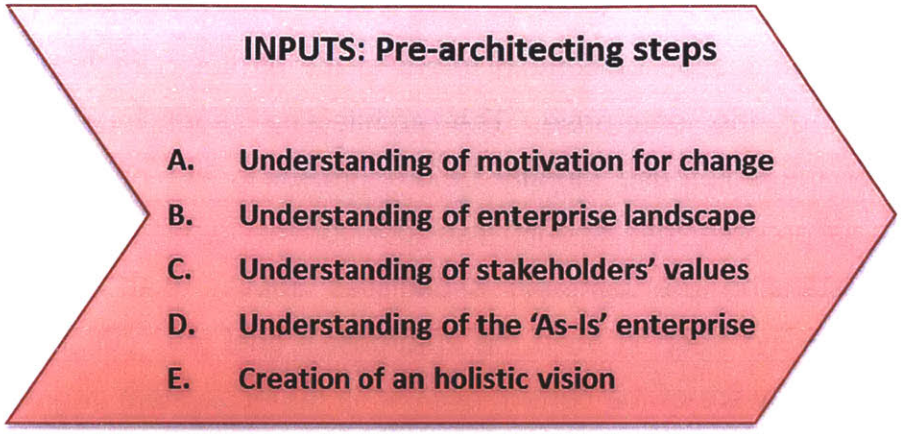


Figure 4-2: Overview of the Pre-architecting steps (Inputs)

As indicated before, this framework is complementary to the EA methodology and requires the completeness of certain pre-architecting steps before beginning the steps of architecture generation. The major pre-architecting steps that are considered inputs to the framework are:

A. Understanding of motivation for change

First, it is critical to understand the drivers that trigger the transformation and the motivation for doing enterprise architecting. This involves also understanding the boundaries of the enterprise (what parts of the organization are involved) and the scope of transformation in terms of scope and timeframe. It also considers an understanding of what is possible to change and what will have to remain fixed.

B. Understanding of enterprise landscape

This involves an understanding of the ecosystem where the enterprise is embedded as well as its internal landscape.

The ecosystem is characterized by the external regulatory, political, economic, industry, market and societal environment in which the enterprise operates and competes (Nightingale & Rhodes, 2012). The use of strategy frameworks such as

Porter's five forces analysis (Porter M. E., 1980) or other similar ones are useful tools to find opportunities and threats to the future enterprise.

The internal landscape or context refers to the strategic imperatives, ideology and core values of the enterprise. The architecting team must have a clear understanding of the enterprise identity before projecting the organization into the future. During this process they should also look for the dynamic capabilities of the company. Understanding the dynamic capabilities implies that management and architects should always keep in mind the enterprise's past (i.e.: culture, core values, weaknesses) and be aware of the current position in the ecosystem in order to be able to project and design's its future state.

C. Understanding of stakeholder values

Stakeholders must be at the center of any successful enterprise transformation (Nightingale & Srinivastan, Beyond the Lean Revolution: achieving succesful and sustainable enterprise transformation, 2011). Before moving into architecting, the key stakeholders have to be identified and prioritized in terms of importance to the enterprise. Performing a value exchange analysis between them and the enterprise is important before moving into architecting, because provides a clear perspective of what they expect from the enterprise, as well as an assessment of the current state of the enterprise in delivering those values. Several tools can be applied in this process, such as the stakeholder salience analysis (Mitchell, Agle, & Wood, 1997) and value elicitation templates (Nightingale & Srinivastan, Beyond the Lean Revolution: achieving succesful and sustainable enterprise transformation, 2011).

D. Understanding of the "As-Is" Enterprise

Another critical pre-step before architecting is to capture the 'As-Is' enterprise to understand the current state and the potential 'gaps' that the enterprise has. Finding a way to solve those issues is an important guidance when thinking in the future architecture.

This process has to be performed from a holistic point of view to allow architects to understand the whole before working on the specifics. It is helpful to perform the analysis through the view of the 10 elements proposed by Nightingale and Rhodes: ecosystem, stakeholders, strategy, organization, processes, knowledge, information, infrastructure, products, and services (Nightingale & Rhodes, 2012). Other tools that can be used are employee surveys and artifacts such as annual reports, media reports or websites.

E. Creation of an Holistic Vision

Finally, a critical step before architecting is creating an holistic vision for the enterprise under consideration. The vision of an enterprise has two major components: core ideology and an envisioned future (Collins & Porras, 1996).

The core ideology "defines the enduring character of an organization – a consistent identity that transcends product or market cycles, technological breakthroughs, management fads and individual leaders" (Collins & Porras, 1996). The core ideology is found inside the organization and cannot be imposed by its leaders.

The second component of the vision is the envisioned future. This is an exercise where the leaders of an enterprise are invited to 'dream' about the future of the organization, defining tangible, energizing and highly focused goals for the next 5 to 10 years. The envisioned future implies a 'creation' of the future and is generated by from the passion and emotions of the leaders. The envisioned future of the enterprise is the equivalent of the 'north star' for the process of designing the future architecture.

With all the inputs collected through the pre-architecting steps, the architecting team can move forward and begin developing candidate architectures for the future state of the enterprise.

4.2. Step 1: Generating Candidate Architectures

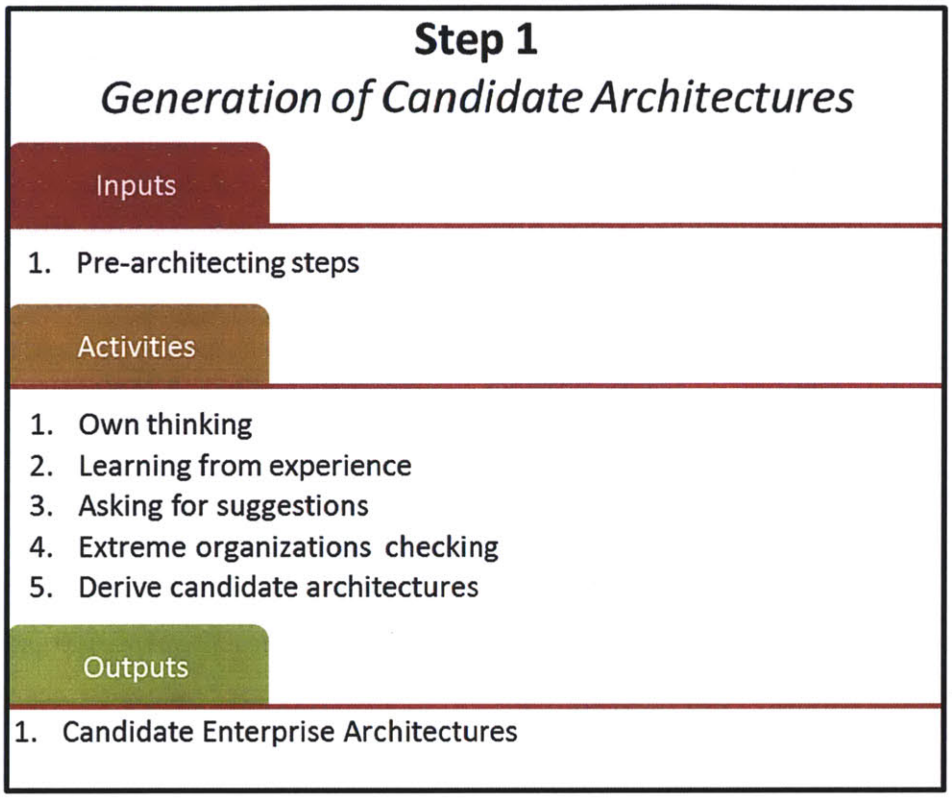


Figure 4-3: Overview of Step 1

The first step of the framework introduces a prescriptive approach to guide managers and architects in developing different candidate architectures for the future state of the enterprise. This step is one of the hardest parts of any transformation process and requires a mixture of science and art. The activities listed in this section do not intend to provide straight answers to the architects, but to offer them with a way for finding them.

Although there is no formal procedure to generate architectural concepts, in the following paragraphs we introduce a set of prescriptive and iterative activities that can help managers and architects in the development of different alternatives.

This process builds over the work done by Francisco Zini in his SDM Master's thesis (Zini, 2012). Figure 5.4.summarizes the proposed major activities for generating candidate alternatives:

Figure 4-4: Major activities for candidate alternatives generation

4.2.1. Activity 1: Team thinking

It is important that that the architecting team begin the iterative process of generation based on their own ideas about how to design the future enterprise. This will allow them to avoid by biased toward solutions or mental models defined by others. Original ideas might be suppressed if they are exposed to experts or consultants before they are fully developed. Team brainstorming sessions and the use of practical tools can be helpful in this process.

The architecting process involves decision about configuration on different areas and levels of abstraction. Similar outputs can be achieved by different configurations; there is not only 'one best way' or 'silver bullet' that will generate the best possible architecture. We suggest starting the brainstorming with these two general questions (Nadler & Tushman, 1997):

- *What kind of structure best enables the enterprise to manage its work in order to meet its strategic objectives?*
- *How will this structure affect other components such as cultural or behavioral patterns of stakeholders?*

To answer those questions and generate multiple alternatives, it is useful to approach the problem from a systems thinking perspective. The 10 elements framework can also help in this process. The following questions/ issues consider several architecting variables that can help as guidance for the generation of architectures:

From a strategic point of view, some of the questions the members of the team should discuss are:

- ***What is the appropriate mean for the enterprise to develop the required expected capabilities? Internal development, external sourcing or both?*** "Internal capabilities refer to creating a new capability within the existence boundaries of the firm by recombining the firm's existing capabilities or creating new ones. Examples include internal training, internal product development, opening new R&D labs and hiring new staff members. External sources means trading in a strategic capability that stems from external sources. This can occur by three means: purchase contracts, alliances and acquisitions" (Capron & Mitchell, 2009). In their study, Capron and Mitchell argue that enterprises that have small capability gaps or already possess a strong position in a particular capability with respect competitors will be more effective through internal development.

- ***What is the right amount of change needed in the enterprise? A reorder of some of the existing competences or a revitalization of what is in its 'core' competences?*** A reorder process the enterprise alters its operations by changing the level of importance to certain competences, setting new priorities and restructuring divisions. In a revitalization, outdated routines are substituted by new competences and processes (Baden-Fuller & Volberda, 1997) This process has a higher impact in the outputs, but is more difficult and risky to implement.

There are also variables related with an organizational point of view. An important part of enterprise design involves decisions about the formal organizational arrangements, including formal structures, processes and systems that make up an organization. Some of the questions the members of the team have to ask themselves are:

- ***Should the boundaries of the enterprise be moved? Should they be based on efficiency, autonomy, competence or to foster enterprise identity?*** Enterprise boundaries are the demarcation between the organization and its environment. There are horizontal boundaries (defined by the scope of product/ markets addressed) and vertical boundaries (defined by the scope of activities in the industry value chain). Depending on the organizational issue that architects want to highlight, there are different answers to this question. Santos and Eisenhardt (2005) argue that there are four conceptions of boundaries depending on what is the focus of the enterprise. For enterprises that are focusing on cost reduction, boundaries should be set at a point that minimizes the cost of governing activities. Organizational boundaries are thus managed an atomistic way through the accumulation of independent make or buy decisions. This is particularly useful for stable and competitive industries where efficiency is crucial.

Enterprises that focus on control and in increasing their organizational influence should set boundaries at the point that maximizes strategic control over crucial external forces (Porter M. , 1985). In terms of vertical boundaries, enterprises expand their participation in the industry value chain by internalizing sources of uncertainty. In terms of horizontal boundaries, enterprises expand their scope of product/ market domains to buffer their core position. Here the central boundary issue is control, not efficiency. (Santos & Eisenhardt, 2005). On the other hand, enterprises determined in exploiting their competences should set the boundaries in a flexible way that allows them to take advantage of the opportunities. They normally adopt a dynamic perspective of boundaries that emphasizes evolution. Finally, enterprises that want to highlight their identity should set their boundaries in a way that emphasize social context for sensemaking. A good example here is Starbucks that has a structure that generate commitment and emotional attachment to the firm.

- ***How should the enterprise strategic grouping for its people be? By activity, by output, by customer segment/ user or mixed?***

There are three basic forms of grouping which can be combined and modified to produce creative variations. Activity grouping gathers people who share similar functions, skills or work processes. The main feature of those groups is that goals, position of influence, rewards and control systems are normally based on performance of specific tasks.

Output grouping gathers people on the basis of the service or products they provide. Groups are composed by staff of different skills, tasks and processes. The objective is that product rewards, promotion and controls are based on the integrity of the final product or service.

Finally, market segment/ user grouping is composed by people who performs different kinds of work and produce different outputs, but serve the same customer, market or market share. The rewards and controls are dominated by user assessment of value and the goals of the group are driven by user needs.

Some questions or issues that architect should consider from the process or knowledge point of view are:

- ***What are the sources of waste in the organization? (E.g. overproduction, unnecessary movements, waiting times, quality issues, over processing, unused capability, inventory issues, etc?)***

- ***Can we improve quality and reduce variations through employee satisfaction? (E.g. shared goals, shared knowledge, mutual respect, frequent & timely communication?)*** (Gittell, 2009)

4.2.2. Activity 2: Learn from experience

Once internal thoughts have been gathered, it is advisable to look at what other successful enterprises have done in similar scenarios. This implies looking not necessarily to competitors, but to the experience that leading enterprises from different industries have had in the past. Applying Von Hippel's concepts of lead users to enterprises, architects should ask the following questions (Von Hippel, 2005):

- **Who are the leading enterprises that have managed successfully the problem we are trying to solve?**
- **Which enterprises are the leaders in the competences we are trying to develop?**
- **How do those enterprises manage their business?**

Associating ideas and making connections from other enterprises and industries would bring new perspectives and help in the development of new ideas to generate candidate architectures. To have a better grasp of what this implies, a McKinsey report suggests asking the following questions (Capozzi, Dye, & Howe, 2011):

- How would Google manage our data?
- How might Disney engage with our customers?
- How would Southwest Airlines cut our costs?
- How would Zara redesign our supply chain?

4.2.3. Activity 3: Ask for suggestions

This activity highlights the importance of asking third party opinions. People at a distance from a problem may see it more clearly, without the conceptual or emotional block the architect team may have (Hammond, Keeney, & Raiffa, 1999). Even more important, these conversations and explanations to others may foster additional creative ideas for the generation of candidate architectures.

Third party opinions could come from formal sources such as consultants, experts or Professors, or from more informal sources such as the team's professional network.

4.2.4. Activity 4: Extreme organizations

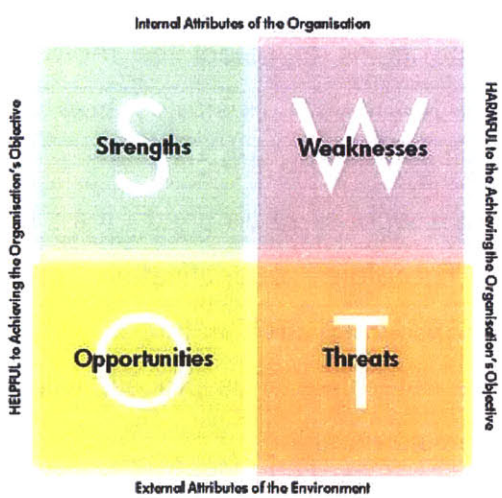
Another way to foster creativity is to consider options that would imply an extreme situation for the enterprise. Analyzing extreme options help in the understanding of the tradeoffs that affect the enterprise. One way to do this exercise is imposing artificial constrains to the current business model. This type of thinking foster creative thinking as it forces the architect to get out of their comfort zone. Some example of auto imposed constrains are (Capozzi, Dye, & Howe, 2011):

- Your largest channel disappears overnight
- You have to move from business to customer (B2C) to business to business (B2B) or vice-versa.
- You have to offer your value proposition with a partner company
- You can serve only one consumer segment

4.2.5. Activity 5: Derive candidate architectures

Finally, architects should derive candidate architectures from the multiple concepts and ideas generated during the previous activities. A combination of elements of the early concepts can help in generating more developed candidate architectures.

At the same time, the use of a SWOT (strength, weakness, opportunity, threat) analysis can be used as a useful tool to condense alternatives to a recommended 3 to 5 suggested list of candidate architectures. For this analysis, *Strength* refers to characteristics that give the enterprise an advantage over others and *Weakness* for those that place the enterprise at a disadvantage to others. *Opportunity* refers to an external chance to succeed in the enterprise's broader environment (or context) and Threat to those than can cause trouble for the enterprise.



Source: http://www.andyeklund.com
Figure 4-5: SWOT Analysis

4.3. Step 2: Quantification of Effectiveness

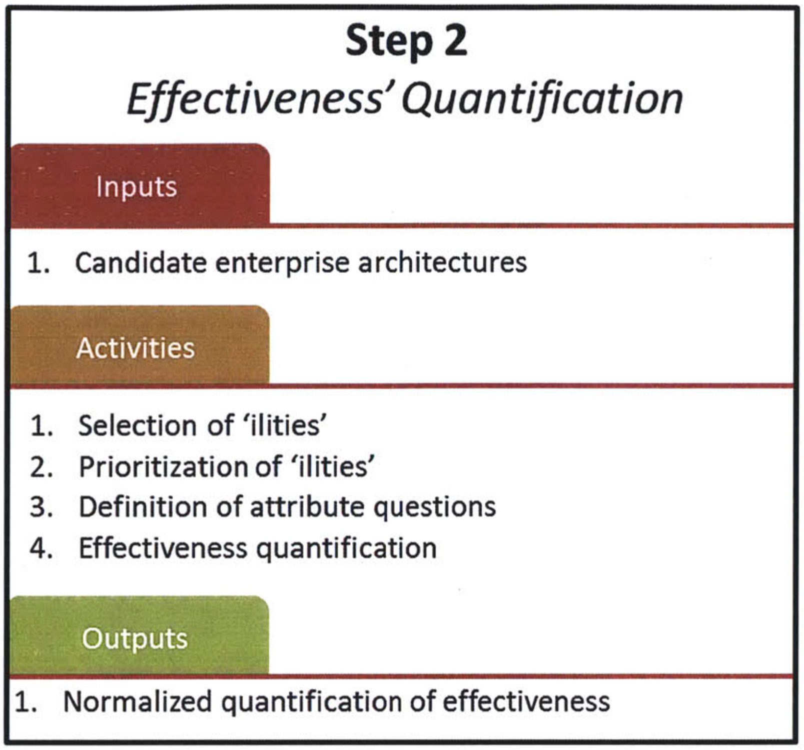


Figure 4-6: Overview of Step 2

Once several architecture alternatives have been developed, the next step is to evaluate them in different dimensions to select the more suitable to the envisioned enterprise. This step in the framework illustrates how to compare the candidate architectures in terms of effectiveness. An effective enterprise architecture provides a value proposition that is aligned with its envisioned future state and meets the future needs of its stakeholders. Therefore, quantification of effectiveness provides a estimation on how close each proposed architecture is with respect what is desired to achieve or obtain by the future enterprise. This process will provide architects with the first set of criteria to support decision making.

To measure the effectiveness of an architecture is not an easy task because enterprises are complex socio-technical systems that include several structural and behavioral interconnections (Nightingale & Rhodes, 2004). This level of interaction makes the use of specific metrics inappropriate at the enterprise level. It requires the selection of metrics situated within a holistic view of the enterprise that, at the same time, are easily mapped to their stakeholders' value proposition and future vision. A proved alternative is to apply properties and design issues related with complex systems in the context of enterprises. The current practice of enterprise architecting at MIT, has been using high level system quality attributes or 'ilities' to measure qualitative enterprise desired characteristics.

Table 4.1 illustrates a list of the 20 most used 'ilities' for evaluation of architectures in transformation projects done in recent years in the context of the MIT Enterprise Architecting course.

Table 4-1: More used 'ilities' as metrics for evaluations

Accountability	*Adaptability*	*Agility*
Compatibility	*Competitiveness*	*Efficiency*
Flexibility	*Learnability*	*Manageability*
Modifiability	*Quality*	*Reliability*
Replicability	*Resilience*	*Responsiveness*
Robustness	*Scalability*	*Sustainability*
Simplicity	*Timeliness*	

However, we noticed that in most of this projects selection of these 'ilities' was done based on the result of the "As-Is" analysis and on the current stakeholders' values. The problem with that approach is that architects tend to over-emphasize past strategic factors that might generate sub-optimal architectures in future environments. This effect is known as 'recency effect', where people tend to pay too much attention to recent experiences.

