ÉCOLE DE TECHNOLOGIE SUPÉRIEURE
UNIVERSITÉ DU QUÉBEC

THESIS PRESENTED TO
ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

IN PARTIAL FULFILLEMENT OF THE REQUIREMENTS FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
Ph. D.

BY
CARLOS MONSALVE

REPRESENTATION OF BUSINESS PROCESSES AT MULTIPLE LEVELS OF ABSTRACTION (STRATEGIC, TACTICAL AND OPERATIONAL) DURING THE REQUIREMENTS ELICITATION STAGE OF A SOFTWARE PROJECT, AND THE MEASUREMENT OF THEIR FUNCTIONAL SIZE WITH ISO 19761

MONTREAL, NOVEMBER 5 2012

BOARD OF EXAMINERS

THIS THESIS HAS BEEN EVALUATED

BY THE FOLLOWING BOARD OF EXAMINERS

Mr. Alain April, Thesis Supervisor
Software Engineering and Information Technology Department at École de technologie supérieure

Mr. Alain Abran, Thesis Co-supervisor
Software Engineering and Information Technology Department at École de technologie supérieure

Mr. Kamal Al-Haddad, President of the Board of Examiners
Department of Electrical Engineering at École de technologie supérieure

Mr. Pierre Bourque, Member of the jury
Software Engineering and Information Technology Department at École de technologie supérieure

Mr. Évariste Valéry Bévo, External Evaluator
Centre de formation en technologies de l'information at Sherbrooke University

THIS THESIS WAS PRENSENTED AND DEFENDED

IN THE PRESENCE OF A BOARD OF EXAMINERS AND PUBLIC

ON OCTOBER 3, 2012

AT ÉCOLE DE TECHNOLOGIE SUPÉRIEURE

ACKNOWLEDGMENTS

First, I would like to express my utmost thanks to Dr. Alain April, my thesis supervisor, and Dr. Alain Abran, my thesis co-supervisor, for their motivation, guidance, time and support. Their advice continuously opened new opportunities to improve the outcomes of this research; without their patient support this research would have never been executed.

Thanks to the members of my board of examiners for their time and effort to review this thesis and to provide me with their feedback.

I am grateful to Mohamed Yassine Katiri, Tarik Ben Jillali, Linda Kang, Saadeddine Rachidi, Oscar Sánchez, and Yaneth Sabogal, students of the École de technologie supérieure, for their support in the execution of this research. Special thanks to all my colleagues, volunteers and experts who participated in the various research activities conducted as part of this research. Thanks to the following organizations and their personnel who contributed to the execution of this research: TMQ Group, Intégratik and Experts-Conseils CEP Inc.

My foremost gratitude to my employer: Escuela Superior Politécnica del Litoral (ESPOL) for its academic and financial support for this research.

Finally, I would like to thank my family and friends, especially Pía, my wife, who unconditionally accepted the sacrifice of changing our lives during these last four years to support me with her daily love and sympathy.

REPRESENTATION OF BUSINESS PROCESSES AT MULTIPLE LEVELS OF ABSTRACTION (STRATEGIC, TACTICAL AND OPERATIONAL) DURING THE REQUIREMENTS ELICITATION STAGE OF A SOFTWARE PROJECT, AND THE MEASUREMENT OF THEIR FUNCTIONAL SIZE WITH ISO 19761

Carlos MONSALVE

RÉSUMÉ

Cette thèse vise d'abord à apporter une aide et un soutien aux ingénieurs de logiciels et aux analystes d'affaires afin qu'ils puissent mieux modéliser les processus d'affaires lorsque ces modèles sont destinés à la spécification des exigences logicielles et assignées à la mesure de la taille fonctionnelle à la seule fin que ces personnes puissent estimer correctement tout projet. Quant à la thèse, elle-même, elle vise un but précis: contribuer à la représentation des processus d'affaires lorsqu'ils sont utilisés au moment de la phase d'«élicitation» des exigences logicielles.

Pour atteindre ce but, deux objectifs de recherche ont été clairement définis:
1. Proposer une nouvelle approche de modélisation qui génère des modèles de processus d'affaires qui doivent être utilisés dans une activité d'«élicitation» des exigences logicielles. Mentionnons que l'approche de modélisation ne devrait pas augmenter de manière significative la complexité des notations graphiques utilisées pour représenter les processus d'affaires, pour peu que cette approche doive permettre la participation active des différents acteurs impliqués dans un projet de logiciel typique pour représenter, de façon cohérente et structurée, leurs besoins et leurs contraintes.
2. Élaborer une «procédure» afin de pouvoir mesurer la taille fonctionnelle d'une application logicielle à partir des modèles de processus d'affaires. Cette «procédure» de mesure doit respecter la norme COSMIC ISO 19761; cette marche à suivre doit pouvoir être appliquée indépendamment de la notation graphique utilisée pour représenter les processus d'affaires.

Afin d'atteindre le premier objectif, cette thèse propose une nouvelle approche de modélisation (surnommée BPM^+) qui offre la possibilité de modéliser des processus d'affaires selon trois niveaux d'abstraction: 1) le niveau stratégique, 2) le niveau tactique et 3) le niveau opérationnel. À partir d'une revue de la littérature, une version *a priori* de BPM^+ a été conçue. Cette version *a priori* a été ensuite améliorée à la suite d'une étude de cas dans le milieu industriel. Cette dernière est devenue plus performante lorsque nous l'avons soumise aux analyses ontologiques pour l'ensemble des concepts des exigences logicielles et que des enquêtes scientifiques ont été élaborées auprès d'experts concernés. Finalement, une version révisée du BPM^+ a été proposée. Cette version révisée a été par la suite évaluée par une deuxième étude de cas. La version finale de BPM^+ a donc été fondée sur plusieurs confirmations et preuves obtenues à partir de diverses sources.

Quant au second objectif, la «procédure» de mesure a été élaborée à partir d'une comparaison analytique entre les spécifications de COSMIC et celles des notations graphiques sélectionnées pour cette recherche (i.e. BPMN et Qualigram). Cette comparaison a permis de définir un ensemble de lignes directrices de modélisation pour le type de logiciels d'affaires. La comparaison analytique a permis également de définir un ensemble de règles de correspondance entre les concepts des notations graphiques et les concepts de COSMIC. En outre, les lignes directrices de modélisation ont été adaptées pour le type de logiciels en temps réel. La «procédure» de mesure a été évaluée en comparant ses résultats à ceux qui ont été obtenus dans des études de cas de référence.

Les résultats obtenus par cette recherche démontrent ce qui suit:
1. BPM^+ permet de générer des modèles de processus d'affaires qui représentent, de façon cohérente et structurée, les besoins des différents acteurs impliqués;
2. La notation Qualigram est mieux adaptée à la conception de BPM^+. De surcroît, la notation Qualigram est plus facile d'utilisation pour les parties prenantes qui ne sont pas impliquées en informatique, tandis que BPMN est plus facile pour celles qui sont impliquées en informatique;
3. La «procédure» de mesure a été appliquée avec succès en utilisant deux différentes notations graphiques: Qualigram et BPMN. Celle-ci a également été mis en application avec succès à deux types différents de logiciels: le type de logiciels d'affaires et le type de logiciels en temps réel;
4. La précision de la «procédure» de mesure a été en conformité avec toutes les règles de la norme ISO /IEC 19761.

Mots-clés: modélisation des processus d'affaires, exigences fonctionnelles, l'analyse de représentation, Qualigram, BPMN, mesure de la taille fonctionnelle, COSMIC, ISO/IEC 19761, points de fonction.

REPRESENTATION OF BUSINESS PROCESSES AT MULTIPLE LEVELS OF ABSTRACTION (STRATEGIC, TACTICAL AND OPERATIONAL) DURING THE REQUIREMENTS ELICITATION STAGE OF A SOFTWARE PROJECT, AND THE MEASUREMENT OF THEIR FUNCTIONAL SIZE WITH ISO 19761

Carlos MONSALVE

ABSTRACT

This thesis aims at helping software engineers and business analysts to better model business processes when those models are meant to be used: for software requirements specification, and for functional size measurement purposes. The research goal of this thesis is to contribute to the representation of business processes for its use during the requirements elicitation stage of a software project.

