MASTER IN COMPUTER SCIENCE

A model-based methodology for early stages of ERP implementations in SMEs

Mordant, Alain

Award date:
2008

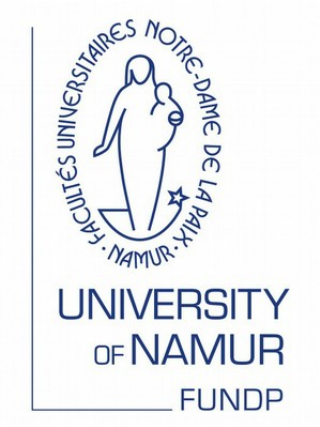
Awarding institution:
University of Namur

University of Namur
Faculty of Computer Science

A model-based methodology for early stages of ERP implementations in SMEs

Alain Mordant

2007-2008



A Thesis Presented to the Faculty of Computer Science
in Partial Fulfillment of the Requirements for the Degree
Master of Computer Science

Faculty of Computer Science - rue Grandgagnage 21, B-5000 Namur, Belgium
Tel. : +32 (0)81/72.50.02 - Fax : +32 (0)81/72.49.67

Abstract

Enterprise Resource Planning (ERP) software implementation in Small or Medium Enterprises (SMEs) is a critical challenge, implying a lot of resources and financial costs. ERP implementations in SMEs involves the whole organization at various levels, and impact many if not all business processes of the enterprise. The decision to adopt such software must be done wisely, and the implementation conducted carefully. Ensuring a clear and common vision of the entire organization is of paramount importance for both the ERP consulting team and the enterprise stakeholders at the very beginning of the project.

This thesis provides an extensive state of the art on ERP software and ERP implementations. It presents a unified life-cycle, outlines Critical Success Factors (CSFs) for a successful implementation, and proposes some improvements to the current modeling methodologies. The earlier stages of the proposed methodology are detailed and applied to a case study. Modeling tasks during these stages, mostly goal and business process modeling, are designed to ensure a safer ERP implementation, and ease the understanding of both the objectives of the enterprise, and the business processes to be covered by the solution.

Keywords: ERP, implementation, methodology, SME, life-cycle, CSF, modeling.

Résumé

L'implémentation d'un Progiciel de Gestion Intégré (PGI, ou ERP en anglais) est un défi majeur pour une Petite ou Moyenne Entreprise (PME), impliquant de lourds coûts financiers et de grands besoins en termes de ressources matérielles ou humaines. L'implémentation d'un ERP au sein d'une PME concerne l'organisation dans son entièreté à différents niveaux, et a un impact sur de nombreux processus business de l'entreprise, voire même sur l'entièreté des processus. La décision d'adopter un ERP doit être mûrement réfléchie, et l'implémentation menée prudemment. Il est particulièrement important de s'assurer une vision claire de l'organisation dans son entièreté, commune à l'équipe de consultance et aux membres de l'entreprise qui vont participer directement au projet, dès le début de celui-ci.

Ce mémoire fournit un état de l'art détaillé sur les logiciels ERP et leurs implémentations. Il présente ensuite un cycle de vie unifié, retrace les Facteurs Critiques de Succès (FCS) pour une implémentation réussie, et propose plusieurs améliorations aux méthodologies de modélisation actuelles. Les premières étapes de cette méthodologie sont ensuite détaillées et appliquées à un cas d'étude. Les tâches de modélisation durant ces étapes sont destinées à assurer une implémentation d'ERP plus sûre, et à faciliter la compréhension des objectifs moteurs du projet, ainsi que des processus business qui devront être couverts par la solution.

Mots-clefs : ERP, implémentation, méthodologie, PME, cycle de vie, FCS, modélisation.

Acknowledgments

I would like to thank my supervisor, Michaël Petit, for his continuous help and availability. His valuable advices and guidance throughout the writing of the thesis helped me to improve the global quality of this work, and he constantly provided me with insightful directions and useful resources.

I would also like to express my gratitude to Claude and Stéphane Hanrez, who gave me the opportunity to work on such an interesting subject, for their welcome and assistance during the internship, and their contribution with the SAP materials.

A very special thanks goes my family for their support and encouragement during my entire studies.

Contents

List of Figures

List of Tables

Introduction

More and more companies tend to automate and streamline their business processes with the support of information technology. Computer software aimed at supporting enterprises operations have been evolving since the early days, continuously trying to reduce the repetitive tasks, improve the productivity, and facilitate the users daily work.

Nowadays, companies willing to improve their information technology support can adopt Enterprise Resource Planning software (ERP) to cover most aspects of their business processes, instead of dealing with different software for every business department. Indeed, ERP software are made to support and integrate many business process of an enterprise. ERP software, as a subset of Commercial, off-the-shelf software (COTS), are provided by the vendor with default functionalities, and must be tailored to match the enterprise business processes. However, this customization step is critical, and should be carefully conducted, implying a high cost for the enterprise if the ERP software does not suit the processes by default.

From large enterprises to SMEs

Most large enterprises are now equipped with ERP software, thus the ERP vendors developed different versions of their solutions, aimed at small and medium enterprises (SMEs). However, a lot of customization costs and problems were tied to ERP implementations among large enterprises in the past, added to some disastrous stories where failed implementations led companies to bankruptcy. The decision to adopt a new ERP software solution in a SME must be carefully planned by the top management, to avoid unexpected problems and minimize the customization tasks wherever it is possible. SMEs are indeed organizational structures particularly vulnerable to wrong investments, due to their higher resource limitations.

All ERP vendors come with their own methods, best practices, and recommendations to implement their solution in the enterprise. On another hand, a lot of academic researches and studies have been made on the ERP implementation topic, but there is a lack of generic and objective methods to determine the mapping of the ERP solutions to the requirements, goals, processes or data of the enterprise. Such determination is of high importance however, especially in SMEs since their resources are limited, although they are likely to encounter the same inadequacy problems that large enterprises have been facing during the last several years.

Typical ERP implementation process

ERP software are implemented in enterprises following predefined vendor-specific methodologies, according to a life-cycle composed of a varying number of stages, suggested by the vendor. During this whole implementation, the most critical part, which always needs to be done, is the alignment between the software solution and the enterprise particular situation.

ERP software are indeed provided as generic packages, which require to be tailored to the enterprise particular needs and constraints, to become the new enterprise IT system. The entire implementation process is supported by methodologies and tools, addressing various modeling aspects of the implementation. Figure 1 illustrates the typical ERP implementation process.

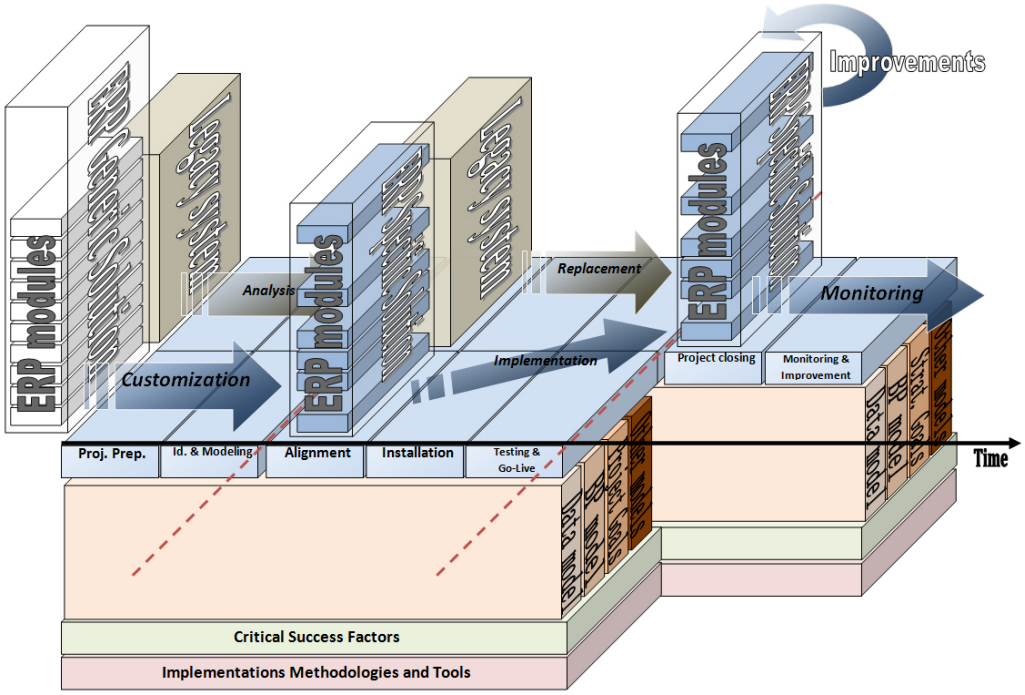


Figure 1: Typical ERP implementation process

A typical ERP software implementation can be viewed as a staged process, supported by particular methodologies and tools, and observing various critical success factors (CSFs) to ensure an easier and safer implementation. A subset of ERP modules available for the ERP solution are initially selected to be implemented in the enterprise. These modules then need to be customized, according to a deep analysis of the enterprise, through various modeling perspectives. The modules resulting from this customization process then form a specific instance of the ERP software solution, tailored to the enterprise considered. The legacy system is replaced by the new system afterward, and will be continuously monitored later, for further improvements.

The challenge of modeling

Modeling techniques are supposed to ease the implementation, and useful to reduce such risks of inadequacy and uncertainty tied to ERP implementation projects. Various specific techniques and tools are available on the vendor's side, to help determining the adequacy of their solution for every new possible customer, and several modeling techniques become useful to ensure a successful ERP implementation. The use of modeling techniques among ERP implementations in SMEs is largely a matter of compromises, even more than in large enterprises, due to the stronger resource limitations mentioned earlier. The most challenging part is to find the advantages and disadvantages of every modeling aspect, while staying aware of the limitations on both the enterprise and modeling techniques constraints. The goal is to find thereafter the best compromise between maximizing the usefulness of the modeling techniques, while wasting the less possible time and resources.

Implementing ERP software can imply several modeling perspectives along the entire life-cycle. A large number of publications have also been realized around the CSFs topic. The CSFs can be assimilated to guidelines, outlining several characteristics, under various perspectives, to ensure a successful ERP implementation.

Research conducted in this thesis

Two problems are addressed in this thesis. The first goal of the thesis is to achieve a clear and accurate vision of ERP implementation related topics. Such an accurate overview of the existing concepts and methodologies is needed before trying to add or improve any aspect of the methodologies. The second goal is to investigate the currently available methodologies, and try to determine if they can be improved through modeling tasks, and if so, in which ways.

This thesis is thus divided into two main parts. The first part proposes a detailed state of the art, around various ERP implementation topics. The meaning of the generic term *ERP*, its origins and main characteristics are investigated, as well the various research topics addressed about ERP implementations. The various life-cycle structures for ERP implementations available in the literature, as well as several reference works on CSFs are also analyzed. The second part of the thesis proposes an integrated methodology, based on the observations from the state of the art, which synthesizes the previous findings and tries to improve the existing methodologies under various modeling perspectives.

The research methodology to find relevant information for the state of the art will be to use several online repositories for high-quality publications, such as the IEEE, Emerald, Aisnet, Elsevier, Sciencedirect or Springer, by using keywords such as 'ERP', 'implementation', 'methodology', 'modeling', and other keywords that could be found to be relevant during the analysis. Google will also be used to find relevant articles describing the ERP situation in the market and the major ERP software vendors. An analysis of the produced state of the art will then be conducted to outline synthesized findings on every topic. A unified life-cycle will be drawn, along with critical success factors for successful ERP implementations. Then the first stages of the methodology will be analyzed in more details, providing modeling improvements for the beginning of the project.

The focus about modeling improvements will mostly be oriented toward goal modeling and business process modeling, as these ones were neglected in the commercial methodology available for this work, namely the *'SAP Business One Accelerated Implementation Program'*. The proposed improvements will be drawn from observations and other modeling techniques reviewed in the state of the art. Every proposed modeling improvement will use a specific language, detailed in the corresponding section.

Contents of this work

The rest of this document is organized as follows : the first part presents a detailed state of the art of various aspects related to ERP implementations. The two first sections present the context of ERP software, by first addressing its origins, characteristics and major vendors. Section 3 presents the various life-cycles, followed by section 4 with the critical success factors. The last sections of the state of the art address the modeling and performance concerns, respectively in sections 5 and 6.

The second part of the thesis presents the proposed methodology and improvements based on the synthesis of the state of the art. Section 7 reviews the existing life-cycles and proposes a unified life-cycle. Section 8 reviews the critical success factors. Section 9 presents general conclusions about modeling concerns, and sections 10 and 11 address the concrete modeling improvements for the first part of an ERP implementation project. Section 12 then presents an illustration example of the proposed methodology for the first stages of ERP implementation projects.

This document is finally concluded in section 13, with a discussion about the advantages of the proposed methodology, the limitations of this thesis, and tracks for future research.

Part I

ERP Implementation Methodologies : a State of the Art.

Section 1 :

The origins of ERP software

> **Content :** *This section will explain where the term ERP comes from, and try to define the main features of ERP software. Finally we address the position of this kind of software in context with COTS software in general.*

1.1 A definition of ERP

There are many different definitions for the term *ERP* but there is however little consensus around it. A definition from the encyclopedia of production and manufacturing management says that ERP is *'a term associated with a multi-module software for managing and controlling a broad set of activities that helps a manufacturer or other businesses' [SWA04]. 'Typically, Enterprise Resource Planning (ERP) are software packages composed of several modules providing cross-organization integration of data through embedded business processes' [EST99].* Klaus, Rosemann and Gable expose in [KLA2K] the difficulties to come to a broadly agreed definition of ERP systems, and present the various existing definitions and meanings behind the term *'ERP'*.

Usually, *'ERP'* is a term designating a class of packaged application providing integration of data throughout the entire set of business processes of the enterprise, and customizable to suit the enterprise needs. It allows the enterprise to seamlessly integrate, manage and control each business process. Another particularity of ERP software is that data must be entered only once in the system, and automatically made available throughout the whole enterprise system. High functionality is also *'one of the main differentiators of ERP' [KLA2K]*, as these software packages are meant to support most, if not all the business functions of an enterprise.

1.2 From MRP to ERP

ERP software derived from material requirements planning (MRP), then Manufacturing Resource Planning (MRP-II) [KLA2K, SWA04, POR06]. Material resource planning software (MRP) were basically automated systems calculating the materials needed in stock, in order to produce a certain quantity of finished goods, based on the master production schedule (MPS).

Around the early 1980s, MRP software gained quite a good reputation throughout production companies, and then were expanded to include more functionalities. MRP became company-wide systems, used to plan and control almost every resource of an organization. Because it

became so different, a new term emerged : *manufacturing resource planning.* (MRP-II)
MRP-II continued to evolve and became enterprise resource planning (ERP), as it was first called so by the Gartner Group of Stamford, Connecticut, USA. [POR06]

Enterprise Resource Planning software are business software packages that support daily business operations and decision making. These packages can integrate and automate basically all the processes of a company.[POR06]

The ERP software is meant to handle the data of every possible business unit of the company, such as financial and accounting, human resource, manufacturing, supply chain, to customer relationship information. It provides the employees with all the necessary data to manage and control the business processes all along the processes chain, right to the final delivery to the customer. It *'allows a quick flow of information through the supply chain and provides centralized or de-centralized accounting functions with the ability to do activity-based costing' [SWA04].*

1.3 ERP systems as a subset of COTS systems

ERP software are not developed from scratch in organizations, these software consist in a subset of what is more usually called the commercial off-the-shelf (COTS) software.

These are packaged solutions proposed by different vendors, and basically sold *as-is.* COTS software have precise functionalities, and are provided by an external company. Because each organization is different, the amount of customization will vary for each implementation project, to be the most accurate to the business processes of the enterprise.

COTS software must not be confused with reuse software. While COTS software and reuse software share similar benefits and risks factors, COTS software components are more a subset of reuse software components [ABT97]. The main differences between the two types of software are :

- reuse components are not necessarily able to operate as stand-alone entities.
- reuse software is generally acquired internally within the software developing organization.
- reuse software usually requires access to the source code, whereas COTS components access to the source code is rare.

ERP software is a COTS software in itself, because it is composed of various modules interacting with each other to integrate the entire process chain of the enterprise. But it is generally provided without any access to the source code, published by an external vendor and capable of operating as a stand-alone entity. As a result of ERP software being a subset of COTS software, generic off-the-shelf requirements engineering processes can be used as a basis to elicit the requirements of ERP software, as seen in [DAN03]

Section 2 :

ERP Software Typicals

> **Content :** *In the preceding section, we addressed the origins and historical context of ERP software solutions. This chapter will be oriented towards the existing solutions nowadays, their most commonly implemented modules, why ERP software can help enterprises and how. It will also present the major ERP vendors in the market.*

2.1 ERP typical modules and functions

This section will address the most commonly seen modules (or business functions) in ERP software, as well as some special functionalities often implemented to enhance the business processes in these application domains.

Many ERP software vendors decline their solutions in modules to elicit the various supported functions. SAP, Compiere, or Microsoft for example, use that approach to present their ERP software solutions. On another hand, some other major vendors prefer to define their ERP solutions as a set of various applications which can be integrated with each other, letting the user choose which departments need the help of the software, thus avoiding a typical *One Size Fits All* solution. Infor Global Solutions is one of such vendors [EI08a].

Figure 2.1: Compiere Capabilities, as shown in [COM08b]

ERP software can be used to support many processes in the enterprise, but one of the problems to differentiate the various offers is that the terms used to define these processes vary

from one vendor to another. there is no global terminology to name the modules or business functions supported. An attempt et presenting this problem is illustrated in Appendix A, where the software solution SAP Business One (SAP BO) is opposed to Compiere and Microsoft NAV.

For example, the process of procurement, which *'covers the business process used for creating purchase orders, processing invoices received from vendors, and generating payments' [COM08b]*, takes three different names in these solutions. Only one difference is really noticeable about these main modules : the lack of project management support for SAP BO. Otherwise all the main processes such as Financial Management, Human Resources Management, Sales, Purchase or Production are supported by the various solutions, under different names.

2.2 Various architectures for ERP software

About the various types of architectures for ERP software, two main possibilities are available. Either on a platform independent (or dependent in some cases) client application or, following the growing philosophy of Service-Oriented Architecture (SOA), stored on a web space, simplifying the possible installation or architecture problems in the customer's enterprise. For example, Compiere proposes the two solutions for its ERP software [COM08a], while SAP proposes one client-server based solution (namely SAP Business One) and one web-based solution (SAP ByDesign) for SMEs.

In [BRE05], the authors go even further about the ERP architecture and propose a distributed architecture based on web services and peer-to-peer network technology. Starting by pointing out the flaws of the conventional solutions, such as a high customization, heavy hardware costs, or limitations in functionalities for *Mini-ERP systems*, such as SAP Business One [BRE05], they come to the conclusions that classical ERP solutions are inappropriate, and propose a new concept, not only based on web services, which are commonly used now in Software as a Service (SaaS) ERP solutions, but also on peer-to-peer technology. This would give such ERP software some important properties such as components being both servers and clients, Scalability, autonomy and decentralization, which would drastically lower the total cost of ownership for SMEs.

Except for the 'peer-to-peer' aspect, Application Service Provider (ASP) or Software as a Service (SaaS) exist on the market and are more and more widespread among SMEs. The main characteristics of SaaS ERP solutions are the access to the various modules through an internet connection and web-based interface. This allows the SMEs to pay *on-demand* and not for a software which would only be used at a third of its capabilities. Furthermore, it simplifies the implementation on a technological approach, both on the hardware and security costs, because the components and server are hosted by the vendor, seriously lowering the total cost of ownership for the customer [EI08c]. Many vendors also decline their solution in SaaS nowadays. For example SAP ByDesign is the SaaS ERP software solution from SAP.

2.3 Major ERP Software Vendors

Listing all the ERP vendors would go beyond the scope of the thesis. Instead, each following subsection will present one of the major ERP vendor on the market, and one of its main solutions.

2.3.1 SAP

According to the main website of the company, *Systems Applications and Products in Data Processing* (the former name of SAP) was founded in 1972 by five former IBM employees. [SAP07] Their first objective was to develop standard application software for real-time business processing. Their first release was mostly based on what would become later their Financial Accounting module, and called the *R/1 System*. About ten years later, the *R/2 system* is created and SAP continues to grow rapidly, to finally change its name to *SAP AG*, which is the present name of the company.

For more than ten years, SAP has been leading the ERP market for Large Enterprises with its SAP R/3 ERP solution, which is now called SAP Business Suite. Nowadays, SAP tries to invest the SMEs market share, by developing new ERP products such as SAP Business One, their first solution designed specifically for SMEs, or SAP Business All-in-One which could be defined as the intermediary solution between medium and big enterprises, and more recently SAP Business ByDesign, their Software as a Service (SaaS) ERP solution.

2.3.2 Oracle

Oracle is the most important competitor against SAP as a world leader in ERP software solutions. Not only they have a strong experience in database management systems, but they also make several acquisitions every year in order to increase their market share. One of the most noticeable being the acquisition of PeopleSoft in January, 2005, which itself had acquired J.D. Edwards the year before, both being active ERP vendors among the market[ORACLE08].

The most important ERP solution from Oracle is the Oracle E-business Suite,which is composed of nearly 150 modules, and has been declined in a version for SMEs at the beginning of the year 2003. They also have their hosted solution, named *Oracle On Demand*, and propose a methodology for faster implementation, named Oracle Accelerate.

2.3.3 Infor Global Solution

Infor Global Solutions, created in 2002, acquired several companies, including Baan (former important ERP vendor), and is now one of the most important ERP vendors, actually the third behind SAP and Oracle [EW08b]. Infor global solution offers several software line of products instead of offering one or two major solutions. Beside the usual ERP products, like for

example *Infor ERP LN*, Infor offers specific software solutions for Enterprise Asset Management (EAM), Customer Relationship Management (CRM), Financial Management System (FMS), just to mention some of it.

Several ERP solutions are proposed on the Infor website, going from their most general, largest solution, namely *Infor ERP LN*, to more industry-specific software such as *Infor ERP TRANS4M*, for automotive manufacturers, or *Infor ERP Adage* designed specifically to meet the needs of process industries such as Food and Beverage, Chemical, or Pharmaceutical for example [INF08]. As a result, Infor Global Solutions offers several verticalized solutions, with pre-customized solutions, and shorter implementation times [EI08a].

2.3.4 Microsoft Dynamics

Microsoft Dynamics, formerly Microsoft Business Solutions, is the line of software for business solutions by Microsoft. Their main advantage is the high integration possibilities with other Microsoft software, such as the Office Suite, mail clients, or even the operating system itself. They propose Financial management systems solutions with Microsoft Dynamics AX, CRM solutions with Microsoft Dynamics CRM, and ERP software solutions with Microsoft Dynamics NAV (formerly named Navision, from the name of the original editor which was bought back by Microsoft [EI08b]) [MSD08].

2.3.5 Open Source Software - Compiere

There is a list of several hundreds of open source solutions available on the market, but this document can not detail all of them. The interested reader could refer to [OEG08] for an exhaustive list of Open Source ERP software solutions. Among the most famous open source ERP software, we can cite Compiere, Openbravo, Opentaps, Adempiere, TinyERP, Postbooks, xTuple, or Neogia.

The most famous among them is Compiere, even if, on a strict open source point of view, the software has some detractors, because it relies on the Oracle database – which is not open source – and in some sort lost its contact with the community around it [EW08a]. This led to the creation of a spin-off ERP software solution : *Adempiere*, which is meant to be more community-driven and open source-oriented than its former version. The first actions being to enable an open source database support, namely Postgresql, and build a fresh, new open source community around it.

Compiere, on the other hand, has successfully managed to gather a large community, and to establish a strong network of implementation partners world-wide. Anyway, due to the problem stated previously, Compiere is more now *a hybrid software between open source and commercial products* than a strict open source software [OEG08].

Section 3 :

Life-cycle of an ERP project

> **Content :** *Before describing the main concerns for every stage of an ERP implementation project, the various definitions of the life-cycles must be addressed. This chapter will cover the main representations for ERP implementations projects, and the several stages composing the life-cycles on both the academic and the commercial point-of-view, through the analysis of the academic literature, and the implementation program for SAP Business One.*

While numerous researches have been made about unification on other topics, such as the ERP implementations Critical Success Factors (CSFs) for example, little has been done to unify the implementation methodologies used in ERP projects and the several stages composing the life-cycle. Instead, each vendor or consultancy enterprise comes with its own strategy, with various criteria.[HEC04]

These methodologies divide the life-cycle of an ERP software in several steps going from the very first beginning to the end of the project, but even the project definitions, beginning and ending milestones vary for every life-cycle. The differences are many : when is the beginning, before the choice of the future software or once the software is chosen ? How many steps can be elicitated, and when can the project be considered as *achieved* ? Before analyzing deeper the various phases of the implementation and their composition, it is important to have a global vision of the life-cycle of an ERP implementation project. The rest of this chapter will present the various ERP life-cycle models presented in the academic literature, followed by the life-cycle used for SAP Business One implementations.

3.1 Process-oriented ERP life-cycle model from Markus and Tanis

One of the first and most renowned life-cycles for ERP implementation project was proposed by Markus and Tanis (2000). Their life-cycle is divided in four steps [MAR2K].

Figure 3.1: The ERP life-cycle from Markus and Tanis (2000)

1. The Chartering Phase.
   This phase contains the activities which lead to the beginning of the project, mainly the Business Case, the adoption of a precise ERP software solution and the budgetization of the project.

2. The Project Phase
   This phase leads to the system *'up and running into one or more business units' [MAR2K].* This phase takes a lot of time because many important points are taken in consideration during this stage. The main configuration, integration, testing and data management are done during this phase, and even the first users training start here.

3. The Shakedown Phase
   This phase is the *cooldown* phase, going from the time the ERP is live to the moment the whole organization get used to it enough to *'normal operations'* or *'routine use'* under the terms of [MAR2K]. Most bugfixing is done during this phase, and while the first issues are resolved, new problems can arise during this phase.

4. The Onward and Upward Phase
   This last phase is going from the moment the organization got used to the system live on production, to the moment it is replaced or upgraded. Many things have already been done at this point, and the essential is now to improve the process where it is still possible, or give the users some more training. This is the phase where the enterprise must realize the major part of the *'business benefits, if any' [MAR2K].*

3.2 ERP stages from Ross and Vitale

[ROS2K] is another of the first academic papers addressing ERP implementation stages, it compares the ERP implementation to *'the journey of a prisoner escaping from an island prison' [ROS2K].* These stages are illustrated in Figure 3.2 and detailed as follows :

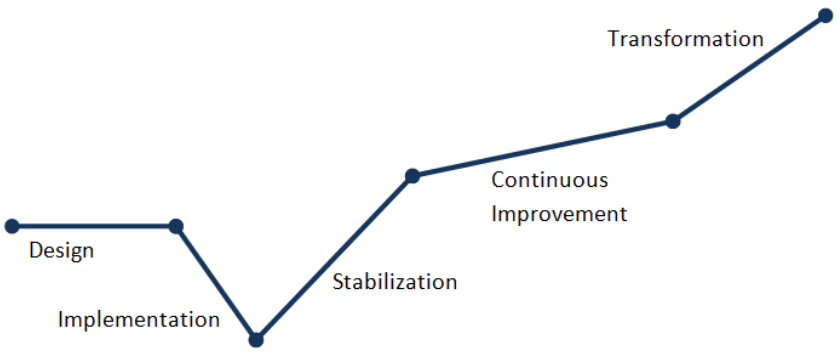


Figure 3.2: *'Stages in the ERP Journey'* from [ROS2K]

1. The approach : ERP design
   First stage of the life-cycle, comprising *'two important decisions, one about process change and another about process standardization' [ROS2K].* The ERP software must be configured to the enterprise needs, and the enterprise should choose the best ways to adapt its organization to the software, or the other way round.

2. The dive : Implementation
   The installation of the software in the relevant business units, or the entire organization, introducing major organizational change, and ultimately the Go-live.

3. Resurfacing : Stabilization
   A period *'immediately following implementation'*, during which the enterprise attempts to *'clean its processes and data and adjust to the new environment'*.

4. Swimming : Continuous improvement
   Following stabilization, enterprises enter *'a stage in which they are adding functionality through new modules or bolt-ons from third-party vendors.'* This is the stage during which enterprises are supposed to realize significant operating benefits.

5. Transformation :
   Last stage of this life-cycle, the transformation is the future of the system, ahead of the continuous improvement. During this stage, the enterprise top management anticipates the next important stages in the IT system and the enterprise structure, with major evolutions for its ERP software, or for the enterprise organization.

3.3 Process management life-cycle

In [POR06] is presented a life-cycle oriented towards the business processes of the enterprise. It introduces a process management life-cycle which will be exposed in this section. The ERP implementation project life-cycle starts when an organization realizes the need for ERP systems, which leads to the vendor selection process. A solution is chosen to suit the needs of the enterprise.

Figure 3.3: Process-oriented ERP life-cycle

Then the organization will have to decide on the implementation approach. This will include for example a reflection about going for a *big-bang* type implementation, or a *phased* type implementation like, for example, one business unit after another. The implementation of the process in itself then begins. It is divided in several steps, as shown in figure 3.3.

This methodology is centered around the business processes of the enterprise all the way from the very beginning of the project to the monitoring phase. Right after the choice has been made about the implementation approach, starts the identification phase which is crucial. During this phase, the purpose is to identify which processes will require the most improvement or reengineering to be the most effective and take the most advantage out of the ERP software.

Process Identification

Several ways exist to identify the most important processes to be reengineered or improved. One of these mechanisms is to compare the need to reorganize in regard to the value of the

process, and classify the processes into four subsets as shown in Figure 3.4[POR06].

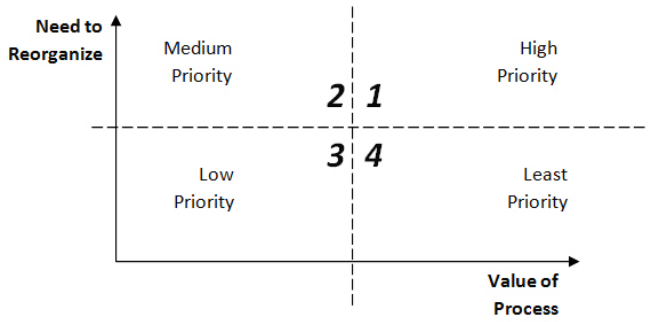


Figure 3.4: Value of Process Vs Need to Reorganize

Another one proposed by Davenport in [DAV93], as cited in [POR06] use similar criteria, but adds the notion of 'steps' to the method. The first step should be to list all the major processes in the organization, then to clearly define the limits of each of these processes. Thirdly, the strategic relevance of these processes must be determined, and their health must be analysed. Finally, the background, cultural and political implications of each of these processes should be qualified, and only the processes filtered out as unhealthy will be taken in consideration for reengineering. [POR06, page 8]

The most important processes can also be selected in regard to their impact on the strategic goals of the enterprise. By identifying the strategic goals of the considered enterprise, and deriving the Key Performance Indicators (KPIs) that will help achieving these objectives, the implementation team is available to determine the most important processes to take into account. The more KPIs are impacted by the process, the more attention it will need. The KPIS are performance indicators which can be used to measure the performance of a given process. This topic will be described in more details further in this document.

Process Modeling

After the process has been identified, the next phase of the implementation is the modeling phase. During this phase, we model the current process of the enterprise. The modeling of the current process, namely the *as-is* process, should not be neglected. Not only it gives everyone a common vision of the processes, but it can also be used as a basis to model the *to-be* process, and to verify its completeness afterward.

Process Analysis

The process can then be analysed in the next phase. During the identification phase, the processes which needed reengineering were identified. The analysis goes further, and points out all the problems in the various steps of the process itself. These problems can then be measured in terms of impact and seriousness on the enterprise, and potential solutions can begin to be imagined to get to the transformation phase.

Process Transformation

The process transformation phase can then begin. There are several ways to improve the business processes of an enterprise. Kalakota and Robinson identify three dimensions to be taken in consideration : the degree of digitisation, the scope of the process integration, and the types of interaction that go on during the process. [POR06]

They also identify three categories for the transformation process execution. The efforts in the reengineering could be focused on process improvement, which are the first kind of improvement that people may think about. This involves reducing the time between the different steps, or the number of steps during the process, adapting the process with the new system in order to reduce the overall cost and time of the process. On the other hand, the focus can be put on the strategic improvement of processes, which does not stop at reducing the end-to-end time of a process, but also try to reduce or modify the life-cycle of much bigger processes among the enterprise [POR06]. It could be for example to modify the way the orders are made, in order to center the strategic process on the customer, and not on the enterprise stock. And at last, the final way to undertake the transformation process is business transformation, which means *changing the way the industry as a whole operates[POR06].* This last way to execute the transformation is far more rare though, and goes beyond the usual ERP implementation projects.

Process Implementation

Two different points of view can be considered for the process implementation phase. These are the *'organizational'*, and *'information technology'* view [POR06]. The organizational point of view is focused on the various people involved in the process, and how convenient is the process with their way to execute it. On the other hand, the information technology point of view focuses on the new infrastructure needed for the new implementation of the system. For example, it can be developing addons for the ERP software or tailoring it to specific needs which are not included into the base package.

The implementation process uses the results from the preceding stages to replace the older processes with the new ones, either directly or more smoothly, according to the previously defined implementation strategy. New issues could possibly occur during this stage, but most potential problems should have been addressed earlier and thus avoided in this stage. This should result in an implementation process as smooth as possible, with only minor issues. The only new problems which could occur during this stage of the implementation should be precise problems, posing minor threats and quickly solved, to finally obtain a satisfying, *'frozen'* state of the system [POR06].

Process monitoring and control

The last stage of this process management life-cycle is the *Process monitoring and control.* During the process monitoring, a special attention should be turned at benchmarking the new processes regarding the most significant key performance indicators (KPIs). One of the important KPIs is the *work time vs. idle time* [POR06]. The benchmarking period can last various amounts of time, but at the end of a predefined period, the KPI must cope with the desired

goal in the strategic vision of the enterprise. Otherwise, the end of the monitoring and control phase can match with the beginning of a new life cycle.

3.4 ERP life-cycle model from Somers and Nelson

In [SOM04], Somers and Nelson present a process view of ERP implementations, based on the ERP implementation framing from [RAJ02]. This life-cycle for ERP implementation projects is based on the six stages of IT implementations projects from [KWO87] and [COO90]. This six-stage model is shown in Figure 3.5

Figure 3.5: ERP project life cycle stages from [SOM04]

The details of the six stages are :

1. Initiation : First stage of the life-cycle, during which a match between the enterprise needs and IT solution must be found. *'Active and/or passive scanning of organizational problems/opportunities and IT solutions are undertaken' [COO90].*

2. Adoption : During this stage, negotiations should be conducted to determine the resources to be dedicated to the project. The investment decisions and cost-benefits analysis related to the implementation should be carried out during this stage.