To avoid this effect we suggest to follow a method that leverage the use of 'ilities' as enterprise metrics, but that looks into the future to select and weight them. Inspired in previous work on the field, as well as in the software and systems architecture fields, we consider the following activities to quantify effectiveness of a set of candidate architectures:

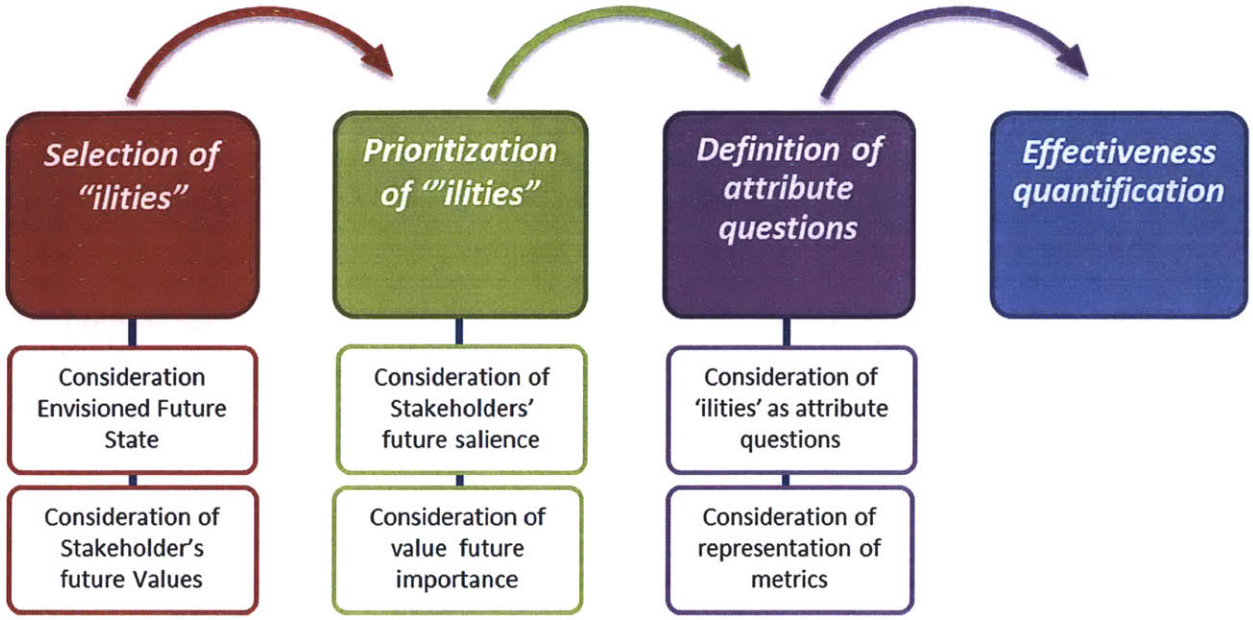


Figure 4-7: Steps to quantify effectiveness of architectures

4.3.1. Activity 1: Selection of 'ilities'

As shown in the figure below, the selection of "ilities" is based on a traceable process that begins in the envisioned future state and on the expected values of the stakeholders of the future enterprise.

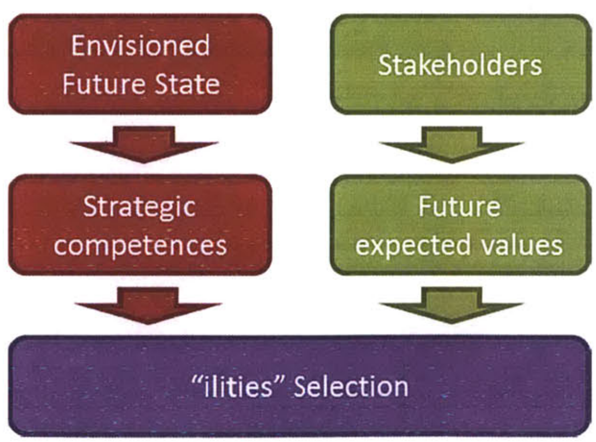


Figure 4-8: "Ilities" selection overview

We begin with the proposition that in order to be effective, a candidate architecture must be aligned with the envisioned future state defined by top managers for the enterprise – this is way the transformation is taken place!

As defined by Nightingale and Srinivasan (2011), the envisioned future state represents "an image of what the enterprise would look like and be like in its future state". It considers aspects such as gaps/opportunities for improving current enterprise performance as well as strategic competencies that may have to be developed by the enterprise to improve performance in future environments. It is always challenging for top managers and architects to identify and define which strategic competencies to develop, because they require time, resources and also because decisions are normally made in a setting characterized by uncertainty, complexity and intra-organizational conflicts ***(Amit & Schoemaker, 1993)***.

Selecting the 'ilities' based on the envisioned future state requires moving in a sequence from: envisioned future state – to future strategic competencies – to 'ilities'. Figure 4.9, illustrates an example of mapping future strategic competencies into 'ilities'.

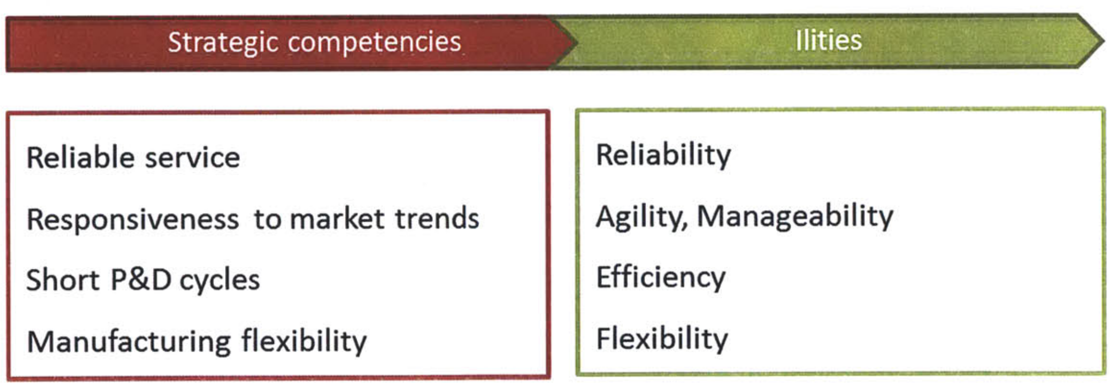


Figure 4-9: Example of mapping 'strategic competencies' into 'ilities'

In addition to be aligned with the envisioned future state, an effective architecture should deliver value to the stakeholders of the future enterprise. This involves moving forward from the value analysis performed in the As-Is analysis because requires the consideration of two questions:

- Will the enterprise interact with the same group of stakeholders defined in the stakeholder analysis?
- Are there any changes in the values these stakeholders will expect from the future enterprise?

As in the previous case, once these questions are examined, the next step is to map the expected future values in terms of 'ilities'. The selection of 'ilities' requires moving in a sequence from: stakeholders – to future values – to 'ilities'. Figure 4.10 illustrates an example of mapping future strategic competencies into 'ilities'.

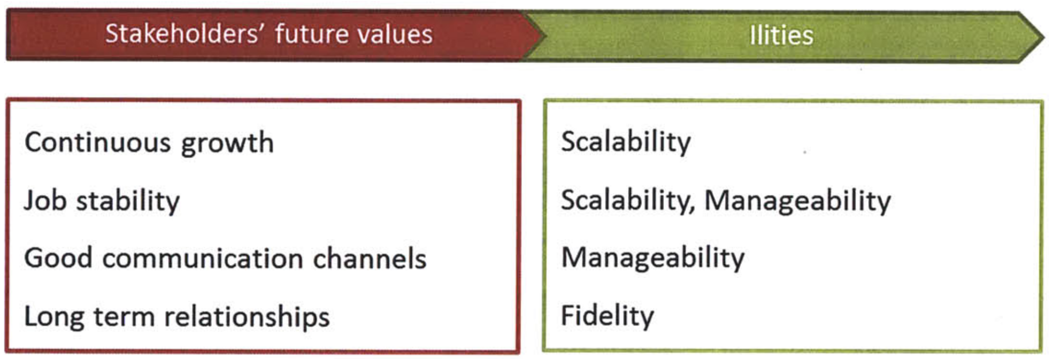


Figure 4-10: Example of mapping stakeholders' future values into 'ilities'

The 'ilities' obtained from the stakeholders' future value analysis, complement the ones previously selected and provide architects with a set of high level enterprise metrics for effectiveness quantification.

4.3.2. Activity 2: Prioritization of "ilities"

Once the set of "ilities" is selected, the next step is to assign weights to them in terms of its impact to what is defined as effective for the future enterprise. One way to do it is considering the importance that different stakeholders assign to the strategic competences or values associated with them. For example, top managers and shareholders will certainly assign more importance to those closely related to the envisioned future state of the enterprise. On the other hand, other stakeholders might have different prioritization according to their own interests and expectations. Some factors to consider for the specification of importance are: how much utility the fulfillment of the value will bring, the adverse that will occur if the value is unmet, the degree of which the value is currently fulfilled and how quickly the value needs to be fulfilled (Crawley, 2011).

In the process of weighting "ilities" is also important to consider the importance that each stakeholder has to the future enterprise. One of the tools that exist to measure tradeoffs among various stakeholders is through the notion of stakeholder *salience*. Under this perspective, the importance of the stakeholders to the enterprise is defined based on their: '*Power*' – to influence the enterprise, '*Legitimacy*' – of the relationship or stake, and '*Critically*' – for the enterprise's strategy and operation (Mitchell, Agle, & Wood, 1997). Therefore, based on the number and intensity these attributes, stakeholders can be classified in three categories (Nightingale, Stanke, & Bryan, 2008):

- Definitive Stakeholders: Their values must be met by an effective architecture.
- Expectant Stakeholders: Their values should be met by an architecture.
- Latent Stakeholder: Their values could potentially be met.

Taking into account the two considerations explained above, we implemented a combined quantification table for "ilities" prioritization that considers both the stakeholder salience as well as an "ility" importance scale for each of them.

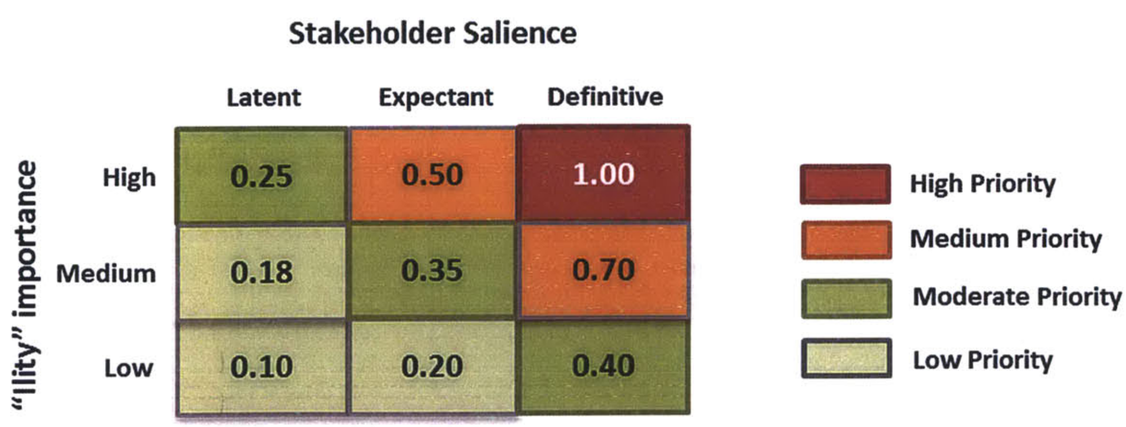


Figure 4-11: Combined quantification table for "ilities" prioritization

The two scales that are shown in the figure 4.11 are:

- *Stakeholder Salience*: The three salience categories have a fundamentally non-linear sense to them. After consulting other test cases and experiences in the field of system architecture, we decided to set the ratio at 2, as can be seen in figure 4.11. This implies for example, that at equivalent importance ratings, a 'Definitive Stakeholder' is twice more influent than an 'Expectant' one.

- *"Ility" Importance*: We defined three categories of "ility" importance (High, Medium, Low) with a linear relation among them when having the same salience category.

The use of this combined quantification table allows the prioritization of the selected "ilities" using numerical metrics. If each selected "ility" is mapped to a stakeholder using a matrix, a numerical input is assigned to it based on the weights shown in figure 4.11.

At the end of the process, a prioritization of "ilities" can be obtained based on their final cumulative scores. Table 4.2 illustrates a simplified example of the method.

Table 4-2: Example of "ilities" prioritization

Stakeholder	Salience	Reliability		Agility		Manageability		Efficiency		Flexibility		Scalability		Fidelity	
Top Management	Definite	High	1.00	High	1.00	Medium	0.70	High	1.00	Medium	0.70	Medium	0.70	Low	0.40
Shareholders	Definite	Medium	0.70	High	1.00	Medium	0.70	High	1.00	Medium	0.70	High	1.00		-
Customers	Espectant	High	0.50	High	0.50	-	-	Medium	0.35	-	-	-	-		-
Project Managers	Expectant	Medium	0.35	Medium	0.35	High	0.50	High	0.50	Medium	0.35	High	0.50	Medium	0.35
Employees	Expectant	Low	0.20	Low	0.20	Medium	0.35	Medium	0.35	High	0.50	High	0.50	Medium	0.35
Partners	Latent	-	-	-	-	High	0.25	Medium	0.18	High	0.25	-	-	High	0.25
Total per "Ility"			2.75		3.05		2.50		3.38		2.50		2.70		1.35
Average Weight			15%		17%		14%		19%		14%		15%		7%

4.3.3. Definition of attribute questions

Once the "ilities" have been selected and prioritized, the next step is to decompose them into more concrete and measurable metrics. The goal here is to avoid misinterpretations and assists the process of evaluation.

A. Description of ' ilities' into attribute specific questions

We propose the use of attribute specific questions as concrete metrics to quantify effectiveness of the proposed architectures. This allows architects to translate the original "ilities" into the terminology and the specific taxonomies that they will be applied for evaluation.

At the same time, the relative importance of each quality attribute question within an 'ility' should be defined to provide a more specific evaluation approach. The following table provides an example of the process:

'Ility'	Relative Weight	Attribute Question	Relative Sub-weight
Reliability	**15%**	Does the proposed architecture improve accountability for quality?	40%
		Does the proposed architecture facilitate lower defects?	60%

B. Check the representation of the metrics

Once the evaluation criteria have been defined, it is recommendable to check the representativeness of the metrics. The selection of representative "ilities" and attribute questions should be done considering a holistic point of view and ensuring that the different aspects of an enterprise's architecture are being covered.

There are different references that architects can use to avoid sub optimization of the metrics. One of them could be going through the verification of the four critical categories established in the balanced scorecard approach (Kaplan & D.P., 1996), as shown in figure 5.16.

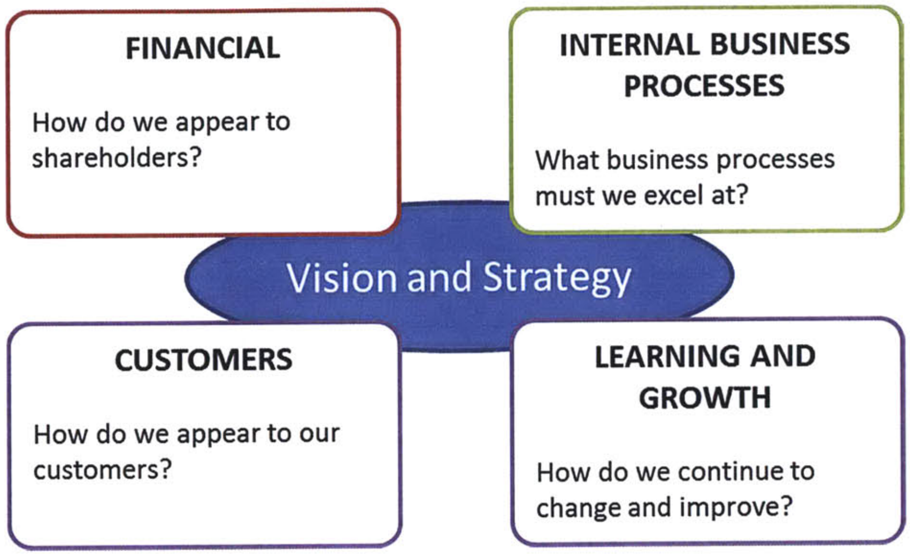


Figure 4-12: Balanced Scorecard approach

Another reference that could be used with this purpose, are the three critical enterprise attributes defined by Valerdi, Nightingale and Blackburn in "Enterprises as Systems: Context, boundaries and practical implications":

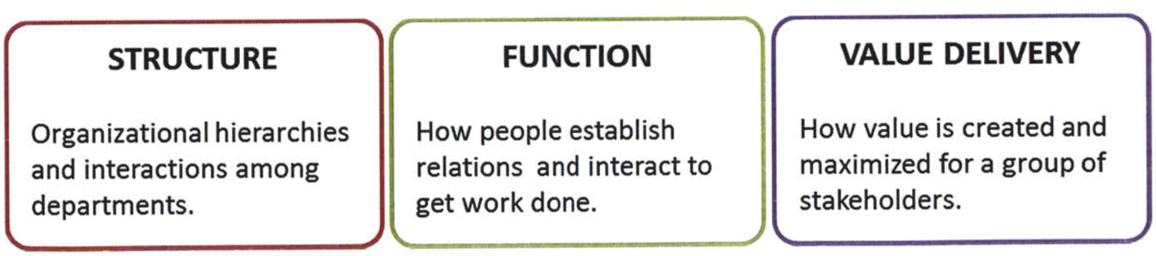


Figure 4-13: Three critical Enterprise attributes

4.3.4. Quantification of Effectiveness

Once the 'ilities', their prioritization and related attribute questions have been defined, the effectiveness quantification for the candidate architectures can be obtained. Scoring matrixes have been usually used to perform architecture evaluations. The matrix incorporates the relative weight of the 'ilities'. Their attribute specific questions are also considered. All the architectures are rated against the reference architecture ('As-Is'). The use of a scale from 1 to 5 is recommended (Ulrich & Eppinger, 2008), as shown in table 5.7.

Table 4-3: Rating scale for the Architecting Scoring Matrix

Relative Performance	Rating
Much worse than reference	1
Worse than reference	2
Same as reference	3
Better than reference	4
Much better than reference	5

Table 4.4 illustrates a simple example of this tool for the quantification of effectiveness.

Table 4-4: Example of application of Architecting Scoring Matrix

Criteria	Weight		Total Weight	Datum AS-IS	Arch. A	Arch. B	Arch. C
Manageability	**40%**						
Does the EA allow for clear accountability in terms of compliance with guidance and timeliness?		40%	16.0%	3	2	4	5
Does the EA facilitate the implementation, use and control of performance metrics?		60%	24.0%	3	2	4	5
Efficiency	**20%**						
Does the EA increase the firm's negotiation power with suppliers?		50%	10.0%	3	3	1	5
Does it minimize redundancy and managerial overhead?		50%	10.0%	3	5	1	5
Agility	**15%**						
Does it increase efficiency and speed of the product development process?		70%	10.5%	3	3	5	4
Does it reduce constrains for internal communication among departments?		30%	4.5%	3	3	3	4
Scalability	**25%**						
Does it support the short and medium term level of growth management wishes to achieve?		80%	20.0%	3	4	1	5
Does it allow room for employee career advancement?		20%	5.0%	3	1	5	3
Total Score				**3.00**	**2.90**	**2.91**	**4.75**
Ranking				**2**	**4**	**3**	**1**

4.3.5. Outputs of step 2

At the end of this second step of the Enterprise Evaluation Framework, the architect will have quantified the effectiveness of each of the candidate architectures in terms of both stakeholders' future values and the envisioned future state for the enterprise. In the process of doing so, the architect will have also identified the major strengths and weaknesses in terms of effectiveness for each architecture. This information can stimulate creative thinking and provide opportunities to improve or correct the initial designs.