To achieve this goal, two research objectives are clearly defined:

1. To propose a novel modeling approach that generates business process models intended to be used in a software requirements elicitation activity. The modeling approach should not significantly increase the complexity of the modeling notations used to represent the business processes; and it must allow the active participation of the various stakeholders involved in a typical software project in order to represent, in a consistent and structured way, their needs and constraints.
2. To develop a procedure to measure the functional size of a software application from the business process models representing it. This measurement procedure should be compatible with the COSMIC ISO 19761 standard; and it should be able to be used independently of the modeling notation used to represent the business process.

To achieve the first objective, this thesis proposes a novel modeling approach (coined BPM^+) that models business processes at three levels of abstraction: strategic, tactical and operational. An *a priori* version of BPM^+ was designed based on the findings of the literature review. This *a priori* version was iteratively refined through a pilot case study in industry, a series of ontological analyses, and a survey of experts. As a result, a reviewed version of BPM^+ was proposed. The reviewed version was evaluated through a second case study in industry. Therefore, the design of BPM^+ has been based on a triangulation of evidences obtained from various sources.

To achieve the second objective, the measurement procedure was developed from an analytical comparison between the specifications of COSMIC and those of the modeling notations selected for this research (i.e. BPMN and Qualigram). This analytical comparison helped to define a set of modeling guidelines for the business application software domain. The comparison also allowed defining a set of mapping rules between the modeling notations' constructs and the COSMIC concepts. In addition, the modeling guidelines were adapted for their application to the real-time software domain. The measurement procedure was evaluated by comparing its measurement results to those obtained in COSMIC reference case studies.

The research results demonstrate that:

1. BPM^+ allows generating business process models that represent in a consistent and structured way the needs of various stakeholders.
2. Qualigram notation is better suited to BPM^+'s design. In addition, Qualigram notation is preferred to be used for non-IT stakeholders, while BPMN is preferred for IT stakeholders.
3. The measurement procedure was successfully applied using two different notations: Qualigram and BPMN, and in two different software domains: the business application domain and the real-time domain.
4. The accuracy of the measurement procedure is in conformity with all the rules of the ISO 19761 standard.

Keywords: business process modeling, software requirements, representational analysis, Qualigram, BPMN, functional size measurement, COSMIC, ISO 19761, function points.

TABLE OF CONTENTS

Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF ABREVIATIONS

ACIS	International Association for Computer and Information Science
ARIS	Architecture of Integrated information Systems
BABOK	Business Analysis Body of Knowledge
BPD	Business Process Diagram
BPEL	Business Process Execution Language
BPM	Business Process Modeling
BPMI	Business Process Management Initiative
BPMN	Business Process Model and Notation
BPMO	Business Process Modeling Ontology
BPMS	Business Process Management System
BPQL	Business Process Query Language
BSC	Balanced Score Card
BWW	Bunge-Wand-Weber
CÉR	ÉTS Ethics Committee for Research
COSMIC	Common Software Measurement International Consortium
CP	Cumulative Percentage
CPM	Cyclical Process Model
E	Entry (data movement)
EPC	Event-driven Process Chain
ERP	Enterprise Resource Planning system
FP	Functional Process
FPA	Function Point Analysis

FSM	Functional Size Measurement
FT	Formal Tropos
FUR	Functional User Requirement
IADIS	International Association for Development of the Information Society
IDEF	Integrated Definition methods
IEC	International Electromechanical Commission
IEEE	Institute of Electrical and Electronics Engineers
IFPUG	International Function Point Users Group
IIBA	International Institute of Business Analysis
IJSEKE	International Journal of Software Engineering and Knowledge Engineering
ISO	International Organization for Standardization
IT	Information Technology
IWSM	International Workshop on Software Measurement
KA	Knowledge Area
KCPM	Klagenfurt Conceptual Pre-design Model
LoA	Logic of Actions
MLA	Multiple Levels of Abstraction
NDG	No Data Group
OMG	Object Management Group
PAIS	Process Aware Information System
R	Read (data movement)
RAD	Role Activity Diagram

SBPM	Semantic Business Process Management
SCOR	Supply Chain Operations Reference model
SERA	Software Engineering Research, Management and Applications
SRE	Software Requirements Elicitation
SRS	Software Requirements Specifications
SWEBOK	Software Engineering Body of Knowledge
UML	Unified Modeling Language
UML AD	UML Activity Diagram
W	Write (data movement)
WS-BPEL	Web Services – Business Process Execution Language
WFMS	Work-Flow Management System
WSEAS	World Scientific and Engineering Academy and Society
X	Exit (data movement)
XML	Extensible Markup Language
XPDL	XML Process Definition Language
YAWL	Yet Another Workflow Language

LIST OF SYMBOLS

χ	Chi-square statistic
CFP	COSMIC Function Point
Δ_i	Absolute difference between two quantities of data movements of type *i*
Δ_{TOTAL}	Total absolute difference between two measurements
Δ_{CFP}	Absolute difference between two COSMIC functional size measurements
F	Frequency
Freq.	Frequency
df	Number of degrees of freedom
N	Number of respondents
%	Percentage
p	Probability of obtaining a test statistic (*p*-value)

INTRODUCTION

Motivation and problem context

Business process management is a promising domain to bring business processes efficiencies into organizations; early publications (Elzinga *et al.*, 1995; Zairi, 1997; Zairi and Sinclair, 1995b) as well as recent publications (Dixon and Jones, 2011; Smith and Fingar, 2007; Spanyi, 2003) have recognized it. Many frameworks, methodologies, modeling notations and tools proposing systematic analysis, design, monitoring and improvement of business processes have arisen during the last decade. Not only there is a growing academic enthusiasm about these topics, but vendors and consultants are also proposing business process management solutions to address the opportunities of this market. Industrial studies show that most organizations see a high importance in adopting and using business process management approaches for their organizations: 93% according to Dwyer (2006, p. 6), 52% according to Harmon and Wolf (2010, p. 13), and over 59% according to Casewise Systems (Casewise, 2011, p. 1). In addition, most organizations are considering “doing more” in the near future in various types of activities related to business process management (Harmon and Wolf, 2010). Therefore, these industrial studies show increased adoption of some sort of business process management within organizations. Moreover, a recent Gartner study (McDonald and Aron, 2011) reports that business process improvement has been consistently identified as one of the top “business expectations of IT” over the past five years.

At the center of business process management are the business processes and their modeling. Business processes are often informal and part of an employees’ experience and competencies. It has been discovered, over the years, that business processes need to be represented formally (i.e. modeled) for many reasons. It may be required to document them, understand them, communicate them, automate them, or improve them (Curtis, Kellner and Over, 1992; Harmon and Wolf, 2011). Business process models are also used, by software engineers and business analysts, for eliciting the software and system requirements of information systems (Demirors, Gencel and Tarhan, 2003; Eriksson and Penker, 2000;

Georgakopoulos, Hornick and Sheth, 1995; Green and Rosemann, 2000; IIBA, 2009; List and Korherr, 2005; Mayr, Kop and Esberger, 2007; Mili *et al.*, 2009). A software requirements elicitation activity requires a good communication between software engineers and all the stakeholders (Abran *et al.*, 2004; Wand and Weber, 2002). For representing and communicating software requirements expressed by different groups of stakeholders, conceptual modeling is considered as a valid approach, and business process modeling (BPM) is one of the popular techniques for performing conceptual modeling (Davies *et al.*, 2006; List and Korherr, 2006). Therefore, in practice, BPM is also often used as part of the software requirements specifications (SRS) document.

A software development project is highly dependent on the quality of the software-requirements elicitation, analysis, specification, and validation activities (Abran *et al.*, 2004; Wand and Weber, 2002). If the SRS has a poor quality, then it is likely that the software development project will face difficulties. Therefore, it is necessary to successfully model the business processes if they are meant to be used as part of the SRS.