3. Adaptation : The ERP software is being installed in the enterprise, the enterprise data is being transferred, and either the software will be adapted to the enterprise, or the enterprise process changed to match the software. The user training also takes place during this stage, and a certain *'resistance should be observed due to the 'inertia' associated with using the previous system' [RAJ02].*

4. Acceptance : Continuous improvements are made to the system, which *'become increasingly available for use in the organization' [RAJ02].* By the end of this stage, the software should be *'available for organizational work' [COO90].*

5. Routinization : The usage of the system *'becomes a regular day-to-day activity' [RAJ02].* The system has been installed in the enterprise and the user accepted the system, now becoming a part of the organization.

6. Infusion : *'The system is used to enhance the performance of the organization' [RAJ02].* During this stage, the ERP system is used at its full potential across all needed business units.

3.5 Esteves and Pastor ERP life-cycle framework

Esteves and Pastor's works such as [EST99, EST07] use another framework structured in phases and dimensions (see Figure 3.6).

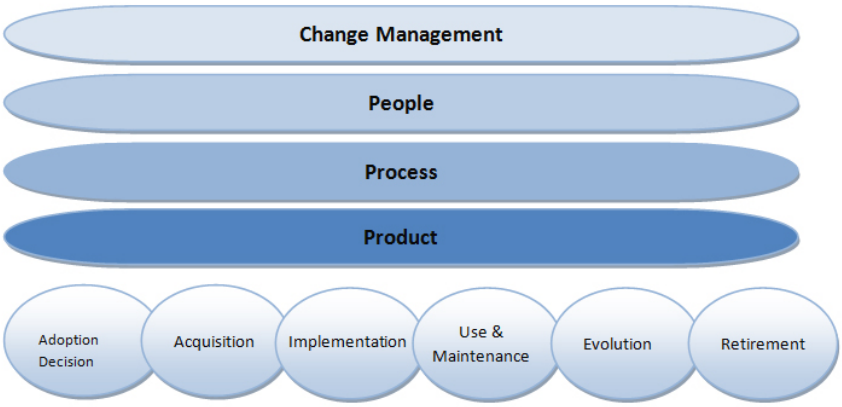


Figure 3.6: The ERP life-cycle framework from Esteves and Pastor

Not only this framework addresses the various stages of an ERP implementation, but it also considers four dimensions, or viewpoints, to analyze the implementation. The structure of this framework is different from the others, not only because of the phases, different in each framework, but also because of the dimensional aspects, taking into account the points of view during the implementation project. [EST99] describes in details the various phases and dimensions of the framework, as well as the various research issues related to it. The summary of these phases and dimensions will be presented in the next two subsections, and is mostly based on the descriptions found in Esteves and Pastor works.

3.5.1 Phases of the life-cycle

1. Adoption decision phase :
   This phase is mostly oriented towards the enterprise. The managers of the enterprise must define the goals and the system requirements, as well as analyze the future objectives and benefits they want to achieve in their enterprise with the help of the ERP software, both on a technical and an organizational point of view.

2. Acquisition phase :
   During this phase, the solution will be selected according to the requirements of the enterprise defined in the previous phase, in order to minimize the customization. The enterprise will also choose a consulting company to help them in the next implementation phases. A contract agreement must be signed, and the authors also recommend to make an analysis of the return on investment of the selected product.

3. Implementation phase :
   During this phase, the selected product will be customized and adapted to the specific needs and constraints of the enterprise, with the help of the consulting company.

4. Use and maintenance phase :
   The enterprise start using the product, gaining some benefits from it, and the maintenance begins, by correcting the last problems and making special optimizations to the various implemented processes.

5. Evolution phase :
   More functionalities will be integrated into the ERP system during this phase, improving various business units among the enterprise.

6. Retirement phase :
   This phase begins when the enterprise wishes to change the ERP solution, because it does not suit the enterprise needs anymore and the only possible solution is to substitute it with another information system. [EST07] shows that this phase is not of critical interest for the research. No publication has been made on this topic at this time, indeed it is unlikely that an ERP solution successfully implemented in an enterprise would be completely replaced, considering the resources invested in such projects.

3.5.2 Dimensions of the life-cycle

The interesting aspect of this framework is in the dimensions, or points of view, spanning the entire ERP implementation process. These viewpoints are indeed rarely modeled in the other frameworks.

Dimensions of the ERP life-cycle

1. Product :
   This focuses on the particular solution considered. A deep understanding of the software solution is needed to correctly implement it, to make the best possible alignment with the enterprise business processes, and achieve the goals intended by the enterprise.

2. Process :
   Each organization has its own requirements which need to be supported by the ERP system. An important part of the investment will be spent on the process reengineering for the ERP solution to match the best possible way the organization processes.

3. People :
   This one is another important viewpoint. The main actors of the project will be the people using the software, and their feeling about the ERP. The way they will use it and adapt to it will be of top importance all along the project.

4. Change management :
   This dimension *'tries to ensure the acceptance and readiness of the new system, allowing the organization to get the benefits of its use' [EST99].* The change management refers to the knowledge needed to make the adaptation smoothly, and within due time and cost.

3.6 Another ERP implementation life-cycle

In [SOJ07], another framework is used in order to classify a list of critical success factors (CSFs) into the various stages of an ERP software implementation. In this work, the differences between the various existing life-cycle models are stressed as well as the number of phases varying for everyone of them. Then the author proposes a framework capturing all the various phases of an ERP implementation project. The phases are organized as follow :

1. Project Organization

2. Training how to manage a company with the use of ERP system

3. Enterprise Analysis

4. Implementation Design
5. Training on ERP system use
6. Detailed project planning
7. Pilot implementation
8. System start
9. Post implementation review

The contents of the various phases are self-explanatory, but the interested reader can find all the details for every implementation phase in [SOJ07].

3.7 Life-cycle from the SAP group for SAP Business One

The life-cycle to be used as a reference by SAP consultants is provided by the SAP group on the SAP PartnerEdge Portal. This life-cycle is twofold. The first part is the *SAP Business One Sales Cycle Navigator* and is designed to cover every aspect prior to the implementation, from the initial contact to the contract signature. The second part is more comparable to the preceding life-cycles, as it represents the methodology recommended by SAP for an easier implementation of their solution. Combined together, these two life-cycles form the entire methodology to sell and implement the SAP Business One ERP solution.

It should be noted to the reader's attention that the life-cycles presented thereafter have been studied as of June, 2008, and have been slightly modified since then. The life-cycles, tools and methodologies from the SAP group are continuously updated on a regular basis.

3.7.1 SAP Business One Sales Cycle

This first of the two life-cycles to be considered while implementing SAP Business One in an enterprise contains six main stages lettered from A to F, and two additional steps. The first of these two steps is the presentation of the consultant to the customer, to be done before anything else, and the last one represents the *Accelerated Implementation Program* of the next section. The six main stages of the Sales Cycle Navigator (See Figure 3.7) contains all the tasks that should be concluded before a contract agreement, and the start of the implementation project in the enterprise.

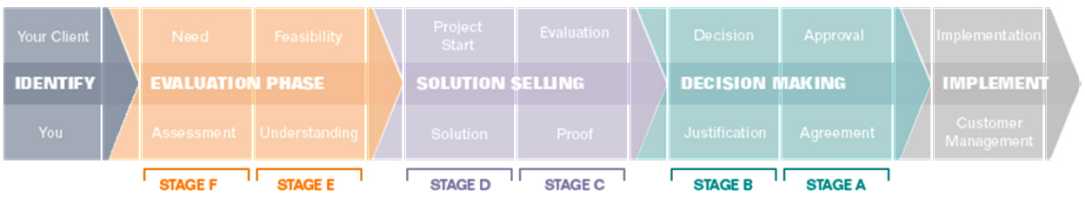


Figure 3.7: SAP Business One Sales Cycle Navigator

Even if more *'marketing-oriented'* than the *Accelerated Implementation Program* itself, the SAP Business One Sales Cycle contains some useful information and important tasks, which makes it important to take into account for a successful implementation of the ERP solution. The purpose of the six main stages of the sales cycle is to sell the solution to the prospect, but in order to do so, this cycle contains several tasks which will help the consultant to determine whether or not the solution is adapted to the enterprise needs.

1. Evaluation phase : stages F and E, namely need assessment and feasibility understanding, aimed at determining the enterprise situation, and if the solution is adapted to the customer's needs.

2. Solution selling : stages D and C, project start and evaluation, aimed at building a personal solution to the enterprise needs and determining the advantages of the ERP solution over the AS-IS system.

3. Decision making : stages B and A, Decision justification and approval agreement, aimed at persuading the customer that the proposed solution is the best for him and obtain a contract agreement.

3.7.2 SAP Business One Accelerated Implementation Program

In December, 2007 was released on the SAP PartnerEdge portal, Documentation Resource Center section, the SAP Business One Accelerated Implementation Program 2.0 (ASAP Business One). This is the adaptation for SAP Business One of the ASAP methodology used with the SAP Business suite (formerly named SAP R/3). This section will expose the life-cycle considered in this methodology.

At first glance, an SAP Business One implementation project using ASAP Business One spans the project among five phases (see figure 3.8). These phases are composed of various milestones and tasks to be done, which will be summarized in the phases descriptions.

Figure 3.8: SAP Business One Implementation Guide

1. Project Preparation
   This phase must provide initial planning and preparation for the project. It is in this step that the stakeholders identify and plan the primary focus areas in the considered enterprise, as well as delivering a first preliminary version on a test system.

2. Business Blueprint
   The purpose of this phase will be to achieve a common understanding of how SAP Business One will precisely help the enterprise to improve each business processes for which the solution will be implemented in the enterprise. The main issues addressed in this section are to elicit the requirements among the business units of the organization, as well as to identify and analyze the various business processes. Furthermore, the original goals must be refined to adapt the schedule consequently. The SAP group even define this step as

the moment to *'Create a Business Blueprint to serve as a technical and functional guide during the subsequent phases of the SAP Business One implementation project'. [SAP08]*

3. Project Realization
   This phase will realize all the previous modeling and analysis, in order to implement the business process and the technical requirements defined earlier. The various configurations are validated and updated, unit and integration tests are conducted.

4. Final Preparation
   This phase is meant to get the various users used to the new system after the realization, and prepare the ERP to be released for production. The final system tests are conducted, as well as the last slight issues that could still occur. In the meantime, the administrators and the end users are trained.

5. Go-Live and Support
   This phase will see the monitoring and control of the business processes, the support for the production environment, and still, some more optimization on the information system performance, where it is still possible.

Section 4 :

Critical Success Factors in ERP Implementation

> **Content :** *Every enterprise willing to adopt a new ERP software with maximum efficiency and minimal costs must keep in mind several important factors. Such factors are called Critical Success Factors (CSFs), and are mandatory in order to implement ERP software within a minimal time, and a minimal cost. This chapter will cover the main CSFs found in the literature and the various ways used to categorize them.*

4.1 Critical Success Factors in ERP implementation projects within SMEs

Now that we have a global vision of the life-cycles used during ERP software implementations, and before detailing the modeling or the performance aspects, we must address the Critical Success Factors. Indeed, many aspects of the methodologies for ERP implementations in major enterprises must be adapted for implementations in SMEs, mainly for cost and time reasons. Modeling, for example, is one of such aspects, but this will be detailed later. The main concerns to successfully implement ERP software are the same in Small, Medium, or Large industries though. These concerns must not be confused with the objectives or the Key Performance Indicators (KPIs) because neither they are precise goals or the enterprise, nor they are measurable indicators for the implemented processes. Instead, the CSFs can be perceived as general guidelines, to be used either during the whole implementation, or more centered around one or two stages of the implementations.

Many researches and studies have been made about the critical success factors for ERP implementation, for example [HOL99, EST2K, MAR2K, NLK01, SOM01, SOM04, ZHA03, ZHA05]. Many CSF models have even been proposed in order to classify them, or unify them in some ways, but there is no general consensus yet about which CSFs are the most relevant success factors along the whole implementation. This chapter will cover the most cited CSFs, as well as various frameworks designed to classify them. The chapter will finally be concluded with an insight on the CSFs use in the ASAP for Business One methodology.

4.2 Strategical and Tactical Success Factors

The Strategical and Tactical Critical Success Factors framework proposed by Holland and Light [HOL99] was one of the first models used to regroup and classify the major concerns during an ERP software implementation process. This model has been fairly referenced in further studies.

The Critical Success Factors model proposed by Holland and Light classifies the CSFs in two categories : the strategical success factors and the tactical success factors. While both types of factors have similar importance, the strategical factors are the ones to keep in mind at the beginning of the process, while the tactical success factors gain importance throughout the progress of the implementation (see Figure 4.1).

ERP Implementation Process	
Strategical	***Tactical***
Legacy Systems Business Vision ERP Strategy Top management support Project schedule and plans	Client Consultation Personnel BPC and software configuration Client Acceptance Monitoring and Feedback Communication Troubleshooting

Figure 4.1: A Critical Success Factors model with strategic and tactical factors, from [HOL99]

4.3 Further works : more perspectives for CSFs

The CSFs from this model were among the first to be identified, but many more have emerged since then. Several possible ways to arrange them have been found, like in [WIE06, p130] where three other possible dimensions are cited :

- Internal vs. External factors,
  on whether the CSF is related to the client or on the vendor and the relation between these two.
- Static vs. Dynamic factors,
  with a distinction on the nature of the CSF, whether it represents a characteristic or a status, or if it expresses an activity.
- Organizational vs. Technical,
  whether the focus of the CSF is aimed at the organization of the enterprise, or on the technical level.

Among further works, the unified model proposed by Esteves and Pastor [EST2K], following a grounded theory research method, resulted in a more sophisticated matrix, presenting a

few more CSFs, arranged in two dimensions : Strategic vs. Technical and Technological vs. Organizational. A presentation of this CSF unified model is illustrated in Figure 4.2

	Strategic	Tactical
Organisational	⇨ Sustained management support ⇨ Effective organisational change management ⇨ Good project scope management ⇨ Adequate project team composition ⇨ Comprehensive business process reengineering ⇨ Adequate project champion role ⇨ User involvement and participation ⇨ Trust between partners	⇨ Dedicated staff and consultants ⇨ Strong communication inwards and outwards ⇨ Formalised project plans/schedule ⇨ Adequate training program ⇨ Reduced trouble shooting ⇨ Appropriate usage of consultants ⇨ Empowered decision-makers
Technological	⇨ Adequate ERP implementation strategy ⇨ Avoid customisation ⇨ Adequate ERP version	⇨ Adequate software configuration ⇨ Legacy systems

Figure 4.2: Unified CSF model [EST2K]

As the studies were progressing on the ERP implementation CSFs matter, it appeared that organizational aspects were considered to be more important than strategical ones [EST2K], the sustained management support being among the most critical CSFs [HOL99, EST2K].

4.4 Contextual Success Factors

Apart from the general success factors, researches have proved that the context is also very important for a successful ERP implementation. (e.g. [ZHA03, EST03]). These studies tend to prove that three more factors must be considered :

- Power distance, which represents the way power is perceived in the organization, in regard to the status of the employees in the organization hierarchy.
- Uncertainty avoidance, which is the level to which an organization is ready to be confident in a project containing ambiguous or uncertain aspects.
- Individualism/Collectivism, depending on the culture of the organization considered, employees will be more or less independant to each other, focusing the importance on the autonomy of a person, or on his importance with respect to a group.

The impact of these specific factors, proposed by Hofstede in 1991 [HOF91], may not be as important as the other more critical factors, but the way people in the organization will react to the ERP implementation project also has consequences. For example, [EST03] relates an ERP implementation project in Portugal, where power distance had a significant importance. Conflicts appeared between senior managers, everyone attempting to impose their view of the project in order to influence their own carriers, and each manager communicated his vision to his subordinates, resulting in even more conflicts of interest. This study also showed a high uncertainty throughout the whole project, people in the middle and top management feared to

lose their jobs consequently to the implementation. There were consequences to this fear, such as delaying decision-making meetings to delay the entire project, or a lack of participation in some tasks. Furthermore, the kind of 'familial' work relationships also brought up some various problems about the confidence between employees and their managers. Employees began to have a lack of trust in their manager, not really knowing where the new ERP software could lead them, and fearing for their jobs.

Other studies such as [ZHA03] or [WWW03] note that the best practice designed in vendor's implementation methods often reflects European or US industry practices, and do not always cope with other markets such as, for example, the asian market. In [ZHA03], the author points out the notion of collectivism which is really important in chinese industries, or the *clash level of the culture embedded in the ERP package with the customer's organizational culture*, which points out the difference in needs of information for the top management, which is a critical matter for western countries, and far less important in China [ZHA03, 4.10].

4.5 Relative importance of the CSFs along the implementation process

Determining the critical success factors of an ERP implementation project is only the first part. It is important to keep in mind the relevance of each CSF regarding the project phase we are processing at a given time. In [NLK01, p. 290] for example, the authors tried to classify the eleven CSFs they identified into the Markus and Tanis process-oriented ERP lifecycle model, in order to define the place of each CSF in a project time life.

On the other hand, Somers and Nelson [SOM01] selected twenty-two CSFs to classify, by the mean of a mailing of 700 surveys sent to specific enterprises in which ERP software was being implemented, or was implemented. Following Somers and Nelson's framework considering six phases, namely the initiation, adoption, adaptation, acceptance, routinization and infusion, they established a score for each CSF. The CSFs considered were then arranged by priority, both for the entire project and for each separate phase. This allowed *not only to determine which CSFs are the most critical in ERP implementations, but also which factors are temporal, i.e. significant in the implementation process for a particular period in time [SOM01].* The study also shows that the most critical steps are situated at the beginning of the project, the *package selection* itself and the *preparation* containing a lot of critical success factors to be considered to ensure that the rest of the project would go on smoothly.

In 2001, Esteves and Pastor, who designed the unified CSF model [EST2K], tried to associate a relevance of each success factor of their model in regard to the various stages of the ASAP implementation methodology [EST01]. It has its own phases and documents, and the goal of the study was to relate the CSFs defined in the unified model with the various processes of the ASAP methodology. By coding the various factors and the implementation phases in a matrix, they were able to derive the most critical success factors for each phase. They proposed further research afterward, one of them being to identify the most critical process of the whole implementation. They insist at the same time that *this step should be done with information provided by cases studies, and not only based in theoretical assumptions.* [EST01, p. 6.]

4.6 Use of CSFs in ASAP BO

This section will expose as an example the use of CSFs during the ASAP implementation methodology for SAP Business One. Before going in the details of every stage, it is important to note that SAP states that *the program was developed based on the requirements of the North American market* [SAP08]. Indeed, numerous studies have been made to elicit the differences among the contextual success factors across various countries, for example in China [ZHA03, ZHA05]. Factors like Power Distance, Individualism, and Uncertainty avoidance are different for each and every country considered. In countries like China, where foreign ERP vendors have taken more than 90 percent of the ERP market share [ZHA05], the environmental and cultural factors can seriously impact the success of the project, from the user satisfaction perspective, or the organizational impact, for example.

Many CSFs are to be considered right at the beginning of the project, and for the whole duration of the life-cycle. The project preparation is the first stage of the implementation in Accelerated SAP for Business One, and logically this phase along with the business blueprint phase should see most of the strategical success factors appearing as guidelines. For example, there is a detailed team composition for each phase of the implementation. Table 4.1 represents the team composition for the first stage of the life-cycle, which illustrates one of the common critical success factors, namely the early inclusion of the end users to the project, and a good composition of the implementation team, mixed with power users, and IT administrators on the enterprise side [NLK01, *ERP Teamwork and composition*, table I, p288].

Team member	**Estimated Time Commitment**
SAP Business One Project Manager	29.5 hours
SAP Business One Consultant	35 hours
SAP Business One Account Manager	26 hours
Client Project Manager	9.5 hours
Client IT Administrator	14 hours
Client Lead - Accounting	6 hours
Client Lead - Sales	6 hours
Client Lead - Operations	6 hours
Client Lead - Purchasing	6 hours
Client Lead - Logistics	6 hours
Client Lead - HR	6 hours
Client Executive Steering Committee (if applicable)	6 hours

Table 4.1: Project Preparation Team Members [SAP08]

Along with the team composition, documents and tasks to be concluded during each phase, there is also a list of *best practice recommendations* corresponding to each stage, which seem at first sight to be what's most close to the CSF defined in this chapter. The next four subsections will detail these best practice recommendations and elicit the various CSFs addressed for every stage.

4.6.1 Project Preparation best-practice recommendations

Some best practices for the first stage of the implementation are to *explicitly confirm hardware, software, and resource availability* or to *include sales team in kick-off meeting* [SAP08], which is a part of defining a global vision of the objectives, common to the organization and the ERP implementation team. Another one is to encourage the client to *create an executive steering committee in order to assure executive management of the support of the implementation project*, which can ensure a strong client investment in the project, which is also an important CSF to successfully implement an ERP software [SOM01, *top management awareness, support, involvement in implementation duties*, page 7].

4.6.2 Business Blueprint best-practice recommendations

This step consists in the best possible adequacy between the business processes or the requirements of the enterprise, and the ERP software which will be implemented. In the best practice recommendations for this phase, some more CSFs are outlined. The SAP group recommends to *conduct both one-on-one and group workshops* to detect *individual requirements as well as overlaps and dependencies between departments.* This shows once again the impact of a good coordination between the vendor and the enterprise, as well as a good communication between the various business units. They also recommend to use the client's chart of accounts instead of creating a new one, in order to reduce the effort needed during the migration of the data. This is the *minimal customization* CSF [SOM01]. Finally, the SAP group recommends to make the client aware that even the slightest further modification to the business blueprint which would be defined could have *significant impact on cost, resources, and timeline of the project.* [SAP08]

4.6.3 Project Realization best-practice recommendations

Unsurprisingly, the first recommendation is to *involve client functional leads constantly*, as it is also advised in most of the literature about the CSFs matter for ERP implementation, by the means of words like *team involvement, co-operation with supplier* [SOJ07], *interdepartemental communication, partnership with vendor* [SOM01], or even *Client consultation* and *Client acceptance* from the CSF model for ERP implementations from Holland et al [HOL99].

Apart from other technical recommendations, as well as recommendations designed specifically for their solution, the SAP group also recommends for this step a *minimal number of changes to setup and scope*, as it can have a serious impact on testing and training.

4.6.4 Final Preparation best-practice recommendations

The main best-practice recommendations for this phase do not specially refer to any CSF except one of them, namely *Communicate training and cut-over plan to all client stakeholders*

with an official, written announcement which refers once again to the clear communication between the vendor and the client.

4.6.5 Go-Live and Support best-practice recommendations

Here again most of the job is done, and there is not a lot of CSFs to be pointed out now. The only thing that could be noted is that the SAP group recommends to perform the *Review and Optimization Conference four to six weeks after project closing meeting.* This in order to ensure the client will be able to have an efficient feedback on the new ERP-enhanced system.

4.7 Preliminary conclusions about the CSFs.

Only a combination of all these various critical success factors can explain a success or a failure in implementation, rather than specific ones, but there is no general consensus yet about the relevance of each and every of them, when considering an ERP implementation project in its whole life-cycle. Some works can serve as strong basis to define such a model, like [EST2K], but this model do not take the context into consideration.

On the other hand, if all the possible critical success factors should be listed in some way, the list would be far too long to be usable, and with every ERP implementation being different, the relevance of each factors would vary from one enterprise to another.

The CSFs are useful to get some notions of what must be kept in mind to obtain some guarantees for a good implementation process. This is not enough however to be able to predict a successful implementation, or to estimate the future needed work. They can at most serve as guidelines for the implementation or clues during the requirement elicitation process, to find the most critical processes or the relevant key performance indicators.

Section 5 :

Modeling during ERP implementations

> **Content :** *In order to fully understand the business processes of the enterprise and to reduce the effort needed to match the ERP software and the enterprise processes, an important modeling process must be conducted. A correct goal and business process modeling will help the enterprise to understand, manage, adapt and explain their main objectives and critical business processes for the ERP implementation project.*

5.1 The importance of ERP modeling.

Every time ERP software is implemented in an enterprise, some alignment process must be undertaken, either by customizing the software to the enterprise needs, or by adapting the enterprise to the software. Most of the time, the latter is preferred, because it should be less expensive for the enterprise, and because the ERP software packages include *reference models* that supposedly reflect the *best business practices* [SOF01]. This approach is also generally preferred because it is less expensive to adapt the business processes to the software, than the other way round, and this is really important from a small or medium enterprise point of view. On the other hand, specific enterprises may be performing some unique business processes, making sensible the investment for a long and expensive implementation project, requiring high software customization or even addons [SOF03]. On both cases, some modeling tasks are needed to support the alignment process during ERP implementations.

5.2 Various Modeling points of view

Several points of view can be used to analyze an enterprise architecture. Several works such as the Zachman framework [ZAC87] for example, detail the various perspectives which can be considered while analyzing an enterprise in order to implement information systems. [ZAC87] refers to several *types of descriptions for a same product*, by the mean of several questions such as :

- *What ?* , for the data, or the business objects
- *How ?* , for the functions, or processes performed by the enterprise

- *Where ?* , for the network, or business locations of the enterprise
- *Who ?* , for the people involved in the project, or the organization itself
- *When ?* , for the time, the schedule or the significant events of the project
- *Why ?* , for the motives, goals and strategies of the project

This section will focus on three main questions about modeling during an ERP implementation project :

- The *What*, about the data to be used by the new system, to find out the gaps between the way data was managed prior to the ERP system, and data handling with the new system.
- The *How*, through the review of the most critical business processes of the enterprise, and the way these are managed by the ERP software. Such analysis will point out the similarities and differences to be worked out.
- The *Why*, through goal modeling in order to elicit the requirements and further establish the goals that ERP software will contribute to achieve.

Other perspectives could be considered but, even if these are not completely neglected, they are instead less relevant to be detailed in this section. For example, considering the *Where*, or the business locations, must be done early while thinking in terms of adaptability and scalability, but does not require a sophisticated modeling step. On another hand, the *Who*, meaning the people of the organization concerned by the project, are generally the entire organization, and the way they interact with each other will be illustrated through the business process model. Finally, the *When*, about the main events and the schedule to be followed during the project are addressed in details inside the life-cycle sections. As said in [ZAC87] : *'there are reasons to expend the resources for developing each architectural representation, and there are risks associated with not developing any of the architectural representations'*. This is again a matter of compromise, particularly important among SMEs, where the resources are even more limited than in large enterprises.

5.3 Goal modeling for ERP implementations

Goal Modeling and goal-driven approaches are the first orientation to consider during ERP implementation projects. As said in [ROL05], *goals have long been recognized to be an essential component involved in the Requirements Engineering process*, and enterprise managers mainly think in terms of goals to motivate the implementation of such new software and its cost. Only after the goals of the implementation are clearly defined, can the implementation team think in terms of data and process management.

In [ROL2K], the authors point out the difficulty to directly match the software functions with the business processes of the enterprise, both because of the amount of data to be maintained and because the enterprise mostly think in terms of objectives and goals than in terms of workflows and business processes. They suggest to abstract the ERP functions in order to see them as ERP goals, thus neglecting the performance issues, and make a first alignment with the enterprise business goals instead of going deep into the processes. (see Figure 5.1)
Illustrating the example with the ERP software SAP R/3, and using a special goal modeling

notation designed on their own, namely the *map* notation [ROL2K], they define *SAP goals* as the strategical objectives, and *strategies* as the various ways to achieve these goals or tasks in the ERP software.

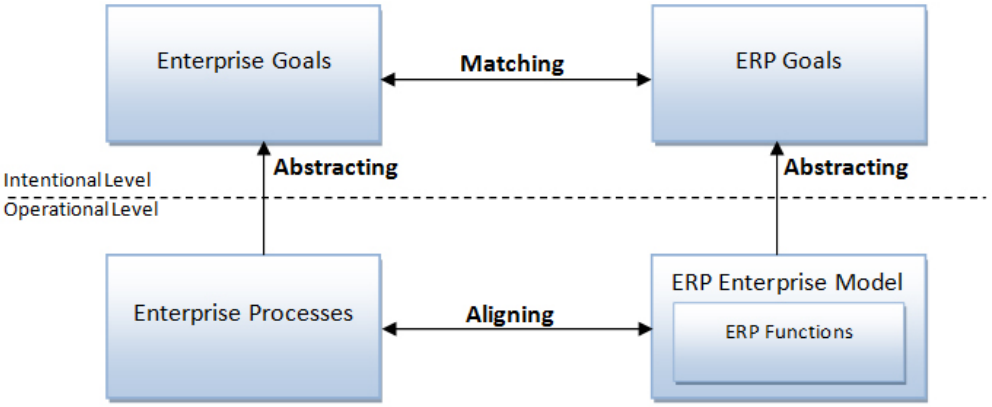


Figure 5.1: Levels of description (from [ROL2K])

Some further research have proved this approach to be effective, for example [SAL07] used the map approach to develop a reuse-based requirements elicitation method for ERP integration, using another ERP software, namely Microsoft Dynamics NAV [MSN08].

Modeling the goals in the first place is the key to determine the adequacy between an ERP software solution and the business needs of an enterprise. By elicitating the goals and objectives of the enterprise, it allows all the stakeholders to share a common good vision of both the ERP main functionalities and the goals of the implementation project. Furthermore, it will provide the implementation team with some clues to identify the most important processes which will be implemented. Preceding studies have been using the MAP formalism to refine the goals, and the authors of [SAL07] even justify this choice by stating their preference against KAOS or I*, being respectively a lack of variability emphasis on KAOS, and no possibility to refine the goals in I*. [ROL2K, SAL07].

The goal-modeling stage should be done to avoid some of the potential problems from the traditional approaches, namely technically good solutions, but inadequate for the enterprise purposes. Only then can the mapping between goals and software objects, events and operations be done. This model must be carefully planned and designed, as it is the first step and an important cause of implementation failures [ROL05].

However, even if it is commonly accepted that goal modeling is an important step for ERP implementation, there is no commonly accepted notation, and most modeling notations are found in the literature rather than in commercial documentations. In [ROL05] the concepts behind goal-based RE approaches are based on modeling both the constraints of the subject world, the naturals laws of the environment for example, and the purpose of the software for the user, or the objectives of the implementation. More information can be found in [ROL05].

The purposes of goal modeling during the early stages of the implementation are many : Requirements elicitation, negotiation, traceability, as well as exploration of design for example.

5.4 Data Modeling for ERP Implementation Projects

Data modeling is another important modeling aspect during ERP Implementations. Indeed, data can be managed under several ways in the software. Similarly, the same business object can be modeled differently from one enterprise to another. The database behind the ERP software is one of the main components and must be configured properly, for every field or column, from the VAT, to the way items are stored, categorized, traced, or the way Business Partners are stored and managed. Anyway, even if data modeling must be conducted in some way to configure the database and the software to the enterprise needs, several means to customize data representation are available in ERP software solutions, such as user-defined fields (UDFs) or various item-grouping and categorizing methods, and these methods and ways to introduce data are generally well-documented by the ERP vendor, thus not needing a lot of data modeling. Comparatively, goal and business process modeling should be conducted earlier, and the alignment of the business processes or enterprise strategies to the ERP software can cause far more trouble.

5.5 Business Process Modeling for ERP Implementations

Modeling the goals is only a first step in the process. A proper ERP model should represent *business processes and their underlying information objects* [SOF03]. However, few scientific publications have been realized on the topic of modeling the business process contents, leaving the major part of the *'reference modeling content science'* to the *'vendors and commercial organizations'* [WAS06].

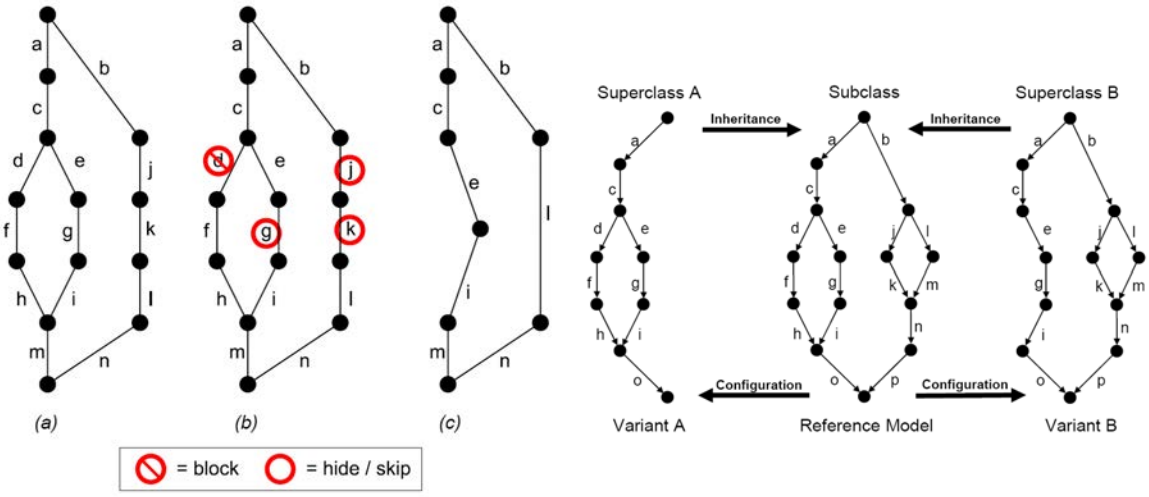
Various notations exist to detail the business processes supported by the ERP software. For example, the SAP Reference Model, as well as the Architecture of Integrated Information Systems (ARIS) use the EPCs as core modeling language [GOT07]. BPMN, Petri nets or UML 2.0 Activity Diagrams could also be reasonably used to model the business processes. Every notation has its strengths and its flaws, and once again, there is no consensus on the best notation to be used for business process modeling. Most of the works thus stay as general as possible, while some other use one notation or another as an illustration. [WOH06] for example, discuss the suitability of BPMN for Business Process Modeling, by using the workflow patterns of [Aal03] as a framework for evaluation, under the control-flow, data, and resource perspectives.

As said in [Aal06], the main advantage of reference models in ERP implementation projects is to *streamline the design of particular models by providing a generic solution*, motivated by the *'Design by Reuse' paradigm*[Aal06, GOT07]. This same work outlines the usefulness of potential configurable process models, to serve as bases for such repositories, which would undoubtedly accelerate the reduction of the gaps between the business processes of the enterprise, and the functionalities of the software. [Aal06] points out that such reference models are usually *'plug and play'* but *'often require some customization/configuration to be adjusted to individual requirements'*, which would be a main advantage of such configurable model. They show an example of such language with Configurable Event Process Chains (C-EPCs). This paper along with [GOT07] also stress the differences between various possible choices in ERP software (or business processes in general). Indeed, most of the languages used to model the business processes only allow one level of choice, but they are many different kinds of choices depicted in [Aal06] :

- Configuration choices, made at build-time for every instance of the process, and possibly impacting the further possibility of run-time choices.
- Run-Time choices, varying for each instance of the process and depending on the context.
- Choices situated *somewhere in between*, for example choices based on the period of the year, or resource availability [GOT07].

The authors also point out the potential difficulty to precisely define the borderline between the two main types of choices. Further details on that matter are given in [Aal06].