4.4. Step 3: Effort Quantification

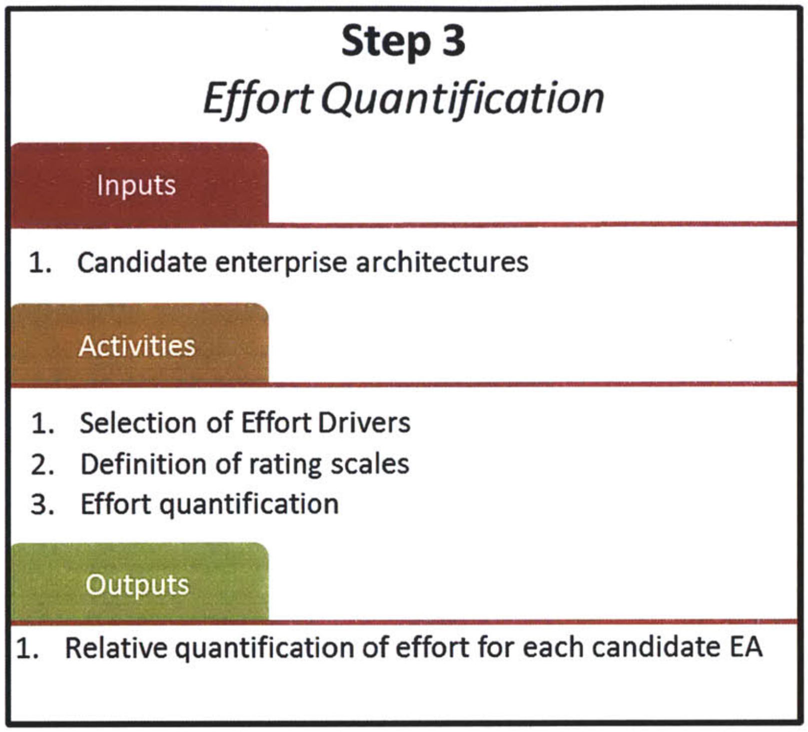


Figure 4-14: Overview of Step 3

The third step in the framework quantifies and compares the relative level of effort required to implement any of the proposed candidate architectures. Architectures will need more or less effort to be implemented depending on the top management motivation, resources required, staff incentives, skills, etc. that will make it happen. It is well known in the field of enterprise transformations that leading change is incredibly difficult to achieve and that eventually most enterprise's initiatives fail to achieve their expected results (Kotter, 1995). The problem is likely to be that the business analyses do not usually address behavioral and internal social concerns to the change. Thus, numerical information that provides a reference for the effort needed to implement the different candidate architectures results critical for decision makers because it allows the consideration of trade-offs that exist among the different options.

No formal approach to compare architecture implementation effort is currently used in the field. In this section we propose the use of a simple parametric model as a first approach in this direction. These types of models have proved to be useful tools to measure qualitative parameters in fields such as software and systems engineering.

Parametric models generate estimations based on mathematical relationships between variables, in this case "effort drivers". These drivers are a set of qualitative factors that have been shown to impact either positively or negatively, the amount of effort required to implement a major change in an enterprise. Assessing the candidate architectures against these drivers it is possible to determine which of those are more or less difficult to implement. For example, architectures that score low in drivers associated with effort savings or high on those associated with effort penalties will naturally end with a higher quantification of total effort.

We consider that the following activities are needed to quantify effort in a set of candidate architectures:

4.4.1. Activity 1: Selection of Effort Drivers

Selection and definition of the effort drivers that influence the ease or difficulty of the transformation is the first step of the process. Although these drivers might be contextual to each particular enterprise, after studying the key literature on enterprise transformation and change management, we identified eleven drivers that were permanently mentioned as relevant by authors and experts that were interviewed. We noticed also that these effort drivers tended to be grouped in four major areas: People, Complexity, Operations and Technology.

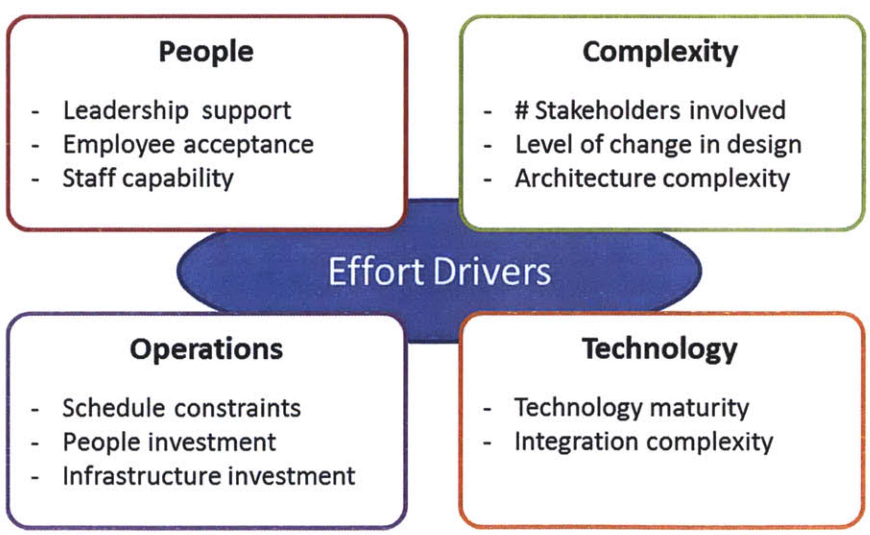


Figure 4-15: Reference Effort Drivers

As shown in the following tables, each driver listed in figure 4.15 was also defined and decomposed in factors that add or reduce effort. The goal of using sub factors is to help architects in assessing the candidate architectures against each effort driver.

Table 4-5: Reference effort drivers related to People

Driver	Factors add/ reduce effort	Definition
Leadership Support	- Motivation/ - Culture compatibility - Familiarity with architecture	Refers to the managers' support for the implementation of the proposed architecture.
Employee Acceptance	- Job losses - Trust in the EA - Culture compatibility - Growth opportunities	Refers to the employee support and willingness to make short term sacrifices for the implementation of the architecture.
Staff capability	- Experience - Competency with architecture	Refers to the level of experience, familiarity or capability of the staff to implement the proposed architecture.

Table 4-6: Reference effort drivers related to Complexity

Driver	Factors add/ reduce effort	Definition
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	Refers to the number and diversity of stakeholders needed to agree on the proposed architecture and its structure.
Level of change in design	- People that change functions - Policy/ procedures change	Refers to the amount of change required in the current structure of the enterprise for the implementation of the new architecture.
Architecture complexity	- Interdependencies - Coordination needed among levels	Refers to the level of complexity in design of the proposed architecture.

Table 4-7: Reference effort drivers related to Operations

Driver	Factors add/ reduce effort	Definition
Schedule constrains	- Implementation time - Time required to show results	Refers to the time constrains of the proposed change.
People investments	- Training - Hiring/ firing expenses	Refers to the upstream investment in human resources needed to implement the change.
Infrastructure investment	- New facilities - New equipment/ technologies	Refers to the upstream investment in facilities, equipment or technologies needed to implement the change.

Table 4-8: Reference effort drivers related to technology

Driver	Factors add/ reduce effort	Definition
Technology maturity	- Mature/ immature methods	Refers to whether the new architecture (in particular the information system) involves well known and mature methods.
Integration complexity	- Integration difficulties	Refers to the extent to which the new technologies/ tools may affect the implementation of a new architecture due to integration difficulties.

4.4.2. Activity 2: Definition of rating scales

In addition to the description, each driver needs to be assigned to a rating scale that rates the degree of impact on the implementation effort. We decided to include five rating levels for each driver: *Very Low, Low, Nominal, High, Very High*. The nominal level represents zero impact on implementation and is therefore assigned a reference value of 1.0. Levels above and below nominal are assigned values above or below 1.0 to reflect their positive or negative impact on the implementation effort.

Great care has to be taken to use consistent interpretation of these words on the different enterprise architectures being estimated. It is critical to define first what is going to be considered as *Nominal*, and based on based on that, the other levels can be defined. Reference levels' definitions for each driver are included on Appendix A. Table 4.9 illustrates an example for the driver *Leadership Support*.

Table 4-9: Example of rating levels for 'Leadership Support' driver

Very Low	Low	Nominal	High	Very High
- Low motivation/ urgency - Mayor cultural change - No familiarity proposed architecture	- Moderate motivation/ urgency - Some alignment internal culture - Some familiarity	- **Willing to generate the change** - **Compatibility with enterprise culture** - **Reasonable familiarity and trust in proposed architecture**	- High motivation/ urgency - Highly compatibility with internal culture - Strong familiarity w/ architecture	- Very high motivation/ urgency - Strong alignment with internal culture - Full familiarity w/ architecture
>> 1.0	**> 1.0**	**1.0**	**< 1.0**	**<< 1.0**

Once the nominal reference is defined, the next step is to define the values of the other levels of the rating scale. The increase or decrease of values along the rating scale will depend on the polarity of each driver (depending if they add or save effort). For example, the *Leadership Support* driver is defined in such a way that if a candidate architecture has a *Very Low* leadership support it will have a penalty on implementation effort. As a result, it will have a multiplier greater than 1.0, such as 1.86, to reflect an 86% effort penalty.

However, assigning ratings for these levels is not straightforward because they are qualitative in nature and require a subjective assessment in order to be rated. With the objective of defining a preliminary reference rating scale we created a questionnaire to rank the reference drivers in terms of their impact into implementation effort.

The questionnaire was sent to MIT researchers as well as experienced managers from different fields that had gone through an architectural change in their organizations. The results of this questionnaire and the application domains of the participants are shown in the figures below.

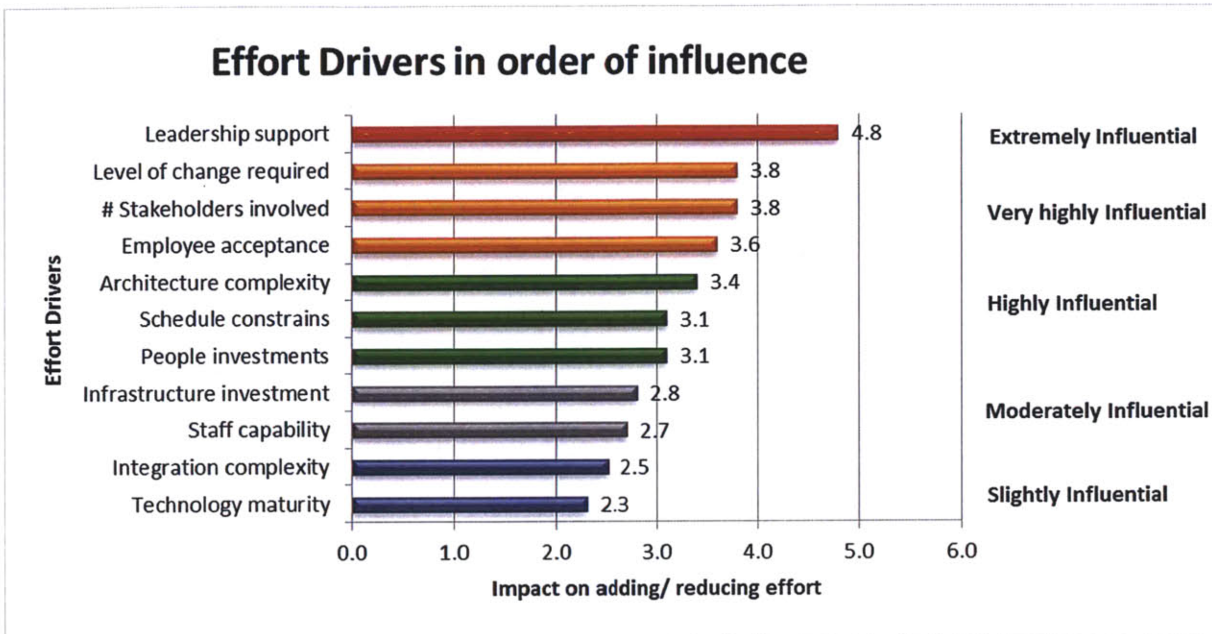


Figure 4-16: Effort drivers in order of impact to implementation effort

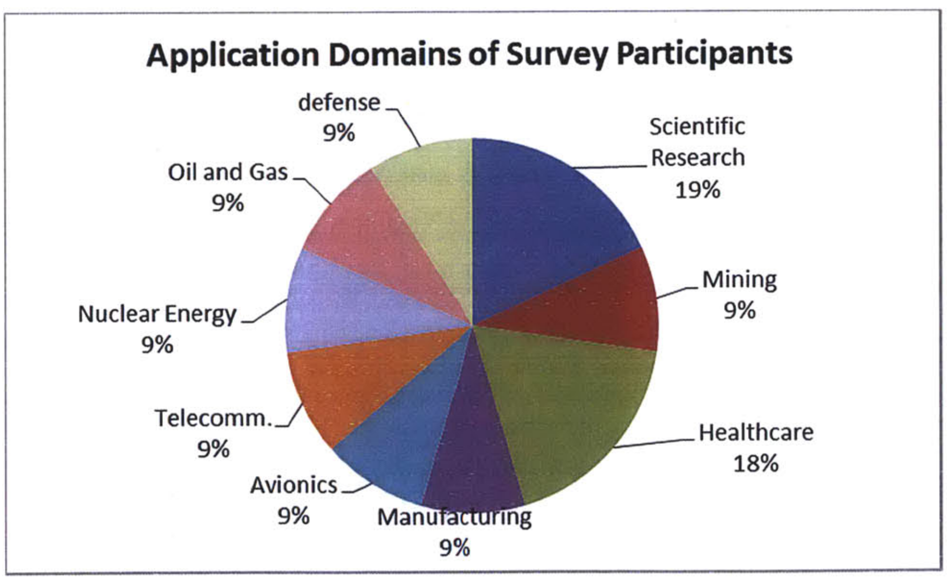


Figure 4-17: Application domains of questionnaire participants

The results of the questionnaire allowed us to identify the cost drivers which have the most influence on implementation effort. As shown in figure 4.16 they were grouped in five categories according to their average level of influence among different domains. This differentiation allowed us to determine a preliminary rating of effort across different levels of each driver. The process was done extrapolating calibrated parameters for equivalent effort drivers from the parametric model COSYSMO (Valerdi R. , 2005), as shown in Appendix B. The eleven effort drivers and their respective rating scales are shown in table 4.10.

Table 4-10: Preliminary rating scale for effort drivers

Effort Driver	Sub-factors	Very Low	Low	Nominal	High	Very High
People Drivers						
Leadership Support	- Motivation - Culture Compatibility - Familiarity/trust EA	**1.86**	**1.37**	**1.0**	**0.78**	**0.61**
Employee Acceptance	- Job losses - Believe/trust EA - Grow opportunities - Culture Compatibility	**1.70**	**1.30**	**1.0**	**0.82**	**0.66**
Staff Capability	- Competency proposed EA - Experience proposed EA	**1.42**	**1.19**	**1.0**	**0.88**	**0.76**
Complexity Drivers						
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Level of Change	- People involved - Policy/procedures change - Structure obstacles	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Architecture Complexity	- Interdependencies - Coordination	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Operational Drivers						
Schedule Constraints	- Time to implement - Time to show results	**0.66**	**0.82**	**1.0**	**1.22**	**1.49**
People Investment	- Training - Hiring/firing expenses	**0.76**	**0.88**	**1.0**	**1.19**	**1.42**
Infrastructure Investment	- Facilities - Equipment - Technologies	**0.76**	**0.88**	**1.0**	**1.19**	**1.42**
Technical Drivers						
Technology maturity	- Methods/ tools maturity	**1.28**	**1.13**	**1.0**	**0.88**	**0.78**
Integration complexity	- Integration difficulties	**0.78**	**0.88**	**1.0**	**1.13**	**1.28**

The nominal ratings for all the drivers are always 1.0, but the polarity of the ratings depend on the definition of the driver. For example, the *Employee acceptance* driver is worded positively since there is an effort saving associated with high or very high levels of acceptance. This is indicated by the values of 0.82 and 0.66, respectively representing a 18% and 34% savings in effort compared to the nominal case. Alternatively, the Architecture complexity driver has an effort penalty of 30% for "High" and 70% for "Very High".

It is important to note that the values shown in table 4.10 are preliminary results that will provide an estimation to decision makers of the relative levels of effort needed to implement the different candidate architectures. They represent a first approach of a model to support decision making during transformations. In its current state, the model has various limitations. The two most important are:

- Use of average results. The rating scale assigned to the different effort drivers was estimated based on the average results of the questionnaire. However, we observed significant effort reporting differences in some of the drivers. One of them was *Time constraints*, which appeared to have a lower impact in more stable industries such as healthcare, defense and mining. Another driver that showed important differences was *Employee acceptance.* In this case, however, the differences appeared to be contextual to the enterprise culture, not to the particular industry.

- Use of COSYSMO rating parameters. The rating of effort across different parameters was done base on the results of the expert questionnaire and reference values provided by COSYSMO for equally influential drivers. We selected COSYSMO as a first reference because it is a proved parametric model that estimates systems engineering effort on big scale projects. These projects have many similarities with transformations projects. However, this action was considered as a first step to determine preliminary values. Future research is needed to refine the rating scale based on historical transformation data and industry participation.

4.4.3. Effort quantification

As previously mentioned, we suggest the use of a simple parametric model to measure the relative effort needed to implement different candidate architectures. At this stage of development, the general form of the model contains only multiplicative parameters, in this case effort drivers. A factor is multiplicative, when it has a global effect across the overall system (Boehm, Valerdi, Lane, & Brown, 2005). Its general form can be seen in the following equation:

$$\boxed{RE = \prod_{i=1}^{n} EM_{i}}$$

Equation 1: Transformation Effort Model

Where,

RE= Relative Effort required for each Enterprise Architecture

EM$_i$= Multiplier values for the i_{th} effort driver. Nominal is 1.0.

n= Number of cost drivers (11)

The following table illustrates two examples of how this model is used. Each candidate is evaluated against each driver and a final score is obtained multiplying the values of each effort driver. Responses for candidate architecture “A” and “B” are highlighted in yellow. The final score for “A” is 2.11, and for “B” is 0.45.

Architecture A:

Table 4-11: Effort quantification example for Architecture "A"

Effort Driver	Sub-factors	Very Low	Low	Nominal	High	Very High
People Drivers						
Leadership Support	- Motivation - Culture Compatibility - Familiarity/trust EA	**1.86**	**1.37**	**1.0**	**0.78**	**0.61**
Employee Acceptance	- Job losses - Believe/trust EA - Grow opportunities - Culture Compatibility	**1.70**	**1.30**	**1.0**	**0.82**	**0.66**
Staff Capability	- Competency proposed EA - Experience proposed EA	**1.42**	**1.19**	**1.0**	**0.88**	**0.76**
Complexity Drivers						
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Level of Change	- People involved - Policy/procedures change - Structure obstacles	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Architecture Complexity	- Interdependencies - Coordination	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Operational Drivers						
Schedule Constraints	- Time to implement - Time to show results	**0.66**	**0.82**	**1.0**	**1.22**	**1.49**
People Investment	- Training - Hiring/firing expenses	**0.76**	**0.88**	**1.0**	**1.19**	**1.42**
Infrastructure Investment	- Facilities - Equipment - Technologies	**0.76**	**0.88**	**1.0**	**1.19**	**1.42**
Technical Drivers						
Technology maturity	- Methods/ tools maturity	**1.28**	**1.13**	**1.0**	**0.88**	**0.78**
Integration complexity	- Integration difficulties	**0.78**	**0.88**	**1.0**	**1.13**	**1.28**
	TOTAL SCORE			**2.11**		

Architecture B

Table 4-12: Effort quantification example for Architecture "B"

Effort Driver	Sub-factors	Very Low	Low	Nominal	High	Very High
People Drivers						
Leadership Support	- Motivation - Culture Compatibility - Familiarity/trust EA	**1.86**	**1.37**	**1.0**	**0.78**	**0.61**
Employee Acceptance	- Job losses - Believe/trust EA - Grow opportunities - Culture Compatibility	**1.70**	**1.30**	**1.0**	**0.82**	**0.66**
Staff Capability	- Competency proposed EA - Experience proposed EA	**1.42**	**1.19**	**1.0**	**0.88**	**0.76**
Complexity Drivers						
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Level of Change	- People involved - Policy/procedures change - Structure obstacles	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Architecture Complexity	- Interdependencies - Coordination	**0.66**	**0.82**	**1.0**	**1.30**	**1.70**
Operational Drivers						
Schedule Constraints	- Time to implement - Time to show results	**0.66**	**0.82**	**1.0**	**1.22**	**1.49**
People Investment	- Training - Hiring/firing expenses	**0.76**	**0.88**	**1.0**	**1.19**	**1.42**
Infrastructure Investment	- Facilities - Equipment - Technologies	**0.76**	**0.88**	**1.0**	**1.19**	**1.42**
Technical Drivers						
Technology maturity	- Methods/ tools maturity	**1.28**	**1.13**	**1.0**	**0.88**	**0.78**
Integration complexity	- Integration difficulties	**0.78**	**0.88**	**1.0**	**1.13**	**1.28**
	TOTAL SCORE			**0.46**		

4.4.4. Output of Step 3

At the end of this second third step of the Enterprise Evaluation Framework, the architect will have quantified the effort needed to implement each of the candidate architectures. The information results particularly relevant for decision makers because it allows the consideration of trade-offs that exist among the different options.