In addition, a SRS is typically used by software engineers as the source of information for measuring the functional size of the software to be developed. Functional size measurement (FSM) provides valuable information for estimating the effort required to develop the measured software. Based on that estimation, software managers can successfully plan resources and estimate costs for the software project (Abran, 2010). Since BPM can be used to elicit the software and system requirements, then a business process model may be a valuable source of information for FSM.

In this context, this thesis addresses two problems associated to the development of business process models for software requirements elicitation. The first problem is related to the necessity of generating business process models that contribute to the success of the software development project; and the second problem is related to the feasibility of using business process models for measuring the functional size of the software that supports (or might support) the business process modeled.

The need for generating high-quality business process models

A high-quality software requirements elicitation activity depends on a good communication between software engineers and end-users for an active participation of all the stakeholders; the result should be a high-quality SRS document (Abran *et al.*, 2004; Wand and Weber, 2002). Modeling business processes that can be successfully used and shared within an organization requires, among other things, the commitment of the top executives and the active participation of all the stakeholders in sharing a common vision of the business processes (Becker, Rosemann and von Uthmann, 2000; Sedera *et al.*, 2004).

Unfortunately, for many organizations business process management is a departmental initiative (Harmon and Wolf, 2010) and business processes may not be consistently documented: according to a recent industrial study (Harmon and Wolf, 2010, pp. 16-17), only 5% of the organizations always document their business processes in a consistent way, and only 3% of the time business process models are very consistent with the information systems designed to support them (p. 20). Moreover, only 13% of the times business process management is an organizational initiative led by top executives of the organization (pp. 31-32); while for 55% of the organizations it is a departmental initiative sometimes led by Information Technology (IT) stakeholders and, at other times, by management stakeholders.

A model corresponds to the point of view of the modeler, and different stakeholders require different perspectives of the business processes being modeled (Berger and Guillard, 2000; Curtis, Kellner and Over, 1992; Indulska, zur Muehlen and Recker, 2009; Lankhorst, 2005; Smith and Fingar, 2007; Van Nuffel and De Backer, 2012; Vara, Sánchez and Pastor, 2008; White, 2004; zur Muehlen and Ho, 2008): it is plausible, then, that IT and management may require different abstractions of the business processes to better represent their specific perspectives. For instance, management typically requires business processes represented at a high level of abstraction (Berger and Guillard, 2000; Van Nuffel and De Backer, 2012), while IT requires a more formal, rigorous, non-ambiguous and detailed description of the business processes because they intend to automate them (Abran *et al.*, 2004; Becker, Rosemann and von Uthmann, 2000; Lind and Seigerroth, 2010; Rosemann and Green, 2000).

Many authors report on the difficulty of choosing a single modeling notation to allow the effective communication and participation of all the stakeholders (Abran *et al.*, 2004; Curtis, Kellner and Over, 1992; Lankhorst, 2005; Lind and Seigerroth, 2010; Van Nuffel and De Backer, 2012). Evidence also shows that the different stakeholders tend to use different notations, conventions and techniques to represent their perspectives of business processes (Berger and Guillard, 2000; Curtis, Kellner and Over, 1992; Indulska, zur Muehlen and Recker, 2009; Lankhorst, 2005; Smith and Fingar, 2007; Vara, Sánchez and Pastor, 2008; zur Muehlen and Ho, 2008). These difficulties often create inefficiencies and duplications when each stakeholder uses his own notation, resulting in numerous communication problems, causing rework, project delays, costs overruns and failure.

Other authors have observed that current BPM notations are highly complex in their attempt to satisfy the different modeling perspectives required by different stakeholders (Indulska, zur Muehlen and Recker, 2009; Recker *et al.*, 2009). This growing complexity has been reported, and corroborated empirically by several authors as one of the key reasons why a modeling notation might not be able to produce effective models (Indulska, zur Muehlen and Recker, 2009; Mendling, Reijers and Cardoso, 2007; Wand and Weber, 2002), hindering the use of the notations and the possibility to reach a common understanding of the resulting models. Despite their growing complexity, BPM notations are still not able to satisfy all the modeling needs required by different stakeholders. As an example, the most popular current BPM notations lack the constructs to appropriately represent all the different requirements of an information system (Lapouchnian, Yu and Mylopoulos, 2007; List and Korherr, 2006; Pavlovski and Zou, 2008; Vara, Sánchez and Pastor, 2008).

Solutions to this problem (i.e. satisfying the various modeling needs required by different stakeholders) have to provide the means for a consistent way of modeling various business process perspectives. Ideally, the solution should be simple and should not significantly increase the complexity of the BPM notations, thereby allowing the business process models to be easily understood and used by different stakeholders.

Using business process models as a source for functional size measurement (FSM)

The use of conceptual models for functional size measurement (FSM) has been studied and analyzed in the research literature. The work of Marín, Giachetti, and Pastor (2008) offers a survey of eleven related works, including their own. In addition to the publications reported in that survey, other works have also studied the use of conceptual models for FSM (Daneva, 1999; Demirors and Gencel, 2004; Lavazza and Bianco, 2009; Sellami and Ben-Abdallah, 2009; van den Berg, Dekkers and Oudshoorn, 2005). Most of these previous works have been based on the use of Unified Modeling Language (UML) (OMG, 2010) diagrams (use case, component, and sequence diagrams), or the use of Event-driven Process Chain (EPC) diagrams (Scheer, Thomas and Adam, 2005) as a source of information for FSM. However, from all these works, only one work (Daneva, 1999) uses BPM for FSM. One of the conclusions of this latter work is that the "application of the counting model" from a business process model requires validation (p. 149). Therefore, there exists only scarce research on the feasibility of using business process models for FSM.

Purpose and research questions of this thesis

The purpose of this thesis is to contribute to the representation of business processes during the software requirements elicitation stage of a software project by proposing novel solutions to:

1. ensure business processes models that: a) take into consideration the needs and constraints from various stakeholders; b) represent, in a consistent way, these needs and constraints; c) allow easy communication of the software requirements to the various stakeholders; and d) can be shared among the various stakeholders;
2. measure the functional size of a software using its business process model representation made during the software requirements elicitation activities.

To achieve this research purpose, the following research question has been formulated: How can a business process be represented to better suit the needs and constraints of the various stakeholders involved in software requirements elicitation activities? This research question is subdivided into the following sub-questions:

1. What are the needs and constraints of the various stakeholders that should be represented by specific business process modeling constructs when conducting modeling during the software requirements elicitation activity?
2. What is the appropriate level of abstraction to represent all these modeling constructs in a business process model? If more than one level of abstraction is required, then what modeling constructs should be represented at each level of abstraction?
3. How well do current business process modeling notations represent these levels of abstraction and modeling constructs?
4. What would be a proposed BPM approach for consistently representing the various needs and constraints at their appropriate level of abstraction?
5. If a business process model represents software functional requirements, then can it be used for measuring the functional size of the software it represents? If so, is there some notation-specific business process modeling guidelines required to allow this measurement?
6. What would be the set of notation-independent business process modeling guidelines for measuring the software functional size?
7. What would be the procedure for measuring functional size using a business process model?

Research goal, objectives and scope

The research goal of this thesis is to contribute to the representation of business processes for its use during the software requirements elicitation stage of a software project. More specifically, this thesis aims at helping software engineers, business analysts, and BPM practitioners to better model business processes when those models are meant to be used: as part of a Software Requirement Specification (SRS) document; and for FSM purposes.

The two research objectives of this thesis are:

1. To propose a novel modeling approach (coined BPM^+) that will generate business process models intended to be used in a software requirements elicitation activity. A measure of the success of this proposal will be that it should not significantly increase the complexity of the BPM notations used to represent the business processes; and it must allow the active participation of the various stakeholders involved in a typical software project in order to represent, in a consistent and structured way, their needs and constraints. The resulting models should be easily understood and shared by the various stakeholders; easing the communication between the various stakeholders as they now can share a common set of models.
2. To develop a procedure to measure the functional size of a software application from its business process model representing its underlying functional requirements. This measurement procedure should be compatible with the COSMIC ISO 19761 FSM method; and it should be able to be used independently of the BPM notation used to represent the business process.