On a theoretical view of configuration, the authors explain the interest of having a global reference model, which would detail all the possible ways to conduct a given process, which could be simplified to give the built-in configuration of the ERP software, as well as the possible choices at run-time. Figure 5.2(a) from [Aal06] gives a good perception of the idea, and [GOT07] goes further by defining such configuration as the *inverse of inheritance*, by defining the various configurations as subclasses, and the reference model as a superclass, as illustrated in [GOT07].



(a) Three labeled transition systems: *(a)* the initial model (e.g., the reference model), *(b)* a particular configuration hiding and blocking specific edges/labels, and *(c)* the resulting model, as seen in [Aal06]

(b) Configuration - the inverse of inheritance, from [GOT07]

Figure 5.2: The Configuration and Choices concept in [Aal06, GOT07]

On another hand, this approach of having global repositories, designed only by the vendors, and commercial organizations, has its limits. They are not always completely accurate, may contain flaws or errors. [MEN06] outlines this problem by analyzing the SAP reference model with the help of a verification tool based on Petri nets, pointing out a minimal percentage of 5.6% of flawed processes in the entire set. They show with this study the problem of keeping such reference models private, and the usefulness of verification tools.

A clear and understandable business process model can provide help to give a common vision of the processes supported by the enterprise, for the entire implementation team, both on the consultant, and the enterprise sides. To acquire a complete reference model of the functions supported by the software is a preliminary step to establish the individual model including the requirements and constraints of the enterprise [GOT07].

5.6 Use of modeling in the ASAP methodology for Business One.

For the SAP group, there are two separate life-cycles for a complete SAP Business One implementation project. The *'Business One Sales process'* on one hand, and the *'Accelerated Implementation Program'* on the other hand. Both life-cycles have their goals and particular tools. While the purpose of the Sales process is to determine whether or not the solution is adapted to the customer's needs, and to determine the first objectives and constraints for the enterprise considered, the implementation itself is done with the help of the second life-cycle.

The implementation life-cycle has been defined in Figure 3.8 and the Sales process is illustrated in Figure 3.7. The next two subsections will address the use of modeling during the ASAP for Business One methodology, as well as the use of modeling before the implementation itself, during the SAP Business One Sales Process.

5.6.1 Use of modeling before the implementation

Goals can partly be elicitated during the Business One Sales Process, before the implementation, and with the help of the tools provided in the *'Sell'* section from the SAP PartnerEdge portal [SAP08]. This is not really goal modeling, but is the closest form of requirements elicitation tool provided for the Sales process of SAP, with the help of, among other things, a *'Qualification Map'*, an *'Opportunity Qualification Tool'*, the *'Business Blueprints'*, and the *'VIP Map'*, which can also be used to determine Key Performance indicators (KPIs).

The 'VIP Map' for *Vision, Indicator and Pain* [SAP08] is a simple tool provided to the SAP Sales consultant to model the enterprise long-term visions, or main objectives, as well the problems they are facing to achieve these goals. They are finally asked to propose an indicator which could be used to monitor the performance of the enterprise for each precise goal. This technique partly meets the objectives of goal modeling, helping the consultant to focus on the enterprise problems and stated objectives. An example of VIP map is illustrated in Figure 5.3. It can be noted that an objective can be hierarchically divided into many indicators, and that the indicators reflect the magnitude of the pain identified for the objective considered.

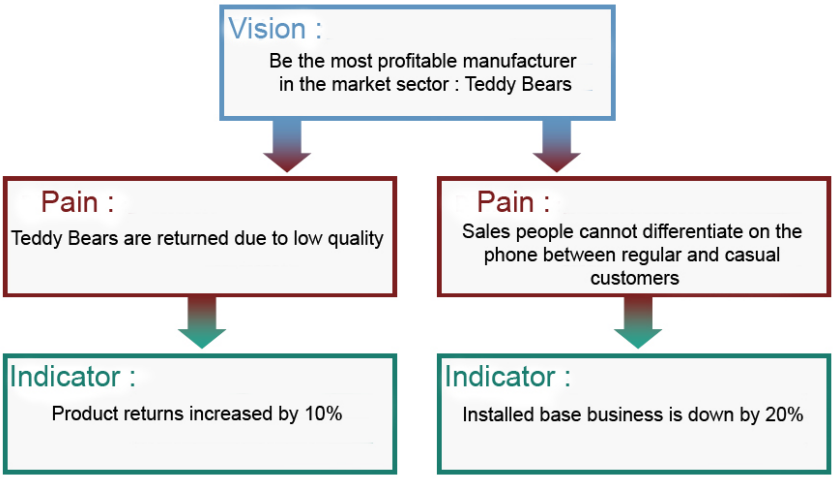


Figure 5.3: VIP Map - Multiple pains and indicators [SAP08]

Along with the VIP maps, the Sales consultant can use the *'Qualification Map'*, the *'Opportunity Qualification Tool'* and the *'SAP Business One Blueprints'*.

The *Qualification Map* is a document which can be used during the sales process to estimate whether the sales opportunity can become a potential customer or not. It basically consists in a table divided in three columns, namely the prospect, the project, and the cost. These three categories are divided in four indicators, each one valuated from 1 to 10. The final note between 12 and 120 gives an estimation about the knowledge of the business case characteristics.

Another tool available to the consultant during the entire sales process is the *Opportunity Qualification Tool*, which is a web-based tool usable to establish the adequacy of the software solution to the business case considered, *'before committing any valuable time and resource'* [SAP08]. This tool could not be analyzed in this thesis though, due to a lack of real case study.

Finally, *SAP Business One blueprints* are, according to SAP, *'a set of paper tools that can be used to show the product's key features, functions, processes and differentiators to really engage and interest our SAP Business One prospects'* [SAP08]. These are a set of A3-format (or US 13x17) posters for the main part, that explain the general functionalities, process flows and technologies used with SAP Business One.

5.6.2 Use of modeling during the implementation

The modeling aspect can be only scarcely found in the SAP Business One Accelerated Implementation Program. There is no or little support in the methodology to precisely model the goals or business processes of the organization. Goals are addressed before the implementation, during the sales process of the ERP solution, and nothing is provided to document the business processes as they are supported by the software. This is a good example of customization aimed at adapting the enterprise to the software, following the *'best practices'*, and not the other way round.

The main 'modeling' work which must be done in the implementation part of the SAP methodology takes place during the second phase of the SAP Business One Accelerated Implementation Program, the *Business Blueprint*. This is not exactly 'true' modeling however, both because it does not use any formalism, and because the main document provided to support the customization stage is a 29-pages template [SAP08, B1AIP 2.0, Business Blueprint]. This is the accurate completion of this document by both the customer and the consultant which will determine how the data and the processes must be handled by the software, and which will also give a common vision and understanding of how the software will support the business processes.

The Business Blueprint template covers 19 sections designed to determine all the customizable aspects from SAP Business One. These sections comprise pure data information such as, for example, the definitions of the general company settings, the currencies, the Chart of Accounts and the General Ledger, or the bank information. But this document also contains some details about the business process variations to determine. These variations are the ones referred as *Configuration Choices* by [Aal06] or [GOT07] for example. some of these configuration choices are : the inventory valuation method for new item file records, the ability to allow a stock release without a defined price, the standard posting periods, customer activity alerts, and many more [SAP08]. The entire document must be fulfilled carefully in order to tailor the ERP software to the enterprise needs, but the SAP methodology used to implement

the SAP Business One solution does not cover any long or important modeling step, apart from the fulfillment of the *'Business Blueprint'* template.

5.7 Preliminary conclusions for the modeling concerns

We reviewed through the former sections that the SMEs particular position, constraints and needs are limiting the modeling aspect to its simplest form, mainly for the cost reasons addressed earlier, both in terms of time, and manpower needed for this part of the implementation. Indeed, most of the modeling seen in the academic literature are aimed at bigger enterprises, and a complete modeling of the business processes could involve a high additional cost, which may be simply too important for smaller companies. Apart from that, the SAP Business One Accelerated Implementation program from [SAP08] summarizes it with a single template covering all the aspects of the software. This is understandable for this precise solution, because the software must be installed in its entirety, and is customizable in limited ways, mainly due to its own nature of being a lighter ERP version, particularly aimed at SMEs. But this can be a flaw to 'force' the customer to buy the ERP solution in its entirety if he does not need the entire set of functionalities available in the software. This is also one of the reasons which helped SaaS solutions such as SAP ByDesign to become more and more popular these days.

Some modeling steps could still be conducted though, without resulting in a too high additional cost, and with a relative importance in terms of both goal and business process modeling :

- Goal Modeling
  Goal Modeling can be conducted in the earlier stages for the various reasons exposed in the corresponding section. This could be used as a tool to clearly and completely elicit the requirements, and refine the main concerns on which the members of the implementation team will have to focus their attention.

- Business Process Modeling
  Business Process Modeling should not be conducted for each business process of the SME. At most it can be used to analyze a precise business process which do not fit the functions supported by the ERP, and could require an add-on to this precise ERP solution [SOF03]. On another hand, what could be an asset for an implementation is a complete and precise repository of the available functions, and more important, configuration and run-time choices of the ERP software, designed by the ERP vendor, and reusable for every implementation project, because it should represent the entire set of possibilities for the process, and give a clear understanding of the process to the implementation team members.

Section 6 :

Performance of ERP implementation projects

> **Content :** *Setting up an ERP system implies high costs and a lot of resources. A failed implementation can have disastrous consequences for a company, and enterprise top management want to have guarantees on the benefits of the ERP system. Two performance evaluations can be conducted about performance concerns. The first is a performance assessment, based on evaluation techniques, prior to the implementation. The second is performance monitoring, in which other techniques can be used to verify how the system performs. This chapter will present these two aspects of the performances of ERP systems.*

6.1 Two stages for ERP implementation performance

The performance assessment of ERP system is presented in this section as a twofold process. A clear distinction must be made between :

- Performance assessment prior to the implementation, conducted to evaluate the adequacy of the ERP solution considered, and determine how the ERP system will answer to the enterprise requirements.
- Performance monitoring, based on the objectives defined earlier, to evaluate how the established ERP system performs in the enterprise, and possibly improve it later.

These two sides of the process will be detailed in the next two sections.

6.2 Performance assessment

The first concern about performance of ERP systems takes place in the first days of the ERP implementation project. Performance assessment techniques can even be conducted before the final adoption of a specific software solution by the enterprise, to determine the adequacy and potential advantages of a software solution over another, or in regard to the present system. Performance assessment prior to the implementation is meant to assess how the software will answer to the enterprise specific needs. As such, it will involve a financial analysis, and an analysis of the enterprise and its goals.

6.2.1 Financial analysis

Little research has been made on the analysis of return on investment and different price models in ERP acquisitions [EST07]. This calculation should be made in some way however, during the acquisition phase. SMEs do not have the same budget as bigger enterprises implementing larger ERP solutions, so the insurance of a good ROI is of paramount importance for them to start a project of this magnitude. There are several methods for evaluating the performance of an ERP system, or an IT system in general, but as pointed out in [SOL04] and [WEI07], financial methods are the first to be taken into consideration by the enterprise top management.

Such financial analysis, like *'turnover, return on investment (ROI), and so on' [WEI07]* will generally need to be conducted, even if conventional financial methods of calculation are generally considered inappropriate for IT benefits calculation on their own [WEI07]. Several flaws in these methods are stated to explain this :

- They lack qualitative analysis and only produce financial results
- They do not estimate the non-financial benefits of the software, such as for example, customer satisfaction or options for future growth [LYP04, WSW05]

An example of ROI calculation tool used in SAP Business One implementations is presented in Figure 6.1.

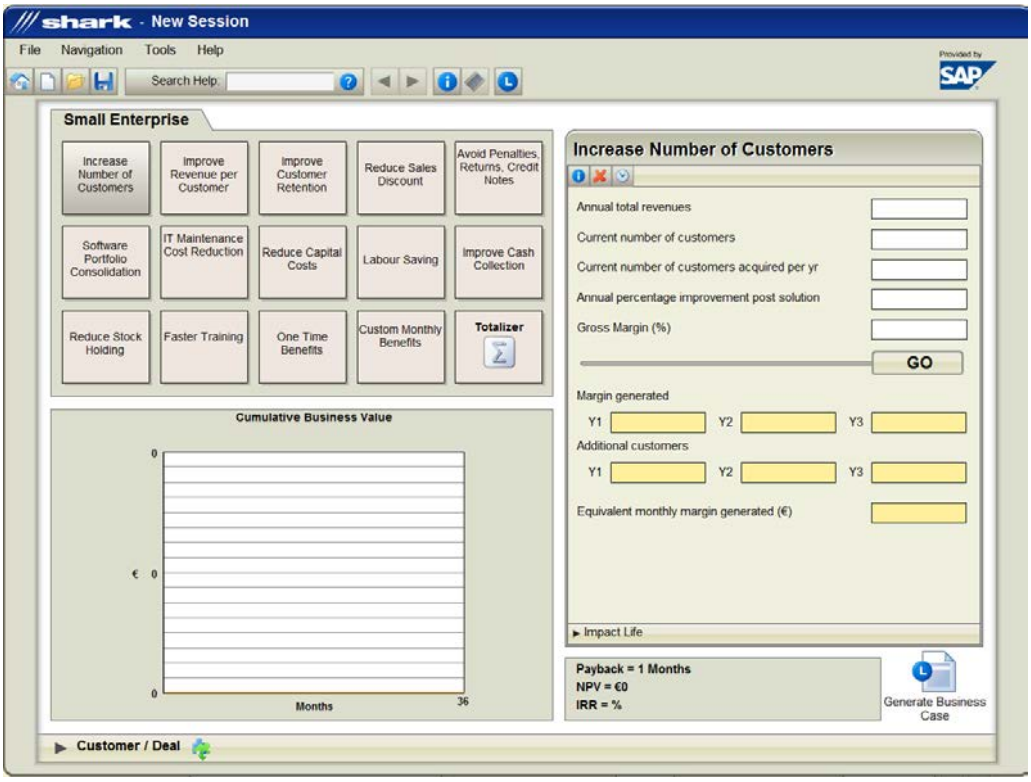


Figure 6.1: ROI Tool for SAP Business One, from [SAP08]

This ROI tool has been designed to produce as quickly as possible an estimation of the financial business case, and estimate the financial benefits of the ERP solution to the SAP customer. This tool is composed of fourteen strategical objectives detailed in Table 6.1, for which informations about the enterprise must be entered. These informations will then be used to calculate projections on the financial benefits for the enterprise, on a given and customizable timeframe, as illustrated in Figure 6.1. The user firstly enters the project general data, such as the review period, the solution cost or the monthly costs, and then needs to fulfill each category from Table 6.1 with the relevant data.

Increase Number of Customers	Improve Revenue per Customer
Improve Customer Retention	Reduce Sales Discount
Avoid Penalties, Returns, Credit Notes	Software Portfolio Consolidation
IT Maintenance Cost Reduction	Reduce Capital Costs
Labour Saving	Improve Cash Collection
Reduce Stock Holding	Faster Training
One Time Benefits	Custom Monthly Benefits

Table 6.1: SAP ROI Tool Categories

By entering information about both the actual and the presumed to-be situation after the implementation, under each section of the tool, it calculates the financial benefits month after month and produces the respectives figures, as well as the total financial benefit, and a graphical representation.

Financial methods are insufficient to produce a relevant performance assessment of a future ERP system. For previous stated reasons, other techniques should be used to evaluate the performance of ERP or other IS. The next section will present an ERP assessment framework highly tied to goal modeling.

6.2.2 Non-financial analysis

The framework proposed in [WEI07] is designed to link the strategic goals of the enterprise with the performance indicators (PI) defined for the project, and its viability is illustrated via an empirical case. As an introduction to the framework, the various IT performance assessment methods are reviewed, reminding the difficulty to financially estimate all the benefits of ERP software implementation. Indeed, the problems encountered by enterprises vary from one case to another. A structural necessity in the enterprise can also be a sound reason to implement an ERP software, and financial benefits may not be the most important way to determine the need to spend money in ERP software [PEE04].

The strategic goals considered by an enterprise can thus be different for every project, and the success of the ERP implementation will depend on several criteria, according to their specific needs and strategic objectives. The author also points out several common problems for a unified performance theory about ERP software, such as performance indicators inconsistently defined, or unappropriate calculation methods like the financial methods addressed earlier [WEI07].

Only specific goals, parallel to the business goals, are always the same : the goals related to the project management. Thus when evaluating the success of the implementation, the cost and time of the project must be kept in mind, along with the other business goals. These two objectives can be treated apart from the other business goals and process indicators. This

shows that the success of the implementation partly relies on costs and benefits during the whole project, along with business goals. The implementation will only be considered successful if both the intended business goals and the more direct-costs goals are satisfied. Figure 6.2 illustrates the way used in [WEI07] to categorize the objectives in regard to their Performance Indicators and relative importance for the whole project.

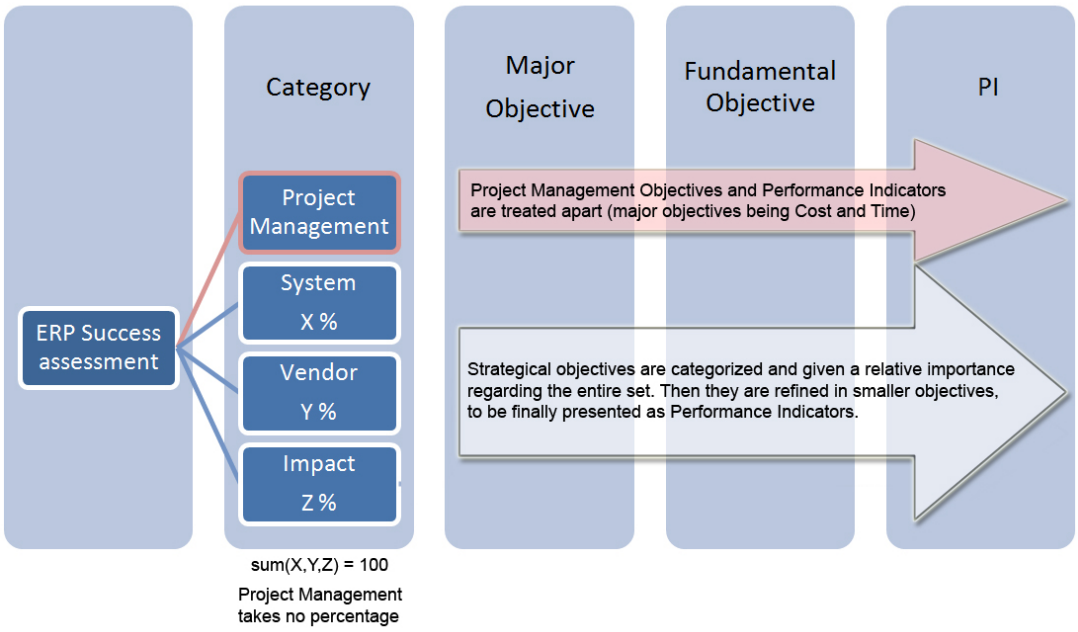


Figure 6.2: Illustration of the categories of objectives and PIs in [WEI07]

Such framework requires a goal analysis at several levels, and the intended goals should be categorized and measured. Several questions then arise for such analysis, like how to categorize the goals, and under which dimensions. The next section will present the goal taxonomy proposed by Magnus Gammelgård [GAM06] and the balanced scorecards by Kaplan and Norton [KAP92, KAP96], which partly answers that questions.

6.2.3 Goal assessment for IT investments

Few publications have been made to detail generic goals for SMEs or enterprises in general. However, Magnus Gammelgård propose such a taxonomy in [GAM06]. The author points out the differences between the business vision and the technical vision. Indeed, when considering goals to achieve, the business management mainly think in terms of *'higher margins'*, or *'customer satisfaction'*, while the technical vision is more oriented towards technical qualities, such as *'availability'*, *'performance'* or *'security'* [GAM06].

Through an extended literature study, the author proposes a categorization of *'IS/IT-investments benefits'* which results in twenty-five categories of business values, organized in three major groups. These categories, designed to cover all possible business benefits from IT/IS investments, are illustrated in Table 6.2. These are reused in a further work, as a part of a management assessment framework, to evaluate enterprise architecture scenarios [GAM07].

In [GAM07], the business organization is assessed under these three main dimensions, as well as the IT Organization and the various IT systems of the organization, to compare various

Business as a black box	Resources of the business	Structure of the business organization
Inbound logistics	Decision making	Strategy formulation and planning
Supplier Relations	Learning and knowledge	Efficiency
Customer relations	Organizational culture	Productivity
Lock-in effect/switching costs	Information	Cost reductions
Competitor relations	Technology/tools	Communications
New products/services		Flow of products/services
Differentiations in products/services		Control and follow up
Quality of products/services		Change management
Deliveries		Integration and coordination
Third party relations		Flexibility

Table 6.2: Gammelgard Business Value Categorization, from [GAM06]

enterprise architecture scenarios.

The three main categories to classify the business values are firstly detailed in [GAM06], and more recently described in [GAM07] as :

- Inputs and outputs of the organization : the business is viewed as a black box, and the goals related to this dimension are either related to the inputs (products and services from suppliers) or the outputs of the enterprise (products and services the enterprise delivers to its customers). Examples of goals within this dimension are : improved products quality, improved supplier relations or improved customer relations.

- Organizational resources : two types of resources are outlined : human-related resources and non-human resources. Examples of goals within this dimension are : improved decision making, improved learning and knowledge for the first type, and improved information, in terms or availability or accuracy for example, for the second one.

- Organizational structure : related to the departments and business units of the enterprise, their structure and how they are linked to each other. These goals can be related to individual components, or several components and their relations. For example, define a more efficient business process, achieve improved communications between several business units, better control over them, and so on. Finally, when considering these relations between the components, there is a last group of goals, related to the structure change management.

These three main dimensions, used to assess the business organization performance, can serve as a basis to determine strategic objectives. The interested reader can find detailed information about all the strategic goals in [GAM06]. Another technique which deals with the goal analysis concerns is the balanced scorecard.

6.2.4 The Balanced Scorecard

Balanced scorecard (BSC) is a performance planning and measurement framework proposed by Robert S. Kaplan and David P. Norton in 1992. The authors proposed this framework as a solution to the shortcomings of traditional financial measuring methods. It relies on financial and non-financial measures through four different perspectives in order to define and monitor the strategical objectives of the enterprise. BSC should not be seen as *a replacement for financial measures*, but as the *complement* for such methods. [KAP96]

These four perspectives are :

- *Financial Perspective :* usual way to measure the performance of an enterprise, generally the first perspective taken into consideration.
- *Customer Perspective :* related to the way the enterprise is perceived by its customers.
- *Internal Business Perspective :* related to the business processes of the enterprise.
- *Innovation and Learning Perspective :* considering the possible means to improve and create value in the future.

For each perspective, there is one column *'Measures'* in front of the column *'Goals'* (see Figure 6.3). Every objective defined in one of the perspectives of the balanced scorecard must be controlled by an indicator which will be used through the entire process to measure the performance of the enterprise under this precise goal [KAP92].

One of the most interesting aspects of the balanced scorecards is in the so-called *'Balanced Scorecard approach'* illustrated in Figure 6.4. This approach consists of defining all the relevant objectives for the enterprise under the four perspectives of the balanced scorecard, and then breaking them down into several, lower level objectives, through business units and *eventually each person* in the enterprise. [GRA07]

It can be hard to simply break down the measures used to assess the strategic goals into lower level measures however, either if the goal itself cannot be broken down, or if the organization of the enterprise does not allow it. For example, a 20% ROI at a strategic level can not simply be broken down in a 20% ROI for each business unit in the enterprise. Some departments like consultancy are able to produce indeed a far higher ROI than other units like real estate. [GRA07]

The risks are high if the establishment of the BSC is not correctly conducted, but thanks to such analysis, the top management produces a complete controlling system over the enterprise and its objectives. Furthermore, the BSC logic aimed at dividing the strategic goals into lower level objectives can help the *personal goals* to be perceived as their own objectives by the people of the organization, for the greater good of the enterprise, and not just to satisfy a top management decision.

Originally perceived as a performance monitoring tool in the early publications of 1992, the BSC started to be considered as a strategic management system in the second major publication by Kaplan and Norton, in 1996. [KAP96] The authors indeed studied more than a hundred organizations, and realized that the top executives in the considered organizations used the BSC as a framework and focus to manage their business processes. That is an achievement that *'no purely financial framework could do'* [KAP96].

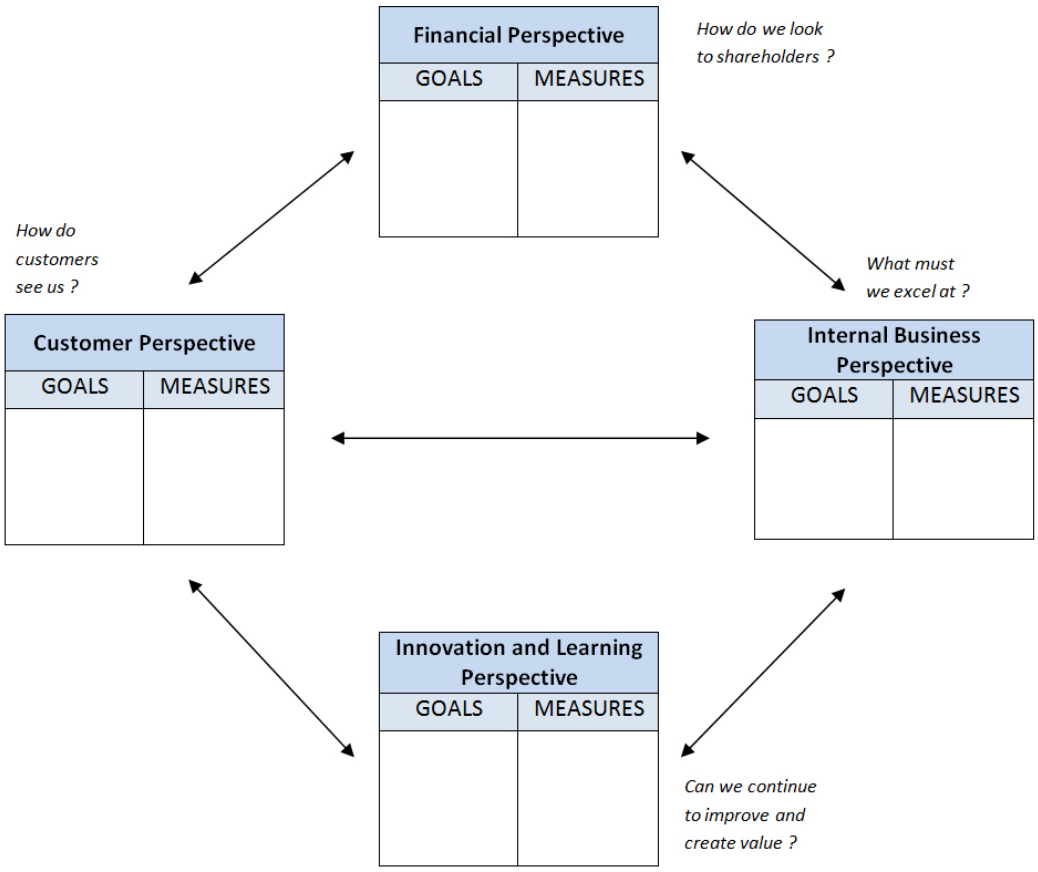


Figure 6.3: The Balanced Scorecard [KAP92]

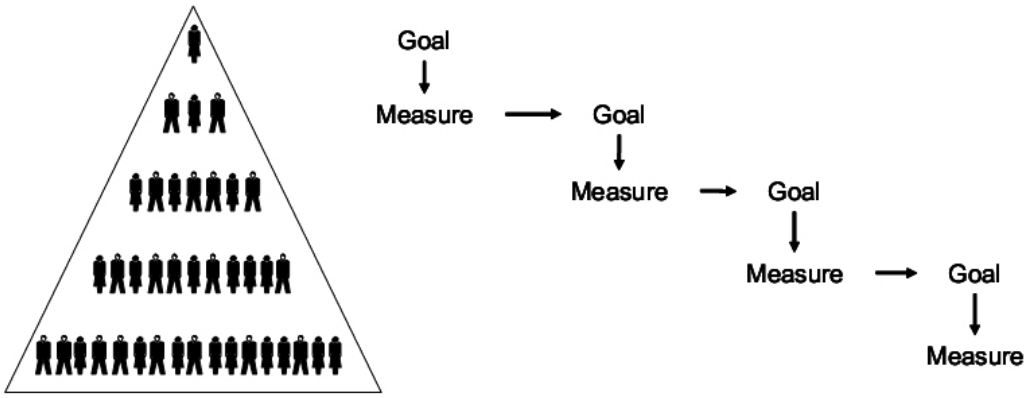


Figure 6.4: Several levels of abstraction for BSC, as illustrated in [GRA07]

6.2.5 Non-Financial analysis for SAP Business One

The whole performance assessment study must be conducted during the first part of the SAP Business One methodology, namely the 'Sales Navigator', during steps C, D, E and F illustrated in Figure 3.7. The available tools for these steps presented as modeling tools in the previous chapter partly serves the adequacy determination and performance assessment tasks :

- The *VIP maps* can be used to determine the strategic goals and partly establish the Key Performance Indicators (KPIs).
- The *Qualification Map* determines the adequacy of the software solution to the business case, through the means of the final score calculated by the information provided.
- The *Opportunity Qualification Tool* helps to determine the adequacy of the solution by answering questions over the industry segment, the organization size, or the number of modules to be used.

6.3 Performance monitoring

Performance monitoring is another aspect of performance measurement in ERP implementations projects. Performance assessment and goal modeling should be conducted by the beginning of the project, and be followed later by performance monitoring. Adequate performance monitoring during and after the implementation will give a mean to control the achievements of the planned objectives, and verify that the project can be qualified as a success.

This other kind of performance issues can be addressed by non-financial techniques addressed earlier. The balanced scorecards by Kaplan and Norton [KAP92, KAP96] indeed proposes to provide precise measures for every goal in the scorecard, and [WEI07] suggests to value each goal in the lowest level with a corresponding Performance Indicator (PI). On the SAP side however, there is less support for performance monitoring.

There is no recommendation in the *Business One Accelerated Implementation Program* to monitor the business processes of the enterprise, or to use measurement or KPIs in any forms during the implementation itself. The only stage close to that is one of the project tasks in the *'Go-Live and Support'* part of the methodology. SAP considers a task of *'Continuous improvement'*, composed of both *'Monitor the system'* and *'Conduct Review and Optimization Conference'*. This part must be concluded at the end of the *Go-Live and Support* part, just before the end of the project. For this stage, the consultant is provided with a *'Post Go-Live Questionnaire'* to be fulfilled by the customer to *'document the results and possible issues of the initial SAP Business One Go-Live'* [SAP08, Post Go-Live Questionnaire]. This questionnaire goes through several components of the ERP implementation success, such as the results for the project management, system management, system administration, user acceptance, user experience or documentation. Apart from that evaluation, there is no recommendation about continuous monitoring of the system, in order to monitor the achievements or propose some improvements.

Part II

Methodology for ERP Implementation in Small and Medium Businesses

Section 7 :

Life-cycle of an ERP project

> **Content :** *After reviewing all the possible structures for ERP Implementation life-cycles, this section will elicit the pros and cons of every possible framework and provide a synthesis with a unified life-cycle model.*

7.1 Proposed unified life-cycle

ERP life-cycles have been continuously evolving in the academic literature since the first ERP life-cycle from Markus and Tanis [MAR2K] to the life-cycle exposed in Esteves and Pastor's works. Several things vary from one life-cycle to another : the number of phases, the tasks to be conducted during every of them, or the notion of dimensions, to represent the various points of view to be taken into account while considering the implementation of ERP software.

Furthermore, in parallel to the theoretical life-cycles proposed in the academic works, every ERP vendor comes with its own techniques, methodologies and schedules, even changing from one ERP version to another.

The life-cycle presented in this section, and illustrated in Figure 7.1, has been designed through the comparison and synthesis of seven life-cycles, six of them being theoretical academic life-cycles [EST99, MAR2K, ROS2K, SOM04, SWA04, SOJ07], and one vendor-specific life-cycle, the ASAP life-cycle for SAP Business One [SAP08]. Two milestones were common to

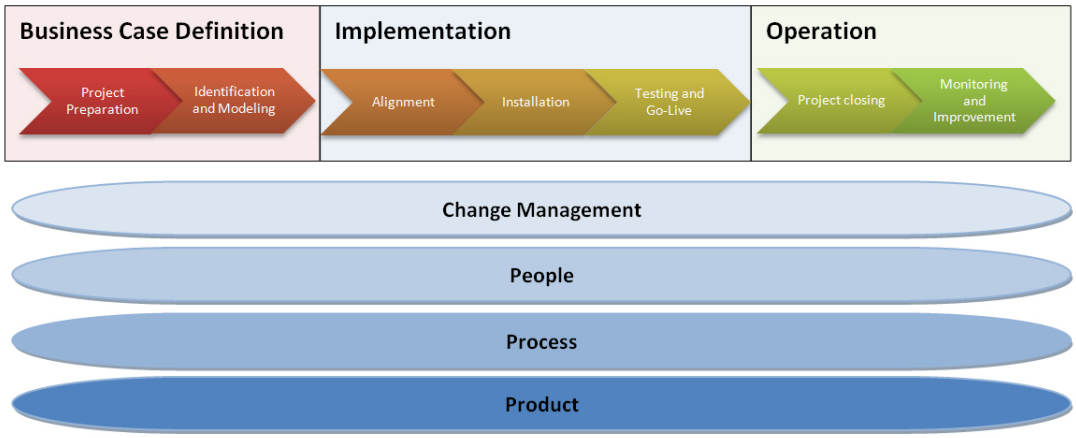


Figure 7.1: Life-cycle for ERP software implementation

every life-cycle : the first being the definition of the business case and schedule, prior to harder expenses and installation in itself, and the second being the most important deadline, when the ERP goes live in production. These two divisions served as the base for this unified life-cycle with the first three main parts of an ERP implementation project, as shown in Figure 7.2.

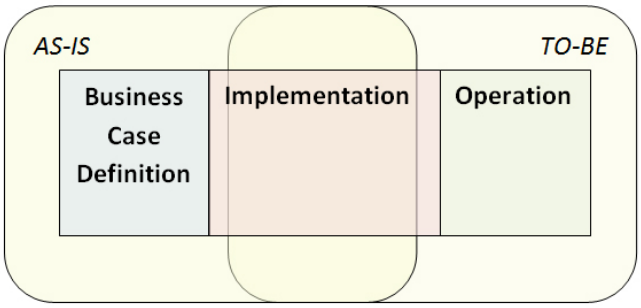


Figure 7.2: The three main parts of an ERP implementation project

In the enterprise architecture world, terms commonly used to describe the system states are the TO-BE and the AS-IS situations. The AS-IS situation describes the system as it is prior to the Information System implementation, with its flawed processes and problems, while the TO-BE situation is the intended situation to be achieved by the implementation team. The problem with this description is that some implementation methodologies consider running the two systems together, in order to determine how the ERP performs in the enterprise, in regard to the established system.