4.5. Step 3: Quantification of Risk

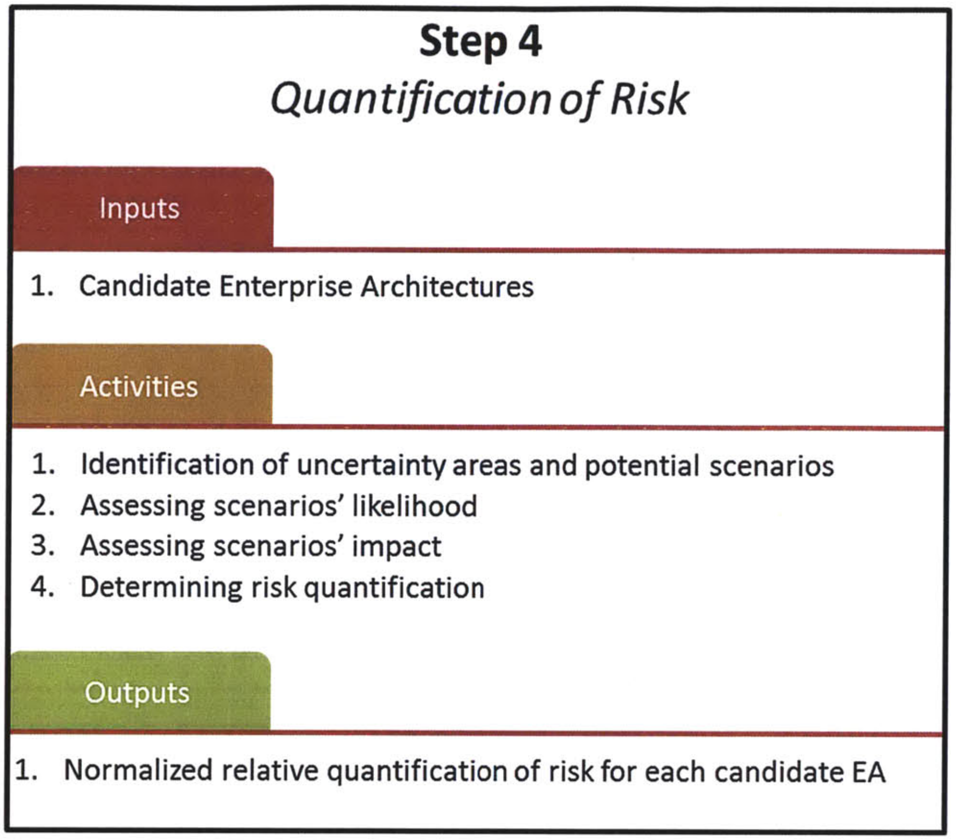


Figure 4-18: Overview of Step 4

The development of the future enterprise is a process of decision making where uncertainty and incomplete information is always present (Saha, 2006). Unexpected events or changes in the internal or external context can affect the performance of the enterprise both positively or negatively. Depending on the enterprise structure, these events can have different levels of likelihood or impact; making certain architectures more risky than others.

In this section we introduce a method to assess the level of risk associated with each candidate architecture. This step of the framework provides key complementary information to decision makers because "no guarantee exists that a system [architecture] will remain best in the face of changing objectives or contexts" (Carlson & Doyle, 2000).

Normally, enterprises are able to assess their vulnerabilities by answering these basic questions (Sheffi Y. , The Resilient Enterprise: Overcoming Vulnerability for Competitive Advantage, 2007):

1. What can go wrong?
2. What is the likelihood of that happening?
3. What are the consequences or impact if it does happen?

The wide literature that existing in risk management indicates that the process of answering those questions and quantifying the risk level for each alternative seems to involve at least four activities:

- Identification of uncertainty areas and potential events
- Assessing events' likelihood
- Assessing events' impact
- Determining a Risk quantification

4.5.1. Identification of Uncertainty Areas and Potential Events

The first activity in the process is to identify and classify the areas of uncertainty that can cause potential context changes or unexpected events to the future enterprise. Simply put, "uncertainties are things that are not known, or known only imprecisely" (Hastings & Mc Manus, 2004).

Uncertainties can normally be classified as internal or external to the enterprise. At the same time, they tend to be slip in different areas, making it difficult to decision makers to know what form they will take. For example, the past three years alone brought a series of external unexpected events that many organizations failed to consider. The subprime mortgage crisis, the earthquake and tsunami in Japan, and populist uprisings in Egypt, Libya and neighboring countries have already affected enterprises with different levels of impact.

The enterprise risk management (ERM) literature, as well as the work done by MIT professors like Sheffi (2007, 2012) and De Neufville (2004) provided us with some insights to identify and classify the major areas of uncertainty faced by enterprises.

Figure 4.19 illustrates a concentric vulnerability map to identify external and internal uncertainties. It classifies uncertainties in three major areas: financial, strategic and operational uncertainties. The illustration is an adaptation to the field of enterprise architecture of a concentric vulnerability map used at General Motors by their Enterprise Risk Management Team (Sheffi Y. , 2007).

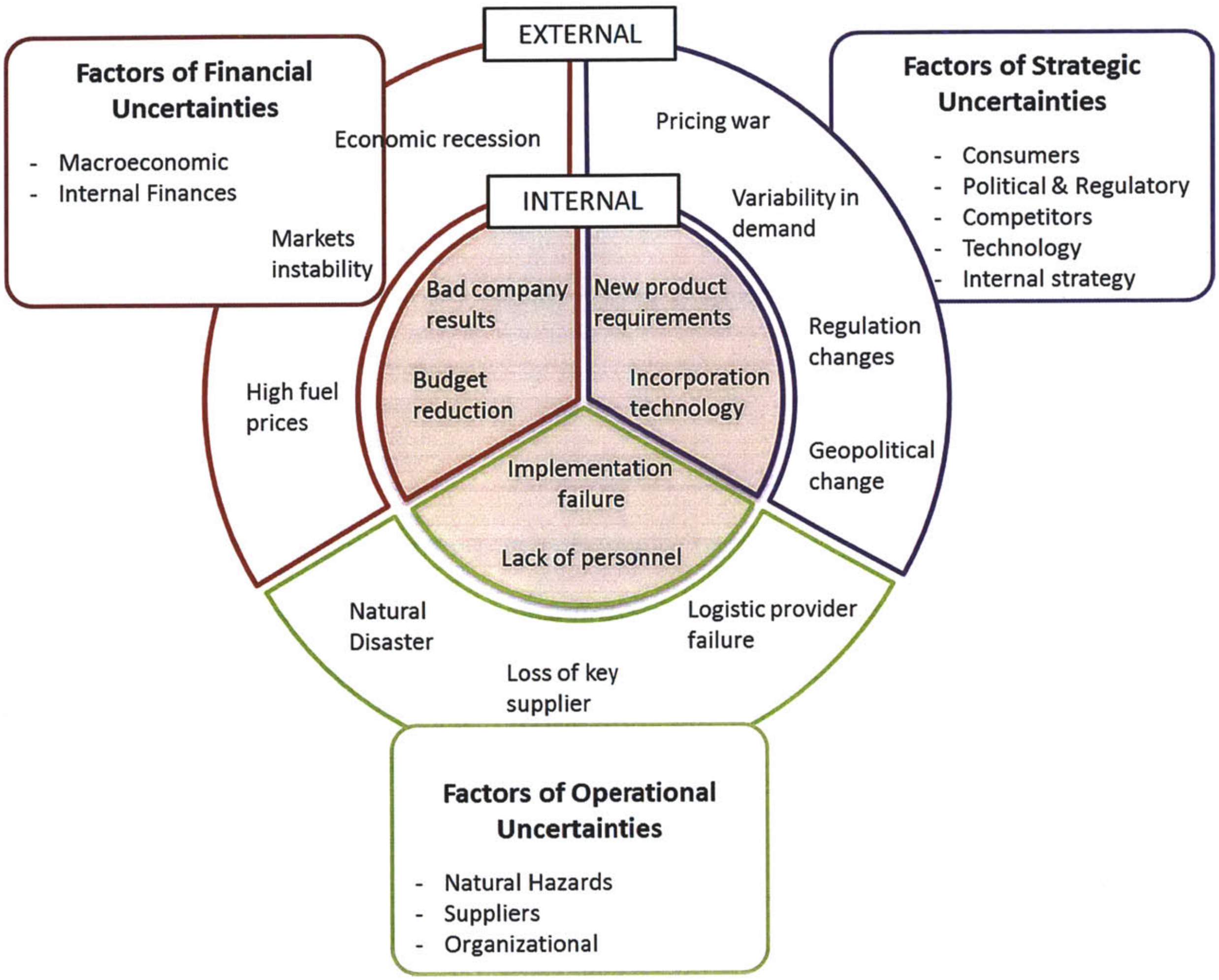


Figure 4-19: Classification of uncertainty areas

Our goal in presenting this categorization is not to offer an exhaustive list of uncertainties, but rather suggesting a starting point to help architects in their thinking to identify the potential unexpected scenarios their candidate EA might face during its implementation and operation.

4.5.2. Assessing Scenarios' Likelihood

The next step is to estimate the likelihood of occurrence of each of them within a discrete time frame (typically 5 years for our case). This step is difficult because generally neither models nor historical data that are inputs to those models are available. Enterprises can overcome these difficulties by starting with a qualitative analysis to classify likelihood of occurrence within ratings of probabilities. Table 4.13 provides the categories and related probability levels typically used in business planning and risk prioritization processes (Fraser & Simkins, 2010).

Table 4-13: Categories probability levels for assigning likelihood

Likelihood	Probability in Planning Period (5 years)
Very Likely (VL)	> 95%
Likely (L)	95% to 65%
Medium (M)	65% to 25%
Unlikely (U)	25% to 5%
Remote (R)	< 5%

In general terms, likelihoods will tend to be the same for external scenarios for which the enterprise has no or limited influence. (e.g. downturns in the economy or changes in regulation). However, probabilities of occurrence of other scenarios, in particular those more related with the enterprise structure, might change for all the different candidate architectures being evaluated. Differences in size, structure and the way they are interconnected would make some of them more vulnerable to the occurrence of certain events. For example, architectures that consider major changes to the current EA are more susceptible to implementation failures due to internal resistance or key skill shortages.

The identified potential scenarios and their estimated likelihood for each architecture alternative can be listed in a scenario evaluation matrix like the one in table 4.14.

Table 4-14: Event evaluation matrix (likelihood)

Uncertainty Area	Scenario	Architecture A		Architecture B		Architecture C	
		Likelihood		**Likelihood**		**Likelihood**	
Financial	Economic recession	M		M		M	
	High fuel prices	L		L		L	
Strategic	Variability in demand	L		M		M	
	Geopolitical risk	VL		M		U	
Operational	Implementation failure	VL		L		M	
	Logistic provider failure	M		L		M	

4.5.3. Assessing Scenarios' Impact

Once likelihoods are estimated, the next activity involves assessing the scenarios' impact over the performance of each EA. Uncertainty scenarios can negatively or positively impact the proper performance of an enterprise and its architecture. The negative impacts generate outcomes such as bad operating results, schedule deviations or failure to achieve proposed goals. On the other hand, positive impacts are associated to the ability to an enterprise to exploit some of the opportunities of associated with these scenarios and increase their planned performance.

As before, there are no hard rules with respect to measuring impacts, but experience has shown that a 1-5 scale provides enough gradations for most evaluations (Fraser & Simkins, 2010). Based on the existing risk management best practices, impact of uncertainty events on architectures could be assessed using the following example criteria:

Table 4-15: Example criteria to assess impact in architectures

Catastrophically Negative (Cat.)	Impact threats the survival of the company in its current form.
Severe Negative (Sev.)	Fundamental threat to operating results. (e.g. serious financial loss that leads to a loss on productivity and ROI).
Marginally Negative (Mar.)	Noticeable deterioration in the achievement of results (e.g. minor financial losses in a line of business).
Negligible Negative (Neg.)	Almost none negative impact on the achievement of results. (e.g. minimal impact on ability to deliver value to customers).
Positive Impact (Pos.)	Positive impact on the achievement of results due to its ability to exploit opportunities of uncertainty's scenario.

Using these criteria, the identified potential scenarios, their estimated likelihood and their potential impact for each architecture alternative can be listed in the same event evaluation matrix, as shown in Table 4.16.

Table 4-16: Event Evaluation Matrix (likelihood and impact)

Uncertainty Area	**Scenario**	**Architecture A**		**Architecture B**		**Architecture C**	
		Likelihood	**Impact**	**Likelihood**	**Impact**	**Likelihood**	**Impact**
Financial	Economic recession	**M**	**Cat.**	**M**	**Mod.**	**M**	**Pos.**
	High fuel prices	**L**	**Sev.**	**L**	**Sev.**	**L**	**Sev.**
Strategic	Variability in demand	**L**	**Sev.**	**M**	**Neg.**	**M**	**Pos.**
	Geopolitical risk	**VL**	**Mod.**	**M**	**Sev.**	**U**	**Mod.**
Operational	Implementation failure	**VL**	**Sev.**	**M**	**Sev.**	**U**	**Sev.**
	Logistic provider failure	**M**	**Sev.**	**M**	**Mod.**	**M**	**Mod.**

4.5.4. Determining Risk Quantification

Combining the likelihood and impact factors described in the above sections it is possible to estimate the risks associated with each event, considering the different candidate architectures. Risk has normally a negative connotation and is often expressed in terms of a combination of impact and associated likelihood of occurrence. To assess the risk that each potential scenario presents to the different architectures, the use of a two dimensional "risk space" is recommended.

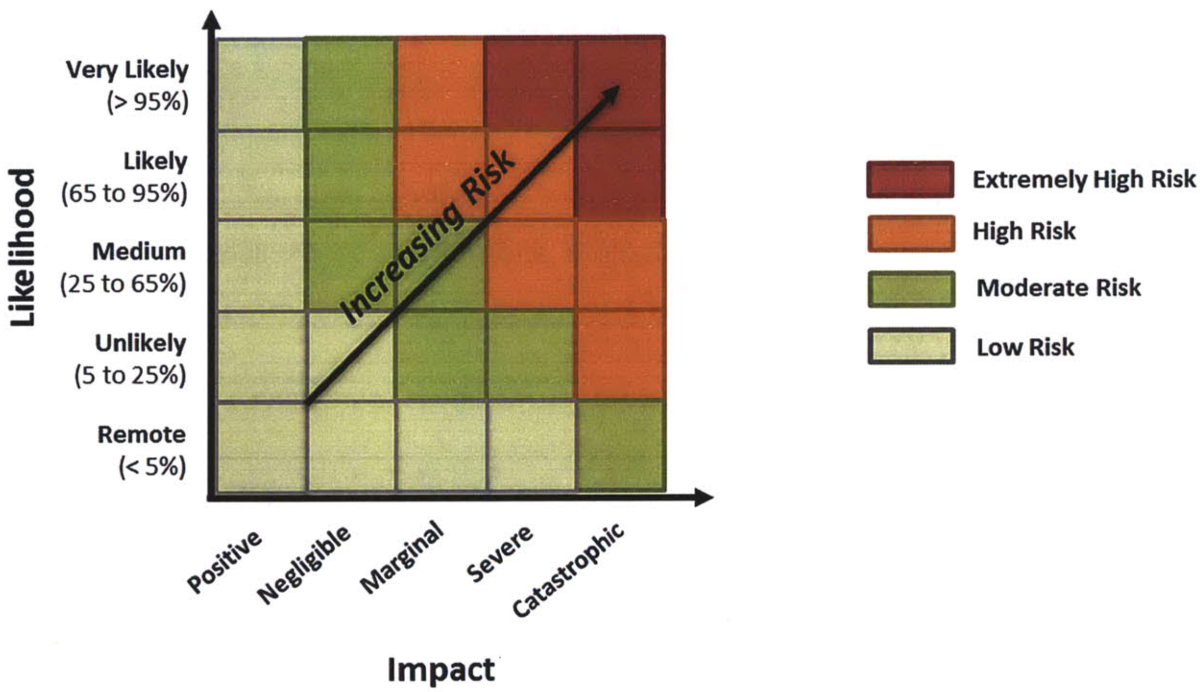


Figure 4-20: Risk space to classify risk levels

Applying this classification scheme to the results shown in table 4.16, a risk comparison matrix can be obtained, as shown in table 4.17. Moreover, applying a 0/1/3/7 severity scale to the risk scales obtained for each scenario, it is possible to quantify the relative global level of risk that each candidate architecture has when considering potential future events that might change the internal or external context of the enterprise. In that quantification process, 7 represents extremely high risk, 3 represents high risk, 1 represents moderate risk and 0 does it for low risk.

Table 4-17: Risk Quantification Matrix

Uncertainty Area	Scenario	Architecture A	Architecture B	Architecture C
Financial	Economic recession	***High Risk***	***Moderate Risk***	***Low Risk***
	High fuel prices	***High Risk***	***High Risk***	***High Risk***
Strategic	Variability in demand	***High Risk***	***Moderate Risk***	***Low Risk***
	Geopolitical risk	***High Risk***	***High Risk***	***Moderate Risk***
Operational	Implementation failure	***Extremely High Risk***	***High Risk***	***Moderate Risk***
	Logistic provider failure	***High Risk***	***Moderate Risk***	***Moderate Risk***
Quantification of Risk		**22**	**12**	**6**
Risk Profile		***High***	***Medium***	***Low***

Table 4.17 illustrates a risk analysis performed to three fictitious candidate architectures. The results obtained in the analysis provide fundaments to architects to classify the candidate architectures in different risk categories: ***"High"***, ***"Medium", "Low" or "Very low"***. The criteria to select which category to select will depend on the results obtained in the analysis and on the risk tolerance culture of the enterprise conducting the transformation.

4.5.5. Output of Step 4

The output of Step 4 of the Framework, provide architects with a third dimension of results to support managers in their decision making process on the selection of the future architecture. It should be noted that the value of this simple risk quantification process is only as good as the accuracy of the methods used to quantify likelihood and impact of the various unplanned scenarios. For a more detailed analysis, the use of appropriate models is more desirable than the use of qualitative approaches.

4.6. Outcome: Architecture Trade-off Matrix

The central outcome of performing an evaluation on different dimensions is to uncover key facts that result critical to supporting decision making. At the end of the four steps of the Enterprise Architecture Evaluation Framework, architects will be able to generate an "Architecture Trade-offs Matrix", a visual representation that facilitates decision making. Various decision analyses could quickly be done by looking at the results shown on the matrix. Figure 4.21 illustrates a reference "Architecture Trade-offs Matrix".

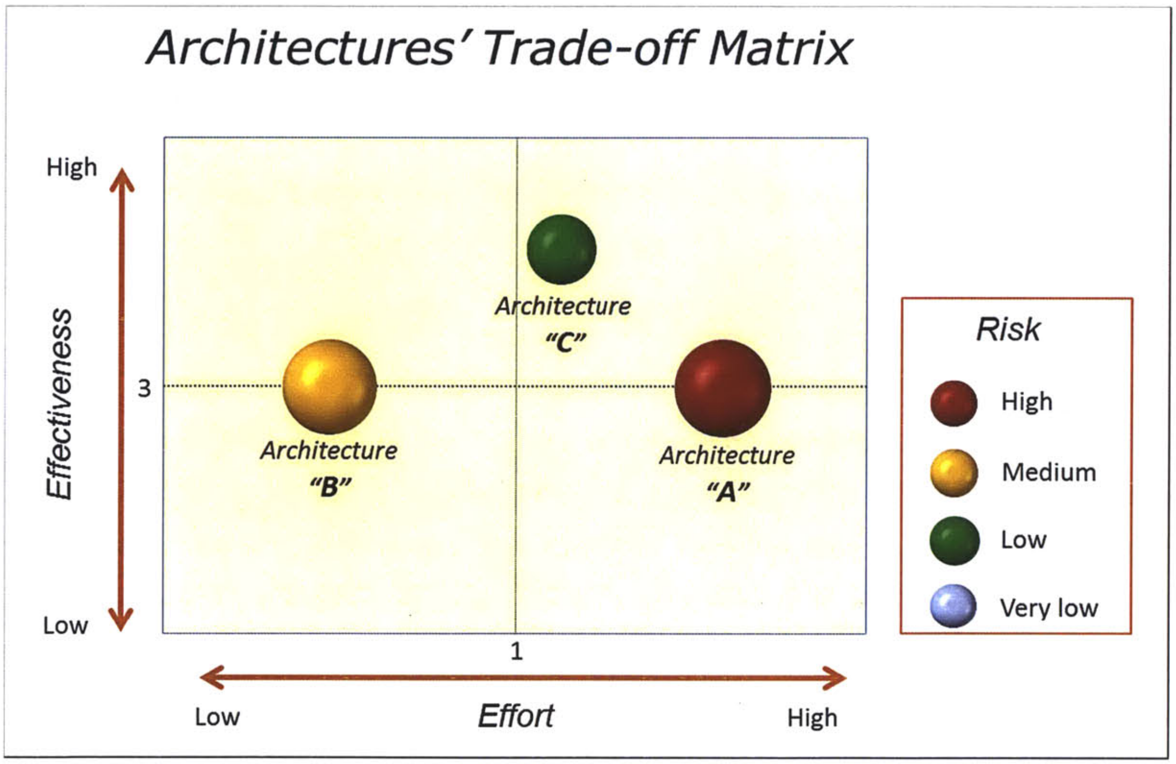


Figure 4-21: Architecting tradeoff matrix

Having a representation like this is helpful for decision making because it allows an easy visualization of the trade-offs that exists among the different alternatives. Important decisions usually have conflicting objectives and, since you can't normally fulfill all of them simultaneously, you have to make tradeoffs. Usually decision makers have to give up something on one dimension to achieve more in terms of another.