To achieve the first research objective the following specific research sub-objectives are defined:

- To identify the relevant Software Requirements Elicitation (SRE) concepts published in the Guide to the Software Engineering Body of Knowledge (SWEBOK) as well as in the Guide to the Business Analysis Body of Knowledge (BABOK) that should be considered when modeling a business process.
- To determine the appropriate levels of abstraction to represent the relevant SRE concepts in a business process model.
- To determine the modeling concepts that should be used at each level of abstraction.
- To assess how well current BPM notations represent these levels of abstraction.
- To identify the modeling constructs required to represent the relevant SRE concepts.

To achieve the second research objective the following research specific sub-objectives are defined:

- To develop a set of business process modeling guidelines to allow functional size measurement.
- To define how to identify the notion of data movements in a business process model.
- To evaluate the accuracy of this novel measurement procedure.

The scope of this research work is limited by three factors: 1) the type of BPM notations used to represent the business processes; 2) the perspectives to be modeled; and 3) the FSM method to be used for elaborating the procedure to measure the functional size of a software application from its business process models. The next paragraphs discuss each of these factors.

As a consequence of the growing popularity of business process management, a growing number of BPM languages and notations have been proposed to model business processes. Ko, Lee and Lee (2009) have proposed classifying BPM notations into one of the following four categories:

1. graphical notations (e.g. Business Process Model and Notation (BPMN));
2. execution notations (e.g. Business Process Execution Language (BPEL));
3. interchange notations (e.g. XML Process Definition Language (XPDL)); and
4. diagnosis notations (e.g. Business Process Query Language (BPQL)).

Of these four categories, this thesis focuses on the graphical notations category, because this is typically the category of BPM notations that will allow a software project stakeholder to represent and communicate his business processes in graphical form. However, the focus of this research is not on designing a new BPM notation but on developing a novel BPM approach that, based on selected BPM graphical notations, will allow to consistently represent the needs and constraints of the various stakeholders.

For the purpose of this thesis, a business process modeling perspective is given by the stakeholder who is willing to model (or who is the target of) the business process, and the purpose of modeling the business process (Rosemann and Green, 2000). The purpose of modeling depends on the task the stakeholder has to perform based on the business process model. The rationale behind this understanding of a business process modeling perspective is that the same stakeholder might present different needs according to the uses to be given to the business process models at a specific moment of time. In addition, for a given purpose of modeling there might be variations of the modeling needs according to the different types of stakeholders involved in the project (See Figure 0.1). The stakeholders considered for the scope of this thesis are the software engineers and the business analysts. The purpose of modeling is to use business process models for software requirements elicitation.

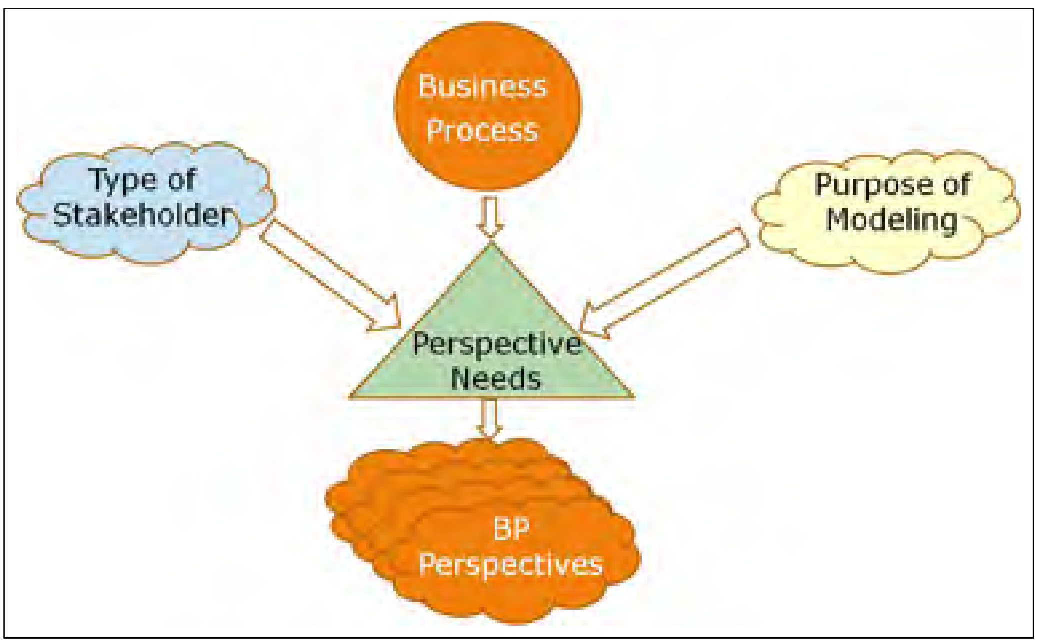

Figure 0.1 Business process perspectives

There are currently five FSM methods approved by the International Organization for Standardization (ISO/IEC, 2006; 2010). From these FSM methods approved by the ISO, this thesis uses the one proposed by the Common Software Measurement International Consortium (COSMIC): the COSMIC FSM method (COSMIC, 2009). COSMIC has been accepted since 2003 as international standard ISO/IEC 19761:2011 "Software engineering – COSMIC: A functional size measurement method" (ISO/IEC, 2011). COSMIC was designed to be applied in various functional domains: 1) business application software; 2) real-time software; and 3) a combination of the two. It is completely open and available in multiple languages (COSMIC, 2009). From the possible functional domains where COSMIC FSM method can be applied, this thesis covers the business application and the real-time software domains with an emphasis in the former domain.

Thesis organization

This thesis is structured in 4 chapters, the thesis' conclusions, one annex and 11 appendices. Following this introduction, Chapter 1 entitled LITERATURE REVIEW, presents a review of related work and establishes the theoretical framework for this research. The focus of the literature review is on the identification of:

1. what has already been published that attempts to solve the research sub-questions formulated in this thesis;
2. issues that have not been solved by the academia, the industry, or by other BPM research efforts; and
3. accepted academic techniques and approaches that could contribute to solving our research sub-questions.

Chapter 2 entitled RESEARCH METHODOLOGY, ACTIVITIES AND EXPECTED RESULTS, presents the research methodology used to address the research sub-questions. It first introduces a clear definition of the problems and the design of a proposal to address them; next, it follows with the description of the research plan, research activities and their execution; and it concludes with the interpretation step of the results obtained from the execution of the research activities. The research deliverables and outcomes are presented. Chapter 2 also describes each of the main research methods/techniques used during the execution of the research activities; the overall research design and the validity issues of each of the research methods/techniques are presented.

Chapter 3 entitled BUILDING THE BPM+ APPROACH, presents the development process of the novel BPM+ approach proposed as one of the two main contributions of this thesis. An *a priori* version of this modeling approach is drawn upon the results of the literature review (CHAPTER 1). This *a priori* version is iteratively reviewed and improved through a number of research methods/techniques (i.e. case study, representational analyses, and survey) until a final version is proposed. Therefore, BPM+ is designed based on a theoretical framework that has been evaluated trough triangulation of evidences (Dahlander, 2005; Miller, 2008; Paré, 2002; Runeson and Höster, 2009) obtained from various sources.

Chapter 4 entitled MEASURING FUNCTIONAL SIZE FROM BUSINESS PROCESS MODELS WITH COSMIC FSM METHOD, presents the development process of the procedure to measure the functional size of a software application from a business process model. The FSM procedure proposed is the second main contribution of this thesis. This procedure is applied using two modeling notations: Qualigram and BPMN. It is also applied both in a business application domain context and in a real-time domain context. Based on the results obtained, a set of notation-independent BPM guidelines for FSM is proposed. The measurement accuracy is evaluated next by comparing the results obtained to those obtained by reference case studies published in the COSMIC literature.

Finally, the conclusions chapter of the thesis summarizes the main contributions of this research, pointing out their originality. We revisit the research sub-questions that have been formulated, and how they have been addressed. We present also the expected impacts of this research work, and analyze its limitations. Finally, some recommendations for future research work are proposed. These recommendations aim to motivate the undertaking of new research or innovative applications that build on or develop the contributions to the knowledge generated in this research.