There is thus three major stages during an implementation life-cycle, a first part while the enterprise system does not change but must be analyzed, the final part when the older system has been replaced by the new ERP system, and an intermediary state, being the implementation while the system is progressively replaced, according to the implementation strategy defined at the beginning.

Refining these three main parts in a second step was a harder task. The comparison table is illustrated in Appendix B to show the various relations between all the life-cycles. Every life-cycle had several stages, ranging from four to nine, depending on the level of details. Furthermore, these stages often have different names and for some of them, stages meaning the same tasks have different names. The *project* stage from [MAR2K] and the *implementation* stage from [EST99] contains the same tasks and milestones, but do not have the same names.

The first division illustrated in Figure 7.2 was pushed further to detect a more detailed vision for every of the three main parts. The goal was to find a compromise between the least detailed life-cycle from [MAR2K] and the most divided life-cycle from [SOJ07], not to mention the methodology from SAP, which requires two distinct life-cycles. All the stages of all the life-cycles were arranged in the table of appendix B according to the division stated earlier, and analyzed to find a second level of details in the structure. The respective analysis of each of the three parts are detailed in the next section.

7.2 Stages of the life-cycle

As for now, the unified life-cycle only contains two distinct major milestones. The first thing to do is to precisely define when the project begins and ends, as even these vary from one

life-cycle to another. Next, other milestones during the project must be found, to establish a second level of details throughout the project, by the means of the analysis and comparison of all life-cycles.

7.2.1 Business Case Definition stages

Some life-cycles do not question the *need* of a new ERP system while considering the implementation. Indeed it is not a part of the implementation itself, although some important concerns such as *'the definitions of systems requirements, its goals and benefits and an analysis of the impact at a business and organizational level' [EST07]* should be addressed while considering ERP software. This comes from the various perceptions of the term 'implementation'. As reviewed in the state of the art, SAP considers two life-cycles for a complete implementation of its SAP Business One solution, the first one treating business visions and enterprise modeling issues, while the second one only deals with the implementation concerns [SAP08]. Clearly, the ERP implementation project starts during the first of these life-cycles, with some important requirements elicitation tasks. The questioning about whether or not adopting a new ERP software solution must be done by the enterprise before any contact with the ERP vendor company. After that, the goals and flaws can be elicitated together with the vendor consultant, but at this stage, when the two stakeholders start to think together about the impact of the software, some work is already being done, and can be considered as a part of the implementation. For these reasons, the proposed life-cycle will include such requirements analysis tasks, and begins with the first requirements elicitation tasks. The distinction for the business case definitions stages is illustrated in Figure 7.3.

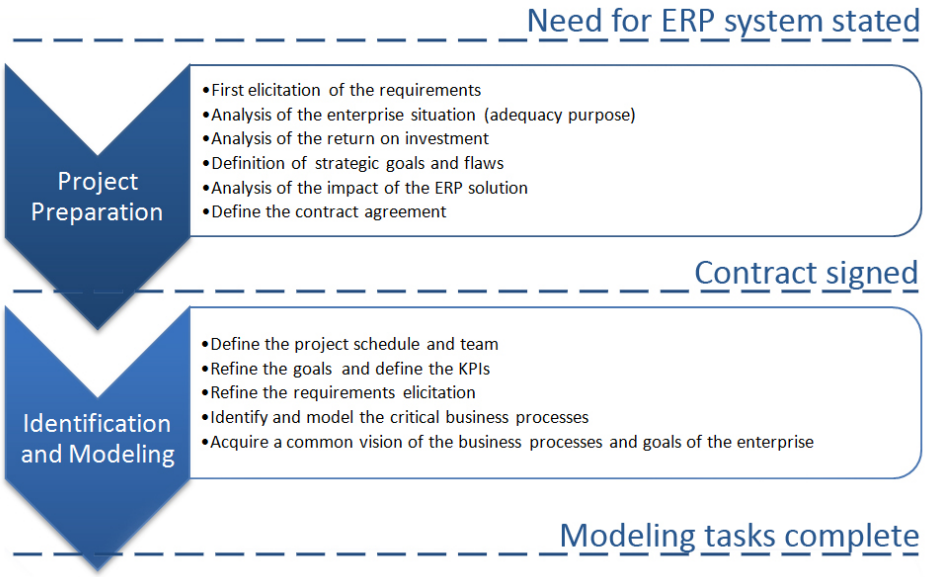


Figure 7.3: Business case definition implementation stages

The business case definition ranges from the time ERP adoption questioning has been made, through the complete definition of the business enterprise parameters and customization. Two levels of detail can be distinguished during this part of the project. Indeed, the modeling tasks are distributed before and after the contract signature. The tasks to be conducted before the

contractual agreement are focusing on getting the *'big picture'*, while the second are dealing with all the detailed aspects of the enterprise to be modeled, according to the customization which needs to be done.

This distinction based on the contract signature seems relevant while considering the various life-cycles reviewed earlier. The *chartering* phase from Markus and Tanis for example, contains all the *'decisions leading up to the funding of an enterprise system' [MAR2K]*. [EST99] also make this clear distinction in the *acquisition*, while the *'contractual agreement is defined'* after selecting the product that *'best fits the requirements of the organization'*. The SAP Business One Sales life-cycle also ends with the signature, and the start of the *Accelerated Implementation Program* [SAP08]. It should be noted that the modeling stages needed after the contract signature are sometimes considered as part of the large *'implementation'* or *'project'* stage [MAR2K, EST99]. In this work, the distinction is made between the modeling tasks which needs to be done to precisely define the Business Case to customize the software, and the implementation tasks which are going to be done after the whole system has been modeled.

7.2.2 Implementation stages

This part of the implementation goes from the time the business case is defined, to the time the ERP solution goes live in production. Based on the modeling realized during the earlier stages, the software must be tailored to the enterprise needs, and the enterprise must adapt its business processes to the new software.

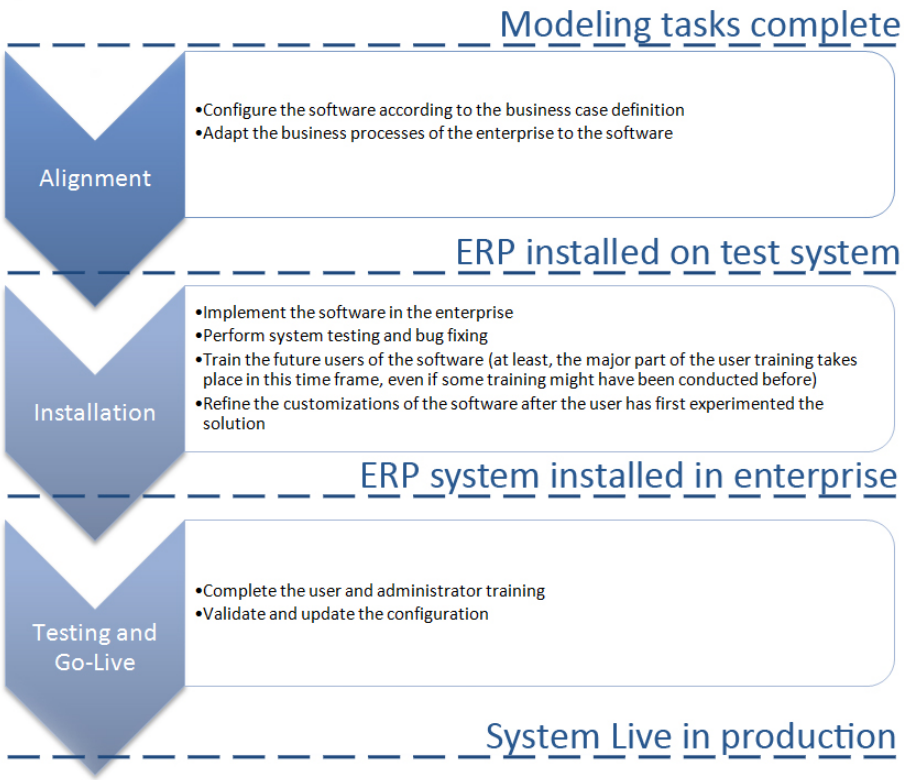


Figure 7.4: ERP Implementation stages

The precise limits between the stages of the implementation are harder to define, because most life-cycles consider the implementation as a whole and do not make any division. The order of the tasks to be done are undefined or varies from one life-cycle to another, some life-cycles consider tasks that others do not, such as the general training on ERP systems from [SOJ07] for example. The chosen distinction is illustrated in Figure 7.4 and detailed as follows : first the software must be configured to adapt to the business case definition, then the ERP software should be installed in the enterprise IT system, to finally enters a last stage during which the whole organization should be prepared for the *Go-Live*.

7.2.3 Operation stages

The possible division between the *operation* tasks was easier to find out. It relies on the period of time after the *Go-Live* during which the system does not show its full potential, and needs some minor adjustments for the enterprise to *'come to grips to the new enterprise system'[MAR2K]*. The distinction for the final stages of the ERP implementation life-cycle is illustrated in Figure 7.5.

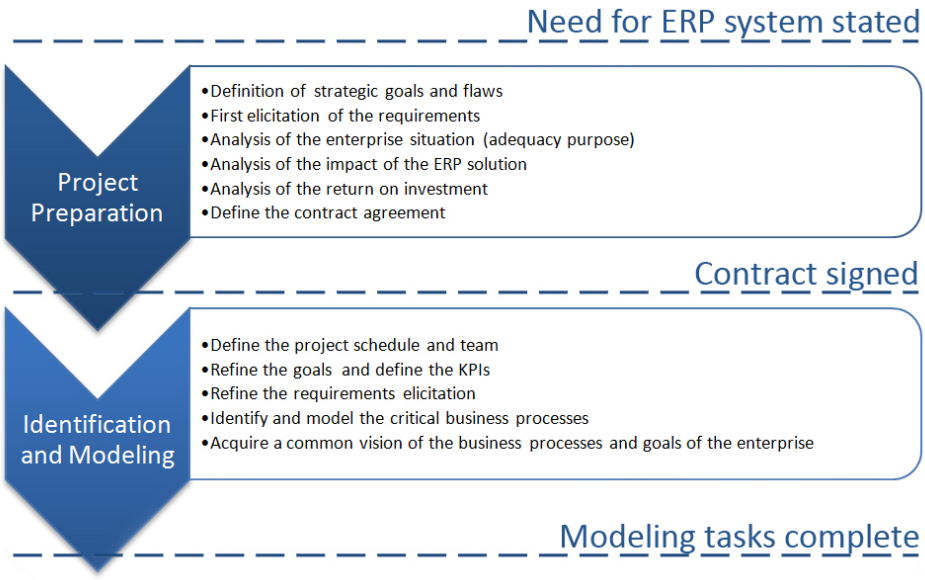


Figure 7.5: Operation implementation phases

During this part of the implementation, the system has been launched and is used by the enterprise in production. The implementation is not completely finished however, because a few more tasks need to be conducted. Right after the implementation, some more user training could be conducted again if new adjustments should be made, or usage difficulties should continue to be encountered. Right after the go-live, there is always a short period while the users get used to the system [MAR2K, EST99, SAP08], and the ERP system newly implemented does not show all its potential. This period, named *'shakedown' [MAR2K]* or *'system start'[SOJ07]* should be relatively short though, and minor issues should be quickly solved. The implementation consultant must also be available to provide immediate *Go-Live support* during the first days of use of the ERP software, as this period is critical in the process for the usage reasons explained earlier.

After the implementation of the software itself is completed, and according to the contract agreement defined for the project, begins the Monitoring and Improvement phase. The tasks to be conducted within this time frame are mainly to collect information about how the software is performing in the enterprise, and to improve it accordingly, whenever it is possible. The length of this period as well as the means used to collect process performance information can vary, depending on the service level agreement.

7.2.4 Dimensions of the life-cycle

While the other life-cycles presented in the state of the art do not express the dimensions of the various stages, and refer to various dimensions without mentioning them precisely, the works from Esteves and Pastor use such notion of *'dimensions'* in their representation of the ERP implementation life-cycle. These are particularly useful when trying to categorize various concepts, like for example the CSFs, or a research agenda on ERP topics, like in [EST07]. The various life-cycle dimensions presented in [EST99] for ERP implementation projects thus need to be considered for an integrated life-cycle. These four dimensions represent the main perspectives for the entire implementation.

1. the *Product* dimension considers the ERP solution which is being evaluated, focussing on issues like the alignment of the software, the technical constraints implied on the organization, or the availability of precise functionalities or processes.

2. the *Process* dimension considers the enterprise processes, and every process reengineering which will have to be done to implement the software more easily.

3. the *People* dimension considering the users, the implementation team on the enterprise side, and the way they perceive the ERP solution. The support of both the users and the management is of paramount importance to facilitate the implementation.

4. the *Change Management* dimension must be considered right from the beginning of the implementation to the end. The implementation of an ERP software solution in an enterprise is a particularly important project, and generally implies a lot of resources, and a lot of change in the business processes of the enterprise. It is then important to be assured of the acceptance of the ERP solution and process changes by the users and managers within the enterprise.

7.3 Discussion on the unified life-cycle

This life-cycle does not pretend to replace the other works from which it is inspired, but proposes a complete view of the various steps to be done, synthesizing all the other life-cycles to draw the best of every of them, while staying as short and clear as possible. To the various steps composing the life-cycle, the *dimensions* as seen in [EST99] must be added too, in order to achieve a complete vision of the system and to be able to refer to every different point of view for each stage of the life-cycle. This make sense if one tries to place a particular notion in regard to the life-cycle stages and perspectives. The CSFs in the next section are a good illustration for such classification.

Section 8 :

Critical Success Factors in ERP Implementations

> **Content :** *The chapter covering the CSFs in the first part of this document addressed the main CSFs to be used during ERP implementation projects, as well as the various ways to categorize them. Furthermore, it addressed the use of CSFs in SAP Business One implementation projects. This part will try to elicit the important CSFs to consider in SMEs, and to associate them to one or more stages in the life-cycle, according to their relevance.*

8.1 CSFs use in SME projects

Through the reading of the various CSFs unified models, such as [HOL99, EST2K], as well as the studies around that matter such as, for example, [EST03, NLK01, SOM01, ZHA03, ZHA05], it can be noted that some CSFs are more important than others, and that the CSFs in general must not be neglected while dealing with ERP implementation among SMEs. The most important CSFs will be detailed in this section, while the next section will try to associate them with each phase of the life-cycle, according to their relevance.

However, the statement made in [EST01] must be reminded to the reader : the identification of the most critical process of the whole implementation *should be done with information provided by cases studies, and not only based in theoretical assumptions [EST01].* Consequently, the most critical CSFs for an entire ERP implementation project cannot be deducted from purely theoretical works.

8.1.1 Contextual factors

The importance of the context in which the SME stands is mentioned in several works and must be taken into account before the more critical, internal factors. Generally, it is not such a big problem in European SMEs than for SMES situated in the Asian market for example [ZHA03, WWW03], because the implementation methodologies are designed for the European or US context. However, studies like [EST03] have proved the need to analyze the context of the SME considered, even in Europe or in the United States.

The three main environmental factors to be considered here have been addressed in the state of the art :

- Power distance
- Uncertainty avoidance
- Individualism/Collectivism

These factors proposed by Hofstede in 1991 [HOF91] have been studied in works like [EST03], pointing out the need to observe them to reinforce the success chances of the project. These factors, related to the people dimension are far more less studied then the other more *'classical'* factors however. Their scope ranges through the whole life-cycle, but their impact on the implementation success is harder to define. These factors can have serious consequences on the project characteristics, such as its duration, or the team composition for example.

8.1.2 CSFs to be considered during ERP implementations in SMEs

After reviewing the contextual factors, we can consider the CSFs and the various unified CSF models. The most famous unified CSF model is the one from Esteves and Pastor [EST2K] illustrated in Figure 4.2. Considering this unified CSF model, along with [HOL99] and the best practices addressed in the ASAP BO methodology, we can draw a table to compare the CSFs appearing in every of these works. This comparison is presented in Table 8.1.

		[HOL99]	[EST2K]	[SOM04]	[ASAP BO]
1	Legacy Systems	X	X		
2	Business Vision	X	X	X	X
3	ERP Strategy	X	X		
4	Top management support	X	X	X	
5	Formalised project schedule and plans	X	X	X	
6	Client Consultation	X	(user involvement and participation)	X	X
7	Personnel	X	(adequate team composition)	(project team competence)	
8	BPC and software configuration	X	X		
9	Client Acceptance	X	X	X	X
10	Monitoring and Feedback	X		X	
11	Good Communication inwards and outwards	X	X		X
12	Reduced troubleshooting	X	X		
13	Project champion		X	X	
14	Adequate project team composition		X	X	X
15	Adequate ERP version		X	(Careful Package Selection)	X
16	Effective organisational change management		X	X	
17	Good project scope management		X	X	
18	Comprehensive business process reengineering		X	X	
19	Avoid customization		X	X	
20	Dedicated staff and consultants		X	X	
21	Adequate training program		X	X	
22	Appropriate usage of consultants		X	X	
23	Empowered decision-makers		X		
24	Vendor Support			X	
25	Data analysis and conversion			X	

Table 8.1: Comparison Table for Critical Success Factors for ERP Implementations

Even if the identification of the most critical CSFs should be deducted by real case data, and not by purely theoretical works, some observations can be done through the analysis of this comparison. From this table, 9 CSFs can be identified as being mentioned in at least three of the four references. These factors numbered 2, 4, 5, 6, 7, 9, 11, 14, and 15 are generally considered in the various works as being the most critical among all the identified CSFs. These factors are :

- A clear Business vision : a common understanding of the goals and objectives aimed by the enterprise after the ERP implementation project is mandatory to succeed.
- Top management support : The top management support during the entire process is of paramount importance. The project could not be achieved without such support on the higher positions on the enterprise side. [SOM04] even states that *'no single factor is as predictive of ERP project success.'*.
- Formalized project schedule and plans : The limited resources of SMEs in general must not be confused with a lack of formalism. The project schedule and plans at least must be formalized in order to be clear and unambiguous for both the consultant and the enterprise sides of the implementation team.
- Client consultation : this CSF can be related to *Good communication inwards and outwards*. Indeed, an easy and quick way to contact the partners in the enterprise can facilitate and accelerate the resolutions of the problems encountered, and ensures to agree with the client during the whole implementation process.
- Personnel : This generic CSF from [HOL99], involving the personnel of the enterprise where the ERP has to be implemented, has been refined in later works into several other CSFs like, for example, an *adequate team composition*, *project team competence*, *client consultation* and *involvement*.
- Client Acceptance : Related to the Change Management topic, this CSF recommends to have the approval of the client for every stage and to avoid conflicts of interests. This CSF is not mentioned 'as-is' in [EST2K] but is a part of *Effective Business Change Management*, which is needed to *'ensure the acceptance and readiness of the new system' [EST2K]*.
- Good communication inwards and outwards : Here again, a good communication is mandatory within the implementation team members, and with the implementation team and the enterprise departments and final users.
- Adequate project team composition : The implementation team must be correctly composed, according to the resources available in the enterprise, and with an adequate proportion of people of the enterprise.
- Adequate ERP version : Of course, the ERP software solution considered must fit the best way possible with the needs, requirements, business processes and functions of the SME. This CSF also mean a careful selection of the modules to be implemented.

The remaining CSFs also need to be considered, but these are harder to categorize according to their impact on the entire project. Most of them also have important consequences, though a few ones are less relevant for ERP implementations in SMEs.

Indeed the particular context and characteristics of SMEs must be taken into account while considering the CSFs. *Flexibility*, *limited resources* and *technical heterogeneity* for example, as cited in [HOY07], must not be neglected and will help to differentiate the remaining CSFs and categorize them.

8.1.3 Other CSFs to consider during ERP implementations in SMEs

As stated earlier, the remaining CSFs are harder to categorize according to their impact on the project. Some of them can have serious impact on the project just like the preceding ones, while other CSFs could be less relevant for SMEs. Indeed, most research and studies on the CSFs topic were conducted with data from large enterprises. This is the case for the works cited in this section. The remaining CSFs will thus be presented on the same level, but with the SMEs particular characteristics in mind. The whole set will then be categorized in the next section, according to their relative importance in the various stages of the life-cycle from the previous section.

- Good project scope management : The project scope management deals with both *scope definition and subsequent scope control [EST2K].* This concern in ERP implementations is not addressed in [HOL99], and not cited as a *'best practice recommendations'* in the SAP BO methodology, but anyway, this CSF is related to the *clear business vision*, and it is recommended to agree on the goals to achieve by the end of the implementation. The scope management can thus be considered as an important CSF too, in order to satisfy the targeted objectives and being able to qualify the implementation as a success.
- Legacy systems : Even if the implementation of a new ERP system will undoubtedly involve an investment in new hardware, and maybe later the purchase of other software related to the ERP solution, the limited resources in SMEs encourages to consider the legacy systems and think about the possible reusability of its former components.
- Reduced troubleshooting : This CSF is logically considered as much in large enterprises as in SMEs. Indeed it is preferable to avoid the problems than to spend too much time to resolve them.
- Avoid customization : This CSF is very important from a SME point of view, because this part of the implementation, when the alignment between the new ERP system and the business processes of enterprise, is one of the most expensive tasks. Customization must be reduced at its minimum, in order to reduce the cost and the difficulty to implement the ERP system.
- Adequate training program : It seems obvious that the training sessions for the new system must be correctly conducted, in order to receive a correct acceptance from the new system by the enterprise's end users. This is true for large enterprises, but also for smaller organizational structures like SMEs. On another hand, *'adequate training program'* also means that this must not last for a too long time. The limited resources prevent the consultant team to impose a huge training time to the enterprise employees. Indeed, while the users train to the new system, they do not use this time to perform their every day tasks, and this should also be avoided. An adequate compromise for this concern is needed for ERP implementations in SMEs.
- Appropriate usage of consultants : The resources available for the implementation should be used properly, and this also relates to the consultant, of course. This is especially true in the SMEs context.
- Empowered decision-makers : The part of the implementation team from the enterprise will often be limited, due to the very nature of the SMEs. Whenever possible and reasonable for the consultant, the people constituting this side of the team must also feel

that their decisions and advices are important. Even if they are not to be given the most strategically important decisions, they should not be limited only to the simplest tasks. Implying them in some important parts of the process will ensure a better motivation on their side, even if the control of the implementation must obviously stay on the consultant side.

- Dedicated staff and consultants : This CSF is very important, especially in SMEs. According to [EST2K], *'the time dedicated to the implementation project is shared with other activities'*, and it is *'important that the staff believes in the project success'*. This is even more critical if the SME context, with this kind of organic structures, where people have a lot of contacts with each other, and can quickly communicate their beliefs and fears. Furthermore, due to the limited amount of human resources available in SMEs to help the implementation, it is important for them to be fully interested and motivated by the project too.

- BPC and software configuration : The particular flexibility of SMEs, *preferring simplicity and flexibility regarding their process and organizational structures [HOY07]*, eases the Business Process Change management and the software configuration

- Monitoring and feedback : Obviously, monitoring the implementation and asking for, or receiving feedback is important to react if a problem had to occur during the implementation, regarding the plans and schedule established during the earlier stages. However it is not as important as in larger, more complex organizations. We already pointed out the differences between the rigidity of large enterprises structures, and the flexibility of smaller, more organic structures among SMEs. A constant monitoring must be undertaken, but in a smaller scale, and without mobilizing as many resources as in larger enterprises.

- Project champion : Because of the reduced size of implementation teams in SMEs, and unless the structure would be so small that only one person from the SME could help the implementation process, it is not reasonable to think about entirely mobilizing one people in the SME as *project champion*, due to both the limitations of human resources available, and to avoid taking all the needed information from only one person. On another hand, the role of *project champion*, being to motivate the people in the organization and interesting the final users to the new system, being a sort of link between the consultant part of the team and the enterprise side, can be assumed by the whole enterprise side of the implementation team, instead of just in one person's responsibility.

- Effective organizational management : An effective organizational management is important, but as exposed previously, the organization is generally more flexible in a SME than in larger enterprises, so logically the organizational management CSF can be placed at a secondary level, when compared to its relative importance for large enterprises.

- Comprehensive business process reengineering : As exposed previously, it is better to modify the business processes of the enterprise to the software, than to try tailoring the software the other way round. This is important not to neglect this aspect of the alignment, of course, but this alignment poses generally less problems in SMEs due to their flexibility and reactivity, than in bigger organizations. A correct business process reengineering strategy must however been agreed upon, at the beginning of the project.

8.2 Relative importance of CSFs

Using the life-cycle from the previous section, it is possible to categorize the various CSFs in the different phases, according to their respective relevance. Figure 8.1 illustrates the alignment between the CSFs and their corresponding life-cycle stages and dimensions. A larger version is illustrated in Appendix D as Figure 3 to express more clearly the various dimensions. Figure 8.1 is presented here instead of Figure 3 for reading concerns, and because Figure 8.1 has been produced through the comparisons and first categorization from Figure 3. In this Figure, the stress is made on the relevance for the various stages, while the Figure presented in Appendix puts the stress on the dimensions first. The remaining of this section will summarize the conclusions about this distribution.

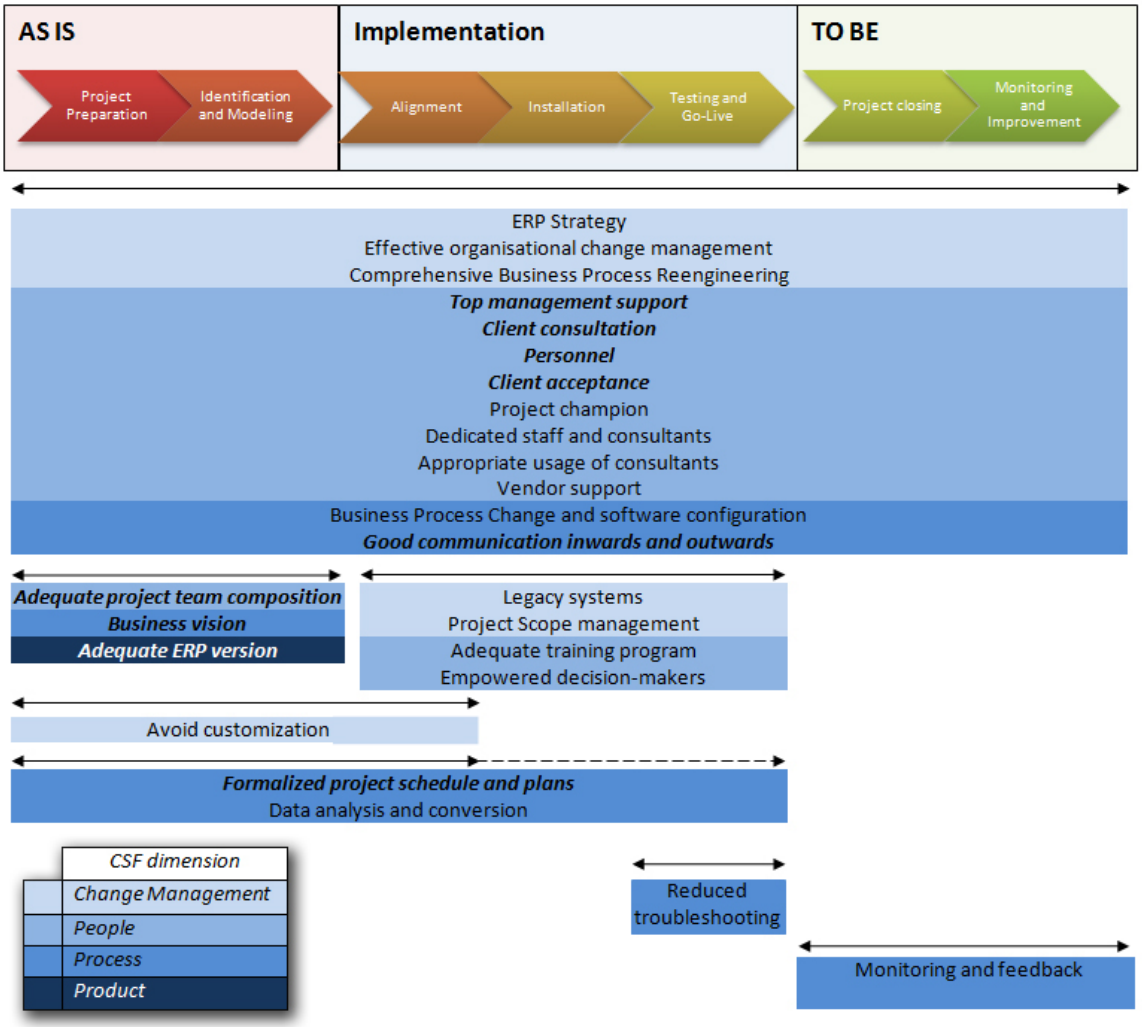


Figure 8.1: CSFs relevance in SMEs according to the life-cycle

Some CSFs need to be considered during the entire implementation process, like the *good communication inwards and outwards*, *regular client consultation*, or the *continuous management support*. Some other ones, on another hand, have a specific time range and are applicable only during that time. *Adequate training program*, or *adequate project team* for example, are only applicable during particular tasks which must be concluded at a particular time during

the project. Generally the dimensions of the life-cycle contains both types of CSFs, in different proportions. For a deep analysis of every CSF relevance regarding every stage of the classical ASAP methodology for SAP Business Suite, the interested reader is invited to consult [EST01] which classifies every critical success factor of [EST2K].

8.2.1 Change Management dimension

The CSFs related to change management are generally centered around the implementation part of the life-cycle. The *legacy systems* must be taken into account mainly during this step, while considering the installation of the new system, the earlier phases of design and project preparation do not deal with this matter. On the same level, the project scope management starts when the project scope has been defined, and this is done during the *AS-IS* part.

On another hand, it is recommended to *avoid customization* during the entire project. [EST01] states that it should *'always'* be *'taken into account while managers are making decisions.'*

8.2.2 People dimension

This dimension contains more CSFs than any other, and most of them span through the entire project. *Continuous management support* has been frequently pointed out as an essential CSF, as well as *regular client consultation*, which must be maintained during the entire project. *Client acceptance*, as well as *dedicated staff and consultants* are also desirable during the entire project. The *appropriate usage of consultants* is also a people-related CSF, as well as an *adequate project team composition*, but these two must be restricted to the *AS-IS* part of the implementation. Indeed the composition is decided during the early stages of the project, and the consultant usage is *'especially relevant in the first phase when the managers have to decide the how, when and number of consultants that they will incorporate in the project team' [EST01].* Finally, the *adequate training program* could be limited to the testing and go-live part, but it would be better to consider it spanning during the entire *Implementation* part, because it must be planned earlier, and the training in general begins before the training of the end users, with the power users and the enterprise side of the implementation team. The review of these CSFs and the fact that the people dimension takes the biggest number of them suggest that this part of ERP implementation projects in SME is one of the most, if not the most, important concern to consider while implementing such software.

8.2.3 Process dimension

Except for the *good communication inwards and outwards* which must be maintained during the entire project, and essential for ERP implementation projects to succeed, all the CSFs addressing the process dimension are relevant only during selected parts of the implementation. Indeed, the process dimension addresses the way things must be done during the project, so logically each stage of the process has its own recommendations and guidelines. For example, it is recommended to ensure a *clear and common business vision* during the early stage of the

implementation, in order to agree on the objectives to achieve and to plan the further steps of the implementation. At the same time, the project schedule and plans must be formalized to avoid being ambiguous and to ensure a better planning of the further activities of the implementation. However, the relevance of this CSF *'decreases during the implementation project'*, as shown in [EST01]. Finally, the troubleshooting steps, hopefully minimal, must occur during the Testing and Go-live step of the implementation.

8.2.4 Product dimension

This last dimension of the life-cycle only refers to one CSF to be considered during the early stages : choosing the adequate ERP version for the enterprise. Indeed, choosing an ERP solution matching the objectives of the SME, and easily customizable to the enterprise requirements, will ensure a reduction of time and effort during the alignment process, the identification and modeling, and all the preliminary steps of the project. Apart from this CSF, no other one has such an important impact on the product dimension.

8.3 Conclusions

CSFs for a successful ERP implementation must be perceived as *guidelines* or *recommendations* more than goals or milestones. Missing one or more CSFs will not necessarily mean a failed implementation but, on another hand, following each recommendation will ensure an easier and safer implementation. Other previous works such as [EST2K, EST01, NLK01] have been done to unify and classify the various CSFs along the relevant implementation phases, for different CSFs and different life-cycles. There is still no consensus in the academic literature on a strict, defined topology on CSFs for ERP implementations, or on a way to classify them, but while this work is not intended to be the universal solution, it provides another look on a classification of relevant CSFs for a successful implementation in SMEs, pointing out the importance of the people involved during the whole implementation and their needed characteristics, with the *People* dimension being the most populated of the four dimensions, directly followed by the *Change Management* and the *Process* dimensions. This should tend to prove that ERP projects are driven by the people conducting it, before everything else. This classification also shows the same results as previous works considering the relevance of CSFs for every implementation stage, the first stages containing far more CSFs than the last ones, when the system has gone 'live', which seems obvious because most problems could be encountered and avoided before the 'Go-Live' step.

Section 9 :

Modeling techniques improvements

> **Content :** *This chapter is aimed at describing the modeling improvements proposed for ERP implementations in SMEs, presenting the proposed modeling concepts and techniques which could be used to improve ERP implementation methodologies.*

9.1 Context and schedule to use modeling techniques

Modeling techniques in general must be used from the beginning of the project to positively impact the implementation. Several advantages will be drawn later during the project, but logically all the modeling work must be done early, during the AS-IS state of the system.

More concretely, the two stages of the AS-IS state of the system are the *Project Preparation* and the *Identification and Modeling* stages, and even if the modeling tasks can have a very positive impact on later implementation stages such as the alignment, the context used to conduct the modeling tasks will be these two stages of the AS-IS part of the life-cycle.

The *project preparation* consists in a first approach of the enterprise system, and as such, the modeling tasks during this part should be minimal and must stay general, while efficient in a small amount of time. The main purpose at this stage is to give a general idea of the possible improvements that must be brought to the enterprise information system, not to get into the details. As illustrated in Figure 7.3, some of the main tasks to be conducted during the project preparation are the *definition of goals and flaws*, a *first elicitation of the requirements* and the *analysis of the impact of the ERP solution*. These tend to confirm the aim to keep an abstract point of view.

On another hand, during the next stage, namely *identification and modeling*, the implementation team must *refine the goals*, *define the KPIs*, *identify* and *model the critical business processes*, among other tasks. The focus must go deeper in the analysis of the enterprise and by the end of this stage, all the modeling tasks, which could ensure a better implementation during the next steps, should have been conducted.

9.2 Modeling techniques

9.2.1 Stressed perspectives for modeling improvements

As stated earlier in this document, several modeling points of view such as the ones from Zachman's framework can be outlined while trying to model an enterprise. The two main

perspectives which require more focus during the early stages of the life-cycle will be the *Why* and the *How* from the Zachman framework [ZAC87] : the goals and the business processes. The other perspectives, such as the *What*, or *data* perspective, are indeed generally largely covered during the implementation, by the enterprise specific methodologies.