"Decisions with multiple dimensions cannot be resolved by focusing on any one dimension" (Hammond, Keeney, & Raiffa, 1999).

For example, in the example shown in figure 4.21, there is no architecture that is better than other two in all the dimensions. Architecture "A" dominates "C" and shows the best results in terms of effectiveness. However, it requires more effort to be implemented than "B". Therefore, decision makers have to make a trade-off to decide between "B" and "C. The final decision will depend on the evaluation strategies and priorities defined by decision makers. The attitude toward risk is also important. As stated by Ross and Rhodes (2008), "no absolute correct or best design exists without subjectively specifying the best strategy for evaluation". Strategies can include maximum effectiveness, minimum effort, minimum risk or any combination among others.

The output of the framework facilitates the identification of the best alternative for the strategy selected by the enterprise's decision makers, as well as the understanding of the most important tradeoffs inherent in the candidate architectures. It helps decision makers and architects reason about architectural decision by showing informed trade-offs caused by the interaction of multiple elements.

5. Application of the Framework: iSoftware Case Study

In this chapter, the AFE Framework is applied to a transformation project of a real enterprise. All the steps are illustrative applied to one of the business units of iSoftware, a world leading and profitable networking company. Please note, that the real names and numbers of the case study have been deceived for this report. At the same time, in order to keep the case study brief, an abridged version of the report is provided.

5.1. Enterprise Background

5.1.1. iSoftware

iSoftware Systems is a company that sells software, high tech products and monitoring services to businesses of all sizes, governments, service providers and consumers. The company has been consistently increasing sales over the last ten years and has achieved presence in three continents around the world. Its vision highlights the importance of technology for the actual world and recognizes innovation and operational excellence as their core values. In order to achieve those values, they invest heavily in research and development. At the same time, its emphasis on operational excellence leads top managers to a constant revision of opportunity improvements within its many business units.

This case study is focused on an "enterprise architecting" project of one of *iSoftware* business units within its Technical Services Area, *I-Software Intelligent Software Services Area (ISSA)*.

5.1.2. Intelligent Software Service Area (ISSA)

iSoftware ISSA's main task is to develop software and provide the architecture that supports the Intelligent Software products portfolio, one of the "Technical Services" offered by the company. The *"Intelligent Sofware Service Package*" is a technical service offered to companies that enables a proactive and remote network monitoring, checkups, repairs and technical support for the entire customer's iSoftware network.

The ISSA organization is composed of around 360 people that are mainly located in the US. There is only a small team located in India of around 20 developers that belongs to the company. On the other hand, there are 5 different companies (or partners) where ISSA outsources the programming of products. The total number of outsourced programmers varies depending on the load, but on average ISSA has 390 outsourced people working on its projects at any time. Those partners are located in China, India, Eastern Europe and the US. A graphical view of the structure is included in Appendix C.

The product development cycle is as follow: First, a product requirement document (PRD) is written by the Technology Services Product Management, which is responsible for regularly gathering customer requirements. The PRD passes through a gate called "executive committee" that is composed by executives from product and function groups who decide what PRD to approve and start working on. Five parallel work streams are then initiated simultaneously and resources are started to be allocated. Although there's one overall product manager, he does not carry out the outsourcing decisions. Those decisions are made by the manager of the four mayor teams of the organization. A flow chart of product development is also included in Appendix C. At any one time, there are multiple PRDs being worked on by these teams. They look at the availability of resources, deliverables already in process, time estimation, and work with product and the project manager to make sure the schedule is met.

Outsourcing is a relevant part of the product development process, mainly to achieve cost objectives (typically a project has to be 30-40% outsourced because of cost). Outsourcing is also relevant to fulfill skills (e.g. a particular programming language) and schedule requirements.

5.2. Pre-Architecting Steps

5.2.1. Understanding ISSA's motivation for change

In 2010, top management of ISSA realized that the unit needed to improve its internal processes to remain competitive within the company. ISSA's project managers were spending almost 50% of their time talking and coordinating with the software development outsourcing partners over the phone. At the same time, it was common for partners to complain about not having enough information about the product they had to produce. There was a feeling **that a better outsourcing structure was needed to improve lead time and cost savings.** However, this feeling was not evident to some department managers, who appeared to be comfortable with the flexibility of the current structure. The lack of a clear "burning platform" in the organization made a holistic and unbiased assessment of the outsourcing process a very attractive option for ISSA.

Our "enterprise architecting" work was sponsored by a process improvement manager of the Operational Excellence division of the enterprise, working as an internal consultant for ISSA. We decided to conduct a general assessment of ISSA's outsourcing process and provide short to medium range recommendations that considered a 3 to 5 years horizon.

5.2.2. Understanding of enterprise landscape

a) *Ecosystem (External Context)*

The company has achieved a leadership position in several of the markets where it competes. In the market of products associated with the services provided by ISSA, the company had a worldwide leadership position with a market share of almost 55%. The rest of the market was fragmented in smaller competitors. That market share discouraged new entrants in this specific niche. The competitive position of iSoftware and its monitoring service provided by ISSA was even stronger considering the high switching cost for their customers. However, this strong position was not a reality on all the markets the company was present. In the last year, iSoftware was suffering product substitution in products lines targeted mainly to private customers.

At the time of this case study, the latest demand trend for remote monitoring services indicated that the market was growing fast and revealed an attractive scenario for technological and service suppliers. Despite these numbers, sales of ISSA had been growing at a slower pace compared to the market (9%). This was mainly to the fact that smaller competitors have been able to react faster to new market demands. Their product development cycles are proven to be shorter and therefore their time to market is faster.

In term of suppliers, the most important suppliers for ISSA products were third party software developers. In the ISSA's scenario those supplier enjoyed a relative strong position compared to the business unit mainly due to their size (over 5,000 engineers) and because of the fragmented outsourcing strategy of our enterprise that ISSA was employing. In addition, they enjoyed certain privileges because the switching costs were high for ISSA and because it takes time to reach high levels of productivity to develop the software ISSA was providing (know-how, training and development of communication channels).

ISSA customers were mainly conformed by governments, information technology industries and other businesses, where ISSA products were still identified as the market leader in terms of maturity, reliability and brand recognition. However the speed of change of technology was increasing and new ways of offering monitoring services could have become available in the short-term.

b) *Internal landscape (Internal Context)*

As explained before, the iSoftware was facing strong competition in some of its markets, especially in those related to consumers market. For this reason, major internal changes were being implemented. The new company's strategy was moving the enterprise toward its core products and in particular toward services provided to other businesses and governments. Although changes caused layoffs and increased the stress level of iSoftware employee, the restructuring process was favorable to ISSA. Being part of the strategy of the firm will potentiate its importance and projection of future demand was positive. ISSA' director had a special motivation to respond to this new challenge.

5.2.3. Understanding of stakeholder values

Our initial task was to identify the major stakeholders of iSoftware ISSA who can provide us with insights and information relevant to the internal and outsourcing processes. The ISSA major stakeholder groups are summarized in the figure below.

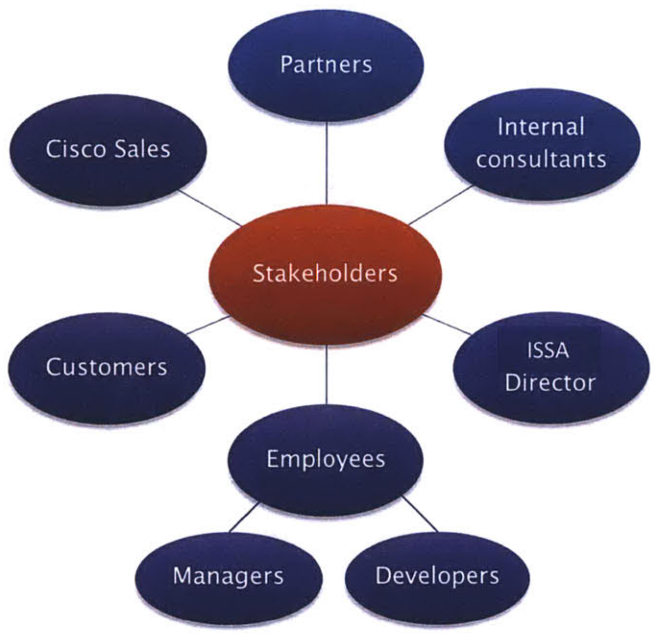


Figure 5-1: Stakeholder Identification

The following are some of the most relevant answers we received from the Project Managers and Team leaders within ISSA. They encompass the average results considering the answers from Project, Engineering, Architecture and QI Managers. Similar analyses were done with all stakeholders. For the sake of brevity they are not all included in this section. Figure 5.2 illustrates a value exchange representation from two types of general managers within ISSA that shows results in terms of current performance vs relative importance for each of them. Figure 5.3 illustrates a consolidated value exchange table among different stakeholders and identifies those with greater improvement opportunities.

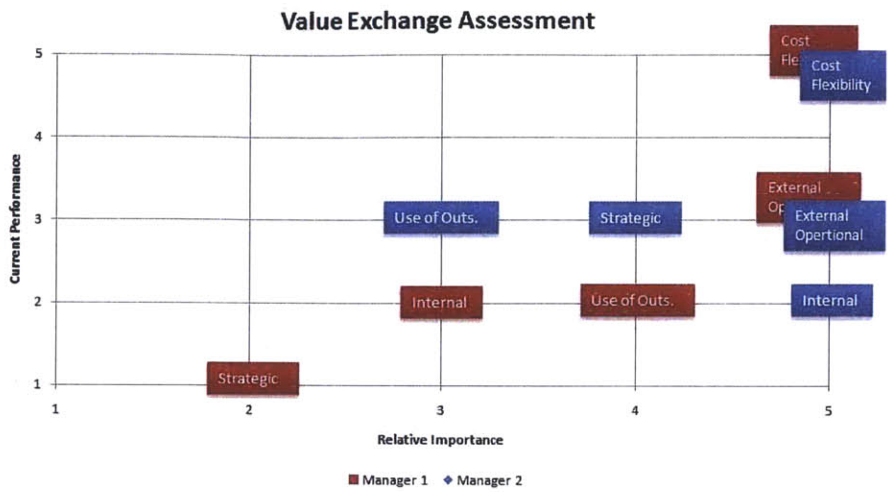


Figure 5-2: Managers' value exchange assessment

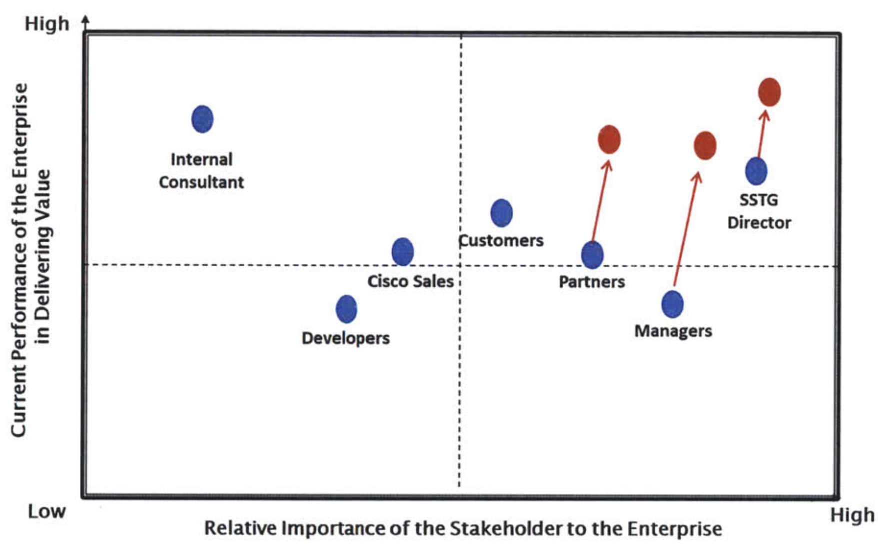


Figure 5-3: Consolidated stakeholders' value exchange

5.2.4. Understanding of the 'As-Is' enterprise

In order to have a better understanding of the "current enterprise" we performed a holistic assessment using the 10 elements/ 8 views framework proposed by Nightingale and Rhodes (Nightingale & Rhodes, 2012). The main artifacts used to collect information were formal interviews, questionnaires and company reports. Some of our key findings are summarized in figure 5.4.

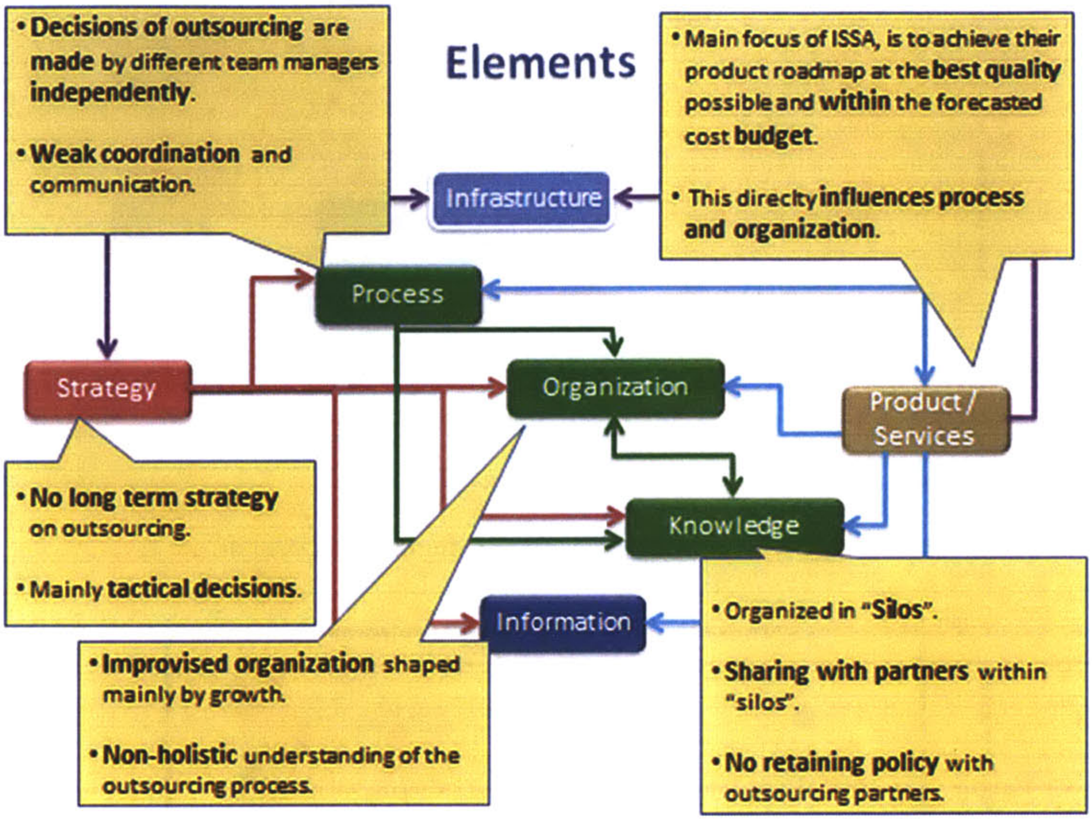


Figure 5-4: 'As-Is' key findings using 10 elements framework

Another tool we used to better understand the organization was a simple system dynamic model. We performed some simulations adding more people and projects to the current organization. Its main conclusion was that the current architecture was not scalable because more people and projects will impact negatively the current productivity. Not having defined procedures, metrics and network will end up having negative effects over productivity; therefore, increasing delays in projects rather than improving lead times.

Based on the results of the Stakeholders Analysis, the 10 Elements framework, the dynamic model, as well as from the discussions we had with our sponsors at *iSoftware*, our main conclusions were:

- ISSA has grown a lot in the past years and has been successful in managing the new requirements through an ad hoc organization. However, there are indicators that show that the current outsourcing structure is about to reach capacity:
    - Time to market is taking longer (compare to customer expectations)
    - Communication with partners is becoming cumbersome
    - Current structure is "burning out some teams" to keep productivity levels, with negative effects on moral.

- There is no clear owner of the outsourcing process as a whole (holistic view). This lead to a weak strategy view of the outsourcing process (tactical decisions), with little coordination among departments, no standard procedures and no use of performance metrics to evaluate outsourcing partners.

- Finally, we perceived that an important paradigm in the organization is 'cost reduction'. This has helped them to bring cost down and improve efficiency among the eyes of shareholders, but with the negative effect of losing sight of the other sight of the coin: the generation of value to different stakeholders.

5.2.5. Creation of an Holistic Vision

a) Core ideology

The company has a strong culture on innovation and operational excellence. We worked with the senior management to reinforce this enduring character in the organization. ISSA's vision was then defined as:

"To develop highly reliable, affordable and timely solutions to our customers based on operational excellence".

b) <u>Envisioned future State</u>

Based on ISSA's strategic plan, the drivers of the organization and the stakeholder values, in this transformation effort the senior leadership of ISSA described their envisioned future for the business unit as follows:

December 31, 2016 – *"Today, software as a service has become the standard for the technology business. This fact and our ability to adapt to the new requirements made our business unit to growth faster than any in iSoftware and we expect to be the largest service unit of the company within the next two years. We had been able to keep our operational excellence while expanding our sales more than 20% year after year. Our ability to adapt quickly to new scenarios and strategic alliances with our partners had set this business apart, providing the highest customer satisfaction of the market. We have also matured our processes and technologies, so that we expanded our portfolio package from software to cloud computer".*

c) <u>Gaps & Opportunities Identification</u>

Based on the envisioned future state and the issues found in the 'As-Is' analysis, some of the 'gaps & opportunities' ISSA should address are:

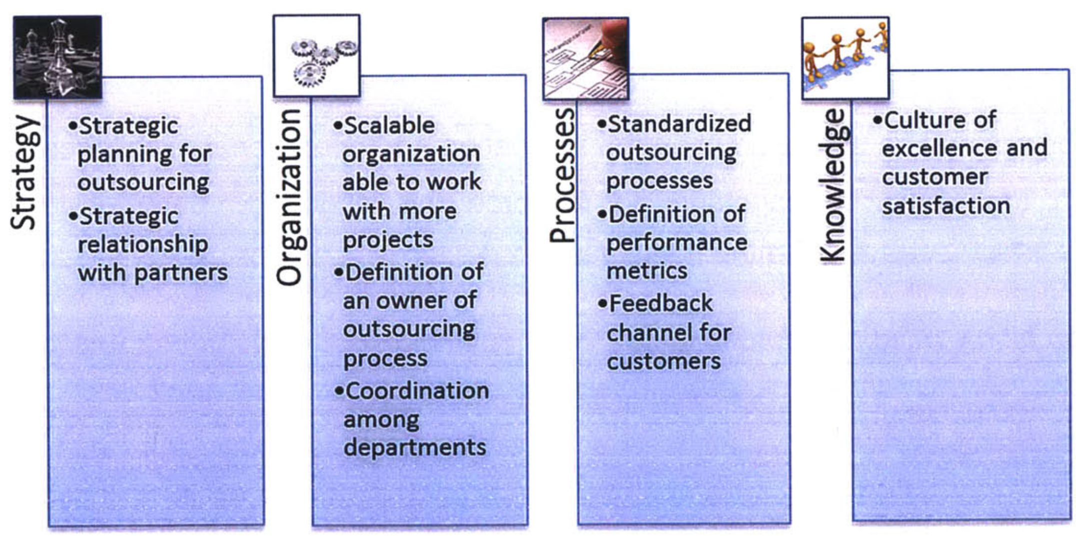


Figure 5-5: Gaps that ISSA should address

5.3. Step 1: Generation of Candidate Architectures

Once the pre-architecting steps were completed, we followed the prescriptive approach described in Step 1 of the EA Evaluation Framework to develop and refine different candidate architectures.