CHAPTER 1

LITERATURE REVIEW

In the literature it is possible to find multiple definitions or conceptions of the terminology related to this research (e.g. business process, business process model, etc.). Therefore, before proposing any new solution it is necessary to review and analyze the various definitions given to this terminology in the literature in order to determine the definitions that will be used as a basis in this thesis. The definition of business process is covered in section 1.1; then, in section 1.2, we define business process management. Finally, section 1.3 provides the definitions of business process modeling and business process model.

Over the past 20 years, a growing number of notations for modeling business processes have been proposed, illustrating the growing popularity of business process management. One of the critical factors for the success of a BPM project is the right selection of the modeling notation (Bandara and Rosemann, 2005). Therefore, it is necessary to assess the various BPM notations currently available. Subsections 1.3.2 and 1.3.3 review some of the BPM notations currently available.

The literature proposes various frameworks for assessing BPM notations (Aguilar-Saven, 2004; Daoudi and Nurcan, 2007; Giaglis, 2001; Hommes and Van Reijswoud, 2000; Kaschek *et al.*, 2007; List and Korherr, 2006; Luo and Tung, 1999; Nysetvold and Krogstie, 2005). The authors of each framework define the characteristics that they consider as critical for the assessment of a BPM notation. For example, Luo and Tung (1999) consider critical to assess the formality, scalability, enactment-ability, and ease of use of a BPM notation; while Aguilar-Saven (2004) considers critical the adaptability of a BPM notation. These frameworks are not contradictory but complementary. Also, it is possible to find in the literature various techniques and methods for assessing particular characteristics of BPM notations (Green and Rosemann, 2000; Indulska, zur Muehlen and Recker, 2009; List and Korherr, 2006; Mendling and Strembeck, 2008; Russell *et al.*, 2004; Sarshar and Loos, 2005;

van der Aalst, ter Hofstede and Dumas, 2005; zur Muehlen and Rosemann, 1998). This thesis: 1) focuses in the capability of a BPM notation to represent specific modeling needs and constraints, and 2) follows an assessment technique that is based in a well-established ontology (Rosemann *et al.*, 2009). Section 1.7 reviews this ontology-based assessment technique.

Modeling a business process is not a trivial task: a business process involves many types of elements (e.g. activities, roles, events, etc.). Each stakeholder presents specific needs for a type of element that should be modeled in a business process model. In addition, the purpose of modeling also dictates the types of elements that should be modeled (Luo and Tung, 1999; Rosemann and Green, 2000). A business process model aiming at simply documenting a business process might look different to a business process model that aims at the automation of the same business process. Therefore, the correct selection of the types of elements to be represented by a business process model depends on the needs of a specific stakeholder when performing a specific task. This thesis focuses on the needs of software engineers and business analysts when performing requirements elicitation. Subsection 1.6.1 present the references used in this thesis for identifying those needs; and subsection 1.6.2 present the various proposals found in the literature where BPM has been specifically used for requirements elicitation.

The difficulty on representing all the possible required types of elements using one single BPM notation has been reported in the literature (Dreiling *et al.*, 2008; Van Nuffel and De Backer, 2012) and multiple approaches have been proposed by various authors as solutions to this difficulty. Some of these approaches involve the representation of several BPM perspectives (see section 1.4) while others propose the use of multiple levels of abstraction (see section 1.5) to address this problem.

Our research does not only aim to model business processes in order to successfully document them as part of a SRS document, but also to use the models generated as a basis for measuring the functional size of the software they represent. Therefore, it is also necessary to review the functional size measurement (FSM) method used in this research for addressing our second research objective: the COSMIC FSM method (see subsection 1.8.1); and to review previous related work where conceptual modeling and business process models more specifically, have been used as a source for FSM (see subsection 1.8.2).

1.1 What is a business process?

Many definitions of business process can be found in the literature. Each definition varies depending on the viewpoint of the author and depending on the focus of the publication. Many authors (Curtis, Kellner and Over, 1992; Davenport, 1993; Dumas, van der Aalst and Ter Hofstede, 2005; Green and Rosemann, 2000; Zairi, 1997) make little distinction between the terms process and business process. This research has considered any definition that helps to identify the different types of elements that contribute to a business process.

Curtis, Kellner and Over (1992, p. 76) define a business process as: "one or more agents acting in defined roles to enact the [business] process steps that collectively accomplish the goals for which the [business] process was designed". This definition highlights the importance of actors and their roles. Medina-Mora *et al.* (1992) also place the actors at the center of the process. Hammer and Champy (1993, p. 35; quoted in Ko, 2009, p. 12; Lindsay, Downs and Lunn, 2003, p. 1017) provide a more comprehensive definition: "a collection of activities that takes one or more kinds of input and creates an output that is of value to the customer. A business process has a goal and is affected by events occurring in the external world or in other processes". One common viewpoint shared between Curtis' and Hammer's definitions is the importance placed on the business process need to reach a goal. However, Hammer's definition brings additional viewpoints into play: the activities, the internal and external events, the transformation of inputs into outputs, and the generated value. Davenport (1993) adds the notions of time, place and structure for the enactment of the activities:

> "A [business] process is simply a structured, measured set of activities designed to produce a specified output for a particular customer or market. It implies a strong emphasis on how work is done within an organization, in contrast to a product focus's emphasis on what. A

> [business] process is thus a specific ordering of work activities across time and place, with a beginning, an end, and clearly identified inputs and outputs" (Davenport, 1993, p. 5).

Another important contribution of Davenport's and Hammer's definitions is the target of the result of the business process: it is not just the execution of a group of activities, but aiming at a particular customer or market. Zairi (1997, p. 64) shares Hammer's viewpoint of considering a business process as a transformation of inputs into outputs: "an approach for converting inputs into outputs". Green and Rosemann (2000, p. 78) add the notion that there is a transformation of a business-relevant object in a business process: "the sequence of functions that are necessary to transform a business-relevant object". Gulledge and Sommer (2002) also contribute to the definition of a business process highlighting the notion that a business process crosses the functional boundaries of an organization. Spanyi (2003, p. 24) emphasizes the need to clearly understand the difference between a business process (using Davenport's definition) and an "Enterprise Business Process" which is "the end-to-end (cross-departmental, and often, cross-company) coordination of work activities that create and deliver ultimate value to customers". More recently, Sharp and McDermott (2009, p. 56) add additional types of elements to the definition: "a way for an enterprise to organize work and resources (people, equipment, information, and so forth) to accomplish its aims". This last definition highlights the use of two kinds of resources: tangibles (equipments), and intangibles (information).

In summary, from these many definitions we learn that a business process involves many types of elements. To be effective, a business process should aim at a goal; it typically includes a series of structured activities that transform inputs into outputs bringing some value to a customer or to the market. An activity might be triggered by an internal or external event, and it is executed by actors (employees) playing specific roles and using resources (tangibles and intangibles) of the organization. A business process often crosses the functional boundaries of a specific corporate function covering organizations end-to-end. As a result of the execution of an activity, business-relevant objects are transformed and value can be assessed. Figure 1.1 summarizes these findings.