This is a quite strong assumption, and this is also true that some other perspectives could benefit from such analysis, but this document will be focused on the the *Why* and the *How*, for the modeling tasks related to the first implementation stages. Obviously the enterprise data structures, formats and objects must be carefully analyzed and the ERP software tailored accordingly, but this is already covered by simple methods and tools provided by the ERP vendor for each solution, with a relative low time and resource investment, in regard to the solution considered and the complexity of the enterprise. Another perspective which could be analyzed and improved by modeling techniques, for example, would be the *Where*, for scalability concerns.

9.2.2 Remarks on data modeling

The stress is not laid on that part of the modeling tasks in this document, but it does not mean that this part of the requirements elicitation is less important. This short section explains in more details the reasons to put aside heavy data modeling techniques while keeping the goals and business processes on the front line.

During ERP software installations, most data formats can be adapted to suit the enterprise needs, and customized fields can also be added to any document used in the enterprise, under the form of User-Defined Fields (UDFs) for SAP Business One, for example. Furthermore, SMEs can not support a deep analysis of every data object in every business which would be supported by the ERP software.

The analysis of the data objects and structures used in the enterprise must be done in some way however, but the notation must be questioned. The notation must be chosen according to the need to formalize the data objects, thus according to the importance of the business processes where the data object intervenes. A deep data formalization will require a class diagram for a large amount of business objects, and the important constraint to keep in mind here is the resource limitations induced by the SME structure. It would be very time-consuming to realize a complete data model for every business object of every business process which would be supported by the ERP software solution. Depending on the data objects to model, it could also be hard to understand from the enterprise point-of-view. On another hand, other UML notations can be used for a higher level of comprehension, like package diagrams for example. This will vary for every different enterprise and can have a serious impact on the time and resources involved, so the data analysis must be conducted carefully, while keeping in mind the resource limitations.

Finally, the data analysis is generally included and fully supported by the vendor methodologies and tools. For example, the *Business Blueprint* document from the *SAP Business One Accelerated Implementation Program* [SAP08] contains a complete range of questions about data formats, average amount of data transferred by business unit over given periods, and every data aspect which will help to configure the system on both software and hardware aspects. This document has the advantage to treat every data concern while staying as fast as possible.

The most critical part of an ERP software implementation project is the alignment process, and this alignment will be driven by the goals, and measured in terms of efforts by the gap

between the *AS-IS* business processes in the enterprise, and the *TO-BE* processes.

9.3 Modeling the Generic and the Specific

To draw a significant advantage from modeling techniques, it is essential to avoid repeating the entire modeling work for every implementation. Instead, a better option would be to benefit from reference models, which would be generic and reusable as starting points for every ERP implementation. These generic reference models would then be used to derive the specific models, tailored to the enterprise needs. This is an important condition for modeling tasks to be efficient and not too much time consuming during the implementation process. Figure 9.1 illustrates the relations between generic and specific modeling, in regard to the whole implementation process.

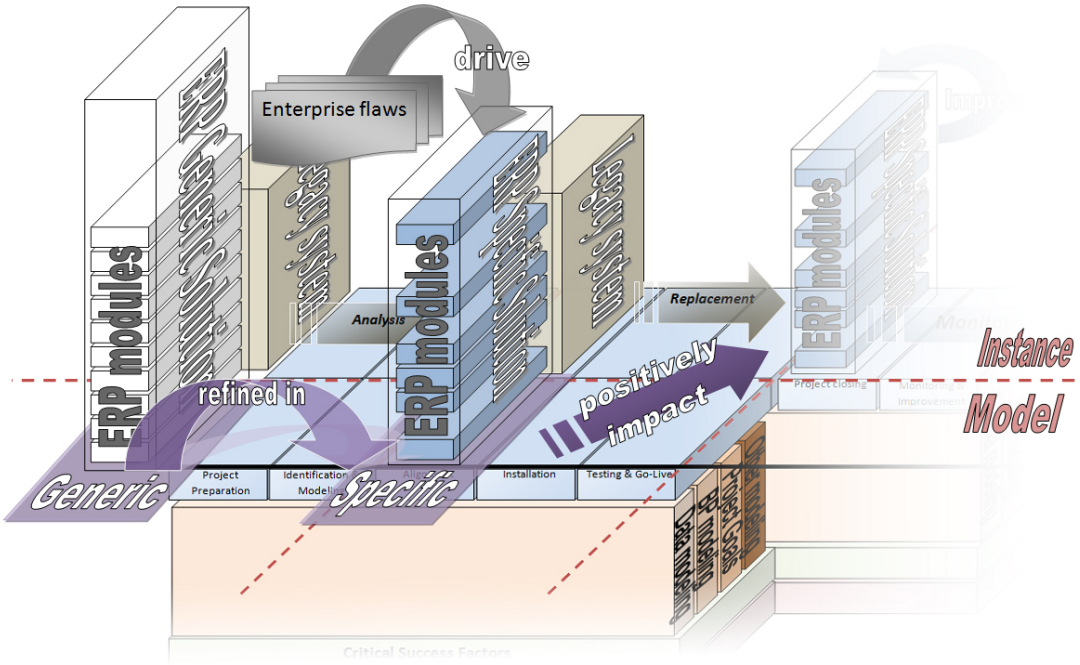


Figure 9.1: Modeling the Generic and the Specific

Generic reference models should be available to the consultant, such as generic goals, or generic business process models, illustrating the business processes support of the ERP solution to be implemented in the enterprise. These two reference models could then be used to derive the specific, customized models, aligned to the enterprise needs.

9.3.1 Performance estimation and performance measurement

The evaluation of the ERP system performance should be twofold, as explained earlier. First, an estimation of the expected improvements brought by the ERP software must be conducted before the adoption of the software, to show the solution's potential to answer to the SME

problems pointed out by the top management. Then there is a need to monitor the performance of the system, according to the specifications defined earlier, to verify how the software answers to the problems the SME was facing prior to the ERP implementation. Only when the milestones considered at the beginning of the project will be encountered by the performances of the system, might the ERP implementation project considered as a true 'success'.

From now on, we stress the two implementation stages from the beginning of the project, during the business case definition. Each one of these stages needs a different level of details, for every modeling technique considered, and the next two chapters will detail the proposed improvements in terms of goal and process modeling, during the two first stages of the ERP implementation.

Section 10 :

Modeling during the Project Preparation stage

> **Content :** *The purpose of this stage is to find an agreement between the enterprise management and the consulting team. Ultimately, the contract should be signed by the end of this stage. The recommended techniques will thus keep a general view of the system. Before detailing the proposed goal modeling technique, the team should ensure the adequacy of the ERP solution before everything else, and a financial analysis of the strategic goals will generally also be done. Then goal modeling is the main modeling technique considered seriously, during the project preparation. Indeed, the generic goals will help to determine the steering objectives of the enterprise : the strategic goals.*

10.1 Preliminary tasks

The goal analysis will generally not be conducted directly. Both the enterprise and the consulting team must first ensure the adequacy of the ERP solution, and the top management will generally ask for a financial analysis of the benefits of the ERP solution adoption. While the relevance of such financial analysis can be questioned, this will often be a prerequisite to discuss about the ERP implementation. These two preliminary steps are presented in the rest of this section.

10.1.1 Adequacy estimation

It is important to identify the problems the enterprise is facing, and to align them with the functions offered by the software. The characterization of the SME considered must be done before everything else, in order to estimate whether or not the software solution fits with the requirements of the SME. Such methods should be customized according to the software, and tools to fulfill these tasks are generally provided by the ERP vendor. For example, the *'Opportunity Qualification Tool'* from the SAP group [SAP08] proposes several generic questions covering topics like :

- the industry segment : for example Retail, Wholesale/distribution or Professional Services
- the industry vertical : for example Food, Chemicals, Banking or Consulting

- the number of employees
- the planned annual growth
- the modules to be used
- the number of add-ons which could be needed
- the average number of Sales documents per year as well as the average number of lines in them.

The adequacy of the software solutions to the enterprise situation will be assessed, according to each answer. This task will also help to determine the enterprise characteristics, and allow the team to go further, once the ERP solution adequacy has been assumed.

10.1.2 Financial assessment

Another task to be done during the early days of ERP implementation projects will be to financially value the advantages brought by the ERP software solution. Indeed, most of the time, top managers of SMEs firstly think in terms of costs and financial benefits of an ERP implementation. Of course this is understandable for them to reason about the financial benefits, and generally *'calculating cost and benefits is a prerequisite for decision-making' [SOL04].* On another hand, [WEI07] pointed out the lack of efficiency in calculating the performance through conventional financial methods, and the need to take the other benefits into account, such as information sharing, or user satisfaction.

This task will generally be a prerequisite to discuss with the top management though, and this can be achieved by the mean of a ROI tool like the one illustrated in Figure 6.1, for example, quickly evaluating the strategical benefits of the adoption of the software solution, based on questions about strategic advantages aimed by the top management. Such tool will allow the team to financially value the strategic goals, on a very high level.

10.2 Language for modeling during this stage

Goal modeling is an important step for ERP implementations mainly for two reasons :

- Clearly establish the goals intended for the implementation of the ERP software by the top management, in order to focus the attention of the implementation team on the problems the enterprise is facing, and not only on the ERP solution. This makes goal modeling an important step of requirements elicitation for ERP software implementations.
- Being able to track down later the performance of the ERP software implementation in regard to the goals stated.

During this stage of the implementation, only the first of these two points is interesting, because the team needs to get the 'big picture'. The means and techniques to track the performance will be detailed later on.

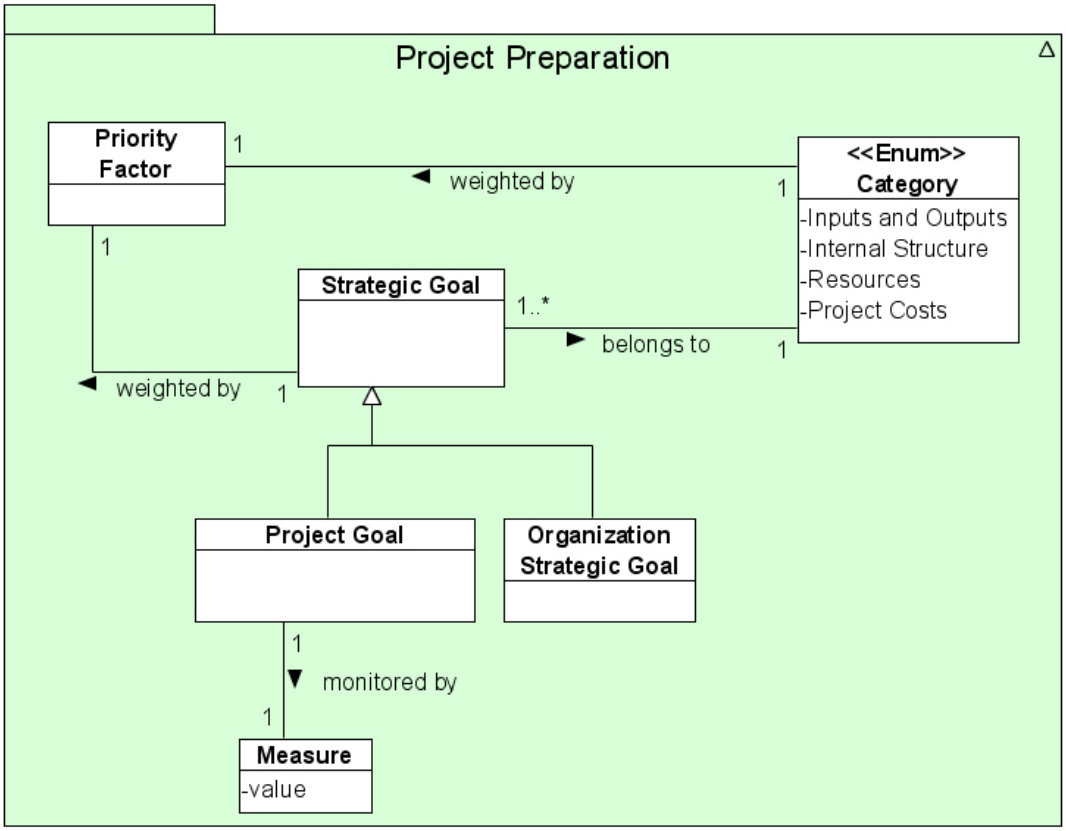


Figure 10.1: Metamodel for this stage

Goal modeling is the main technique considered during this stage of the implementation. The various concepts are illustrated in the metamodel of Figure 10.1.

- Strategic Goal : Generic goals of the enterprise, drawn from [GAM06]. Each strategic goal belongs to one category and is weighted by a priority factor.
- Category : *Inputs and Outputs*, *Internal Structure*, *Resources*, or *Project Costs*. These four categories are used together to classify all the Strategic Goals of the enterprise that the ERP solution will help to achieve. The *Project Costs* category is weighted by a Priority Factor of 100%, and the sum of the other three categories priority factors is 100%.
- Priority Factor : Percentage of relative importance for each goal regarding the other ones of the same level, and each category regarding the other categories. The *Project Costs* Priority Factor is always 100%, to deal with the project costs in parallel to the other goal categories. The sum of all priority factors of the other three categories is 100%, as no other category can be used.
- Organization Strategic Goal : Goal belonging to one of the three categories : *Inputs and Outputs*, *Internal Structure*, and *Resources*.
- Project Goal : Goals that belong to the *Project Costs* category. These are the project cost goals, in terms of both money and time. These goals are measured by a Measure.
- Measure : Value to asses the achievement of the project cost goals, always expressed in terms of gap between scheduled situation and present situation.

10.2.1 Discussion about the concepts

The VIP Map from the SAP group, illustrated in Figure 5.3, could have been chosen as a starting point to elicit the strategic goals of the enterprise. Indeed, a parallelism could be made with the *Visions* from SAP with strategic goals, the *Indicators* being the mean to verify the achievement of each goal, and the *Pains* being the flaws, which needs to be solved, in order to achieve the goal. Furthermore, the objectives defined in the VIP maps could be divided in several pains, which would represent the various flaws behind the operational goals, which together allow the achievement of the higher-level goal.

The problem with the VIP maps however was a lack of general reference to be analyzed, or general structure, from which the strategic objectives could be sorted out. To mitigate this problem, the analysis was pushed toward the Business Scorecards (BSC) from Kaplan and Norton [KAP92, KAP96], or the works from Magnus Gammelgård [GAM06, GAM07] which categorize the strategic goals. An important choice was made at this point to categorize the strategic goals, to be able to establish a defined generic model, composed of a predefined number of goals. These goals could then be categorized, and reused later to derive the specific goal model of the enterprise.

Gammelgård identifies three different business value dimensions, namely the *Inputs and outputs of the organization*, the *Organizational resources*, and the *Organizational structure* [GAM07]. These dimensions and goals are illustrated in Table 6.2. The BSC on another hand identifies four perspectives to analyze an enterprise performance : the *Financial*, *Customer*, *Internal business*, and *Innovation and learning* perspectives. (see Figure 6.3)

Each of these categorizations has its advantages and disadvantages, and the VIP maps seemed promising, but here are the points to keep in mind while trying to find an adequate compromise solution :

- The VIP maps are a clear tool and stay simple and expressive, with the availability of a high level of details if required.
- The Gammelgård management assessment framework, in its method to analyze the business organization, has the advantage to be built on a IT/IS investments benefits categorization, which in turns relies on an important literature study, covering all the possible kinds of benefits for IT investments. [GAM06]
- The balanced scorecard classifies the financial objectives in a separate perspective, making it possible to directly reuse the data from the financial evaluation conducted earlier with for example a ROI tool like the one illustrated in Figure 6.1. Furthermore, balanced scorecards can be hierarchically structured, and each goal in the BSC can be valued with a corresponding measure, like in the *VIP map* notation, for example.

10.2.2 Suggested categorization

At first sight, it is hard to support one categorization over the other to define a categorization for the language to use during this stage. However some of the perspectives used in BSCs are found as subdimensions in the Gammelgård categorization : the *Customer* perspective is a

part of the *Business as a black box*, and the *Learning and Growth* perspective is both about the resources and the structure of the organization. Indeed the (*'Learning and Knowledge'* perspective is referenced by the Gammelgård categorization under the *Resources of the business* dimension, and the future growth possibilities are taken into consideration in the *Structure of the organization* dimension, with business values such as *'Strategy formulation and planning'*, *'Change management'*, or *'Control and follow up'* [GAM06]. From the four perspectives of the BSC, only the *Internal business* perspective can not be compared to a subset of the Gammelgård categorization, but can be assimilated as the *'structure of the business organization'*.

This leaves the *Financial* perspective aside, which could be added as a parallel dimension in the categorization, to allow the top management to monitor the financial objectives of the project in parallel to the other ones. It is important to note that the objectives defined in ROI calculation tools, like the one illustrated in Figure 6.1, will not be measurable during the implementation, and shall only be fulfilled after several months of use of the new system. Thus they can not be placed in the financial perspective, but in their respective other dimensions. The Financial dimension is thus more like a 'controlling' dimension, to monitor the expenses of the implementation project. It is composed of the project objectives defined in terms of time and cost expenses, which should also be monitored later, and achieved along with the other strategic goals, to allow the team to finally qualify the implementation as a success, as explained in [WEI07].

This reasoning would result in the following four categories in our language :

- Inputs and Outputs : matching the *Business as a black box* dimension from [GAM07].
- Business Resources : composed of the objectives related to the human and non-human resources, detailed in [GAM07].
- Internal Structure : identical to the dimension of the same name from [GAM07], detailed previously.
- Project costs : divided in time and cash costs, as illustrated in [WEI07].

At this point, the implementation team can now fulfill a table with these four dimensions, with the commonly accepted financial objectives and the strategic objectives defined for the implementation project.

10.2.3 About the priority of the strategic goals

The *categories* used in the language have a special property : the priority factor. Every strategic goal has its relevance through the eyes of the top management, and generally they will need to achieve every strategic objective considered, but valuing each of them will prioritize them. Following the methodology used in [WEI07], it is possible to value each strategic objective in regard to its dimension, and each dimension in regard to the other ones. Such notion is included in the language to allow the implementation team to prioritize the strategic goals of the project.

The project-related objectives need a special attention, and must be dealt apart from the other ones. Indeed the costs implied by the project must be monitored continuously, while the other objectives will only be achieved some time after the *Go-Live*. The project Costs goals should not be valued according to (or in competition with) the strategic objectives of

the implementation. These special characteristics of the *Project Costs* goals are illustrated by a static priority factor of 100% in the language, to fix their priority apart from the other enterprise strategic goals.

10.3 Syntax of the language

The syntax proposed for the presentation of the language is based on the tabular arrangements from [WEI07, KAP92]. Each of the four categories is valued by a priority against each other, then the respective rows are divided according to the goals of each category. Each of these goals is valued in the next column by its priority factor, and the last column provides the measure for the project costs goals. Table 10.1 illustrates the general arrangement of the categories, goals, priority factors and measures.

	Total Priority	Goal	Priority	Measure
Inputs and Outputs	Sum = 100%	Strategic goals for *Inputs and Outputs*	Goal Priorities Sum = 100 %	Measures are only added to the project costs goals below, as the other strategic goals of the enterprises will be measured later.
Resources		Strategic goals for *Resources*	Sum = 100 %	
Internal Structure		Strategic goals for *Internal Structure*	Sum = 100 %	
Project Costs	100%	Strategic goals for *Project Costs*	Sum = 100 %	Measures for all project costs goals

Table 10.1: Syntax of the goal model

10.4 Generic model

A generic model can be built from the language and the syntax defined earlier. By inserting the twenty-five generic goals for IT/IS investments from [GAM06] in the syntax, the generic model illustrated in Table 10.2 can be drawn.

The generic model include details about project costs goals and their measures, similarly to their representation in [WEI07] (see Appendix E), considering the two main constraints of the project management, being cost and time.

- Cost constraints : this includes the four main costs, namely the *software*, *infrastructure*, *maintenance* and *consultant* costs.
- Time constraint : the time planned for the completion of the project.

	Total Priority	Goal	Priority	Measure
Inputs and Outputs		Inbound logistics	%	
		Supplier Relations	%	
		Customer relations	%	
		Lock-in effect/switching costs	%	
		Competitor relations	%	
		New products/services	%	
		Differentiations in products/services	%	
		Quality of products/services	%	
		Deliveries	%	
	%	Third party relations	%	
Resources		Decision making	%	
		Learning and knowledge	%	
		Organizational culture	%	
		Information	%	
	%	Technology/tools	%	
Internal Structure		Strategy formulation and planning	%	
		Efficiency	%	
		Productivity	%	
		Cost reductions	%	
		Communications	%	
		Flow of products/services	%	
		Control and follow up	%	
		Change management	%	
		Integration and coordination	%	
	%	Flexibility	%	
Project Costs		Cost	%	Gap between estimation and real expense
	100 %	Time	%	Gap between estimation and real schedule

Table 10.2: Generic goal model

The measure for the project management objectives is the gap between what has been planned and the real expenses, being cost, or time expenses. This is true for all implementation projects, though the cost goals from Table 10.2 could possibly be more detailed for precision concerns, according to the four cost constraints detailed earlier. The project costs goals being focused on time and costs constraints, these two generic goals are included in the generic model.

10.4.1 Methodology

After the adequacy of the software solution has been assessed through the analysis of the enterprise characteristics, the consulting team and the top management must both come to a customized goal model. In order to do so, they will proceed as follows :

1. Examine the twenty-five strategic goals and choose which one needs attention
2. Prioritize every category, and every strategic goal of the subset

Table 10.3 is a simple example of what the strategic objectives table could look like. The first financial estimation which could be made at this point is still subject to change, after further analysis of the business processes, data, and structure of the organization. Direct indicators will not be assigned to the strategic goals, as these ones will often need to be divided in a few operational goals impacting together on their respective higher level, strategic goal.

	Total Priority	Goal	Priority	Measure
Inputs and Outputs	50 %	Improve customer relations	30 %	
		Improve quality of products	30 %	
		Improve deliveries	40 %	
Resources	20 %	Improve decision making	100%	
Internal Structure	30 %	Improve flow of products	60 %	
		Improve communication	40 %	
Project Costs	100 %	Cost	60 %	Gap between estimation and real expense
		Time	40 %	Gap between estimation and real schedule

Table 10.3: Example of strategic goals classification.

At this point of the implementation, it is too early to conduct a deep analysis of the business processes, but the business process reference models stay available to the consulting team, to briefly illustrate the support for one or another process, if it would become necessary. Analyzing any business process or being too *'technical'* at this point can be inappropriate. That would be done during the tasks of the *Identification and Modeling* stage coming next, and detailed in the next chapter. The top-level processes can still be discussed and addressed, while trying to assess the strategic objectives that the ERP implementation will contribute to achieve. The business processes can be addressed on a high level during this stage, in order to show how the ERP solution will cover them, but detailed business process modeling should come during the next stage though, after the contractual agreement.

Section 11 :

Modeling during the Identification and Modeling stage

> **Content :** *During this stage, the ERP implementation team must achieve a complete and precise business case definition. The tasks conducted during the project preparation helped to achieve a common vision and a general idea of the strategic objectives to achieve, and the business case requirements and constraints. This stage constitutes a deeper analysis of the enterprise characteristics, in order to determine the complete configuration and customization of the ERP software solution.*

11.1 Language for modeling during this stage

Regarding the goal modeling concerns, the concepts proposed during the previous stage will be reused to refine the strategic goals into several operational goals. Business process modeling notions will be added to the previous stage, and this will result in the metamodel presented in Figure 11.1. The concepts to be used during this stage are thus the ones described in Section 10.2, augmented with the following notions :

- Goal : Abstract entity, used in the metamodel to represent both the strategic and the sub-level goals. Goals are weighted by a priority factor and impact on one or more generic reference processes.
- Sub-Level Goal : Goal obtained from a strategic goal through a refinement process. The sub-level goals are situated in a lower level than the strategic goals.
- Intermediary Goal : Goal situated in a level between the strategic and the operational goals. An intermediary goal is a sub-level goal which is not an operational goal, but will be refined later and linked to one to several operational goals.
- Operational Goal : Goal situated in the lowest level of the hierarchy. They can be linked to an intermediary goal, or directly derived from a strategic goal. An operational goal is monitored by a Key Performance Indicator.
- Key Performance Indicator : Measure used to monitor the achievement of an operational goal. a KPI must be measured by the mean of a defined value.

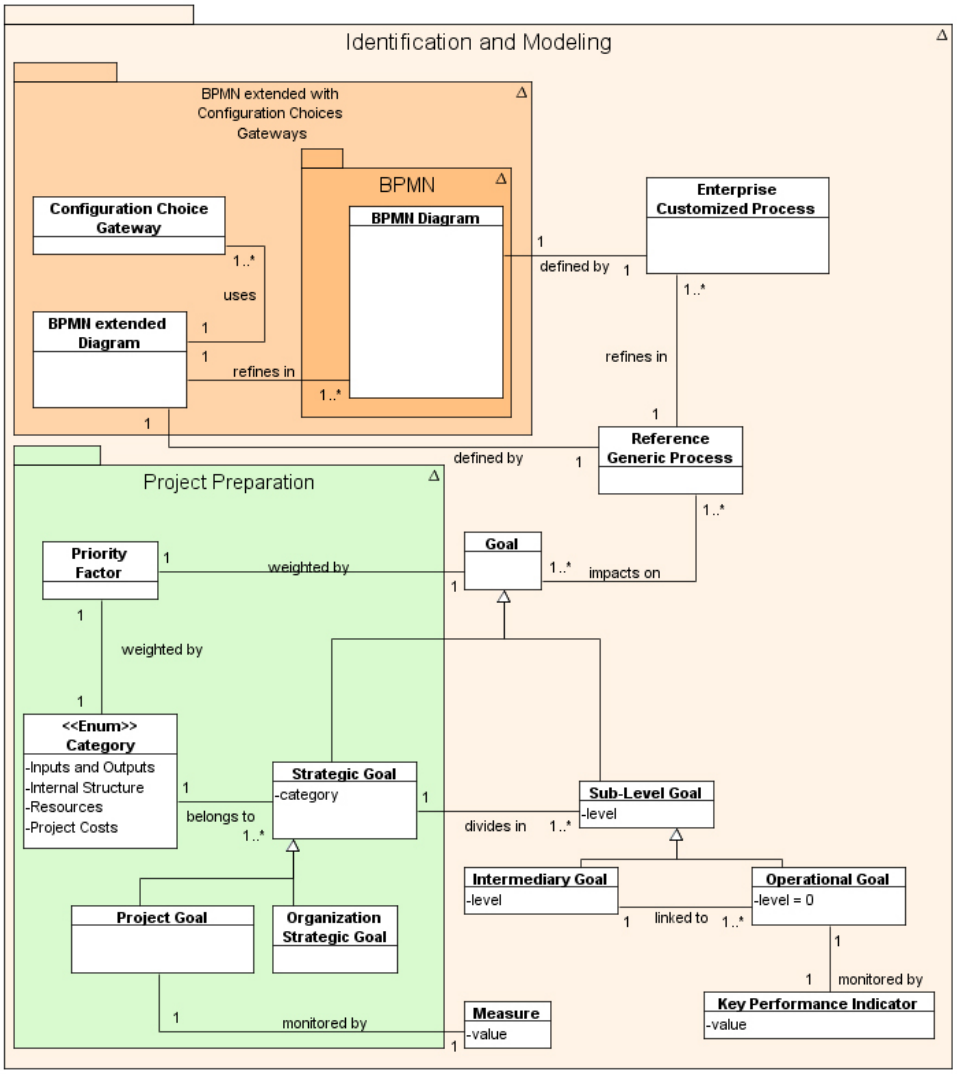


Figure 11.1: Metamodel for this stage

- Reference Generic Process : Reference Model showing all the possible ways to support a defined business process by the ERP software solution. A reference generic process model is defined by a BPMN Extended Diagram.

- Enterprise Customized Process : Process resulting from the refinement of the reference generic process, through the elimination of the configuration choices gateways. An enterprise customized process is defined by a standard BPMN diagram, following the specifications of [BPMN08].

- BPMN Extended Diagram : Diagram used to model the reference generic processes supported by an ERP software solution. BPMN extended diagrams use the standard BPMN notations defined in [BPMN08], and configuration choices gateways.

- Configuration Choices Gateway : notation added to the standard BPMN from [BPMN08], to represent the possible configuration choices. This notation allows to model the various configuration choices available in the software solution, described in details in Section 11.2.2, and illustrated in Figure 11.3.

- BPMN Diagram : Standard Business Process Modeling Notation diagram, following the specifications from [BPMN08].

It should also be noted that the entity *'Measure'* from the preceding metamodel was moved from the project preparation to the identification and modeling, as this measure can possibly be adjusted after a more complete analysis of the enterprise considered and its business processes.

11.1.1 Discussion about the concepts

The review of the literature about business process modeling during ERP implementations allowed to point out several important points of this modeling aspect :

- Various notations exist to model the business processes of an enterprise, each one with its advantages and disadvantages. It goes from EPCs in the SAP Reference Model for *SAP Business Suite* [SAP08] to Petri nets or UML 2.0 Activity diagrams.
- There is a serious advantage in owning a reference model in ERP implementation projects. Such reference model eases the understanding of the way business processes are treated by the software and help the design of the particular models aimed at an individual SME, through the 'Design by Reuse' paradigm [Aal06, GOT07].
- Several levels of choices in business process models do exist, and this must be taken into account when designing the general reference model, as explained in [Aal06] or [GOT07]. The choice levels can go from the *Configuration choices* which must be made once and only once during the installation, to the *Run-Time choices* which can represent the various alternatives to a given process which could turn out differently for every instance of the process considered.
- Such reference models, when they are available, are designed by the vendors of the software solution, and this may cause some problems. Studies like [MEN06] have proved that such models can be flawed, and show the interest of verification tools to ensure the correctness and completeness of such models.
- The analysis of the *'ASAP for Business One'* methodology points out the lack of business process modeling support within the methodology. The configuration choices are determined while fulfilling the 'Business Blueprint' template. Even if this step is at first glance faster than analyzing a business process model, it does not use any formalism or notation at all, thus there is no official step, or support material, dedicated to the analysis of the business processes of the enterprise, or the various alternatives available to make them fit with the software functionalities.

11.2 Syntax of the language

11.2.1 Goal modeling

The syntax for the goal modeling part of this stage is also mainly based on [WEI07], and follows the balanced-scorecards approach. Each strategic goal outlined, thanks to the generic model

illustrated in Table 10.2, will be divided into several sub-level goals. These will then be arranged in cells within a hierarchical tree structure, along with their priority factors, as illustrated in Figure 11.2.

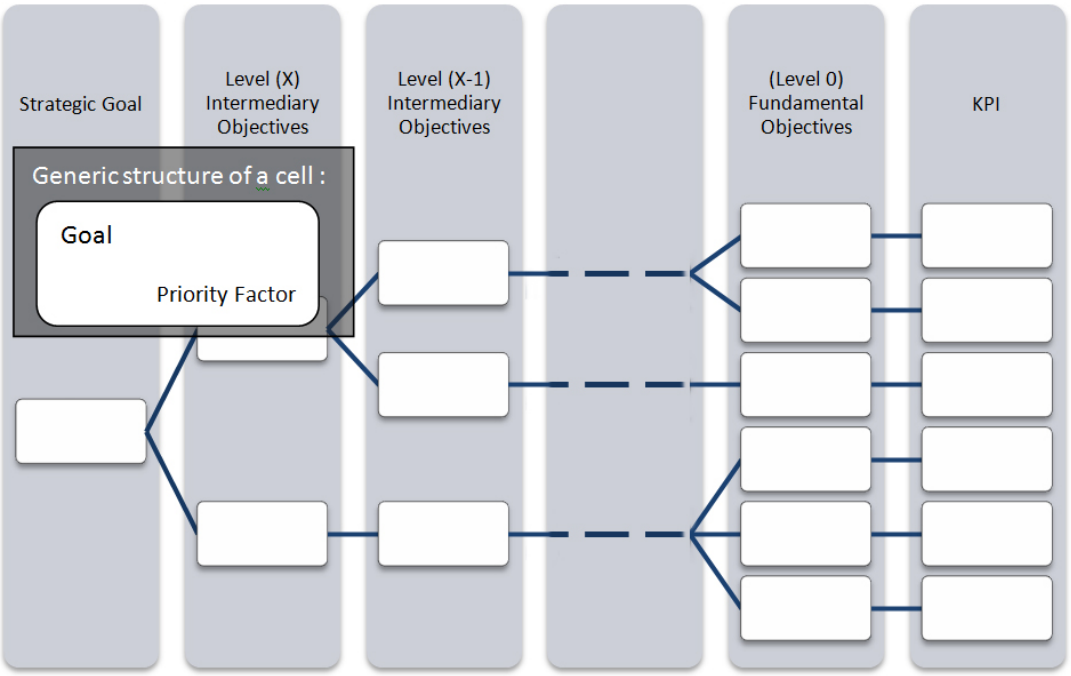


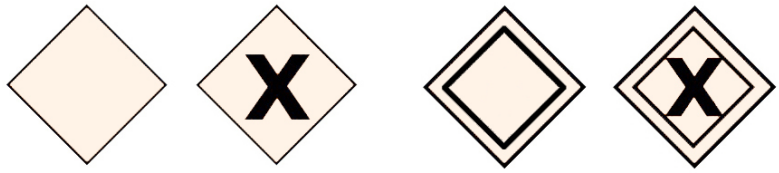
Figure 11.2: Syntax for goal modeling

The choice of a hierarchical tree structure has been made because it eases the decomposition process, and keep the sub-levels clear. The choice to realize a separate tree for each strategic goal is made in this work to illustrate the concepts. Such division can be made at the category level if the number of strategic goals is limited, or at a sub-level if a given strategic goals has a high number of intermediary goals.

11.2.2 Business Process modeling

According to the discussion initiated in [Aal06] and [GOT07], several levels of choice should be used while dealing with the business process models for ERP software. Most notations lack such different levels of choices, but some can be extended to support them.

In this work, the mainly used notation is the Business Process Modeling Notation, or BPMN, from the OMG group [BPMN08]. Indeed, BPMN allows the modeler to express every business process modeling concept, except a different level for the *Gateway* symbols, required to illustrate the configuration choices. According to the specifications of the BPMN language, BPMN can be extended with additional symbols and *'non standard elements to satisfy a specific need'* [BPMN08, p.16]. The gateways presented in the BPMN language can thus logically be extended, to satisfy the need to model particular choices : in this case the configuration choices, requiring attention during the software installation. An example of the BPMN extension proposed in this work is presented in Figure 11.3. This Figure illustrates the original data-based exclusive-choice gateways, and their configuration choice counterparts.