5.3.1. Activity 1: Own thinking

Initially, we conducted rounds of brainstorming discussing about the best enterprise structures to allow ISSA to manage its outsourcing work and meet its future growing objectives. We tried to address the problem using the more important 'views' for this project.

In terms of strategy, we generated different concepts to develop the required capabilities to close the gaps previously identified. We thought on internal development of these capabilities through a restructuration of its product development process. We also explored the option of reducing the current size of the organization and look for these capabilities through external sourcing (e.g. alliances or acquisition of partners). We explored here different options, moving the vertical boundaries of the organization in different directions.

In terms of organization, we brainstormed about different grouping options for their employee. The current structure is grouped "by activity", which allow them to reduce costs and have more specialization in certain areas. We explored other options to enhance coordination and better relations with partners. We thought in particular on the grouping "by output" option that would allow more control and closer relationships with partners or in combinations of "activity and output" options.

In terms of processes, we focus on generating ideas that could improve the current coordination problems and reduce cycle times. Improving outsourcing control and using standardized procedures were some of the options we discussed.

5.3.2. Activity 2: Learn from experience

We looked on “lead users” on different fields to bring new perspectives to some of the issues ISSA was trying to solve. For example, we looked at Toyota to understand more how they were managing their relationships with their suppliers or partners. Learning from the Toyota principles to manage supply chain (Womack, Jones, & Roos, 1991), we came out with some takeaways for ISSA and iSoftware. Some of them were:

- Work with few partners/suppliers, but invest on them (training, relationship) and think in long term partnerships.
- Look more on profits sharing rather than cost savings.
- Share people with suppliers; have iSoftware engineers at partners' locations and viceversa.
- Look for continuous improvement opportunities (kaizen) with partners.
- Increase trust with them. Allow them to design and have better understanding of complete package production.

5.3.3. Activity 3: Ask for suggestions

We talked also to MIT researchers about best outsourcing practices. There is a wide literature on this topic about practices that work better for some industries and others that don't. We reviewed many of those papers and refined our initial concepts.

We tried also to explore outsourcing practices used by other business units at iSoftware. Unfortunately, due to confidentiality issues, we were not granted access to them.

5.3.4. Activity 4: Extreme organizations checking

During the process of generating concepts, we also considered ideas that would imply moving the enterprise into more extreme situations. The two more important options that we discussed were:

(i) the implementation of a strong outsourcing strategy that will move the core of the product development cycle to outside partners;

(ii) The implementation of a strong in house policy that would reduce dependency on suppliers to the minimum. Thinking about these alternatives allowed us to better understand the tradeoffs and implications of the outsourcing process.

5.3.5. Activity 5: Derive candidate architectures

From the concepts generated in the previous steps, we generated different candidate architectures. After a pre-selection process using SWOT analysis, we decided to continue with four potential alternatives. In the following paragraphs we described the four candidates at a macro level. We include also a reference chart to show the organizational changes in the organization, where ISSA's team is white, partners are highlighted in grey and outsourcing decision-holders are represented by black boxes.

1. <u>Architecture "A": 'Strong Outsourcing'</u>

One option for SSTG is to strengthen the role of outsourcers in their product development process. This alternative would imply a radical change in the organization because outsourcing partners would take care of the whole process of engineering and testing of the projects with the objective of reducing costs and improve communication. Under this EA, only the Architecture and Project Management teams would remain within the ISSA organization. The Project Management team will be reinforced in order to conduct the additional coordination, evaluation and control tasks. On the other side, the partners would be responsible of delivering the products for ISSA. This architecture is represented in figure 5.6.

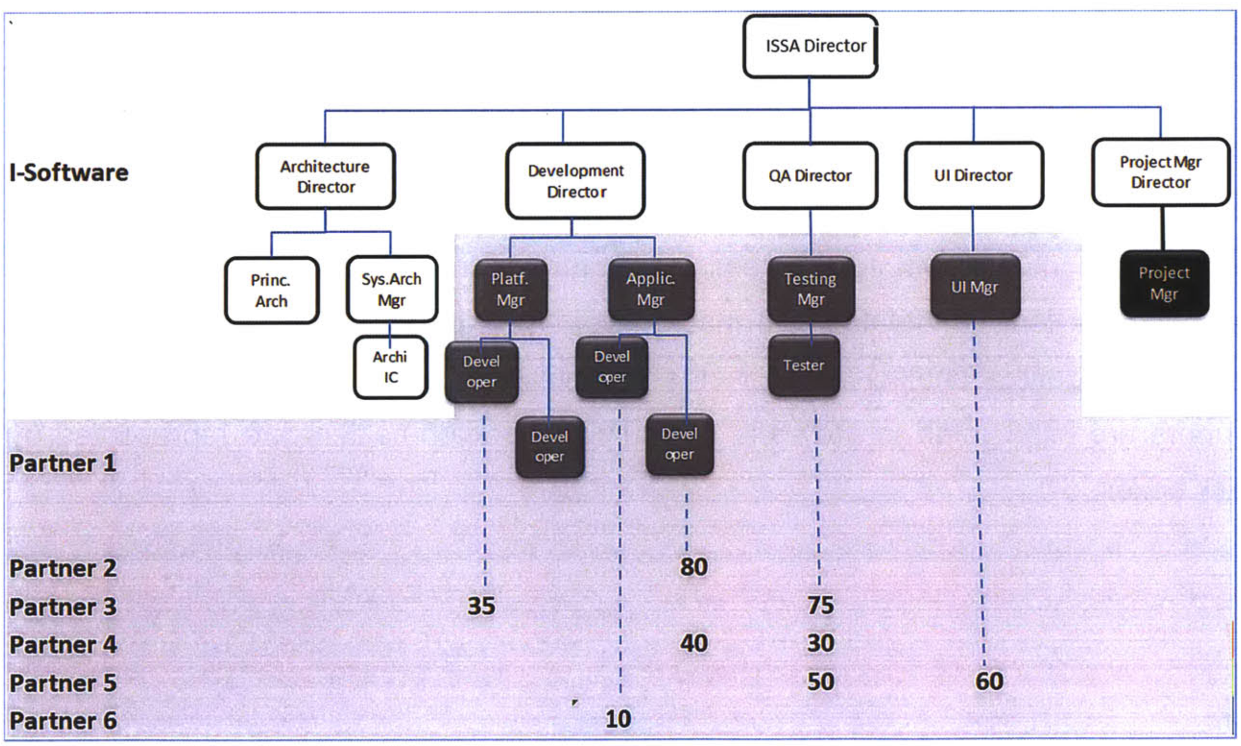


Figure 5-6: Architecture "A", 'Strong outsourcing'

2. <u>Architecture "B": 'Back-sourcing'</u>

This option is exactly the opposite of the previous one. It refers to the action of bringing the existing outsourcing services back to "in house". This would imply to most of the outsourcing activities. There would basically be two ways of doing this:

• Acquire one or two of the current outsourcing partners.

• Gradually hire and create new teams that would be taking the task of programming and testing.

Some the long term benefits that this strategy will bring are: More autonomy and control in the development process, better communication channels, improvement in cycles times and protection if intellectual property. However, this option would increase the operational costs and reduce the flexibility to market changes, in particular demand changes.

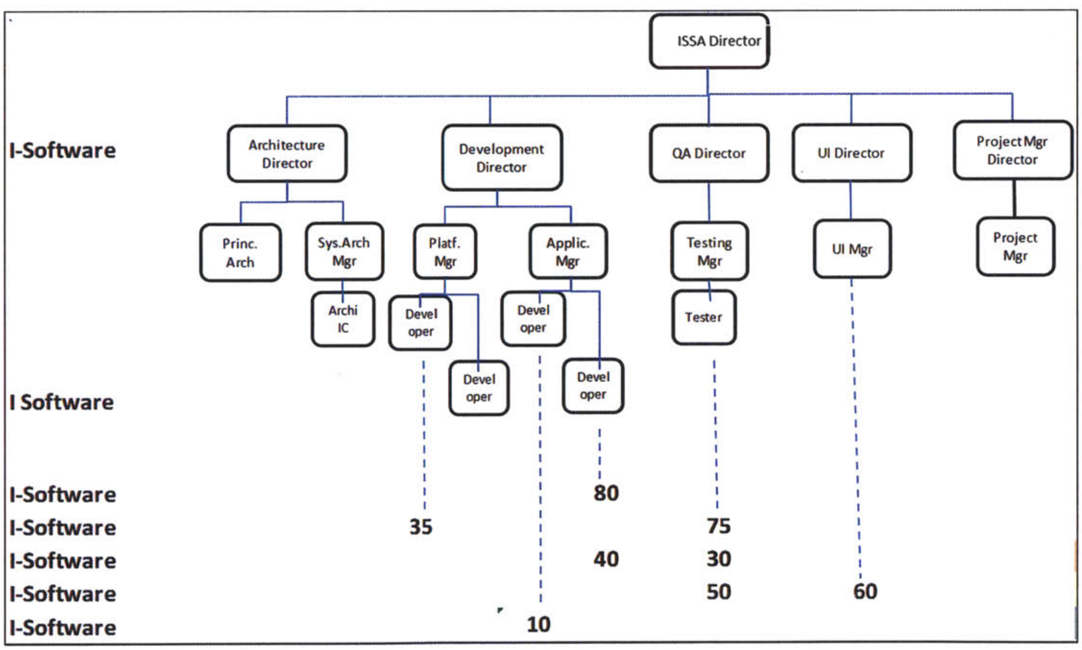


Figure 5-7: Architecture "B", 'Back-sourcing'

3. Architecture "C": 'Outsourcing team'

Another alternative would be to create a new team/department that will be in charge of the procurement process. This would allow ISSA to centralize the expertise on contract definition, channels of communication and bargaining power. For example, with the current process, different areas could be simultaneously conducting outsourcing with the same partner without using their bargaining power. At the same time, this group would be able to have dedicated employees to monitor and manage the performance of outsourcers.

On the other hand, this would imply a high implementation cost and would increase the head-count of the current organization. Additionally, it will certainly add an extra-layer of people (example: engineering manager, outsourcing manager, partner) that could lead to longer lead times at the beginning (learning curve and acceptance of new structure by employees). However, if successfully implemented, having a specialized team doing the outsourcing coordination and monitoring would give better governance and control to the outsourcing activities. It can also collaborate in reducing cost and increasing value to customers through the implementation of better practices to manage relationship with partners.

It is also important to note, that the company has procurement structures in other business units. Thus, the implementation of an outsourcing team would not be something completely new to the organization.

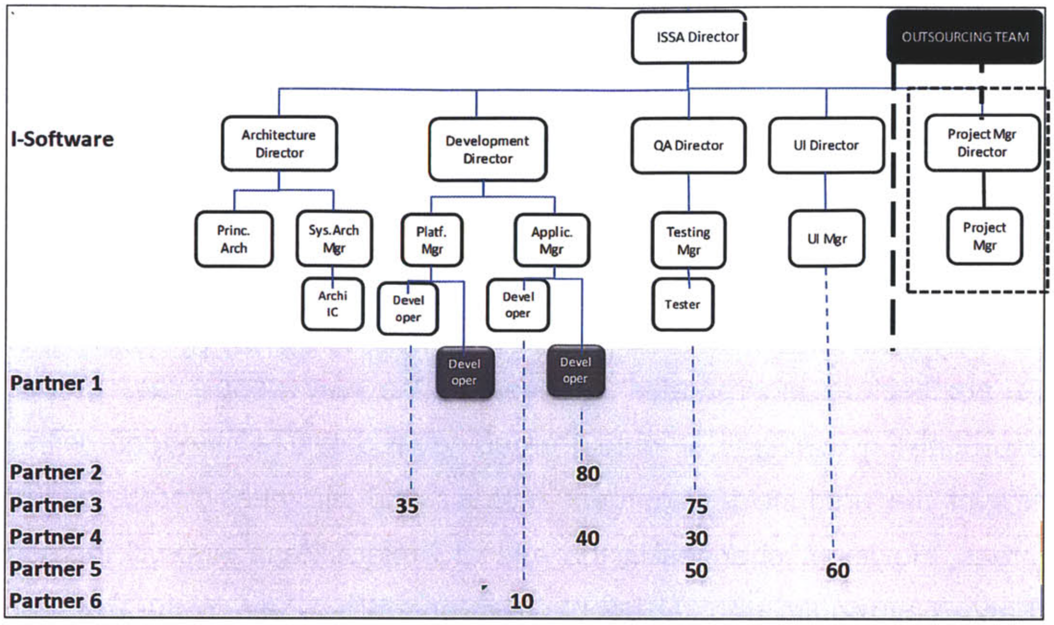


Figure 5-8: Architecture "C", 'Outsourcing Team'

4. <u>Architecture D: 'Process Owner"</u>

The current structure of ISSA includes a Project Managers team. Each Project Manager is responsible of the process and schedule governance through the entire product development cycle of each project. However, they are not involved in the outsourcing decisions. The main idea of this architecture is to empower the Project Managers in order to have an end-to-end responsibility and authority in the process of product development.

This type of architecture implies a shift from department goals to process goals, where managers from different areas would need to change from a department design to a more teamwork oriented process. Having strong process owners allows the organization to be prepared for change and makes people less reluctant to new environments. The process focus would allow ISSA to adapt more quickly to the context and therefore increases the chances of survival in a world of rapid change. Also, it would help aligning different areas and would enhance and holistic view of the process.

The process manager would also define and supervise the outsourcing activities. He should provide better governance of the process and would be responsible for defining the outsourcing partners, the contracts, procedures and the resources allocated to each particular project. This would facilitate the centralization of the high level definitions of the outsourcing process and, therefore, would allow having a stronger bargaining power with the partners.

This architecture incorporates changes in the way people are group in the organization applying concepts of 'output grouping'. It provides managers with a more holistic view of the product development process and facilitates coordination among different areas. However, changes in the role of Project Managers will increase their work load and may require transfer people from operative areas to the PM team. This increase in work load could create scalability constraints in the medium term.

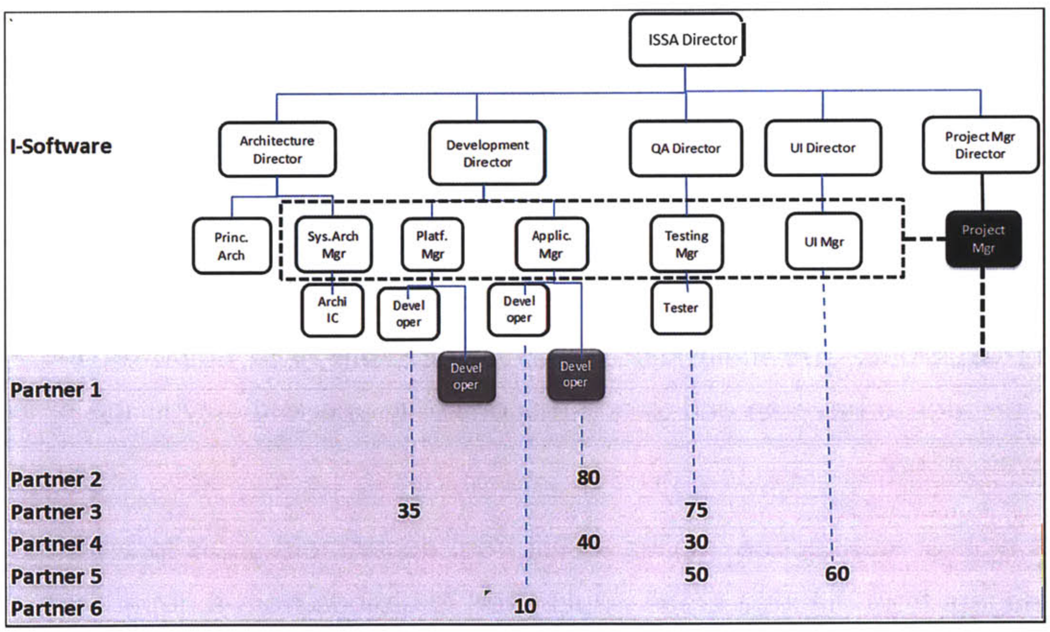


Figure 5-9: Architecture "D", 'Process Owner'

5.4. Step 2: Effectiveness Quantification

Once the candidate architectures we moved to the evaluation steps following the Enterprise Architecting Evaluation Framework. We began with the effectiveness quantification for each of the four options.

5.4.1. Selection of 'ilities'

For the selection of 'ilities' to quantify effectiveness we focused on the core attributes needed to achieve the envisioned future state and on the expected values of the stakeholders of the future enterprise.

a) Envisioned Future State:

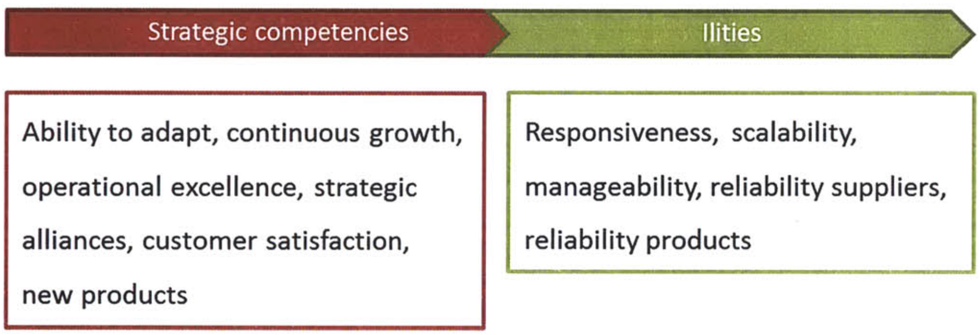


Figure 5-10: Selection of 'ilities' based on Strategic competences

b) Stakeholder's future values:

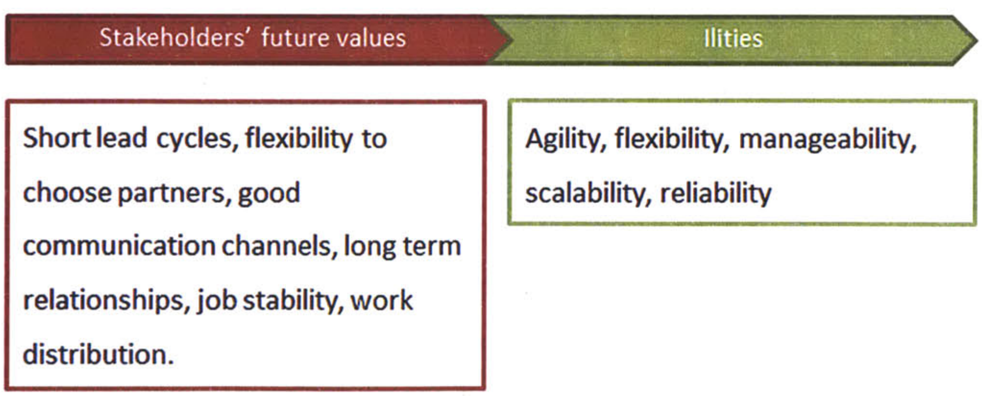


Figure 5-11: Selection of 'ilities' based on stakeholder's future values

5.4.2. Prioritization of 'ilities'

'Ilities' were prioritized using a 'combined quantification table' that considers both the salience of the different stakeholders, as well as the relative importance of the different 'ilities' for each of them.

Table 5-1: 'ilities' prioritization

Stakeholder	Salience	Responsiveness		Agility		Manageability		Reliability product		Flexibility		Scalability		Reliability Supplier	
ISSA Director	Definite	High	1.00	High	1.00	Medium	0.70	High	1.00	Medium	0.70	High	1.00	High	1.00
Project Managers	Expectant	Medium	0.35	High	0.50	High	0.50	High	1.00	High	0.70	Hiigh	0.70	Medium	0.35
Outsourcing partners	Expectant	-	-	-	-	High	0.50	Medium	0.35	-	-	Medium	0.35	High	0.50
Customers	Expectant	High	0.50	High	0.50	-	-	High	0.50	-	-	-	-	-	-
Developers	Latent	Low	0.10	Medium	0.18	High	0.25	Medium	0.18	High	0.25	High	0.25	High	0.25
iSoftware sales	Latent	Medium	0.18	High	0.25	Low	0.10	High	0.25	-	-	-	-	Medium	0.18
Internal Consultants	Latent	-	-	High	0.25	High	0.25	High	0.25	High	0.25	High	0.25	High	0.25
Total per "Ility"			2.13		2.68		2.30		3.53		1.90		2.55		2.53
Average Weight			12%		15%		13%		20%		11%		14%		14%

5.4.3. Definition of attribute questions

Based on the strategic goals and the different interest of the stakeholders, the selected 'ilities' were decomposed on specific questions to quantify effectiveness. Table 6.2 (below) illustrates the questions and relative weight given to each of them.