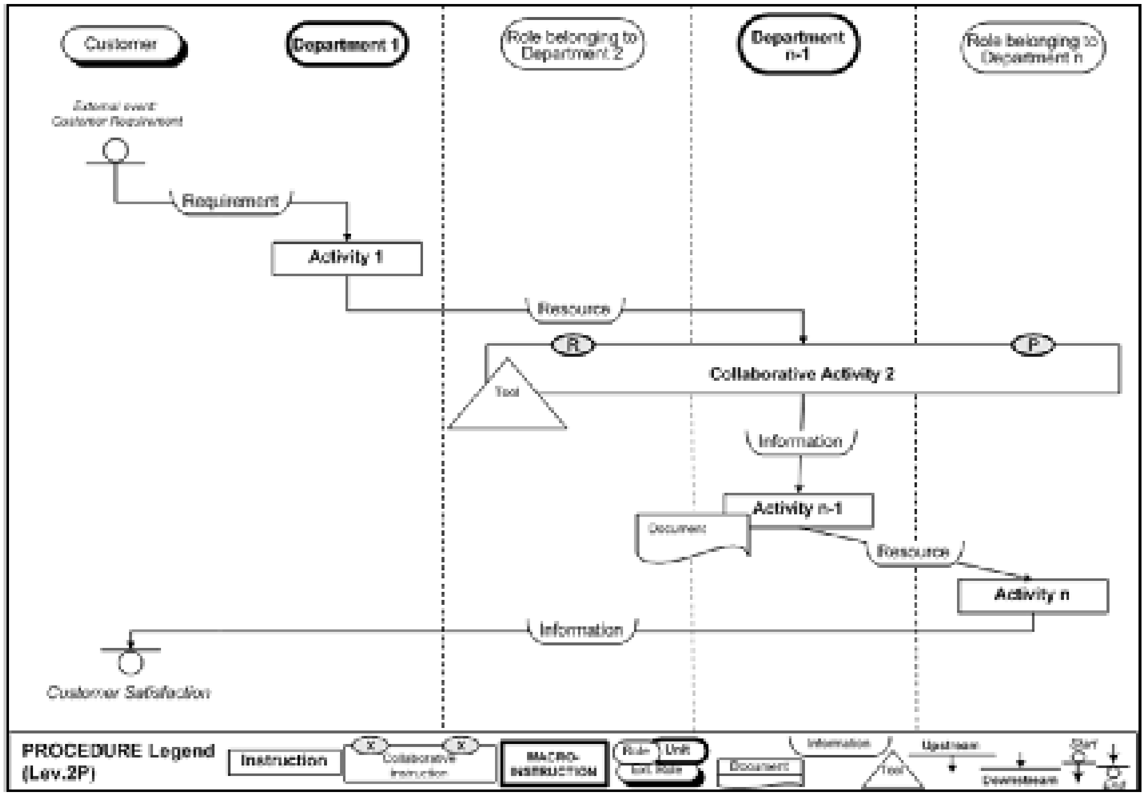

Figure 1.1 Description of a business process

With a clearer comprehension of a business process, we proceed in section 1.2 to present the concept of business process management and some examples of its support systems.

1.2 Business process management and its support systems

Organizations look for enhancing their efficiency and effectiveness to achieve their goals (ISO, 2008); for many of them this translates into achieving revenue improvements, which are directly tied to customer satisfaction and performance improvement. If each of the business processes of the organization is optimized, then the organization should be more efficient and effective to achieve its goals, to satisfy its customers, and finally to improve its revenues. This business process approach is typically referred in the literature by the term business process management. In this section the definition of business process management is first presented examining both its origin and its current perception, and next some examples of support systems and tools for business process management are presented.

1.2.1 What is business process management?

The origin of business process management is related to different management efforts that have been proposed to bring competitiveness to the organizations, either by improving the quality of their products and services (Elzinga *et al.*, 1995), or by improving the performance of their business processes (Zairi and Sinclair, 1995a). Most authors agree that the contributions of process innovation by Davenport (1993) and re-engineering by Hammer and Champy (1993) were the catalysts of the popularity growth of business process management. Business process management is neither a technology nor a type of information system: it is a management approach; therefore, this subsection will not consider those definitions that only present a clear IT-oriented point of view of business process management.

Elzinga *et al.* (1995, p. 119) provide a definition of business process management that highlights the importance of the quality of what is done by the organization: “a systematic, structured approach to analyze, improve, control, and manage [business] processes with the aim of improving quality of products and services”. Zairi’s definition shows more concern for the performance of how things are done within the organization: “[Business process management] is a structured approach to analyze and continually improve fundamental activities such as manufacturing, marketing, communications and other major elements of a company’s operation” (Zairi, 1997, p. 64).

Some recent definitions, for example, Recker *et al.* (2006, p. 2) provide a more general definition: “a structured, coherent and consistent way of understanding, documenting, modeling, analyzing, simulating, executing and continuously changing end-to-end business processes and all involved resources in light of their contribution to business performance”. Later, Smith and Fingar (2007) argue that business process management “is for business people" (p. 14), enabling them to "gain control of the design, implementation and optimization of their business processes".

For this research, the term business process management is a structured management approach aimed at optimizing end-to-end business processes to create value and to contribute to the organization's goals.

As stated at the beginning of this section, business process management is not a technology; however, information technology can support any business process management activity. The next subsection (1.2.2) analyzes some examples of support systems for business process management.

1.2.2 Business process management support systems

When re-engineering and process innovation came into scene, at the beginning of the 1990s, different authors, vendors and practitioners began to support those new management approaches with the use of workflow management systems (WFMS) that had been originally developed for office automation (Dumas, van der Aalst and Ter Hofstede, 2005; zur Muehlen, 2004). Shortly after, the terms business process management and business process management system (BPMS) began to be used, the former one sometimes mistakenly, to refer to either legacy systems or specifically developed systems that supported these new management approaches. As Smart, Maddern and Maull (2009) mention, the term business process management has been continuously used, even after process improvement and re-engineering no longer were considered as relevant concepts. At the beginning of the 2000s there has been an increase of the interest in these latter terms as a way to perform "business process automation". During this time frame, the expression 'process aware information system' (PAIS) appeared as an attempt to lump together every process-oriented information system. This section reviews these business process management support systems.

According to Georgakopoulos, Hornick and Sheth (1995, p. 119), a workflow describes the activities of a business process “at a conceptual level”. The workflow management technologies can support the reengineering of an existing business process. Reengineering a process implies the optimization of an existing business process; therefore, monitoring and controlling the business processes is implicit in the term. A WFMS aims at supporting: “1) business process modeling…; 2) business process reengineering…; and 3) workflow automation”.

Smith and Fingar (2007, p. 233) argue that a BPMS “enables companies to model, deploy and manage mission-critical business processes, that span multiple enterprise applications, corporate departments, and business partners”. After comparing this perception of a BPMS with the definition of business process management given by the same author (See subsection 1.2.1), it is possible to find a parallelism between: model and design; deploy and implement; and, manage and optimize. If the organization wants to optimize a business process it needs some sort of monitoring and control of it.

More recently, Dumas, Van der Aalst and Ter Hofstede (2005, p. 7) propose the acronym PAIS. It is defined as: “a software system that manages and executes operational processes involving people, applications, and/or information sources on the basis of process models”. Note that this definition uses the term ‘operational process' rather than business process. The former term is more general, and includes any kind of process that allows managing the resources and activities within an organization. Because of this more general definition, Dumas, Van der Aalst and Ter Hofstede (2005) consider as examples of PAIS different systems, such as: tracking systems, collaboration systems, and enterprise resource planning systems (ERP). This definition is complemented with a four-phase lifecycle named the PAIS lifecycle: 1) [business] process design; 2) [business] process implementation; 3) [business] process enactment; and 4) diagnosis (See Figure 1.2). In this definition, the diagnosis phase corresponds to the monitoring needed to improve a business process.