(a) Data-Based Exclusive Gateways of BPMN V1.1 [BPMN08]

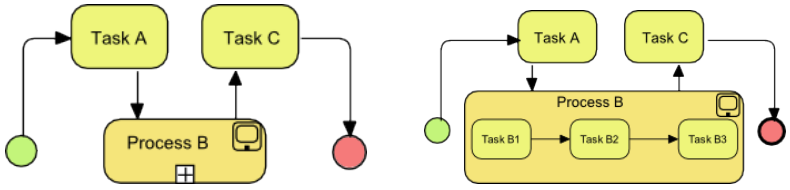
(b) Configuration Choice for Data-Based Exclusive Gateways

Figure 11.3: The original Exclusive Data-Based Gateways and the Configuration Choice Gateway

The particular gateways used to model the configuration choices such as the ones from Figure 11.3(b), also respect the recommendations from the BPMN specifications to *'not alter the footprint of the basic flow elements'* [BPMN08]. While similar to the original gateways, the main difference between original gateways and configuration choice gateways is a double-diamond symbol, which can be easily differentiated at first sight, and do not compromise the understandability of BPMN diagrams.

The addition of *configuration choice gateways* results in an extended BPMN to model the generic business process reference models of the ERP solution. Such notation gives to the modeler both abilities to *'hide'* and *'block'* specific process parts, both being needed properties to configurable reference models, as stated in [Aal06, p. 515].

The ability of *hiding* several parts, is naturally provided by the multiple abstraction levels provided by BPMN. A sub-process can indeed be modeled with all the steps within it, or abstracted to the simple name of the subprocess. Figure 11.4 presents these abstraction levels.



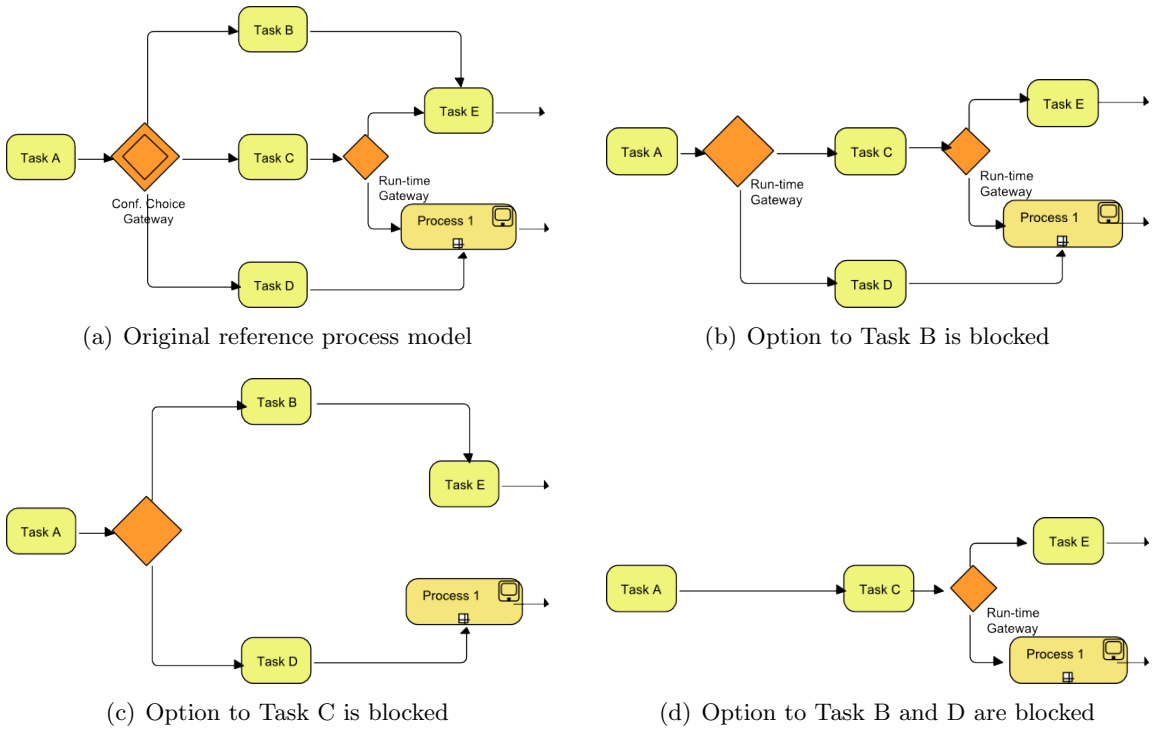
(a) Process B is abstracted

(b) Process B is detailed

Figure 11.4: Abstraction capabilities of BPMN

The ability of *blocking* parts of the reference model is provided by the configuration choice gateways. Each configuration choice gateway should be analyzed by the implementation team prior to the installation, and dealt accordingly to model the future processes of the enterprise.

There are several ways to deal with the configuration choices. As pointed out in [Aal06], configuration choices can block one or another part of the reference model, but there is still another possibility : to leave this choice at run-time. Keeping the original BPMN gateways syntax allows to transform the original reference model into a customized model, matching the business process of the enterprise, and supporting all the possible gateways of BPMN. The customization process requires to deal with each configuration choice gateway, and transform it into the corresponding *'classic'* gateway, in order to leave the decision at run-time (Figures 11.5(b) and 11.5(c)), or just simply replace the gateway with a straight link, if only one possibility for this particular choice should be chosen (Figure 11.5(d)).



(a) Original reference process model

(b) Option to Task B is blocked

(c) Option to Task C is blocked

(d) Option to Task B and D are blocked

Figure 11.5: Configuration process examples

The business process model resulting from the configuration process should be clean from all configuration choice gateways, producing a classic BPMN diagram. It is important for the modeler to check that the syntax of the resulting BPMN diagram stays coherent with the BPMN specifications. For example, a classic exclusive gateway, refined from the configuration process, should not have only one exit. Similarly, if an inclusive gateway, allowing one or more sequence flows for the same gateway ([BPMN08]), should be on a configuration choice level, and then refined in a *'classic'* inclusive gateway ([BPMN08, p. 80]), it should keep the same semantics, allowing one or more sequence flows simultaneously, for a given instance of the process.

11.3 Generic model

11.3.1 Generic goal model

There is no generic goal model at this stage, as the goal model from this stage is derived from the specific goal model created during the *project preparation* stage. No generic goal model can be defined for goals on a lower level than the strategic level, because these goals can vary from one ERP solution to another. This depends on the functionalities specific support for the considered solution. Such a generic hierarchy for every strategic goal could be drawn for one particular solution though, according to the functionalities of the software. However, this should be realized and continuously improved with the support of real case data. Furthermore, such generic tree could also vary from one enterprise to another, considering the same solution. For example, the pick & pack manager from SAP Business One can improve the Sales-to-Delivery process, but if the enterprise top management chooses not to use this functionality for the inventory management, it will have to be deleted from the hierarchical goal model.

11.3.2 Generic Business Process model

The generic business process model must reflect the support of the various business functions of an enterprise by the software. Such generic reference model is thus composed of several business process models, detailing the support of every business process of the enterprise that the ERP system is going to support.

Such reference model includes the notion of *configuration choices* explained earlier, and follows the syntax described in Section 11.2.2. As an illustration, the Sales Process support of SAP Business One, modeled in BPMN extended with configuration choices, is presented in Appendix E.

11.4 Methodology

During this part of the project, goal modeling tasks should be conducted along with business process modeling tasks.

11.4.1 Goal Modeling tasks

We already noted that managers of an enterprise in need of ERP software will first think in terms of goals instead of business processes, or data for example. However, a complete and too exhaustive goal modeling would make the goal modeling tasks inefficient. A compromise solution must be found between no goal modeling at all and method resulting in too much time consumption.

The goal modeling tasks at this point are the following :

1. Refine the strategic objectives for every business unit, following the balanced scorecard approach.

2. Continue to refine the objectives as needed, until the operational level, where corresponding Key Performance Indicators (KPIs) can be determined.

Establishing the goal hierarchy

The idea is to use the strategic objectives defined earlier, to determine the operational objectives which will allow their completion. The related operational objectives must be deduced at some point, to determine the proportions in which the ERP software will be available to help the SME considered. The strategic goals will almost always be refined in several other goals, but for reading concerns, it is possible to consider several intermediary levels to the hierarchy, between the few strategic goals, and the many operational goals. An example of such hierarchy inspired from [WEI07] is illustrated in Figure 11.6.

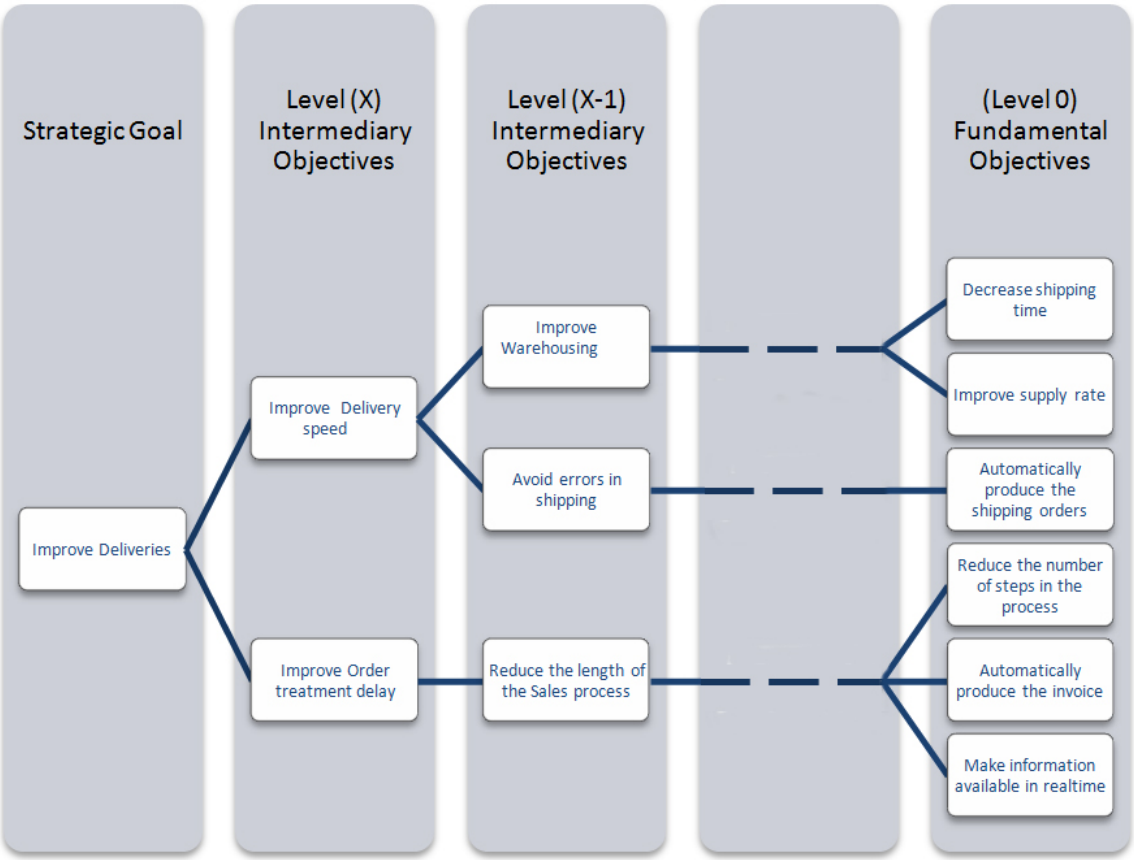


Figure 11.6: Goals hierarchy example

At this point, a complete goal hierarchy is available to the implementation team, and both the strategic and operational goals to achieve are clearly defined. The next step is to assign a KPI to each operational objective, in order to measure the performance of the system.

Assigning the corresponding KPIs

The notation exposed previously should now be augmented, in order to include a measurable indicator to the operational goals, reflecting the actual situation of the enterprise. This would lead to a notation combining both goal modeling and performance monitoring, which still would not be too expensive. This might not be as accurate as a deep analysis of the objectives among every business unit of the whole enterprise however, but such compromise would still fulfill the purpose of a goal analysis, for resource-limited enterprises though.

Measuring the goals achievements

Until now, the performance measurement techniques were only addressing performance estimation, but from now on with the help of the KPIs defined earlier, the implementation team can begin to monitor the performance of the system.

Following the way strategic objectives were valued according to each other, intermediary and

operational goals can also be valued with a proportion factor, to weight them according to the other goals of the same level. This can be done recursively to the highest level, as illustrated in Figure 11.7, and previously for strategic goals in Table 10.3.

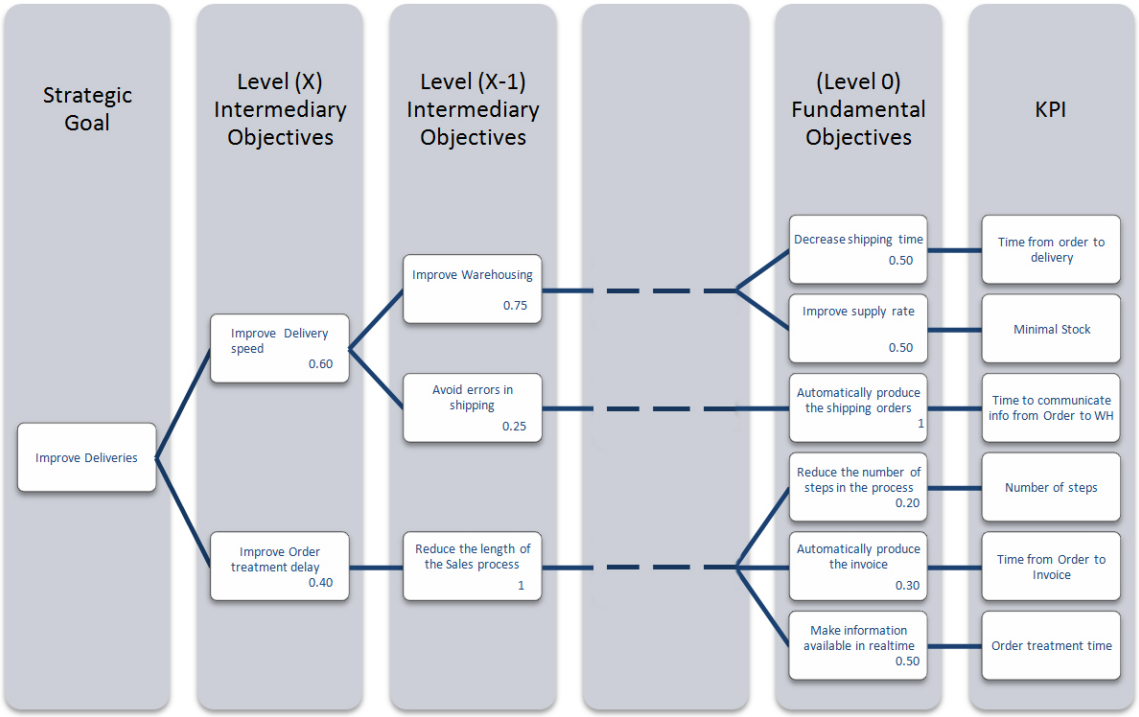


Figure 11.7: Valued goals and KPIs example

When the objectives are defined through the entire hierarchy and valued according to the entire set, one KPI can be derived for each fundamental objective, in order to define the milestone to achieve. Only then can the performance assessment after the project be possible.

Regular performance monitoring

Through the entire project, the KPIs can be measured to value them according to the *to-be* value defined during the requirements elicitation. Performance monitoring could be conducted on a regular basis during the project, to monitor the goals stated earlier, but this process generally faces a cost problem and thus become neglected. For example, the *ASAP for Business One* methodology only recommends to check at the end of the project if the goals intended are met, but there is no recommendation to continuously measure it during the project. Such monitoring could be valuable though, regarding the goals of the *project costs* category, but a correct time frame must be agreed upon, to avoid spending too much time and resources on this task. Such monitoring can then result in updates of the goal hierarchy and achievements of the KPIs. The interest of measuring the KPIs before the entire completion of the project can be questionable, as some goals will only be achieved by the end of the project, or even after some use in the organization, but the use of precise KPIs is undoubtedly an asset after the implementation to precisely determine the ERP implementation benefits.

11.4.2 Business Process Modeling tasks

The addition of business process modeling at this stage of the project gives to the implementation team a clear vision of the future business processes, as they will be supported by the software. The reusability of the business process reference model also allows a deep analysis of the organization important processes, while still keeping the costs in terms of resource and time on a low level.

The main part of the business processes analysis takes place during this stage of the implementation. The proposed tasks are the following :

1. Determine, in parallel with the goal hierarchy, which business processes will be the most critical, and require the most attention. Take advice from the top management about this matter whenever it is needed.
2. For each business process in the reference models to be supported by the ERP solution, refine the generic model to determine the customized business process and the configuration choices, according to the requirements of the enterprise.
3. Use the refined business processes, which both sides of the implementation agreed upon, to configure the handling of the various business processes and functions.

The *Pick and Pack manager* from SAP Business One will be used as an illustration to briefly show the customization process. In SAP Business One 2005, the enterprise has the ability to use a special functionality of the software, named the *Pick and Pack Manager*. The top management can choose to enable this functionality if the inventory is managed by serial numbers, and package units are properly configured. This function of the software allows the system to suggest, for an issued reserve invoice, the matching items to be picked in the warehouse and the package units to ship the inventory items in. This allows to track down the inventory items, thanks to the serial numbers and the information of the *pick and pack manager*, as well as keeping a trail of the items with the packages information. Furthermore, this saves time for both the sales and the warehouse employees. The system automatically suggests the items and the packages units. It also gives information to the warehouse about which items must precisely be included in the shipping. The simplified generic process is illustrated in Figure 11.8.

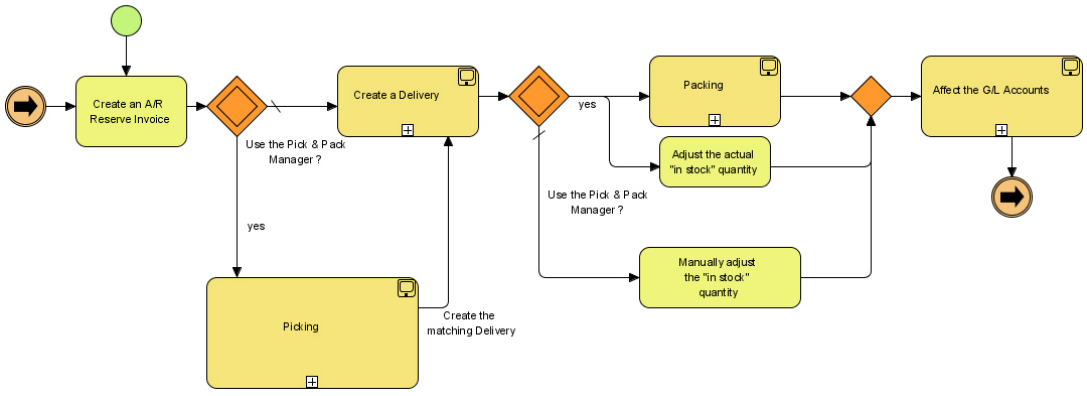
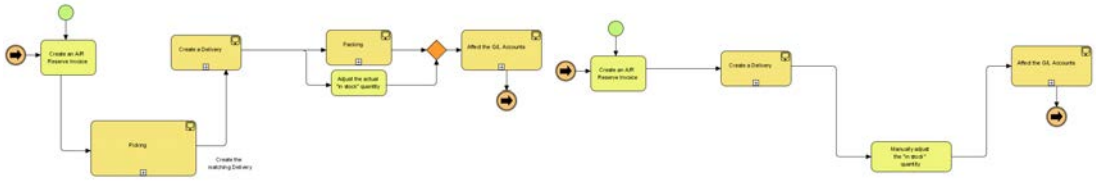


Figure 11.8: Pick and Pack manager generic reference model

The Figures 11.9(a) and 11.9(b) illustrates two possibilities to manage the process : with or without the *pick and pack manager*.



(a) Pick and Pack manager enabled (b) Pick and Pack manager disabled

Figure 11.9: Pick and Pack manager configuration choices

A third possibility is available, as explained in Section 11.2.2. The choice to use the Pick and Pack manager could be left to run-time. This would result in a customized model very similar to Figure 11.8, with the only replacement of the configuration choice by the corresponding exclusive choice gateway from BPMN. The refinement process of the reference model consists in replacing each configuration choice gateway by the chosen support for this part of the process, with respects to the original BPMN gateways semantics, as explained earlier. The customized process is obtained by repeating this action for every configuration choice gateway of the reference model.

Section 12 :

Case study

Content : *This section presents a case study used as an illustration of the methodology and techniques for the early stages of an ERP implementation.*

12.1 Case study context

A real case analysis could not be available for this work. The case study presented in the next sections is thus provided as an illustration of the methodology, rather than a demonstration of its validity. This case study has been inspired from [KAN06] and other observations from academic works, as well as information from SAP materials, since this case study will be applied to the SAP Business One solution.

12.2 Enterprise characteristics

Company A is an important SME in the local IT market, selling personal computers and pieces at low prices for the private individuals or companies. Their recent success in the business and the diversity of their customers locations made them take the decision to broaden their activities into three regional sites. The former main establishment of the enterprise is now used as the operation center and main warehouse for the other two regional sites. This new infrastructure led to important investments for the company, in terms of real estate, recruitment, and business process change. The new architecture of the enterprise also required important IT investments, as their former IT system needed to be upgraded to support the new business processes to be put in place.

The SME's business continued to thrive for some months with the former IT system, more or less tailored to the new infrastructure, but problems rapidly began to arise, such as operational inefficiency, redundancy, inacurracy of information. The enterprise top management had predicted that the communication between every enterprise site and their respective business units would have to be improved. They decided to adopt a new ERP system, in the continuity of their organizational structure improvements.

12.3 Project Preparation

The enterprise realized the need for a new ERP system, and decided to move to the new system as fast as possible, as the enterprise architecture had changed and needed a more efficient IT system to support commercial and processing activities. Some business processes were indeed badly flawed due to inadequacy of information and miscommunication. The customer sales order to deliveries, the inventory management, and financial and accounting information management processes are some examples of such processes.

The SAP Business One consulting team was contacted to present the ERP solution, and eventually short-listed. The project preparation could then begin, by precisely assessing the adequacy of the software solution.

12.3.1 Adequacy assessment

The first task to be conducted is the adequacy assessment. For this study we used the *Opportunity Qualification Tool* designed by the SAP group for their solution SAP Business One. This material being copyrighted, the exact questions could not be referenced in this work, but here are some characteristics of the enterprise that the tool allowed to outline :

- This enterprise is a Retail enterprise.
- This enterprise produces a high volume, with a low margin.
- The enterprise employs forty people but will require less than thirty licenses, as the top management estimates that there is no need for a specific license per employee, but rather per business unit, and role in the enterprise.
- The accounting and financial components of the solution will be used in the enterprise.
- The inventory management functionality will be used as well, and the enterprise will draw advantage from the ERP solution to track down their inventory transactions, by serial numbers, with multibin locations support, for possible future extensions.
- No add-on was required for the support of the IT system.
- Only one warehouse will need to be set up for this implementation.
- the volume of documents, and document lines processed is adequate for the ERP solution capabilities.

The result of the adequacy assessment concluded that the enterprise requirements were a match to the features of the ERP solution, with some recommendations though. These recommendations are dealing with some aspects of the business processes that will be covered by the solution. For example, the consultant should check with the enterprise whether the *SAP Business One* functionality fully supports their retail process, or their needs regarding the multiple warehouses support, or the *Pick and Pack* functionality. These examinations should be done to check if the solution fully supports the enterprise processes, or will require an add-on or additional development.

These recommendations in the adequacy assessment tool seem to confirm the need for the availability of the reference model for this stage already.

12.3.2 ROI calculation

There is a preexisting ROI tool available for the SAP Business One consulting teams. By the end of the ROI analysis, an estimation of the costs were given to the top management, along with a very high level estimation of the return on investment. A license for a power user of SAP Business One is 2,500 €, and a normal user license is 750 €. With these figures in mind, the top management came to a license budget of 27,500 €, with five power user licenses, and twenty normal licenses. Given the enterprise characteristics and the first adequacy assessment, the consulting team estimated the implementation budget to be approximately the same than the license budget,estimating the entire project cost, including licenses, at around 50,000 €. The first financial estimations suggested that the enterprises should get the return on investment within a two years period, which seemed promising, and at the same time, they started the enterprise goal model, to give an insight of the software capabilities to the top management. Financial figures alone were insufficient to convince the top management, but they were interested by a goal analysis of their enterprise and the possible consequences of the software adoption. They proceeded to the next step : the goal modeling.

12.3.3 Goal Modeling

Table 12.1 shows the strategic goal model for the case of enterprise A. To design this model, the top management first had to value the project costs goals priorities. The three other categories were then weighted, according to each other and their respective goals.

	Total Priority	Goal	Priority	Measure
Inputs and Outputs	35 %	Supplier Relations	10%	
		Customer relations	20%	
		Competitor relations	10%	
		Differentiations in products/services	20%	
		Quality of products/services	20%	
		Deliveries	20%	
Resources	20 %	Decision making	40%	
		Information	60%	
Internal Structure	45 %	Efficiency	10%	
		Productivity	10%	
		Cost reductions	15%	
		Communications	15%	
		Flow of products/services	15%	
		Control and follow up	15%	
		Change management	10%	
		Integration and coordination	10%	
Project Costs	100 %	Cost	60 %	Gap between estimation and real expense
		Time	40 %	Gap between estimation and real schedule

Table 12.1: Enterprise A strategic goal model

Project Costs category

The top management of the enterprise chose to put the cost of the project on a slightly higher priority than the time goals. The SME already suffered high costs for the new enterprise architecture, and the top management wanted the cost of the project as low as possible, without neglecting the time goals either. The more time spent on the implementation, the more possible loss and errors due to the legacy system. Considering the context, the project costs goals were valued at 60%, and the project time goals at 40%.

Other categories

After defining the project costs goals, the other three categories needed to be valued according to each other. The *Internal Structure* category received the highest weight, due to the recent internal structure that was drastically changed. This category contains most of the goals that motivated the adoption of a new ERP solution. The enterprise needed an ERP solution adapted to their needs, to continue excelling in internal structure objectives such as the *efficiency*, *cost reductions*, *control and follow up*, or *communications* for example. This category was weighted at 45%, due to its high importance for the enterprise top management.

The second category was the Inputs and Outputs, weighted at 35%. Strategic goals in this category also have a high impact on the enterprise. They need to improve their deliveries, and continue to ensure a high level of quality for their products and services. Customer and supplier relations are also important for them, to keep sales volume at a high level, and quality products at the lowest possible price.

The last category is the resources of the enterprise, valued at 20%. The enterprise needed a better vision of the information on every level of the organization, and better technology and tools to support its business.

Inputs and Outputs

The important strategic goals in this category are for the top management : the *customer relations* and the *deliveries*, which together had made their success in the past, along with the *quality* and *differentiation in their products and services*. Two other objectives were pointed out by the top management, but slightly less important though. These are the *supplier relations*, from which they should continue to obtain the best products at the best possible price, and the *competitor relations*, to watch out for the item prices and offers from the competitors in the market.

Discarded objectives were the *lock-in effects/switching costs*, which was not in the philosophy of the enterprise, the ability to manage more new products and services, the top management assessing the actual business activities to be enough, and the third party relations, irrelevant for this case.

Resources

For this category, the *information* was valued at a factor of 60%. Due to the new enterprise architecture, the information flow was indeed a priority for the enterprise. Improved decision making could also be valuable for the enterprise. The other three strategic goals were discarded. *Learning and knowleddge*, as well as *organizational culture* were considered irrelevant, and no new technology or non-IT tool was going to be adopted by the enterprise.

Internal Structure

This category was the most important for the enterprise top management. As such, nearly all the goals were selected by the top management. Only two goals in this category were discarded : *Strategy formulation and planning* was discarded, and not considered as relevant as the other strategic goals of this category, and *flexibility* was also discarded, as the enterprise context was not subject to frequent changes.

The rest of the strategic goals were then prioritized. The goals related to the connections between the business components were considered more important than the other ones. *Communication*, *flow of products/services*, *control and follow up* were thus outlined, but the top management decided to put the *cost reductions* goal on the same level as these three. *Cost reductions* being a component-related goal, but directly impacting the business activity of the enterprise. The rest of the goals were considered as desirable, but not as important as the four others. The first four goals were weighted by a factor of 15%, and the other four by a factor of 10%.

12.3.4 Conclusions for the Project Preparation

The first stage of the implementation allowed both stakeholders to come to the conclusions that the implementation of the ERP solution could be a real opportunity for both of them. The enterprise was adequate to the solution possibilities, not requiring any add on, or additional development at first sight. The ROI was also acceptable for the top management. Unsurprisingly, addressing the goal model, the stress was made on the internal structure goals, as these were the reasons that drove the enterprise top management to the possibility of a enhanced IT system with the help of the ERP software.

At this point, most tasks of the *project preparation* stage were done :

- First elicitation of the requirements, during the first discussions between the consulting team and the customer.
- Analysis of the enterprise situation during the adequacy assessment.
- Analysis of the ROI.
- Definition of strategic goals and flaws.
- Analysis of the impact of the ERP solution, exposed to the enterprise top management on the basis of the previous analysis.

The contract was signed a few days later, leading to the *Identification and Modeling* stage.

12.4 Identification and Modeling

The implementation team was formed with SAP consultants and power users from the enterprise, chosen between the seniors users in various business units. This allowed to ease the understanding of the business processes and interact with the enterprise future users. Due to concurrent activities, the SME top management was supporting the implementation, but could not be as available as needed during the whole implementation process. To mitigate this problem, a senior user was chosen to help the decision makings during the implementation.

A first schedule of the implementation project was also defined.

12.4.1 Data Modeling

The *SAP Business One blueprint* questionnaire was given to the enterprise power users, to fulfill the information required for data customization. Meanwhile, the SAP consultants established the goal hierarchy, to model the advantages brought by the ERP in terms of operational goals and KPIs.

12.4.2 Goal Modeling

The goal model from Table 12.1 was reused to be refined in a hierarchical goal model with KPIs, composed of various hierarchical trees similar to the one illustrated in Figure 11.7. Figure 12.1 illustrates the goal model of the Resources category for enterprise A. As further illustration, the *Improve customer relations* and *Improve efficiency* goal hierarchies are presented as Figure 17 and Figure 18 in Appendix F.

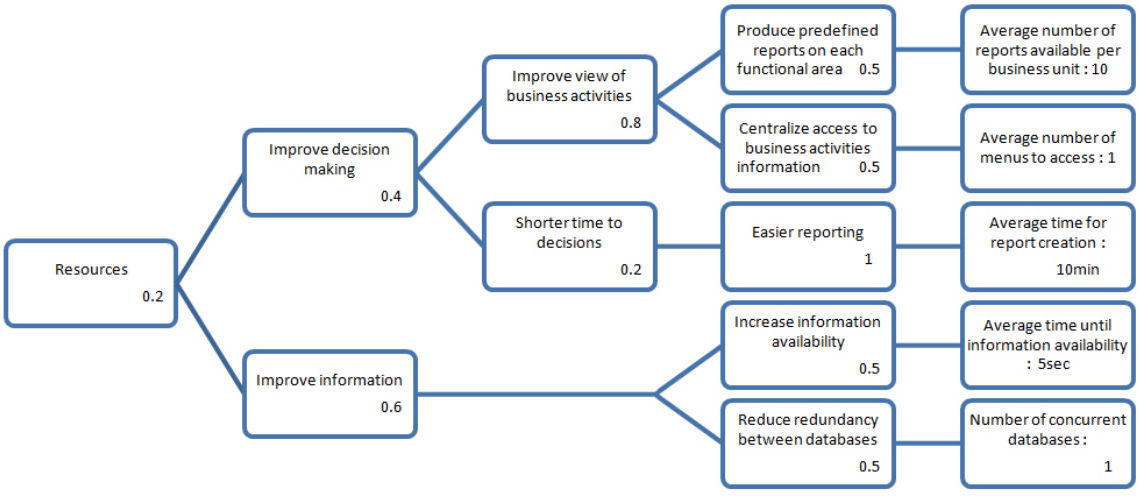


Figure 12.1: Enterprise A goal hierarchy for the Resources category

Following the methodology exposed in Section 11.4.1, the strategic goals defined for every category during the previous stage were refined into lower level goals according to the ERP

software capabilities, until the operational level. KPIs were chosen according to the software capabilities for all the operational goals, and the enterprise senior employee then valued each goal of the hierarchy with the consulting team.

The refinement of the strategic goals into the respective goal hierarchies was initially realized by the consulting side of the team, because it requires a good knowledge of the ERP solution to translate the enterprise strategic goals into operational goals, and specific advantages to be brought by the ERP solution. During the refinement of each strategic goal into operational goals, the consultant should be very careful to include all aspects of the strategic goal and translate these aspects into sub level goals, to be achieved with the help of the ERP software.

The enterprise side of the implementation team was then advised, to verify the validity of the goal model, value the intermediary goals, and possibly acquire precisions on specific functional goals. A final version of the goal model was then defined by both sides of the implementation team, with factors for each goal and values for each KPI.

Several observations were done after the conduction of the various goal hierarchies :

- Some KPIs were hard to define precisely. For example, *average time for report creation* is easier to define than the average *number of additional sales per month* which could be done in the future. (Thanks to the automatic reminders during the sales process, see the *Customer relations* goal hierarchy in Appendix F)
- The number of levels was different from one goal to another.
- Improvements appeared clearly under each considered goal. The KPIs appears as unambiguous, valued goal to achieve, thanks to the ERP implementation.
- New possibilities for the enterprise appeared thanks to the refinement of the strategic goals into operational goals. For example, the amount of possible reports for the *improve decision making* goal, or the customer activities tracking in the *better follow up of customers* goal. The enterprise only had one possible report creation prior to the implementation, which always involved a lot of time to be created by the top management. Furthermore, the enterprise never kept track of its past activities for every customer either, only a sales history was preexisting in their previous sales software.

With The goal hierarchy established for every strategic goal of the enterprise, the objectives to achieve on every aspect of the implementation, as well as their priorities, became clear to the entire team.

12.4.3 Business Process Modeling

In parallel to the goal analysis of the enterprise, the most critical business processes were also being analyzed. The sales process was among the most important processes to the eyes of the enterprise top management. This business process had a direct impact on several of their most important strategic goals, such as improvements in customer and supplier relations, efficiency, productivity, or flow of products and services, for example. This process should also be analyzed for its consequences on the deliveries, or the information flow within the enterprise, as well as the needed paperwork during the process. The implementation team thus decided to proceed in a deeper analysis of the sales process support in their ERP solution.

One of the first observations done through the reference process model analysis was the number of entries for the Sales process. The process could start from many different locations, allowing to create a quotation first or directly start with a Sales Order, or an invoice which was the only mandatory document. Very few configuration choices were present in the reference model, and the most noticeable was the sales order management. The functionalities available in the software to support the sales orders were threefold. The orders could be managed normally, meaning sales of services or products with inventory management, and deliveries in case of product selling. A second possibility was aimed at companies managing low stock inventories, these were the *'dropship'* and *'back-to-back'* order management. Back-to-back order management mean that the sales order creation automatically triggers a message, to create the matching purchase order in the purchase department. Dropship orders are dealt similarly to the back-to-back orders, but also include the delivery in the process. Figure 12.2 illustrates a simplified view of this configuration choice. The complete business process reference model of the Sales process is presented in Appendix E and illustrates the entire reference model.