5.4.4. Quantification of Effectiveness

The quantification of effectiveness was made using a scoring matrix with a rating scale from 1 to 5. Rating 3 was considered as similar in effectiveness to the current 'As-Is' and lower or greater ratings were indications of worse or better performance. Table 6.2 illustrates the results.

Table 5-2: Effectiveness quantification for candidate architectures

Criteria	Weight		Total Weight	Arch. A	Arch. B	Arch. C	Arch. D
Responsiveness	**12%**						
Does the proposed architecture improve the response time to varying customer demands?		60%	7.2%	1	5	4	4
Does it facilitate the introduction of newer products/ services?		40%	4.8%	2	4	3	3
Agility	**15%**						
Does it increase efficiency and speed of the product development process?		50%	7.5%	1	5	4	4
Does it reduce constrains for internal communication among departments?		50%	7.5%	4	3	5	5
Manageability	**13%**						
Does it facilitate the communication channels among different internal levels and outsourcing partners?		60%	7.8%	4	5	5	4
Does the EA facilitate the implementation, use and control of performance metrics?		40%	5.2%	2	4	5	4
Reliability Products	**20%**						
Does the proposed architecture facilitate lower defects through better communication?		40%	8.0%	3	5	4	4
Does it facilitate the implementation of best practices that allow growth while continuing operational excellence?		60%	12.0%	2	5	5	4
Flexibility	**11%**						
Does it provide the flexibility to choose from different suppliers if needed?		80%	8.8%	5	3	4	4
Does it allow room for employee career advancement?		20%	2.2%	1	5	3	3
Scalability	**14%**						
Does it support the short and medium term level of growth management wishes to achieve?		60%	8.4%	5	4	5	4
Does it minimize redundancy and managerial overhead?		40%	5.6%	3	4	4	5
Reliability Suppliers	**14%**						
Does the proposed EA emphasize long term relations with suppliers?		50%	7.0%	4	4	5	4
Does the distribution of outsourcers provide reliability in terms of suppliers' availability?		50%	7.0%	3	5	3	3
Total Effectiveness Score				**2.98**	**4.31**	**4.30**	**3.95**
Ranking				**4**	**1**	**2**	**3**

5.5. Step 3: Effort Quantification

As shown below, using the parametric model template detailed in chapter 5, we estimated the relative effort needed to implement the different candidate architectures.

Table 5-3: Effort quantification for candidate architectures

Architecture A: 'Strong Outsourcing'

Effort Driver	Sub-factors	Very Low	Low	Nominal	High	Very High
People Drivers						
Leadership Support	- Motivation - Culture Compatibility - Familiarity/trust EA	1.86	**1.37**	1.0	0.78	0.61
Employee Acceptance	- Job losses - Believe/trust EA - Grow opportunities - Culture Compatibility	1.70	**1.30**	1.0	0.82	0.66
Staff Capability	- Competency proposed EA - Experience proposed EA	1.42	1.19	**1.0**	0.88	0.76
Complexity Drivers						
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	0.66	0.82	**1.0**	1.30	1.70
Level of Change	- People involved - Policy/procedures change - Structure obstacles	0.66	0.82	1.0	1.30	**1.70**
Architecture Complexity	- Interdependencies - Coordination	0.66	0.82	**1.0**	1.30	1.70
Operational Drivers						
Schedule Constraints	- Time to implement - Time to show results	0.66	0.82	1.0	**1.22**	1.49
People Investment	- Training - Hiring/firing expenses	0.76	0.88	1.0	**1.19**	1.42
Infrastructure Investment	- Facilities - Equipment - Technologies	**0.76**	0.88	1.0	1.19	1.42
Technical Drivers						
Technology maturity	- Methods/ tools maturity	1.28	1.13	**1.0**	0.88	0.78
Integration complexity	- Integration difficulties	0.78	0.88	1.0	**1.13**	1.28
	TOTAL SCORE			4.9		

Architecture B: 'Back-sourcing'

Effort Driver	Sub-factors	Very Low	Low	Nominal	High	Very High
People Drivers						
Leadership Support	- Motivation - Culture Compatibility - Familiarity/trust EA	1.86	1.37	1.0	**0.78**	0.61
Employee Acceptance	- Job losses - Believe/trust EA - Grow opportunities - Culture Compatibility	1.70	1.30	**1.0**	0.82	0.66
Staff Capability	- Competency proposed EA - Experience proposed EA	1.42	1.19	**1.0**	0.88	0.76
Complexity Drivers						
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	0.66	0.82	**1.0**	1.30	1.70
Level of Change	- People involved - Policy/procedures change - Structure obstacles	0.66	0.82	1.0	1.30	**1.70**
Architecture Complexity	- Interdependencies - Coordination	0.66	0.82	1.0	**1.30**	1.70
Operational Drivers						
Schedule Constraints	- Time to implement - Time to show results	0.66	0.82	1.0	**1.22**	1.49
People Investment	- Training - Hiring/firing expenses	0.76	0.88	1.0	1.19	**1.42**
Infrastructure Investment	- Facilities - Equipment - Technologies	0.76	0.88	1.0	**1.19**	1.42
Technical Drivers						
Technology maturity	- Methods/ tools maturity	1.28	1.13	**1.0**	0.88	0.78
Integration complexity	- Integration difficulties	0.78	**0.88**	1.0	1.13	1.28
	TOTAL SCORE			3.1		

Architecture C: 'Outsourcing Team'

Effort Driver	Sub-factors	Very Low	Low	Nominal	High	Very High
People Drivers						
Leadership Support	- Motivation - Culture Compatibility - Familiarity/trust EA	1.86	1.37	1.0	0.78	**0.61**
Employee Acceptance	- Job losses - Believe/trust EA - Grow opportunities - Culture Compatibility	1.70	1.30	**1.0**	0.82	0.66
Staff Capability	- Competency proposed EA - Experience proposed EA	1.42	**1.19**	1.0	0.88	0.76
Complexity Drivers						
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	0.66	0.82	**1.0**	1.30	1.70
Level of Change	- People involved - Policy/procedures change - Structure obstacles	0.66	0.82	**1.0**	1.30	1.70
Architecture Complexity	- Interdependencies - Coordination	0.66	0.82	1.0	**1.30**	1.70
Operational Drivers						
Schedule Constraints	- Time to implement - Time to show results	0.66	0.82	1.0	**1.22**	1.49
People Investment	- Training - Hiring/firing expenses	0.76	0.88	**1.0**	1.19	1.42
Infrastructure Investment	- Facilities - Equipment - Technologies	0.76	0.88	**1.0**	1.19	1.42
Technical Drivers						
Technology maturity	- Methods/ tools maturity	1.28	1.13	**1.0**	0.88	0.78
Integration complexity	- Integration difficulties	0.78	0.88	**1.0**	1.13	1.28
	TOTAL SCORE			1.15		

Architecture D: 'Process Owner'

Effort Driver	Sub-factors	Very Low	Low	Nominal	High	Very High
People Drivers						
Leadership Support	- Motivation - Culture Compatibility - Familiarity/trust EA	1.86	1.37	1.0	0.78	**0.61**
Employee Acceptance	- Job losses - Believe/trust EA - Grow opportunities - Culture Compatibility	1.70	1.30	**1.0**	0.82	0.66
Staff Capability	- Competency proposed EA - Experience proposed EA	1.42	1.19	**1.0**	0.88	0.76
Complexity Drivers						
# Stakeholders involved	- Number of stakeholders - Diversity of stakeholders	0.66	0.82	**1.0**	1.30	1.70
Level of Change	- People involved - Policy/procedures change - Structure obstacles	0.66	**0.82**	1.0	1.30	1.70
Architecture Complexity	- Interdependencies - Coordination	0.66	0.82	**1.0**	1.30	1.70
Operational Drivers						
Schedule Constraints	- Time to implement - Time to show results	0.66	0.82	**1.0**	1.22	1.49
People Investment	- Training - Hiring/firing expenses	0.76	**0.88**	1.0	1.19	1.42
Infrastructure Investment	- Facilities - Equipment - Technologies	0.76	**0.88**	1.0	1.19	1.42
Technical Drivers						
Technology maturity	- Methods/ tools maturity	1.28	1.13	**1.0**	0.88	0.78
Integration complexity	- Integration difficulties	0.78	0.88	**1.0**	1.13	1.28
	TOTAL SCORE			0.50		

Table 5-4: Effort quantification summary tables

	Arch. A	Arch. B	Arch. C	Arch. D
Total Effort Score	4.90	3.10	1.10	0.50
Ranking	4	3	2	1

5.6. Step 4: Risk Quantification

5.6.1. Identification of major uncertainties

In order to quantify how risky were the candidate architectures, we identified first the major uncertainties that might generate unexpected events to the enterprise:

Table 5-5: Identification of major uncertainties

Internal Uncertainties:	- Implementation failure - New products requirements - Intellectual property losses
External Uncertainties:	- Variability in demand - Economic downturn - Loss of key supplier

5.6.2. Estimation of 'likelihood' and 'impact'

Table 5-6: Events evaluation matrix (likelihood and impact)

Uncertainty	Architecture A		Architecture B		Architecture C		Architecture D	
	Likely-hood	Impact	Likely-hood	Impact	Likely-hood	Impact	Likely-hood	Impact
Internal								
Implementation failure	**Very Likely**	**Sev.**	**Likely**	**Sev.**	**Unlikely**	**Sev.**	**Unlikely**	**Sev.**
New products requirements	**Medium**	**Sev.**	**Medium**	**Neg.**	**Medium**	**Neg.**	**Medium**	**Neg.**
Intellectual property losses	**Very Likely**	**Sev.**	**Unlikely**	**Sev.**	**Medium**	**sev.**	**Medium**	**Sev.**
External								
Variability in demand	**Likely**	**Neg.**	**Likely**	**Cat.**	**Likely**	**Neg.**	**Likely**	**Mar.**
Economic downturn	**Medium**	**Neg.**	**Medium**	**Sev.**	**Medium**	**Mar.**	**Medium**	**Mar.**
Loss of supplier	**Likely**	**Neg.**	**Unlikely**	**Neg.**	**Medium**	**Mar.**	**Medium**	**Mar.**

5.6.3. Risk Quantification

Using a two dimensional risk space, the risk levels for each event were determined combining the likelihood and impact factors. Applying also a 0/1/3/7 severity scale to these results, a risk quantification matrix is obtained.

Table 5-7: Risk Quantification Matrix

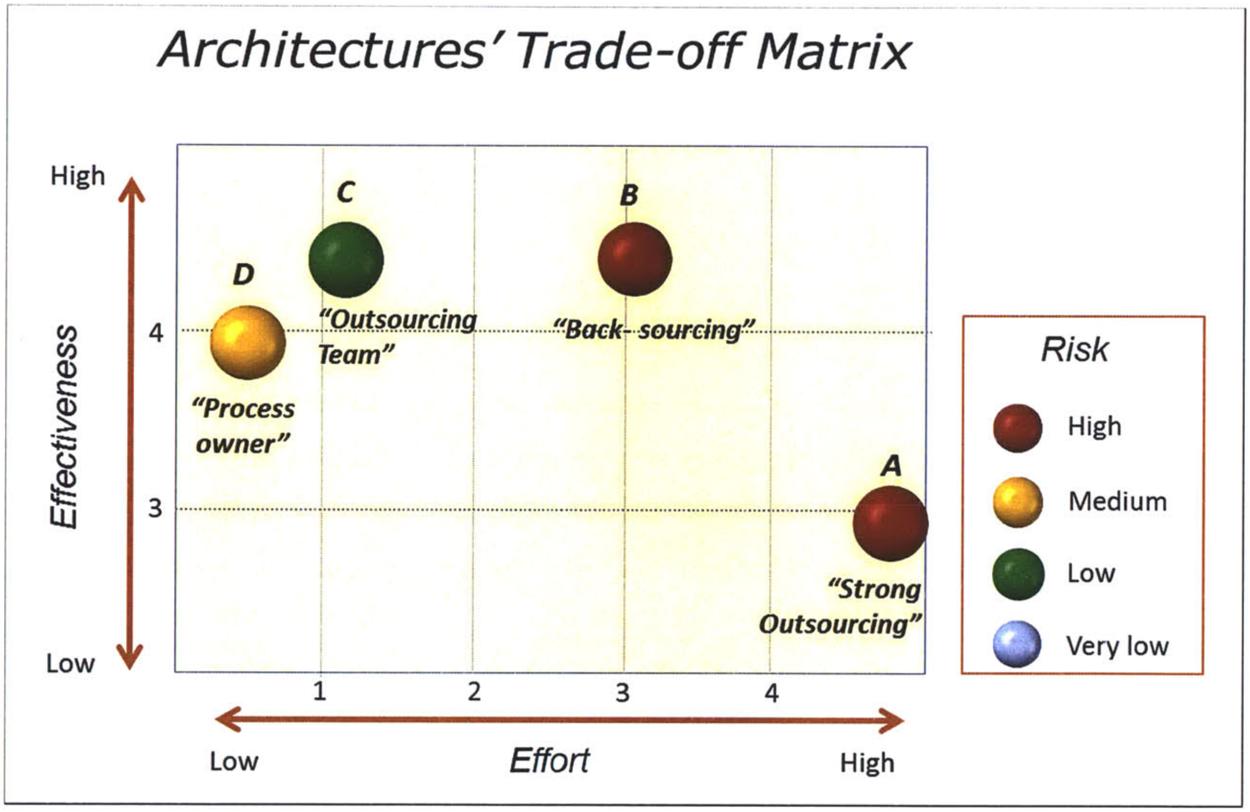
Uncertainty	Architecture A		Architecture B		Architecture C		Architecture D	
	Risk	Score	Risk	Score	Risk	Score	Risk	Score
Internal								
Implementation failure	Extreme high	7	High	3	Moderate	1	Moderate	1
New products requirements	High	3	Moderate	1	Moderate	1	Moderate	1
Intellectual property losses	Extreme high	7.	Moderate	1	High	3	High	3
External								
Variability in demand	Moderate	1	Extreme High	7	Moderate	1	High	3
Economic downturn	Moderate	1	High	3	Moderate	1	Moderate	1
Loss of supplier	Moderate	1	Low	0	Moderate	1	Moderate	1
Quantification	20		15		8		10	
Risk Profile	**High**		**High**		**Low**		**Medium**	

As can be seen from table 5.7, Architecture A 'Strong outsourcing' was ranked as the riskier option, mainly because of its difficulties to implement and its vulnerability to IP losses. lack of flexibility under variations of the external ecosystem. It was closely followed by architecture B, which lack of flexibility make it vulnerable to changes in the external landscape. Architecture C 'Outsourcing Team' was classified as *"Low Risk"* and architecture D 'Process Owner' as *"Medium Risk.*

5.7. Step 5: Outcome: Architecture Trade-off Matrix

After performing an evaluation in three different dimensions, an architecture trade-off matrix was generated to support decision making in the selection of the future architecture.

Figure 5-12: Architecture Trade-off Matrix



5.7.1. General Analysis

The matrix facilitated the analysis of the different alternatives with the top managers of ISSA. Architecture A 'Strong Outsourcing' was quickly discarded as a feasible alternative because of its higher risk and implementation difficulties. Regarding the other three options, trade-offs were considered. Architectures B and C appeared to be equally effective, but B was more risky than C. On the other hand, architecture D was less effective, but easier to implement. It had though, some scalability restriction that made more riskier than C in the long term.

5.7.2. Recommendation

Based on our multidimensional analysis and the selection criteria discussed with the Director, we recommended a two steps transformation process to ISSA: implementing first architecture D "Process Owner" and once further growth is achieved, proceed with architecture C "Outsourcing Team".

Looking only at the dimensions of effectiveness and risk, the preferred architecture should have been B "Outsourcing Team". However, ISSA was facing a challenging external and internal context, where implementation agility and the ability to generate positive results in the medium term were crucial.

For this reason, our recommendation was to proceed initially with architecture C "Process Owner", empowering the project managers in order to have an end to end responsibility and authority in the processes of outsourcing and product development. Transforming ISSA to this structure would generate short term benefits and would also prepare the way for the implementation of a more scalable architecture like Architecture C "Outsourcing Team". Therefore, this strategy would use the "Process Owner" architecture as a bridge to the final architecture "Outsourcing Team".

Finally, if an outsourcing team was to be implemented, our recommendation was also to aim to standardize the practices and procedures with other procurement teams within iSoftware. The implementation plan suggested to iSoftware is not included in this report for being out of the scope of this thesis.

6. Policy Implications, Conclusions and Future Work

As discussed in Chapter 1, businesses need to continuously transform the way they perform, but only a few of these transformations are successful. Enterprise leaders have few tools available that can help them in following a structure process for reasoning in architecting the future enterprise and in assessing the tradeoffs that exist among alternatives. This lead to the questions of *how can the process of architecting the future enterprise increases its repeatability using simple tools? How can decision makers make more informed decisions when evaluating and selecting among different enterprise architecting options?*

A mixed method research approach was used, involving literature review, interviews, expert questionnaires and a case study, to determine the validity of the core proposition of this research – that *simple engineering and management tools can be leveraged to generate a framework that will drive the process of architecting the future enterprise*.

To summarize the results, this chapter includes a discussion on the policy implications of this research, general conclusions, limitations and insights regarding potential areas of follow-on research.

6.1. Policy Implications

The policy implications of this research span across two major areas: (1) its effect on enterprises transformation planning policies and (2) its influence on the leaders' decision making process when selecting an enterprise future state.

First, one of the important policy implications of this research is the effect it can have on successful transformation planning in large enterprises. By providing practitioners a structure for reasoning on a step by step basis, the framework aims to minimize decision making errors caused by the inclination of some managers to 'cut corners' and jump into solutions too fast, without creating a broader solution space where better alternatives might exist. The framework can be used by enterprises of all types to strategize and generate better design solutions during a transformation process.

A second policy implication of this research is the influence it can have on the decision making process normally followed by enterprise leaders to plan, evaluate and select the future architecture of their organizations. The framework leverages the use of simple engineering and management tools to support managers and architects in making more informed decisions by considering a multidimensional analysis of alternatives. The quantification of qualitative factors such as effectiveness, effort and risk, allow the consideration of tradeoffs and enhance the normal practice used by managers for decision making. Ultimately, it does not only allow more informed decisions, but also enables the identification of potential issues in the planning phase, before enterprise resources are committed to the project.

.

6.2. Conclusions

This thesis has introduced a framework to guide the process of architecting the future enterprise in the context of an enterprise transformation. Using a mixed method research approach, the primary objective of this research – to generate a systematic technique that provide architects with a structure for reasoning for the process of architecting the future enterprise using simple engineering and management tools – was accomplished. Three different research stages contributed on this purpose: (i) the identification of current needs of practitioners using enterprise architecting as an instrument for planning transformations (ii) the theoretical development of a framework to architecting the future enterprise, and (iii) the application and testing of the framework.

The first stage, focused on understanding the current needs of practitioners using enterprise architecting as an instrument for planning transformations. The analysis of a proven enterprise architecting approach like the one developed by Nightingale and Rhodes (N&R) proved to be a very insightful starting point for our research. The feedback of 54 previous users of the approach allowed us to identify opportunities for improvement and provided us valuable guidance when structuring our framework. Based on our findings, we focused on addressing three issues: the need for more guidance in the candidate generation process, the need for a more complete candidate evaluation method, and improvement in the toolkit available for practitioners.

In the second stage, results from the literature review on decision making, as well as from the study of heuristics principles developed by the application of the N&R approach in previous case studies, were integrated to develop the Architecting the Future Enterprise (AFE) Framework. The framework incorporates a systems thinking approach to design future states and a multidimensional evaluation process that compares competing architectures in terms of effectiveness, effort and risk. The output of the framework enables the quantitative assessment of the tradeoffs that exists among the competing architectures in a way that is not practicable with the current approach. Therefore, the AFE framework provides a prescriptive structure of reasoning for the processes of generating, evaluating and selecting future states that complement and enhance particular aspects of the N&R approach.

Finally, the application of the AFE Framework to guide the transformation process on the iSoftware case study proved to be a valuable tool to support the processes of generating, evaluating and selecting the future state on a real case. It also provided insights to us to adjust and simplify the numbers of activities initially considered in the framework. In general terms, the use of the AFE framework in this case study resulted on a practical contribution to the practice of enterprise architecting and management decision making. A prescriptive process that highlights alternative generation and multidimensional more quantitative evaluation was followed, leading to a more informed decision when selecting the enterprise future architecture.