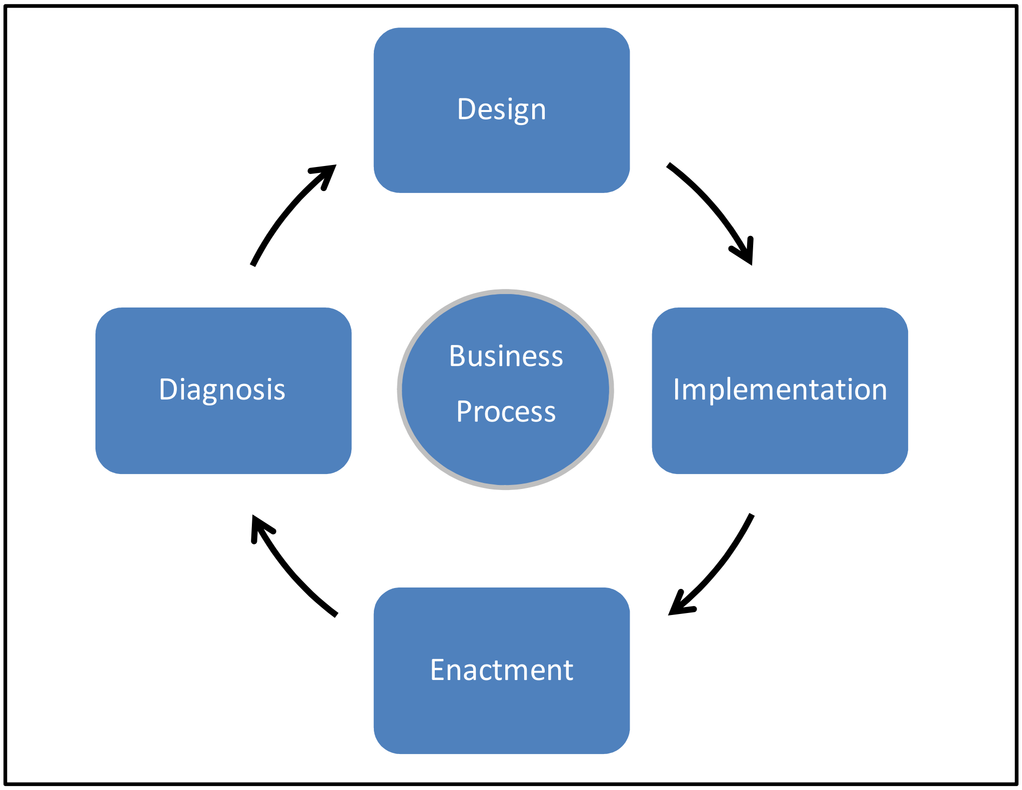

Figure 1.2 The PAIS life cycle
Adapted from Dumas, van der Aalst and Ter Hofstede (2005, p. 12), Copyright © 2005 by John Wiley & Sons, Inc., with permission from John Wiley and Sons

The three different business process management support systems proposed by the authors are compared in Table 1.1 from a lifecycle point of view. In this table the first lifecycle phase (modeling or design) allows for two different situations. The first situation is the modeling or design of a new business process: this usually is referred in the literature as the "as is" business process. The second situation is the re-modeling or redesign of a business process based on the results of a reengineering, optimization or diagnosis phase. This second situation is referred, in the literature, as the "to be" business process.

For some authors these three business process management support systems are identical. For example, zur Muehlen (2004) sees little difference between WFMS and BPMS. However, other authors highlight many differences among them. For example, Dumas, Van der Aalst and Ter Hofstede (2005) present a WFMS as a subset of a PAIS which does not include the diagnosis phase of the lifecycle.

Table 1.1 Comparison of business process management support systems

Authors	**Business process management support systems**	**Lifecycle phases**			
(Georgakopoulos, Hornick and Sheth, 1995)	Work Flow Management System	Modeling	Automating		Reengineering
(Smith and Fingar, 2007)	Business Process Management System	Model / Design	Deploy/Implementation		Manage/ Optimization
(Dumas, van der Aalst and Ter Hofstede, 2005)	Process Aware Information System	Design	Implementation	Enactment	Diagnosis

This thesis will avoid using the term WFMS as there is currently no consensus among the different authors. It will not use the term PAIS because of its generality. This thesis will use the term ‘business process management system’ and its acronym BPMS. It is possible to find languages, notations and tools that are designed for each of the four phases of the BPMS lifecycle (See Figure 1.2). From these four phases, this thesis focuses on the first phase (i.e. business process design or modeling); therefore the next section (section 1.3) will introduce some basic concepts of business process modeling (BPM) and then it will review some of the most popular languages and notations that are used for this first phase.

1.3 Business process model and modeling

Business processes are often informal and part of an employees’ experience and competencies. It has been discovered, over the years, that business processes need to be represented formally at each of the four phases of the BPMS lifecycle. To represent a business process, business process models are often preferred to textual descriptions. This research will mostly focus on the business process modeling needs during the design phase of the BPMS life cycle. In this section, the definition of a business process model is presented first, and then some of the most popular BPM notations are reviewed.

1.3.1 What is a business process model?

A model is an abstraction of the reality (Curtis, Kellner and Over, 1992): it represents only those details that the modeler considers as important for the domain he or she is working for. Thus, a business process model is defined as an abstraction of a real business process. Moreover, a business process model represents those types of elements, of the real business process, that the modeler believes are important for a specific perspective.

Therefore, business process modeling (BPM) is the act of producing abstract representations of actual business processes. In this thesis, BPM is considered directly related to the design phase of the BPMS life cycle. The design phase addresses high-level concerns of a business process. For example, stakeholders, at the design phase, will likely document an existing business process, design a new business process, or modify an existent business process. BPM might also be used during the design phase, by software engineers and business analysts, for gathering the software and system requirements of information systems (Albani and Dietz, 2006; Eriksson and Penker, 2000; Georgakopoulos, Hornick and Sheth, 1995; Green and Rosemann, 2000; List and Korherr, 2005; Mayr, Kop and Esberger, 2007; Mili *et al.*, 2009; Recker *et al.*, 2006). The clarification of the scope of BPM for this thesis is needed because at a lower level (implementation and enactment phases) modeling the business process might also be required. In many publications this lower level modeling is also referred as a business process modeling. For this thesis, we will refer as “business process execution model” any model used at the implementation or enactment phases.

One of the critical factors for the success of a BPM project is the right selection of the modeling notation (Bandara and Rosemann, 2005). The next subsections (1.3.2 and 1.3.3) review some of the most popular BPM notations.

1.3.2 Popular BPM notations

One of the key factors for successfully modeling business processes is the use of an appropriate BPM notation (Sedera *et al.*, 2004). IT and management typically use different notations, conventions and techniques to represent business processes (Curtis, Kellner and Over, 1992; Indulska, zur Muehlen and Recker, 2009; Lankhorst, 2005; Smith and Fingar, 2007; Vara, Sánchez and Pastor, 2008; zur Muehlen and Ho, 2008). Over the last 20 years, a plethora of notations for modeling business processes have been proposed and developed. Based on the lifecycle phases of a BPMS (See Figure 1.2), Ko, Lee and Lee (2009) have proposed classifying BPM notations into one of the following four categories: 1) graphical notations (e.g. BPMN); 2) execution notations (e.g. BPEL); 3) interchange notations (e.g. XPDL); and 4) diagnosis notations (e.g. BPQL). Of these four categories, this thesis focuses on the graphical notations category, because this is typically the category of BPM notations that have been designed for the first phase of the BPMS lifecycle (i.e. design phase), and they allow a stakeholder to represent and communicate the business processes in graphical form.

The notations most frequently used (Harmon and Wolf, 2011; Ko, Lee and Lee, 2009; Mili *et al.*, 2010) for modeling business processes at the design phase are: Business Process Model and Notation (BPMN), Event-driven Process Chains (EPC), Integrated DEFinition methods (IDEF), Petri Nets[1], Role Activity Diagrams (RAD), Unified Modeling Notation (UML)[2], and Yet Another Workflow Language (YAWL).

[1] Petri Nets is a formal mathematical theory. Nevertheless, it has been extensively used for BPM. This thesis follows the treatment given to Petri Nets by (Desel, 2005).

[2] Typically, from all the types of diagrams offered by UML, the Activity Diagrams (UML AD) are the most commonly used for modeling business processes.

Table 1.2 shows the number of citations for each of the aforementioned BPM notations obtained after querying the Compendex and Inspec databases for any publication indexed, during the past five years (i.e. between years 2007 and 2012), using the keywords "business process modeling" or "business process model" and their variants (e.g. "business process modeling) and their abbreviations (i.e. BPM), where the modeling notation appears in a journal, conference article or conference proceedings written in English.

Table 1.2 Compendex and Inspec citations of most common BPM notations[3]

Modeling notation	Compendex	Inspec
BPMN	288	207
EPC	48	27
IDEF	1	1
Petri Nets	117	69
RAD	8	7
UML	110	70
YAWL	17	13

The next subsections (1.3.2.1 to 1.3.2.4) review the four most commonly cited notations in the recent BPM literature (i.e. BPMN, EPC, Petri Nets and UML).