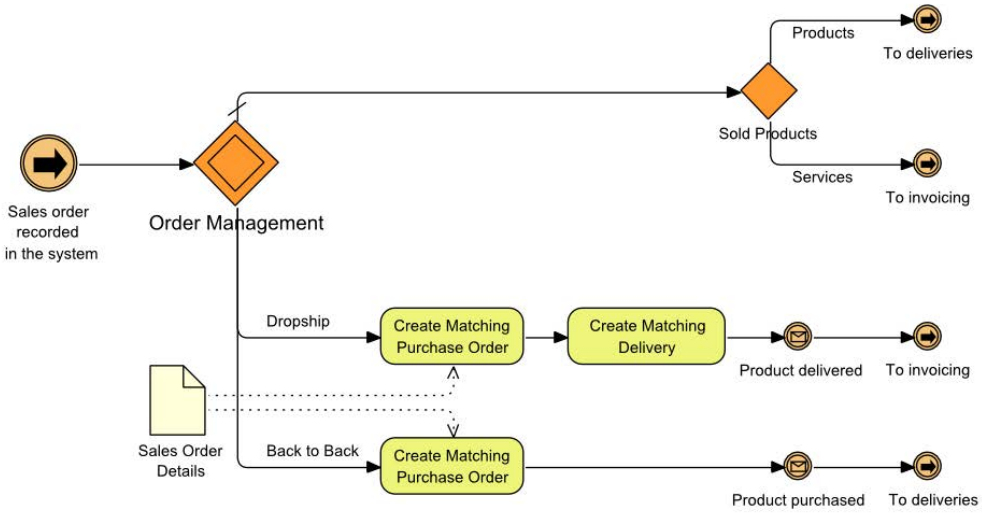


Figure 12.2: Order management configuration choice

The *Order Management* configuration choice is illustrated by a configuration choice related to a simple data-based exclusive gateway. The reference model could be refined like the simple process illustrated in Figure 11.5. The data-based gateway should then result in one permanent choice for all future sales order, or either a subset or the entire set of possibilities. The solution chosen by the enterprise A to manage its sales orders was to use the classic order management by default. However, enterprise A also proposed some special articles to their customers, unavailable in stock due to their quantities and high prices. These articles were mainly specific computer pieces, like high-end graphical boards or flat screens, requiring special orders to their suppliers. This was previously a source of delays or additional work, but this would become easily doable with the new solution. The simplified process, refined to the enterprise requirements, is illustrated in Figure 12.3.

Other components of the Sales process were also investigated, such as the Pick & Pack manager to ease the delivery process and tracking, or the dunning process for a better follow up of customer payments, for example.

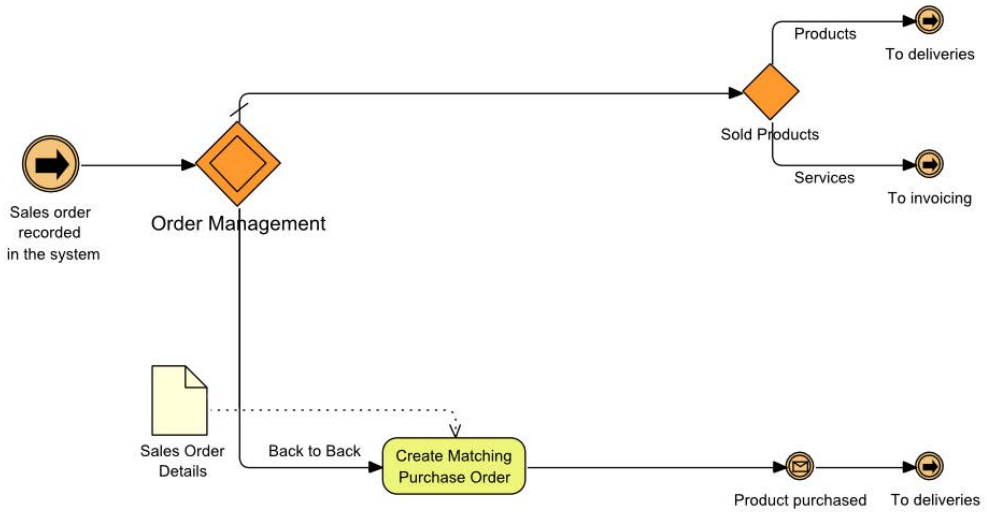


Figure 12.3: Customized sales order management for enterprise A

12.4.4 Conclusions for the Business Case Definition

By the end of the Identification and Modeling stage, the modeling tasks prior to the implementation of the ERP software were achieved, and the ERP software ready to be configured to the enterprise requirements.

The goal model allowed the enterprise to achieve a clear and unambiguous vision of the future benefits of the ERP solution on the enterprise. Even if the implementation costs were high at first sight, the complete goal hierarchy allowed to outline all the functional benefits for every given strategic goal.

The business model was significantly useful to achieve a common vision of the Sales process support, as well as other critical processes. Considering the configuration choice gateways though, these were far less useful in the refinement process of the SAP BO reference model. This can be explained with the very nature of this ERP software solution. SAP BO was developed with the goal to achieve a maximum flexibility in the supported processes, and even if some paths can be blocked, most choices done during the software installation can possibly be undone. Some other specific places in the process could be altered, and additional conditions or notifications could be implemented, but this would require additional coding. Such modifications in the software solution are not considered as configuration choices and are not illustrated in the reference model.

Section 13 :

Conclusion

> **Content :** *This last section presents some conclusions about the advantages of modeling during ERP implementations, as well as some observations and tracks for future improvements or further studies.*

13.1 Conclusions

This thesis was written mainly to achieve two main objectives. The first goal was to acquire a clear and accurate vision of the existing ERP systems, ERP implementations methodologies and tools, and main topics around it. The second goal of the thesis was to synthesize the existing implementation methodologies and possible defaults, and suggest possible modeling improvements to ERP software implementations.

With these objectives in mind, this thesis proposed the following contributions :

- A detailed state of the art on ERP characteristics, main vendors, life-cycles, critical success factors, modeling techniques, and performance assessment and monitoring.
- A synthesis of the ERP implementation topics addressed in the state of the art, providing a unified life cycle for ERP software implementations including stages and dimensions, a review of CSFs for ERP implementations in SMEs, and their relative importance through the implementation process, and a deep analysis of ERP implementations early stages.
- Proposals for modeling improvements, for early stages of ERP implementations in SMEs, under goal modeling and business process modeling perspectives, illustrated in a case study.

The first encountered problem was to acquire the most possibly complete vision of the ERP software characteristics, implementation methodologies and tools. The state of the art presented in this thesis tried to be the most possibly extensive within each addressed topic. It was designed to give a global vision of both ERP systems and ERP implementations, under several aspects of these major topics. Though it was not intended to address every aspect of academic researches around ERP system implementations. ERP software is a broad topic, and other academic works have been published to present the extent of researches around ERP implementations.

Regarding private tools and methodologies, the only vendor-based methodology available for this work was the SAP Business One methodology, so this state of the art is limited on

the observations around the vendor methodologies. From observations over the SAP materials though, it seems that ERP systems implementations are largely covered by their respective vendors, by the means of tools, forms, and materials to be provided to the customer. The analysis of the SAP Business One methodology pointed out a lack of modeling tasks, questioning the use of such tasks in ERP implementations. From both the observations of the academic works and the SAP materials, a new, integrated methodology was proposed. This methodology included a unified life-cycle, CSFs to be used as guidelines for ERP implementation projects in SMEs, and modeling propositions, including modeling techniques unavailable in the SAP Business One implementation methodology.

Producing an integrated, unified life-cycle was done to provide a fresh vision of ERP implementation stages and dimensions. The researches and studies found in the literature use their own life-cycles or one of the several commonly accepted life-cycles, but there is a lack of a single unified proposition. Such life-cycle could be useful for researchers to share the same language in all works. This could allow the researchers in the field to classify their works within each and every stage of an ERP implementation project, under various dimensions. The life-cycle presented in this thesis was inspired from all the other life-cycles encountered, and designed to give the ability for researchers to do such classification. A compromise was found between the four initial stage of [MAR2K] and the nine stages from [SOJ07], based on the milestones and main tasks of all the life-cycles. It tries to represent ERP implementations as clearly and simply as possible. CSFs were then arranged within the life-cycle, to expose the important guidelines to keep in mind during the implementation project.

Finally, the first two stages of the implementation were detailed, and illustrated in a case study, to show the potential of modeling techniques in ERP implementations. The SMEs have stronger constraints than large enterprises, and require even more attention to the costs of the modeling techniques. These modeling techniques were proposed with that idea in mind.

The modeling techniques presented in these sections have the advantages of being rather simple, thus not implying high additional costs for the project, while also providing keys to achieve a common vision of every aspect of the enterprise business, and how the solution will have to cover them. Goal modeling and business process modeling techniques were based on the findings in the literature, to help the implementation stakeholders to share a common vision of the steering objectives of the project, and the business processes of the enterprise.

13.2 Future works

It would be interesting to compare what has been found in the state of the art with methodologies coming from other private vendors, or in the open source world. This would allow a deeper analysis and add information over various topics, like modeling or performance management, for example.

A major part of the future works would be related to the proposed methodology. Indeed the methodology proposed in this work is based on theoretical assumptions and relies mostly on works from the literature, and observations during their analysis. This methodology needs to be tested on a real case study, to show its potential, and the details to be improved. This could not be done during the writing of the thesis, but would ensure the validity of the proposed modeling techniques of the methodology.

For example, it would be interesting to extensively test the goal modeling technique through several enterprises and solutions. This could allow a validation of the methodology, an esti-

mation of the possible reusability level for the goal hierarchies between the enterprises and solutions, and possible improvements to the notation.

Considering the configuration choices notation, a possible additional extension would be to deal differently with the hiding and blocking on one hand, as illustrated in this work, and the *optional* blocking and hiding on another hand, with another notation, different from the double-diamond gateways. This might allow a distinction between the configuration choices which could be deferable to run-time, and the ones which could not. However, the usefulness of such additional notation would also require some verification on real case studies, and multiple reference models for various ERP solutions.

A tool could also be developed to support the refinement process from the generic reference business process model to the specific model of the enterprise. Another useful tool to develop would support the goal hierarchy design, with the ability to support predefined intermediary goals, or predefined hierarchies for a specific solution, or a specific industry vertical.

ERP software is a broad topic, and their implementation, always a complex and risky process. This work can be used as a basis for anyone willing to conduct research on ERP software, or ERP implementations. Numerous academic works were referenced, covering and synthesizing various important aspects around ERP software and their implementations. Commonly accepted consensus are difficult to achieve within most ERP software topics, due to their very nature, ERP software being complex, involving the entire organization with high economic stakes. Furthermore, every ERP software solution has its particularities and every ERP implementation can be motivated by different reasons. With these important constraints in mind, further research should be supported through real data analysis, and the methodology should be experimented in real context to show its potential benefits and flaws, and to be improved in the future.

Bibliography

[Aal03] W.M.P. van der Aalst, A.H.M. Ter Hofstede, B. Kiepuszewski and A. P. Barros, 'Workflow Patterns', *Distrib. Parallel Databases*, Vol. 14, No. 1, pp. 5-51, 2003.

[Aal06] W.M.P. van der Aalst, A. Dreiling, F. Gottschalk, M. Rosemann, and M.H. Jansen-Vullers, 'Configurable Process Models as a Basis for Reference Modeling', in *Business Process Management Workshops*, pp. 512-518, 2006.

[ABT97] C. Abts, B. Boehm, 'COTS Software Integration Cost Modeling Study', Center for Software Engineering, University of Southern California, 1997.

[BPMN08] The OMG Group, 'Business Process Modeling Notation Specification, V1.1', 17/01/2008, retrieved from `http://www.omg.org/spec/BPMN/1.1/PDF` on 04/03/08.

[BRE05] N. Brehm, J. Marx Gomez and C. Rautenstrauch, 'An ERP solution based on web services and peer-to-peer networks for small and medium enterprises', *Int. J. Information Systems and Change Management*, Vol. 1, No. 1, pp. 99111, 2005

[CHA06] Amel Chaabouni, 'Implantation d'un ERP : antécédents et conséquences' , XV$^{\text{ème}}$ Conférence Internationale de Management Stratégique, Annecy / Genève, 13-16 June 2006.

[COM08a] Compiere Open Source ERP and CRM Business Solution, retrieved from `www.compiere.com` on 10/02/2008.

[COM08b] Compiere Open Source ERP and CRM Business Solutions Software Capabilities , retrieved from *http://www.compiere.com/products/capabilities/index.php* on 11/02/2008.

[COO90] R. B. Cooper, R. W. Zmud, 'Information Technology implementation research : a technological diffusion approach', *Management Science*, Vol. 36, No. 2, pp. 123-139, 1990..

[DAN03] M. Daneva, 'Lessons Learnt from Five Years of Experience in ERP Requirements Engineering', *RE '03: Proceedings of the 11th IEEE International Conference on Requirements Engineering*, IEEE Computer Society, Washington, DC, USA, 2003.

[DAV93] Thomas H. Davenport, *Process Innovation : reengineering work through information technology*, Harvard Business School Press, Boston, MA, USA, 1993.

[EW08a] Peter Galli, 'Compiere User Community Splits; Code Forks', *www.eweek.com*, 2006-10-19, retrieved from `http://www.eweek.com/c/a/Enterprise-Apps/Compiere-User-Community-Splits-Code-Forks/` on 04/04/2008.

[EW08b] Renee Boucher Ferguson, 'Oracles X Factor', *www.eweek.com*, 2007-04-16, retrieved from `http://www.eweek.com/c/a/Enterprise-Apps/Oracles-X-Factor/`, on 03/04/2008.

[EI08a] Benoït Herr, 'Le petit géant qui monte : Infor', *www.erpinfos.com*, retrieved from `http://www.erp-infos.com/article_in_mid.php?id_news=184` on 03/04/2008.

[EI08b] René Beretz, 'Conçu pour les PME, Navision a pris de l'ampleur en devenant Microsoft Dynamics NAV', *www.erpinfos.com*, retrieved from `http://www.erp-infos.com/article/m/223/concu-pour-les-pme-navision-a-pris-de-lampleur-en-devenant-microsoft-dynamics-nav.html` on 03/04/2008.

[EI08c] Benoït Herr, 'L'adoption des ERP en mode SaaS reste en retrait, selon une étude Markess International', *www.erpinfos.com*, retrieved from `http://www.erp-infos.com/article/m/212/ladoption-des-erp-en-mode-saas-reste-en-retrait-selon-une-etude-markess-international.html`, on 03/04/2008.

[EI08d] Benoït Herr, *ERP en PME : quadrature du cercle ou panacée ?*, erpinfos.com, retrieved from `http://www.erp-infos.com/article/m/221/erp-en-pme\protect\unhbox\voidb@x\penalty\@M\-quadrature-du-cercle-ou-panacee-premiere-partie.html`, on 03/04/2008.

[EST99] José M. Esteves, Joan A. Pastor, 'An ERP Life-cycle based Research Agenda', *First International workshop in Enterprise Management and Resource Planning : Methods, Tools and Architectures - EMRPS'99*, Venice, Italy, 1999.

[EST2K] José M. Esteves, Joan A. Pastor, 'Towards the unifications of critical success factors for ERP implementations', *10th Annual BIT conference*, Manchester, UK, 2000.

[EST01] José M. Esteves, Joan A. Pastor, 'Analysis of Critical Success Factors Relevance Along SAP Implementation Phases', *Seventh Americas Conference on Information Systems*, 2001.

[EST03] José M. Esteves, Joan A. Pastor, João Carvalho, 'Organizational and national issues of an ERP implementation in a portuguese company', *IFIP 8.2+9.4 conference*, pp. 139-154, 2003.

[EST07] José Esteves, V. Bohorquez, *An Updated ERP Systems Annotated Bibliography: 2001-2005*, Working Papers Economia wp07-04, Instituto de Empresa, Area of Economic Environment, March, 2007.

[GOT07] Florian Gottschalk, Wil M. P. Van der Aalst, Monique H. Jansen-Vullers, 'Configurable Process Models A Foundational Approach', *Reference Modeling*, pp. 59-78, Springer, 2007.

[GAM06] Magnus Gammelgård, 'A Categorization of Benefits From IS/IT Investments', *Proceedings of the 13th European Conference on Information Technology Evaluation*, September 2006.

[GAM07] Magnus Gammelgård, Mårten Simonsson, Åsa Lindström, 'An IT management assessment framework: evaluating enterprise architecture scenarios', *Information Systems and E-Business Management*, Vol. 5, No. 4, pp. 415-435, September 2007.

[GRA07] Michael Grabinski, 'Balanced Scorecards', in *Management Methods and Tools*, in *Balanced Scorecards*, pp. 35-47, Gabler, 2007.

[HEC04] André Tchokogué, Yves Langevin, Nicolas Hien, *Méthodologies d'implémentation des progiciels de gestion intégrée : diversité et similitudes*, Cahier du groupe de recherche CHAINE n° 04-06, Août 2004.

[HOF91] Hofstede, G., *Cultures and Organizations : Software of the Mind*, McGraw-Hill, New York, 1991.

[HOL99] Christopher P. Holland, Ben Light, 'A Critical Success Factors Model For ERP Implementation', *IEEE Software*, Vol. 16, No. 3, pp. 30-36, May, 1999.

[HOY07] V. Hoyer, O. Christ, 'Collaborative e-Business Process Modelling: A Holistic Analysis Framework focused on Small and Medium-Sized Enterprises.', *Lecture Notes in Computer Science (LNCS)(4439)*, Springer, Berlin, 2007.

[INF08] 'Software Solutions for CRM, EAM, ERP, FMS, HCM, PM, PLM, SCM, SRM', `http://www.infor.com`, accessed on 08/01/2008.

[KAN06] V. Kansal, 'Enterprise Resource Planning Implementation: A Case Study.', *Journal of American Academy of Business, Cambridge*, Vol. 9, No. 1, pp. 165-170, March, 2006:

[KAP92] Kaplan, Robert S. and Norton, David P. 'The balanced scorecard - measures that drive performance', Harvard Business Review, Vol. 70, No. 1, pp. 71-79, January-February 1992.

[KAP96] Kaplan, Robert S. and Norton, David P., *The Balanced Scorecard - translating strategy into action*, Harvard Business School Press, Boston, MA, 1996.

[KLA2K] Helmut Klaus, Michael Rosemann and Guy G. Gable, 'What is ERP?', *Information Systems Frontiers*, Vol. 2, No. 2, pp. 141-162, August, 2000.

[KWO87] T. H. Kwon and R. W. Zmud, 'Unifying the fragmented models of information systems implementation', in *Critical issues in information systems research*, John Wiley & Sons, Inc. , New York, NY, USA, pp. 227-251, 1987.

[LYP04] Choon Seong Leem, Chui Young Yoon, Seung Kyu Park, 'A process-Centered IT ROI Analysis with a Case Study', *Information Systems Frontiers*, Vol. 6, No. 4, pp. 369383, 2004.

[MAR2K] M. Markus, C. Tanis, 'The enterprise systems experience - from adoption to Success', in *Framing the Domains of IT Management: Projecting the Future Through the Past*, Robert W. Zmud and Michael F. Price Pinnaflex Educational Resources, Inc. , Cincinnatti, OH, pp. 173-207, 2000.

[MEN06] J. Mendling, M. Moser, G. Neumann, H.M.W. Verbeek, B.F. van Dongen, and W.M.P. van der Aalst, 'Faulty EPCs in the SAP Reference Model', *Lecture Notes in Computer Science : Business Process Management*, Vol. 4102, pp. 451457, 2006.

[MSD08] Microsoft, 'Microsoft Belgique et Luxembourg - Dynamics', retrieved from `http://www.microsoft.com/belux/fr/dynamics/` on 04/03/2008.

[MSN08] Microsoft, *Microsoft Dynamics NAV (formerly Microsoft Navision)* retrieved from `www.microsoft.com/dynamics/nav` on 04/03/2008.

[NLK01] Fiona Fui-Hoon Nah, Janet Lee-Shang Leu and Jinghua Kuang, 'Critical factors for successful implementation of enterprise systems', *Business Process Management Journal*, Vol. 7, No. 3, pp. 285-296, 2001.

[OEG08] *'Open Source ERP Guru'*, retrieved from `http://opensourceerpguru.com/` on 07/01/2008.

[ORACLE08] *Strategic acquisitions*, retrieved from `http://www.oracle.com/corporate/acquisition.html` on 04/03/2008.

[PEE04] Peerstone Research, *ERP ROI : Myth and Reality*, January 2004.

[POR06] V. Portougal, D. Sundaram, *Business Processes: Operational Solutions for SAP Implementation.* IRM Press, 2006.

[RAJ02] P. Rajagopal, 'An innovation-diffusion view of implementation of enterprise resource planning (ERP) systems and development of a research model', *Information and Management*, Vol. 40, No. 2, pp. 87-114, 2002.

[ROL2K] C. Rolland, N. Kaprash, 'Bridging the gap between organizational needs and ERP functionality', *Requirements Engineering Journal*, Vol. 41, pp. 180-193, 2000.

[ROL05] C. Rolland, C. Salinesi, 'Modeling Goals and Reasoning with Them', in *Engineering and Managing Software Requirements*, pp. 189-217, Springer Berlin Heidelberg, 2005.

[ROS2K] Jeanne W. Ross and Michael R. Vitale, 'The ERP Revolution: Surviving vs. Thriving', *Information Systems Frontiers*, Vol. 2, No. 2, , pp. 233-241, 2000.

[SAL07] C. Salinesi, M. Ramzi and E. Elfassy, 'An Experience of Reuse Based Requirements Engineering in ERP Implementation Projects', in *Proceedings of the 11th IEEE International Enterprise Distributed Object Computing Conference*, IEEE Computer Society, 2007.

[SAP07] SAP AG, 'SAP History : From start-up software vendor to global market leader', accessed from `http://www.sap.com/about/company/history.epx`, on 16/11/2007.

[SAP08] SAP AG, 'SAP Educational Portal', accessed from *http://sap.channel.com*, on 16/03/2008.

[SOF01] P. Soffer, B. Golany, D. Dori, Y. Wand, 'Modelling off-the-shelf information systems requirements : an ontological approach', *Requirements Engineering Journal*, Vol. 6, No. 3, pp. 183-199, 2001.

[SOF03] P. Soffer, B. Golany, and D. Dori, 'ERP Modeling : a Comprehensive Approach', *Information Systems Journal*, Vol. 28, No. 6, pp. 673-690, 2003.

[SOJ07] Piotr Soja, 'Success factors in ERP systems implementation: Lessons from practice', in *Advances in Information Systems Development*, pp. 275-286, Springer US, 2007.

[SOL04] Rini van Solingen, 'Measuring the ROI of Software Process Improvement', *IEEE Software*, Vol. 21, No. 3, pp 32-38, May, 2004.

[SOM01] Toni M. Somers and Klara G. Nelson, 'The impact of Critical Success Factors across the Stages of Enterprise Resource Planning Implementations', *hicss*, 34th Hawaii International Conference on Systems Sciences (HICSS-34), Vol. 8, 2001.

[SOM04] Toni M. Somers and Klara G. Nelson, 'A taxonomy of players and activities across the ERP project life cycle', *Information and Management*, Vol. 41, No. 3, pp. 257-278, 2004.

[SWA04] Paul M. Swamidass, *Encyclopedia of production and manufacturing management*, Kluwer Academic Publisher, 2004.

[WAS06] Avi Wasser, Maya Lincoln, and Reuven Karni, 'ERP Reference Process Models: From Generic to Specific', in *Lecture Notes in Computer Science*, Vol. 4103, pp. 45-54, 2006.

[WEI07] Chun-Chin Wei, 'Evaluating the performance of an ERP system based on the knowledge of ERP implementation objectives', *The International Journal of Advanced Manufacturing Technology*, Springer London, 2007, retrieved on `http://www.springerlink.com/content/q127466147661610/?p=56f8269d70294854a9bc1bff5b03ef5b\&pi=0`.

[WIE06] Martin Wiener, *Critical Success Factors of Offshore Software Development Projects*, DUV, Gabler Edition Wissenschaft, 2006.

[WOH06] P. Wohed, W. M. P. van der Aalst, M. Dumas, A. H. M. ter Hofstede and N. Russell, 'On the suitability of BPMN for Business Process Modeling', in *Business Process Management*, pp 161-176, Springer Berlin / Heidelberg, 2006.

[WSW05] Jen-Her Wu, Shin-Shing Shin and Chi-Cheng Wu, *COTS-based Systems: A Methodology for Evaluating Data and Output Misfits*, P*roceedings of the 38th Hawaii International Conference on System Sciences*, 2005.

[WWW03] Jen-Her Wu, Yuh-Min Wang and Hsi-Tze Wan, *Enterprise Resource Planning Experience in Taiwan: An Empirical Study and Comparative Analysis*, *Proceedings of the 36th Hawaii Conference on System Sciences*, 2003.

[ZAC87] John A. Zachman, 'A framework for Information Systems Architecture', *IBM Systems Journal*, Vol. 26, No. 3, 1987.

[ZHA03] Liang Zhang, Matthew K.O. Lee, Zhe Zhang , Probir Banerjee, 'Critical Success Factors of Enterprise Resource Planning Systems Implementation Success in China', *Proceedings of the 36th Hawaii International Conference on System Sciences*, 2003.

[ZHA05] Liang Zhang, Matthew K. O. Lee, Zhe Zhang, Christy M. K. Cheung, 'ERP Systems Implementation Determinants and Success Measures in China : a Case Study Approach', in *Enterprise Information Systems V*, Springer Netherlands, pp. 109-116 , 2005.

Part III

Appendix

Appendix A :

Comparison Table for ERP Capabilities

Comparison Table of the capabilities for Compiere, SAP Business One, and MS NAV ERP solutions.

	Compiere	SAP BO	Microsoft NAV
Partner Relations Management	Partner Relations Management	Business Partner Management	(incl. Customer Relationship Management)
Material Management	Material Management	(incl. Inventory + MRP)	(incl. Resource Planning)
Pricing	Pricing	(incl. Inventory)	(incl. Inventory)
Procurement	Procurement	Purchasing – A/P	Supply Chain
Order Management	Order Management	Sales - A/R	(incl. Management)
Open Item Management	Open Item Management	(incl. Financials + MRP)	(incl. Financial + Resource Planning)
eCommerce	eCommerce	Electronic Commerce	eCommerce modules
Customer Relations Management	Customer Relations Management	Sales Opportunity Management	Customer Relationship Management
Project Management	Project Management	none	Project Management
Financial Management	Financial Management	Financials	Financial Management
Performance Management	Performance Management	Service Management	Service Management
Reporting	Reporting	Reporting	(incl. In the respective modules)
Production	Manufacturing	Production	Production
Inventory	(incl. Pricing)	Inventory	Inventory Management
Material Requirement Planning	(incl. Procurement)	Material Requirement Planning	Resource Planning
Human Resources Management	(incl. Partner Relations Managt)	Human Resources Management	Human Resources
Banking	(incl. Financial Management)	Banking	(incl. Financial Management)
Marketing	(incl Partner and Customer Relations Managementt)	(incl Partner and Customer Relations Management)	Marketing

Legend :

Name of the process in the ERP software	: Available in the ERP software, as a category
Incl. Another process of the software	: Functions available in the software, but with another name in one or several other modules

Figure 1: Comparison Table for ERP Capabilities

Appendix B :

Comparison Table for ERP Life-cycles

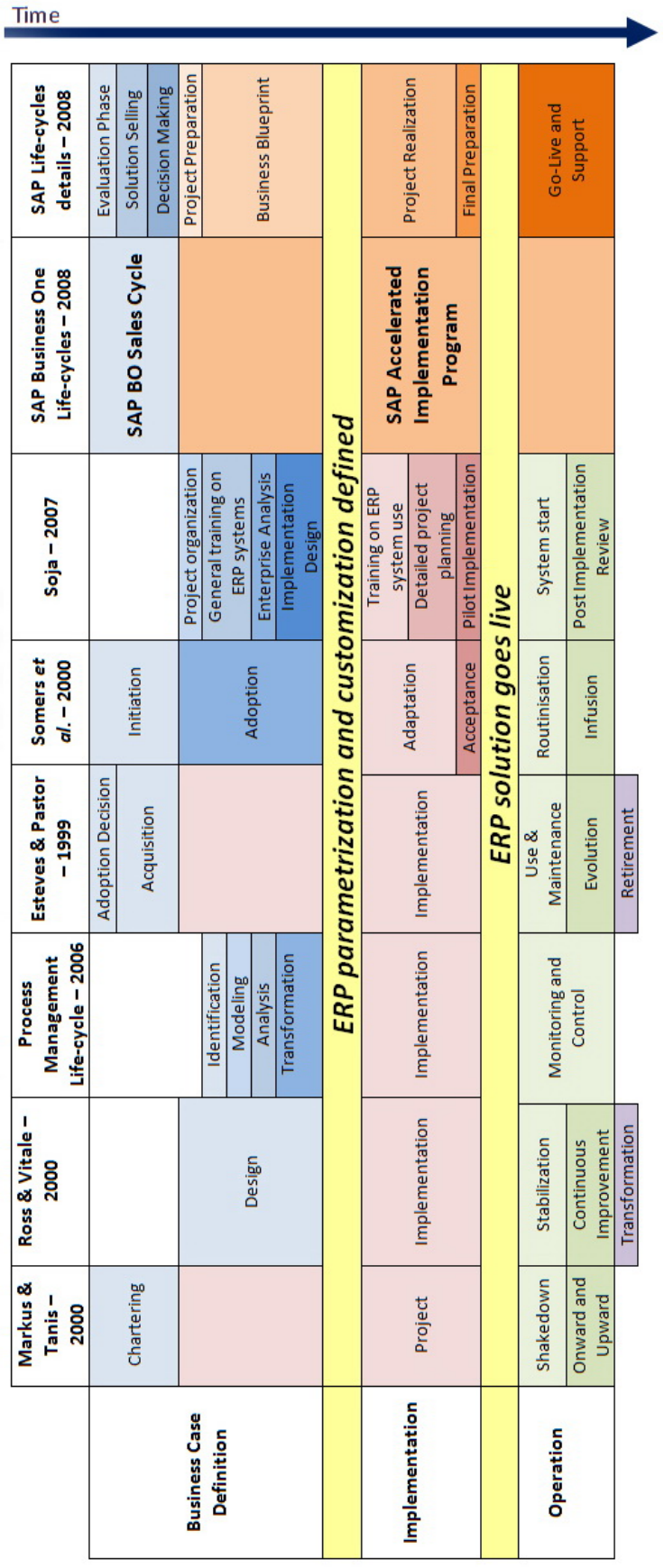


Figure 2: Comparison Table for ERP life-cycles

Appendix C :

Critical Success Factors for ERP implementations in SMEs

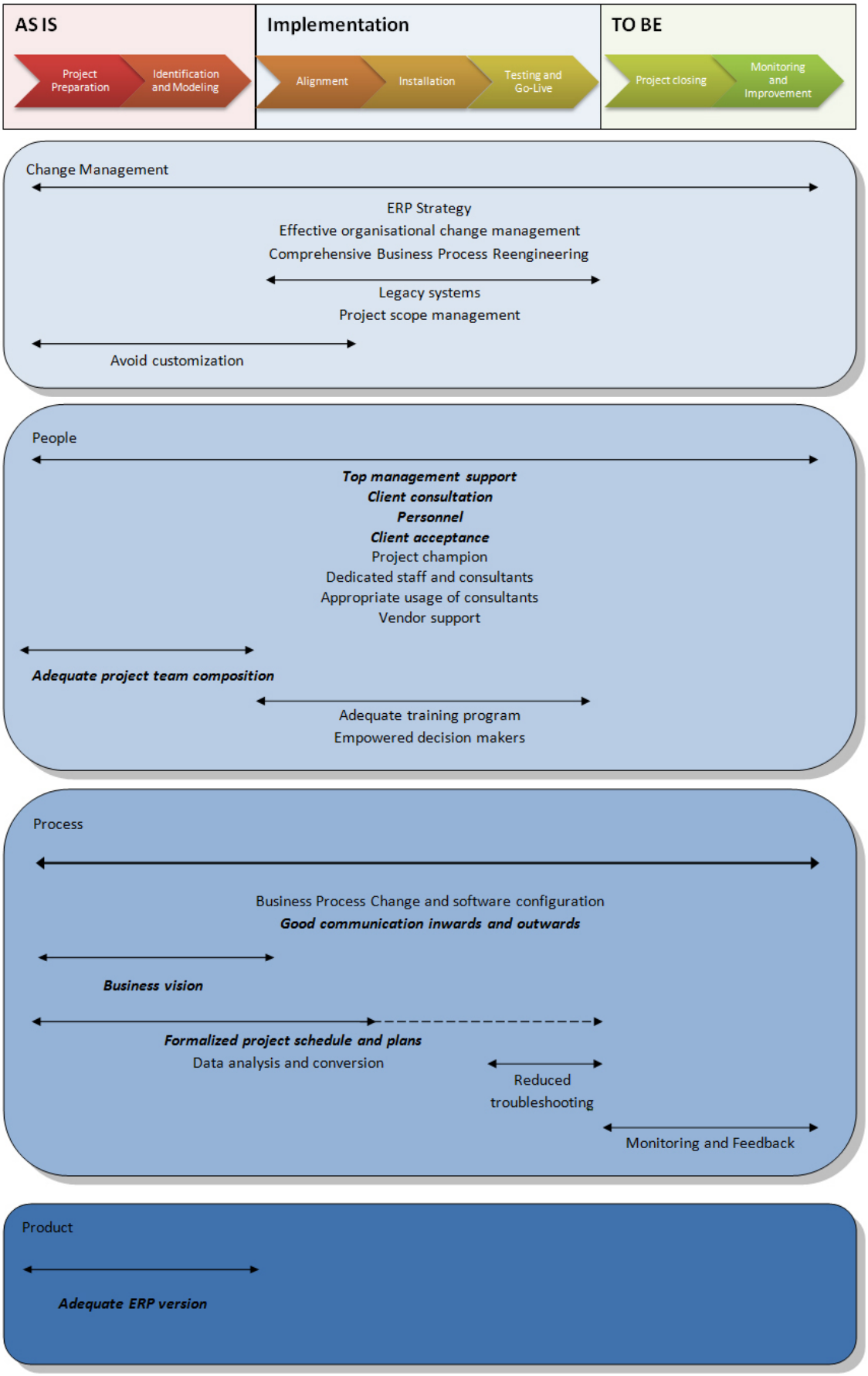


Figure 3: Critical Success Factors for ERP Implementations in SMEs

Appendix D :

ERP PI structure and corresponding weights

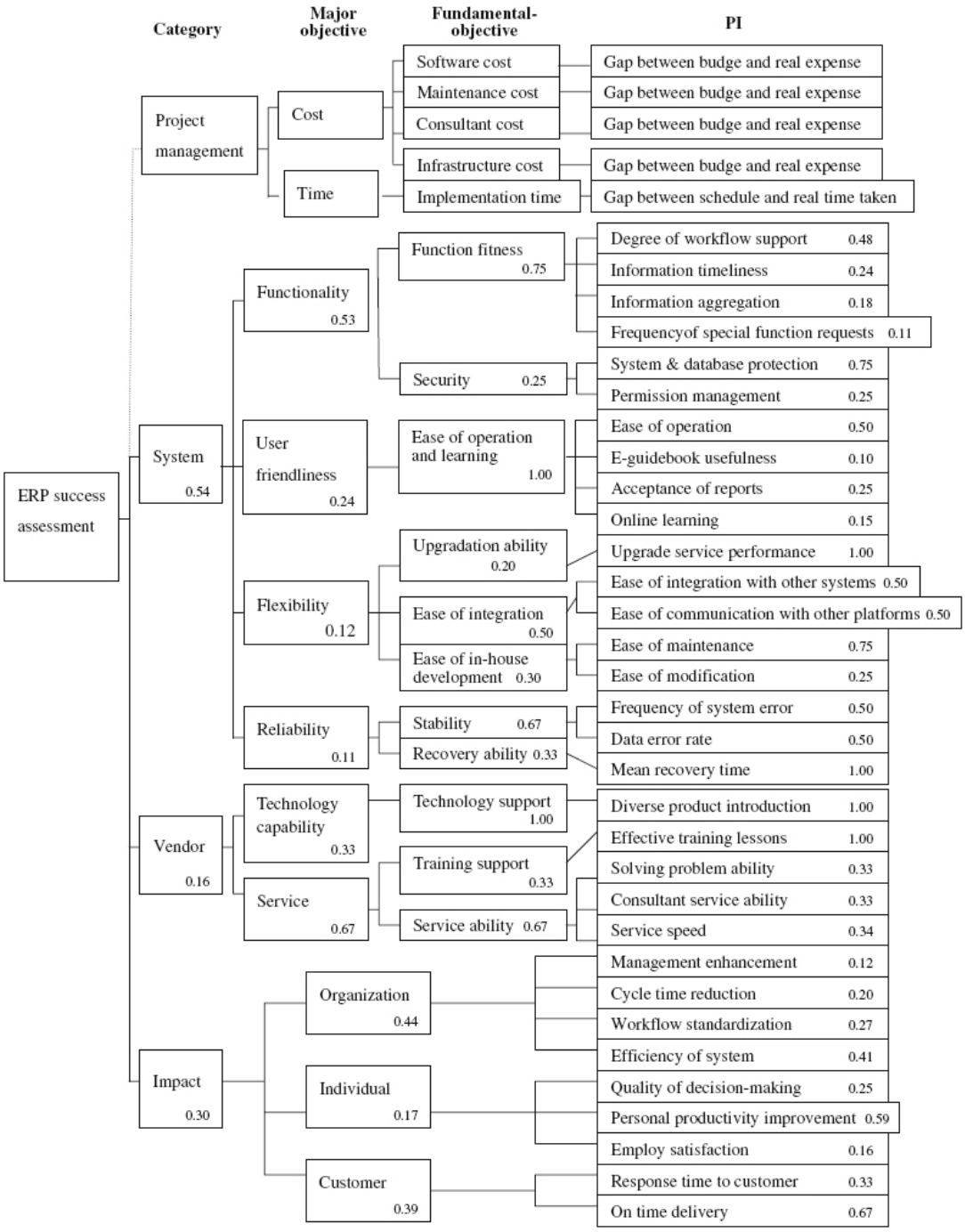


Figure 4: 'ERP PI structure and their corresponding weights', from [WEI07]

Appendix E :

BPMN Diagram of SAP BO Sales Process

SAP BO Sales Process

Figures 5, 6, 7, and 8 illustrates the business process reference model for SAP Business One used during the case study. The main process was divided in these four sub-diagrams for readability concerns. The subprocesses are then detailed in the remaining Figures, following the main process.