6.3. Research Limitations and Future Work

This section identifies the most significant limitations of this research and outlines areas of future research by which they can be reduced.

a) The case study research is limited in its representativeness. After its development, the framework was tested and adjusted using the iSoftware case study. The application of the framework in different transformation projects is needed to validate the research conclusions. One of the areas of future research is to analyze the feedback given by practitioners to identify improvement opportunities.

b) The effort model requires calibration with historical data. The effort model presented in this research is a first approach to compare implementation effort among transformation projects. Further research is needed to refine its rating scale based on historical transformation data and industry participation. This might be an interesting area of future research to provide the first calibrated cost estimation model that provides transformation effort estimation.

c) The framework provides a multidimensional static evaluation of the different alternatives. Future work should incorporate the effect of future opportunities or second layer benefits associated with each alternative, as well as to explore potential dynamic representations of the results.

7. Bibliography

Amit, R., & Schoemaker, P. (1993). Strategic Assets and Organizational Rent. *Strategic Management Journal, Vol. 14*, 33-46.

Baden-Fuller, C., & Volberda, H. (1997). Strategic Renewal in Large Complex Organizations: A Competence-based View. In A. Heene, & R. Sanchez, *Competence-based Strategic Management* (pp. 89-110). John Wiley & Sons Ltd.

Birdsye, H., & White, L. (1994). Business Transformation: The key to long-term survival and success. *International System Dynamics Conference*.

Boehm, B., Valerdi, R., Lane, J., & Brown, A. (2005). COCOMO Suite Methodology and Evolution. *The Journal of Defense Software Engineering*.

Capozzi, M. M., Dye, R., & Howe, A. (2011). *Sparking creativity in teams.* McKinsey Quarterly.

Capron, L., & Mitchell, W. (2009). Selection Capability: How Capability Gaps and Internal Social Frictions Affect Internal and External Strategic Renewal. *Organization Science 20(2)*, 294-312.

Carlson, J. M., & Doyle, J. (2000). Highly Optimized Tolerance: Robustness and Design in Complex Systems. *Physical Review Letters*, 84 (11): 2529-2532.

Collins, J. C., & Porras, J. I. (1996). Building Your Company's Vision. *Harvard Business Review*, 63-77.

de Bono, E. (1970). *Lateral Thinking: a text book of creativity.* New York: Ward Lock Educ.

De Neufville, R. e. (2004). *Uncertainty Management for Engineering Systems Planning and Design.* Cambridge: MIT, ESD Monograph.

Fraser, J., & Simkins, B. (2010). *Enterprise Risk Management.* Hoboken, NJ: John Wiley & Sons, Inc.

Garner, B. (2009). *Black's Law Dictionary.* West, 9 edition.

Gittell, J. (2009). *The Southwest Airlines Way.* New York: McGraw Hill.

Glazner, C. (2011). Enterprise Transformation using Simulation of Enterprise Architecture. *Journal of Enterprise Transformation*, 1: 231-260.

Hammond, J. S., Keeney, R. L., & Raiffa, H. (1999). *Smart Choices.* Boston: Harvard Business School Press.

Hastings, D., & Mc Manus, H. (2004). A Framework for Understanding Uncertainty and Its Mitigation and Explotation in Complex Systems. *Engineering Systems Symposium.* Cambridge, MA: MIT.

ISO. (2009b). *ISO Guide 73:2009 - Risk Management - Vocabulary.* Geneva: International Organization for Standarization.

Kazman, R., Klein, M., & Clements, P. (2000). *ATAM: Method for Architecture Evaluation.* Pittsburgh: Software Engineering Institute, Carnegie Mellon University.

Kotter, J. (1995). Why Transformation Efforts Fail. *Harvard Business Review*, 73(2), 59-67.

Maier, M. W., & Rechtin, E. (2009). *The Art of Systems Architecting.* Boca Raton: CRC Press.

Mc Manus, H., & Hastings, D. (2005, July 10 to 15). A Framework for Understanding Uncertainty and its Mitigation and Explotation in Complex Systems. *Fifteenth Annual International Symposium of the International Council On Systems Engineering (INCOSE).*

Mitchell, R., Agle, B., & Wood, D. (1997). Toward a Theory of Stakeholder Identification and Salience: Defining the Principle of Who and What Really Counts. *Academy of Management Review 22*, 853-886.

Nadler, D. A., & Tushman, M. L. (1997). *Competing by Design.* New York: Oxford University Press.

Nightingale, D. (2009). Principles of Enterprise Systems. *Second International Symposium on Engineering Systems.* Cambridge, MA.

Nightingale, D., & Rhodes, D. (2004). Enterprise systems architecting: Emerging art and science within engineering systems. *MIT Engineering Systems Symposium.* Cambridge.

Nightingale, D., & Rhodes, D. (2012). Enterprise Architecting Course (ESD.38). Cambridge, MA, MIT.

Nightingale, D., & Srinivastan, J. (2011). *Beyond the Lean Revolution: achieving succesful and sustainable enterprise transformation.* New York: American Management Association (AMACOM).

Nightingale, D., Stanke, A., & Bryan, T. (2008). *Enterprise Strategic Analysis and Transformation 2.0.* Cambridege, MA: Lean Advancement Initiative, Massachusetts Institute of Technology.

Porter, M. (1985). *Competitive Advantage: Creating and Sustaining Superior Performance.* New York: Free Press.

Purchase, V., Parry, G., Valerdi, R., Nightingale, D., & Mills, J. (2011). Enterprise Transformation: Why are we interested, what is it, and what are the challenges? *Journal of Enterprise Transformation*, 1:14-33.

Ross, A., & Rhodes, D. (June 2008). Using natural value-centric time scales for conceptualizing system timelines through epoch-era analysis. *INCOSE International Symposium.* Utrecht, the Netherlands.

Rouse, W. B. (2011). Necessary Competencies for Transforming an Enterprise. *Journal of Enterprise Transformation*, Volume 1: 71-92.

Saha, P. (2006). A Real Options Perspective to Enterprise Architecting as an Investment Activity. *Journal of Enterprise Architecting*, Vol. 2. No. 3, 32-52.

Santos, F., & Eisenhardt, K. (2005). Organizational Boundaries and Theories of Organization. *Organization Science 16(5)*, 491-508.

Sheffi, Y. (2007). *The Resilient Enterprise: Overcoming Vulnerability for Competitive Advantage.* Cambridge, Massachusetts: The MIT Press.

Sheffi, Y., Vakil, B., & Griffin, T. (2012). Risk and Disruptions: New Software Tools. *Supply Chain Management Review*, Executive Newsletter April 18th.

Sterman, J. D. (2000). *Business Dynamics: Systems Thinking and Modeling for a Complex World.* McGraw-Hill.

Teece, D. J., Pisano, G., & Amy, S. (1997). Dynamic Capabilities and Strategic Management. *Strategic Management Journal 18*, 509-533.

Valerdi, Nightingale, & Blackburn. (2008). Enterprises as Systems: Context, boundaries and practical implications. *Journal Information-Knowledge-Systems Management*, Vol. 7 Issue 4.

Valerdi, R. (2005). *The Constructive Systems Engineering Cost Model (COSYSMO).* University of Southern California.

Varledi, R. (2011). Heuristics for Sytems Engineering Cost Estimation. *IEEE Systems Journal*, 91-98.

Von Hippel, E. (2005). *Democratizing Innovation.* Cambridge: MIT Press.

Womack, J. P., Jones, D. T., & Roos, D. (1991). *The Machine that Changed the World.* New York: Harper Perennial

Zini, F. (2012). How Do Senior Leaders Conceive and Re-architect their Enterprises? Cambridge, MA: Unpublished Master's Thesis, Massachusetts Institute of Technology.

Appendix A: Reference Level Definition for Effort Drivers

For simplicity and easiness to apply, the reference rating levels for each cost driver are defined in this appendix as Low, Nominal and High. The levels for Vey Low and Very High can be inferred from these definitions.

A. Effort Drivers Related to People:

- ***Leadership Support*** Refers to the managers' support for the implementation of the proposed architecture. Primary sources of added transformation effort are motivation, culture compatibility and familiarity with the architecture.

Low (more effort)	Nominal	High (less effort)
Low motivation and sense of urgency to generate the change	*Willing to generate the proposed change*	*High motivation and sense of urgency to generate the change*
Implies a change in culture paradigm	*Compatible with enterprise culture*	*Strong alignment with enterprise culture*
Minimal familiarity/ trust in the EA	*Reasonable familiarity/ trust in the EA*	*Strong familiarity /trust of the EA*

- ***Employee Acceptance:*** Refers to the employee support and willingness to make short term sacrifices for the implementation of the architecture. Primary sources of added transformation effort are job losses, trust in the EA, culture compatibility and growth opportunities.

Low (more effort)	Nominal	High (less effort)
Many job losses and layoffs	*Some job losses and layoffs*	*Minimal job losses and layoffs*
Lack of believe that the proposed EA is useful and possible.	*Some believe that the proposed EA is useful and possible.*	*Strong believe that the proposed EA is useful and possible.*
Lack of perception of growth opportunities	*Moderate perception of growth opportunities*	*Perception of growth opportunities*
Implies a change in culture paradigm	*Compatible with enterprise culture*	*Strong alignment with enterprise culture*

- ***Staff capability:*** Refers to the level of experience, familiarity or capability of the staff to implement the proposed architecture. Primary sources of added transformation effort are experience and competency with the architecture.

Low (more effort)	*Nominal*	*High (less effort)*
Limited competency to implement the proposed EA.	*Reasonable competency to implement the proposed EA.*	*High competency to implement the proposed EA.*
No familiarity or previous experience with a similar EA.	*Some familiarity or previous experience with a similar EA.*	*Strong familiarity or previous experience with a similar EA.*

B. <u>Effort Drivers related to Complexity</u>*:*

- ***Understanding Complexity:*** Refers to the level of understanding of the new architecture by all the internal stakeholders of the enterprise. Primary sources of added transformation effort are if it is an unprecedented architecture and its difficulty to communicate.

Low (less effort)	*Nominal*	*High (more effort)*
Easy to communicate (ref: less than 5 minutes)	*Reasonable easy to communicate*	*Takes time to communicate (ref: more than 5 minutes)*
Previous experience with a similar EA.	*Some familiarity or used by other BU.*	*Unprecedented in the organization*

- ***Level of change required:*** Refers to the amount of change required in the current structure of the enterprise for the implementation of the new architecture. Primary sources of added transformation effort are people involved, policy/ procedures change and internal structure obstacles.

Low (less effort)	Nominal	High (more effort)
Minimal people movements	*Some change in organizational structure*	*High amount of change in organizational structure*
Minimal change procedures/ policies	*Some change procedures/ policies*	*High amount of change in procedures and policies*
Minimal structural obstacles	*Some structural obstacles*	*Requires removal of "elephants"*

- ***Architecture Complexity:*** Refers to the level of complexity in design of the proposed architecture. Primary sources of added transformation effort are the level of interdependencies in the functions and amount of coordination needed.

Low (less effort)	Nominal	High (more effort)
Some vertical and horizontal coordination	*More complex interdependencies and coordination*	*Complex interdependencies and offsite coordination needed*

C. <u>Effort Drivers related to Operations</u>*:*

- ***Schedule Constraints:*** Refers to the time constrains of the proposed architecture. Primary sources of added transformation effort are extended time needed to implement the change, and required time needed to show results.

Low (less effort)	Nominal	High (more effort)
Less than 12 months to implement	*12-24 months to implement*	*More than 24 months to implement*
Able to generate short term 'wins'	*Reasonable time needed to show results*	*Long time needed to show results*

- ***People investment:*** Refers to the upstream investment in human resources needed to implement the change. Primary sources of added transformation effort are training and hiring/firing expenses.

Low (less effort)	Nominal	High (more effort)
Low investment in training and hiring/ firing expenses.	*Reasonable investment in training and hiring/ firing expenses.*	*Considerable investment in training and hiring/ firing expenses.*

- ***Infrastructure investment:*** Refers to the upstream investment in facilities, equipment or technologies needed to implement change. Primary sources of added transformation effort are facilities, equipment and technologies.

Low (less effort)	Nominal	High (more effort)
Low investment in facilities, equipment or technologies.	*Reasonable investment in facilities, equipment or technologies*	*Considerable investment in facilities, equipment or technologies*

D. <u>Effort Drivers related to Technology</u>*:*

- ***Technology Maturity:*** *Refers to whether the new architecture (in particular the information system) involves well known and mature methods.*

Low (less effort)	Nominal	High (more effort)
Proven methods and tools (widely used in industry)	*Reasonable mature methods and tools*	*New methods/ tools (pilot use)*

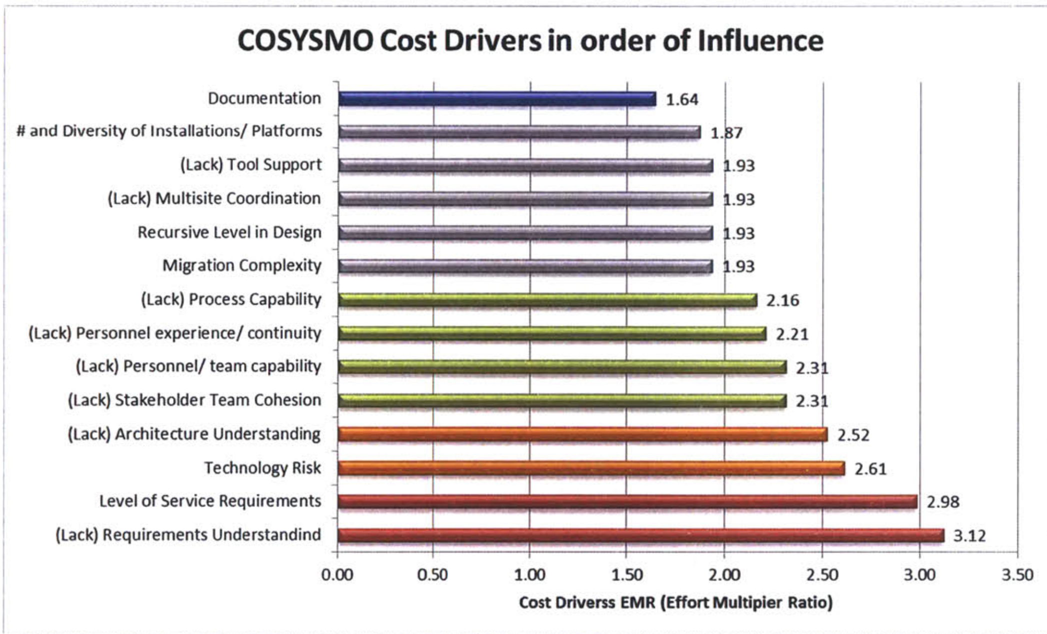
- ***Integration Complexity:*** *Refers to the extent to which the new technologies/ tools may affect the implementation of a new architecture due to integration difficulties.*

Low (less effort)	Nominal	High (more effort)
No integration complexity	*Minimal integration complexity*	*High integration complexity*

Appendix B: Determination of Rating Scales

COSYSMO, the Constructive Systems Engineering Cost Model, is a validated parametric model that helps people reason about their decisions related to system engineering (Valerdi R. , 2005). Among its internal parameters, the model contains fourteen effort multipliers that were calibrated through expert data collected through the Delphi method. COSYSMO's fourteen cost drivers and their respective rating scale in terms of influence are shown in the table 10.1.

Table B-0-1: COSYSMO cost drivers in order of influence



Based on these results, we grouped the COSYSMO's cost drivers in five categories: Extremely, very high, high, moderately high and slightly high influential drivers. Within these groups we calculated the average individual values for their applicable rating scales as shown in table 10.2. We extrapolated those calibrated results to the determination of the transformation effort cost drivers.

Table B-0-2: Average rating scale for the five effort driver categories

Extremely Influential Cosymo Cost Driver	EMR	Very Low	Low	Nominal	High	Very High
(Lack) Requirements Understandind	3.12	0.60	0.77	1.00	1.37	1.87
Level of Service Requirements	2.98	0.62	0.79	1.00	1.36	1.85
Average Rating Scale		**0.61**	**0.78**	**1.00**	**1.37**	**1.86**

Very High Influential Cosymo Cost Driver	EMR	Very Low	Low	Nominal	High	Very High
Technology Risk	2.61	0.67	0.82	1.00	1.32	1.75
(Lack) Architecture Understanding	2.52	0.65	0.81	1.00	1.28	1.64
Average Rating Scale		**0.66**	**0.82**	**1.00**	**1.30**	**1.70**

High Influential Cosymo Cost Driver	EMR	Very Low	Low	Nominal	High	Very High
(Lack) Stakeholder Team Cohesion	2.31	0.65	0.81	1.00	1.22	1.50
(Lack) Personnel/ team capability	2.31	0.65	0.81	1.00	1.22	1.50
(Lack) Personnel experience/ continuity	2.21	0.67	0.82	1.00	1.22	1.48
(Lack) Process Capability	2.16	0.68	0.82	1.00	1.21	1.47
Average Rating Scale		**0.66**	**0.82**	**1.00**	**1.22**	**1.49**

Moderately High Influential Cosymo Cost Driver	EMR	Very Low	Low	Nominal	High	Very High
Migration Complexity*	1.93	-	-	1.00	1.25	1.55
Recursive Level in Design	1.93	0.76	0.87	1.00	1.21	1.47
(Lack) Multisite Coordination	1.93	0.80	0.90	1.00	1.18	1.39
(Lack) Tool Support	1.93	0.72	0.85	1.00	1.18	1.39
# and Diversity of Installations/ Platforms*	1.87	-	-	1.00	1.23	1.52
Average Rating Scale		**0.76**	**0.88**	**1.00**	**1.19**	**1.42**

Slighty High Influential Cosymo Cost Driver	EMR	Very Low	Low	Nominal	High	Very High
Documentation	1.64	0.78	0.88	1.00	1.13	1.28
Average Rating Scale		**0.78**	**0.88**	**1.00**	**1.13**	**1.28**

Appendix C: Survey Rating Effort Cost Drivers

Name	
Organization	
Field of Project/ transformation Experience	

The following is a list of the 11 drivers that we believe, based on literature review and practical experiences, influence positively or negatively in the effort needed to implement an architectural change in an organization. However, some of them have a higher impact on effort to change than others. Based on your experience, please rate them on a scale of 1 (Slight impact on effort to change) to 5 (Extremely high impact on effort to change) to show how you believe these drivers **impact in the effort** needed to implement an architectural change in an organization.

Drivers related to People

Driver	Definition	Slight Impact on effort	Moderately impact on effort	High impact on effort	Very high impact on effort	Extremely high impact on effort
Leadership Support:	Refers to the managers' support for the implementation of the proposed architecture. Factors of this driver are: motivation/ urgency for the change, culture compatibility and familiarity with the architecture.					
Employee Support:	Refers to the employee support and willingness to make short term sacrifices for the implementation of the architecture. Primary sources of added transformation effort are: Job losses, lack of trust/ believe in the proposed EA, lack of growth possibilities, or culture misalignments.					
Staff capability:	Refers to the level of experience, familiarity or capability of the staff to implement the proposed architecture.					

Drivers related to complexity

Driver	Definition	Slight Impact on effort	Moderately impact on effort	High impact on effort	Very high impact on effort	Extremely high impact on effort
Number of Stakeholders involved	Refers to the number and diversity of stakeholders needed to agree on the proposed architecture and its structure.					
Level of change required	Refers to the amount of change required in the current structure of the enterprise for the implementation of the new architecture. Primary sources of added effort are people that change functions, people involved in the transformation, existence of new procedures/ policies, structural obstacles that have to be overcome.					
Architecture complexity	Refers to the level of complexity in design of the proposed architecture. Factors that influence this driver are the level of interdependencies in the functions and the amount of coordination needed among the different levels.					

Drivers related to operations

Driver	Definition	Slight Impact on effort	Moderately impact on effort	High impact on effort	Very high impact on effort	Extremely high impact on effort
Schedule constrains	Refers to the time constrains of the proposed change. Factors that influence this driver include time needed to implement the change and required time needed to show results.					
People investments	Refers to the upstream investment in human resources needed to implement the change. Factors that influence this driver are training and hiring/ firing expenses required.					
Infrastructure investment	Refers to the upstream investment in facilities, equipment or technologies needed to implement the change.					

Technical Drivers

Driver	Definition	Slightly High	Moderately High	High	Very High	Extremely High
Technology maturity	Refers to whether the new architecture (in particular the information system) involves well known and mature methods.					
Integration complexity	Refers to the extent to which the new technologies/ tools may affect the implementation of a new architecture due to integration difficulties.					