The reader should be advised that not all exceptions have been modeled during the business modeling process, to keep this illustration model rather simple. This model is not intended to precisely present all the details of the Sales process support of SAP Business One, but rather an illustration of the configuration choices, applied to the ERP solution available for this work.

For example, the Return goods mechanics have been excluded from the model, for readability concerns. In the complete SAP BO process, products could be undelivered, or unavailable for purchase in a dropship or back to back process, or even returned by the customer during the Sales process. The return processes would have several impacts on various places in the model, in terms of credit memo, roll back in the main process, or compensation and cancellation triggers. We thus made the choice to exclude the Return goods mechanics from this process model.

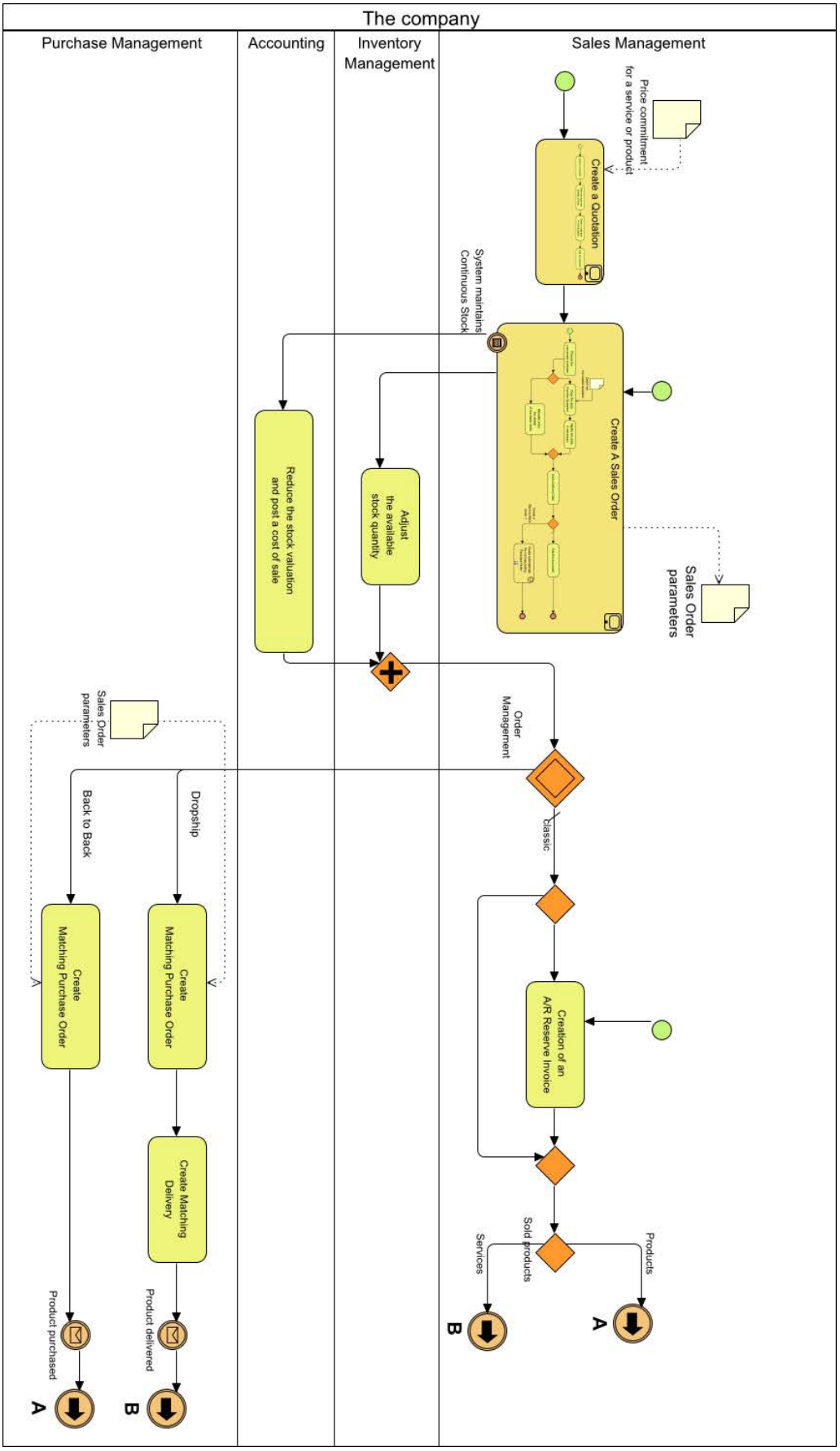


Figure 5: SAP BO Sales Process (1)

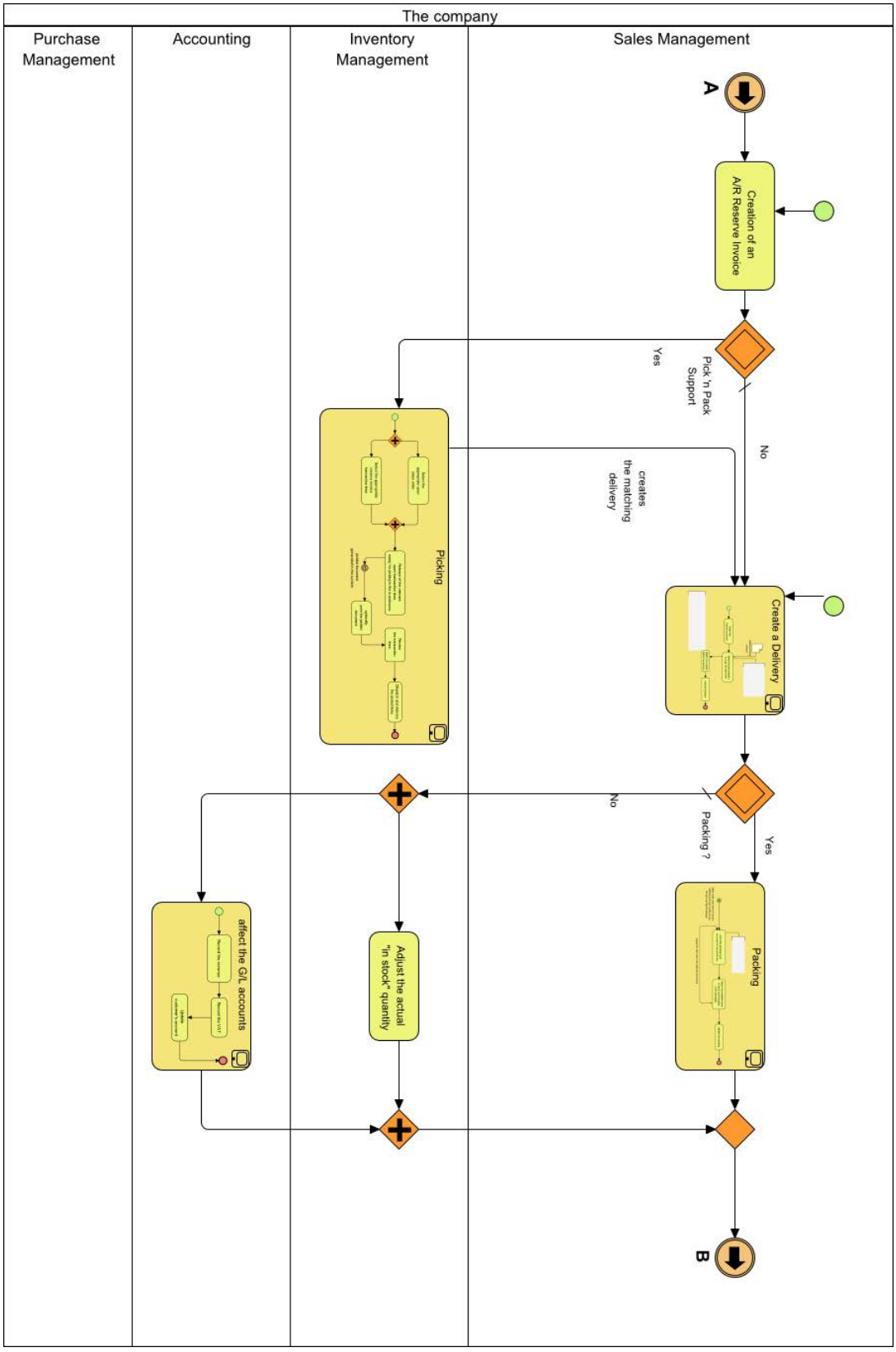


Figure 6: SAP BO Sales Process (2)

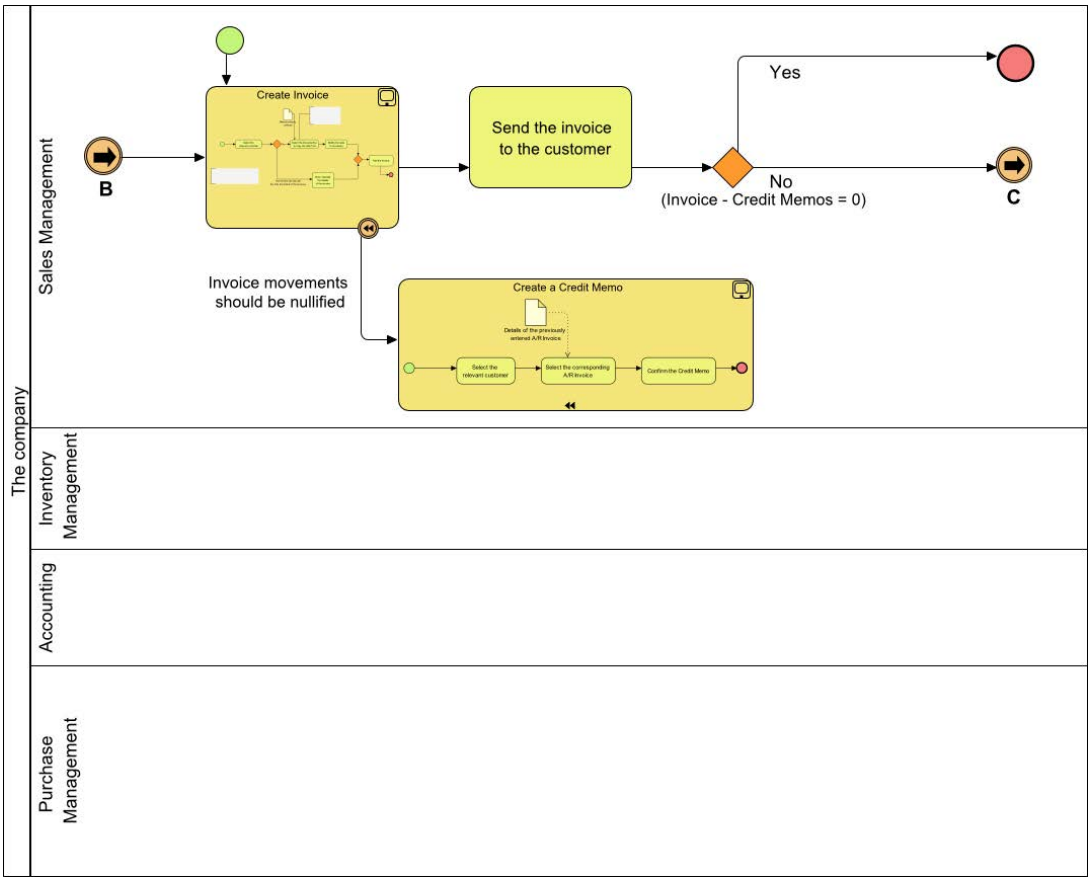


Figure 7: SAP BO Sales Process (3)

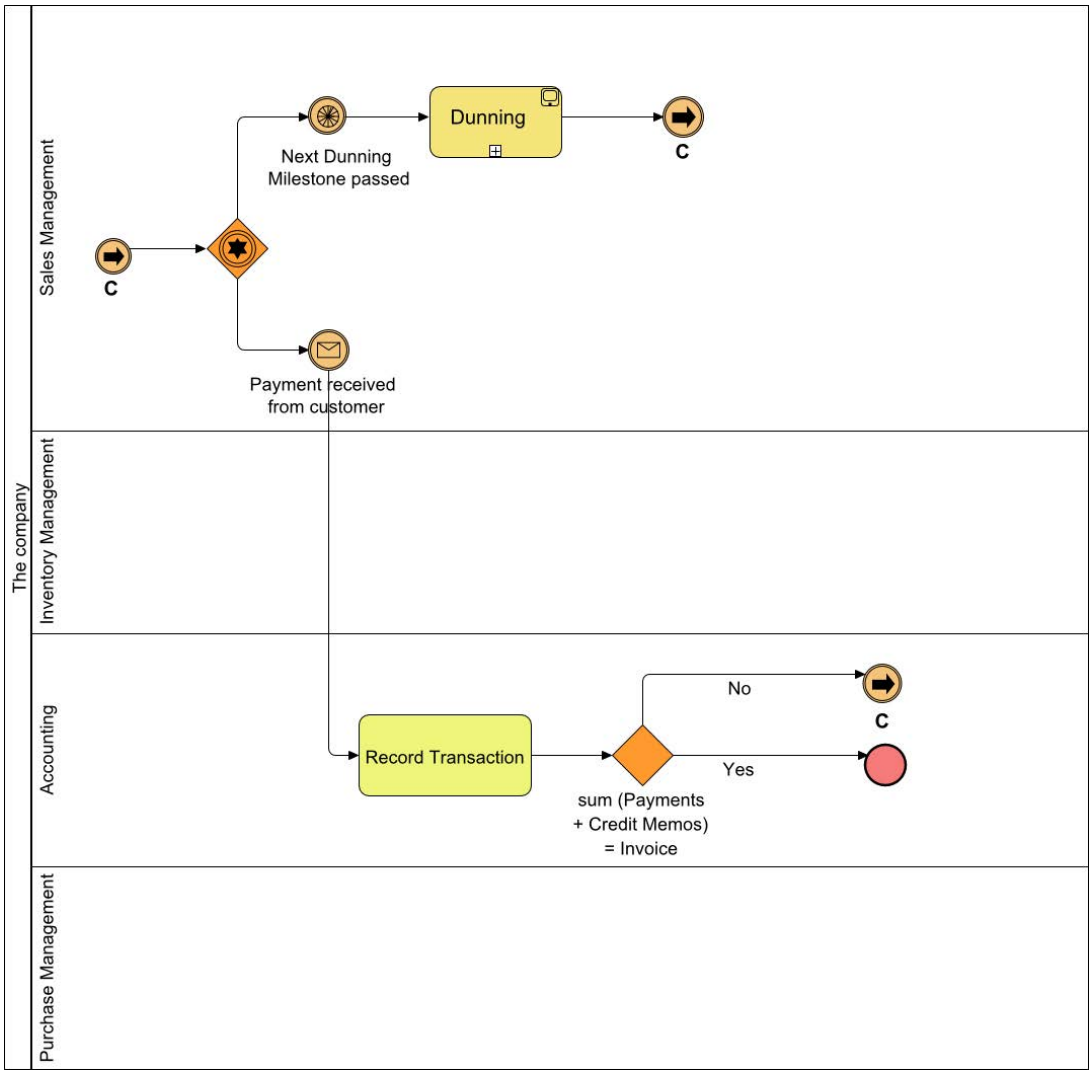


Figure 8: SAP BO Sales Process (4)

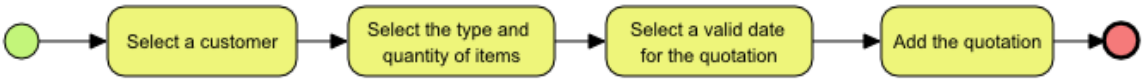


Figure 9: Create a quotation

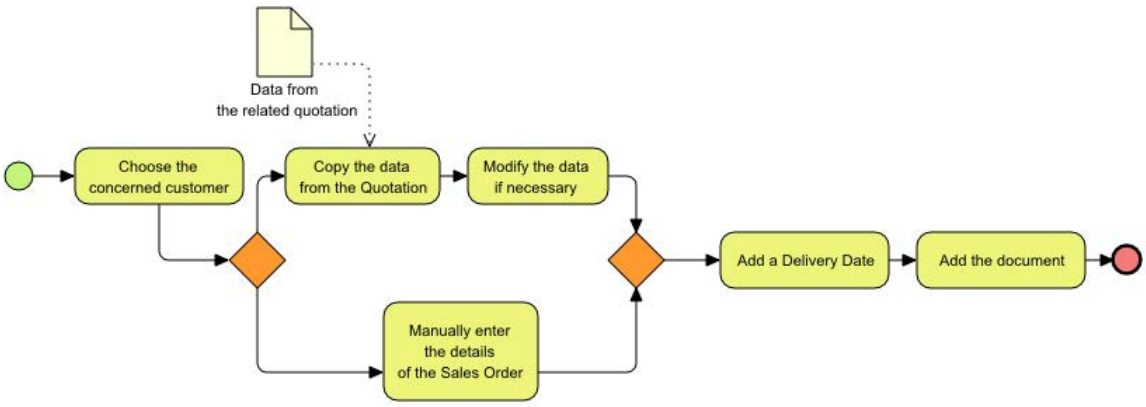


Figure 10: Create a Sales Order

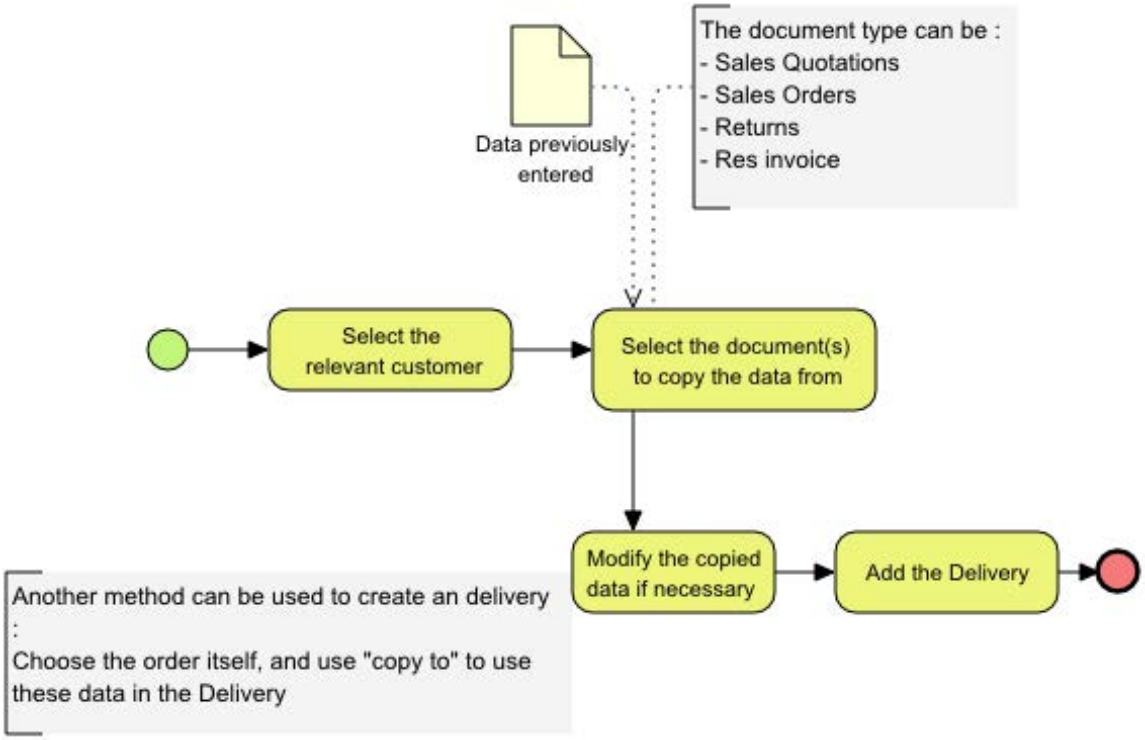


Figure 11: Create a Delivery

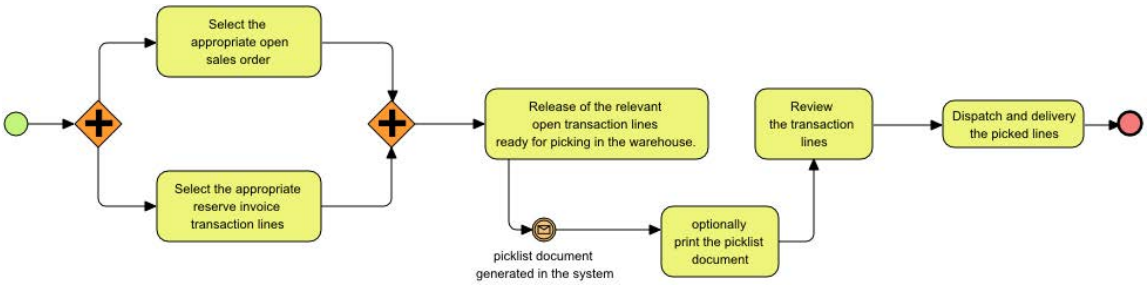


Figure 12: Picking Sub-Process

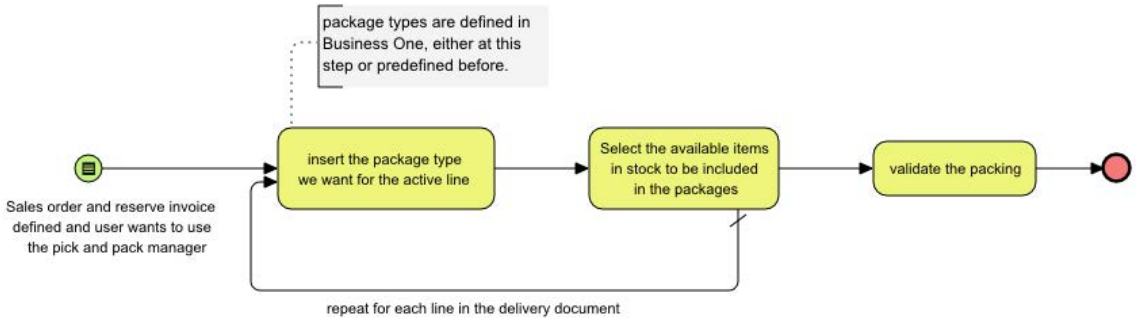


Figure 13: Packing Sub-Process

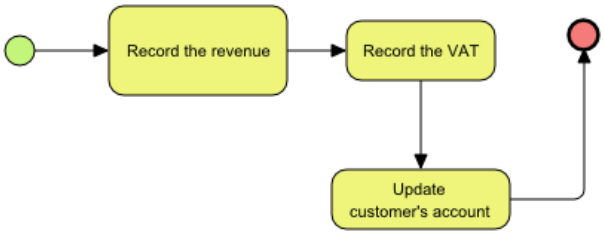


Figure 14: Affect the G/L Accounts

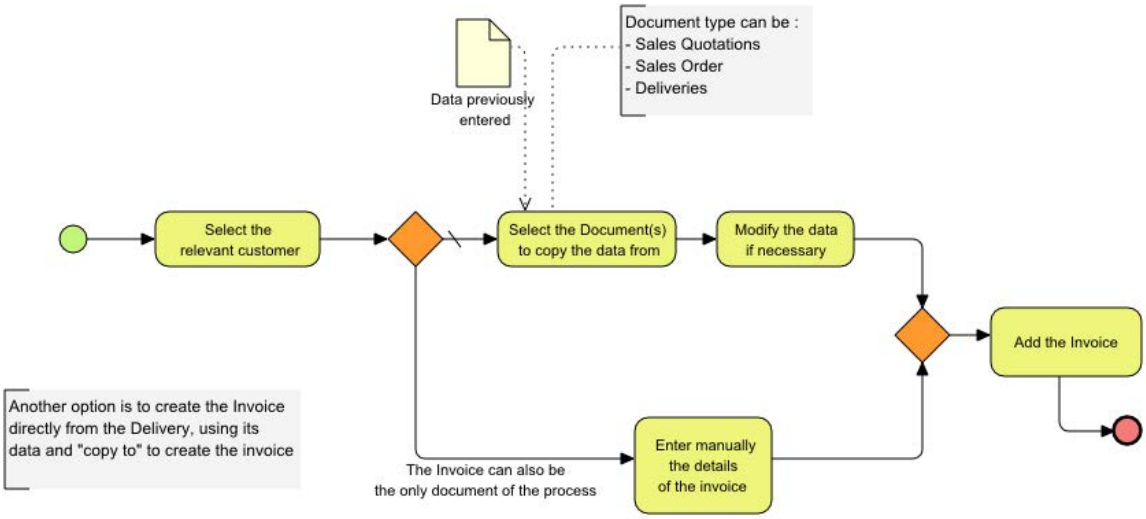


Figure 15: Create an invoice

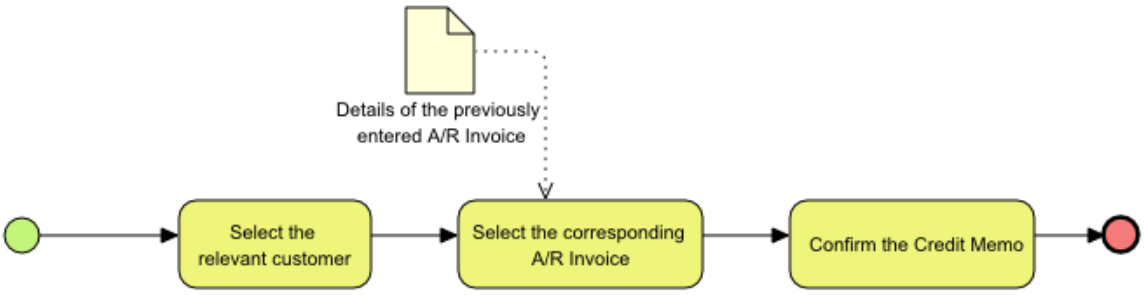


Figure 16: Create a credit memo

Appendix F :

Goal hierarchies for enterprise A

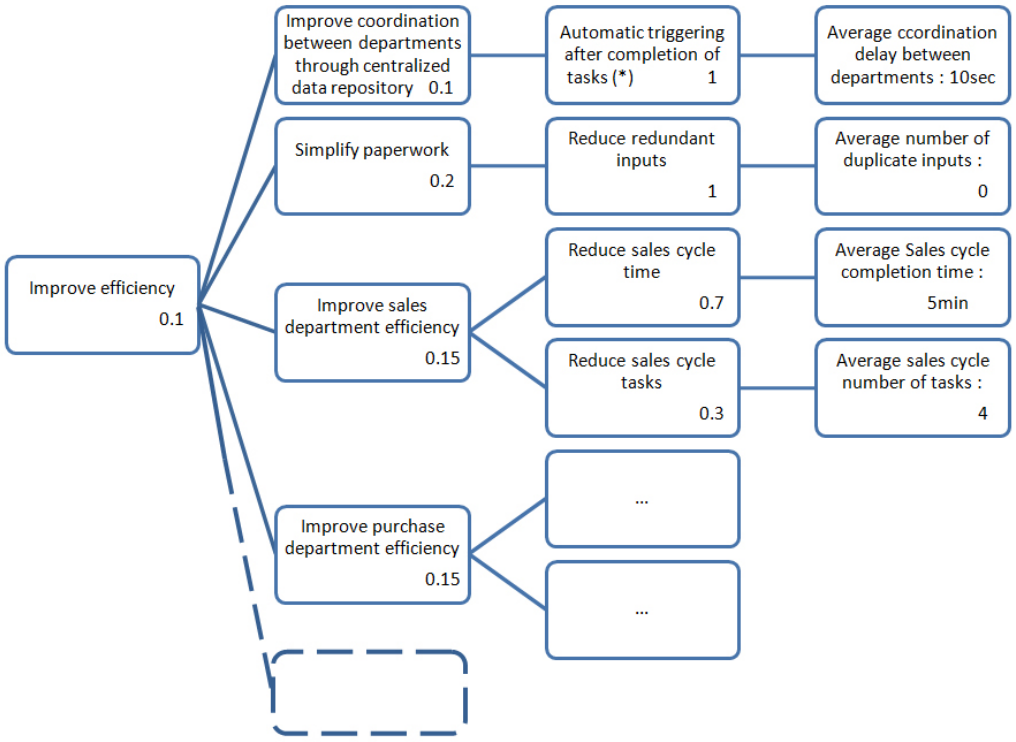


Figure 17: Partial goal hierarchy for the *Improve efficiency* strategic goal for enterprise A.

[GAM06] defines Efficiency as : ' *"Doing as much as before with less resources", e.g. shortening of manufacturing times/lead times/cycle times/work times, simplified/reduced paperwork and administrative tasks, automation of work tasks, staff reductions etc' [GAM06].* This means improvements at a the general level of the enterprise, and improvements in every individual department. All departments of the enterprise should be included in this Figure, and analyzed like illustrated for the sales department. Our case being an illustration case, not all possible departments have been included in Figure 17.

NOTE : Examples of automatic triggering are in this case : automatically trigger a notice in Warehouse to prepare the delivery matching the sales order, or in the Purchase department to replenish the inventory, after completion of sales order.

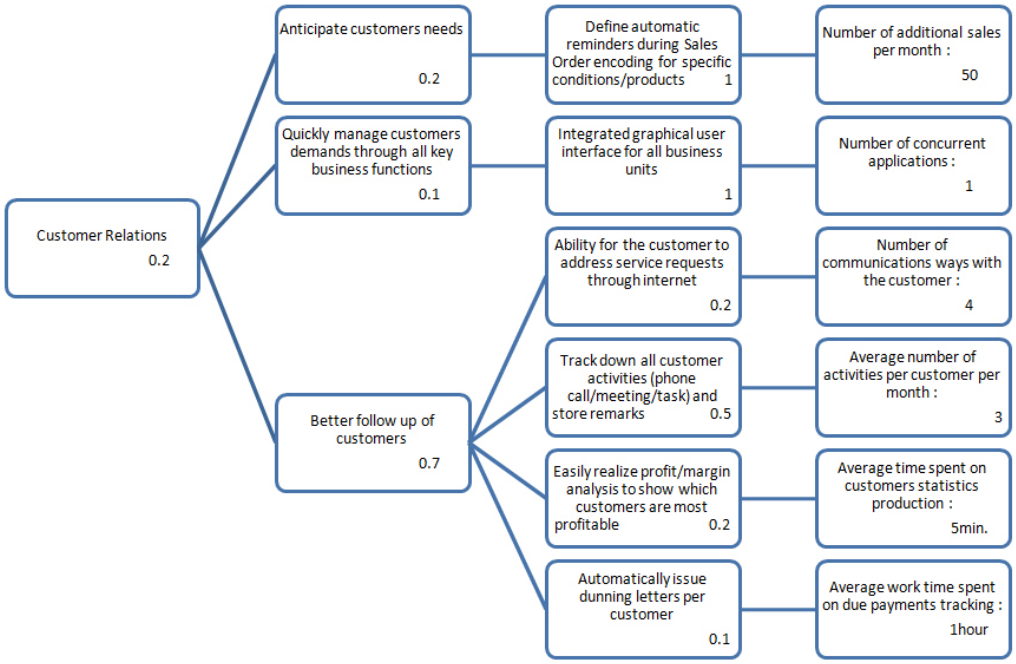


Figure 18: Goal hierarchy for the *Improve customer relations* strategic goal for enterprise A.