1.3.2.1 Business Process Model and Notation (BPMN)

BPMN is currently a standard of the Object Management Group (OMG, 2011). It was initially developed by the Business Process Management Initiative (BPMI), and in 2004 the specification of BPMN 1.0 was released (White, 2004). Later on, in 2005, there was a merger of BPMI and OMG, BPMN being adopted by the latter.

[3] Retrieved February 6, 2012 from: http://www.engineeringvillage2.org/controller/servlet/Controller?CID=expertSearch&database=3

BPMN was designed with the aim of providing a unified notation to be used by both IT and management stakeholders, with the characteristics of being easy to understand, but at the same time having a formal basis (Ami and Sommer, 2007; Recker *et al.*, 2006; Recker, 2008; Silver, 2009; Smith and Fingar, 2007; White, 2004). To achieve this goal, the version 1.2 of the standard (OMG, 2009a) includes a basic set of constructs named “Business Process Diagram (BPD) Core Element Set” (i.e. Core Set)[4], and a more complete set, “BPD Extended Set” (i.e. Extended Set)[5]. The first set is intended for documentation and communication purposes, and the second one for developing more detailed models, appropriated for BPMS implementations (OMG, 2009a; White, 2004). Harmon and Wolf (2010) and Recker *et al.* (2006) show evidences that BPMN is being used by more and more organizations, showing a fast rate of increase in its use.

BPMN is a modeling notation rich in modeling constructs for representing various types of control flow and events. Rosemann *et al*. (2009) have presented a study of how the various BPM notations have evolved to become more complete over time. Their results show that BPMN is the most complete of all the BPM graphical notations studied (Rosemann *et al.*, 2009).

BPMN has a high degree of expressiveness, but at the same time is highly complex (Recker *et al.*, 2009). According to (Muehlen and Recker, 2008), of all the modeling constructs offered by BPMN, a typical business process model created in industry uses only nine. The selection of the nine constructs varies from one BPM initiative to another. However, only four modeling constructs were always observed in all the business process models studied, and some of the BPMN modeling constructs have never been used in practice.

---

[4] BPMN 2.0 defines a set of “basic BPMN modeling elements” (OMG, 2011, pp. 28-30) that is very similar to the BPD core element set; being the main difference the inclusion of a *message* modeling element which does not impact the results presented in this thesis.

[5] BPMN 2.0 defines a set of “extended BPMN modeling elements” (OMG, 2011, pp. 30-41) similar to the BPD extended set. The main differences are related to the capability of BPMN 2.0 to represent choreography diagrams that are out of the scope of this thesis. Therefore, the changes introduced in BPMN 2.0 do not impact the results of this thesis.

BPMN is one of the BPM notations selected for the execution of this thesis because: 1) its popularity is growing; 2) it is considered as a standard by the OMG; and 3) it has a high degree of completeness. The current version is BPMN 2.0 (OMG, 2011); however, this thesis uses BPMN version 1.2 (OMG, 2009a) because version 2.0 was still considered a Beta 2 version at the time of executing this research. When the term BPMN is used in this thesis without any reference to either version, we are referring to the version 1.2. The implications of not using BPMN version 2.0 are briefly discussed along the various sections of this thesis when this is considered necessary (see subsections 1.5.2.2, 2.3, 3.2.2.2, 3.5.1, 3.6.3, and 4.2.2)[6]. In general terms, the BPMN version does not affect neither the meaning of the thesis-author's findings nor the results of this thesis.

1.3.2.2 Event-driven process chains (EPC)

Event-driven Process Chains (EPC) is a BPM notation used by ARIS (ARchitecture of Integrated information Systems), a popular architecture that aims at providing an integral solution for implementing information systems from the business processes of the organization. It has been developed by IDS Scheer (Scheer, 2012) based on the results of some research of using stochastic networks and Petri Nets for BPM (Scheer, Thomas and Adam, 2005). Since its origin, EPC has been associated to SAP (SAP, 2008). Currently, ARIS also supports BPMN and UML as BPM notations (Scheer, 2009). EPC has been frequently used for BPM, especially in Europe (Harmon and Wolf, 2011; Recker, 2006). Its popularity might be related to its association to SAP, to its apparent simplicity while being based on complex modeling theory (Scheer, Thomas and Adam, 2005), or to its capability to model from different perspectives in an integrated way (Russell *et al.*, 2006b). Lankhorst (2005, p. 37) has reported that EPC has "quite a learning curve".

---

[6] The BPMN version 2.0 specifications (OMG, 2011) defines the notation and semantics of four types of modeling conformances: 1) Process modeling conformance that aims at representing "a standard process or an orchestration process" (p. 315); 2) Process execution conformance that aims at defining the "execution semantics for orchestrations" (p. 425); 3) BPEL process execution conformance that aims at defining the "mapping of a BPMN model to WS-BPEL" (p. 445) ; and 4) Choreography modeling conformance that aims at representing "the way participants coordinate their interactions" (p. 315). From these four types of modeling conformances only the "process modeling conformance" is within the scope of this thesis.

ARIS defines five enterprise perspectives, each of them represented at three description levels: requirements definition, design specification, and implementation description. From a business process management point of view, ARIS works at three levels: 1) strategy level, 2) design, control and optimization level, and 3) execution level (Davis, 2008; Scheer, Thomas and Adam, 2005). In addition, ARIS suggests a BPM hierarchy composed of three abstraction levels: high level process, functions, and tasks (Davis, 2008).

1.3.2.3 Petri Nets

Petri Nets is a formal specification and description technique based on a mathematical theory. It has its roots in systems engineering for modeling discrete dynamic systems. Nevertheless, due to its simplicity and formal background, it has been used to model systems where a control flow is important (Desel, 2005; Giaglis, 2001). Regarding BPM, Petri Nets has been used to directly model business processes as well as to provide the theoretical background for the development of BPM notations.

Desel (2005, pp. 148-154) explains the different uses that academia, modelers, industry and vendors can give to Petri Nets. Desel mentions that, depending on the use, Petri Nets can be categorized "as a visual language, as a mathematical theory, and as a formal language". According to this categorization, if Petri Nets is used for BPM, then it has to be treated as a "visual language". If Petri Nets is used as the theoretical background of a modeling notation, then Petri Nets can be considered as a "mathematical theory". Finally, if Petri Nets is needed for "simulation and analysis techniques", it has to be treated as a "formal language". Therefore, in this thesis we will consider Petri Nets as a visual or graphical BPM notation.

Aguilar-Saven (2004) and Giaglis (2001) mention that the weaknesses of Petri Nets are its lack of data concepts and its lack of establishing a business process hierarchy. Therefore, it was unavoidable that business process "models often became excessively large" (Aguilar-Saven, 2004, p. 138).

There are different variations of the original Petri Nets; some of them have been used for BPM, as for example Colored Petri Nets and Stochastic Petri Nets (Aguilar-Saven, 2004; Desel, 2005; Giaglis, 2001). Some authors point out that these variations introduce modeling improvements without compromising the qualities of the original Petri Nets (Aguilar-Saven, 2004; Giaglis, 2001).

1.3.2.4 Unified Modeling Language (UML)

UML is actually a family of thirteen object-oriented modeling diagrams that can be grouped as: structure, behavior, and interaction diagrams. Probably the most commonly used for BPM is the Activity Diagram (UML AD), one of the three available behavior diagrams. Even after BPMN being adopted by the OMG as a standard, this organization still mentions that UML might be used for BPM (OMG, 2009b).

One of the characteristics of UML is that it provides three mechanisms to allow its extension: 1) stereotypes that permit to create new kinds of constructs; 2) tagged values that allow including modeling notes; and 3) constraints that permit to refine the semantics of UML. Several authors have used these mechanisms to create extensions to UML in order to model the various perspectives of BPM demanded by different stakeholders (Engels *et al.*, 2005; Eriksson and Penker, 2000; Lankhorst, 2005; OMG, 2009b). In subsection 1.6.2.2 some of these extensions are reviewed.

In summary, all the reviewed BPM notations have some weaknesses reported. The approaches to support various modeling perspectives vary for each modeling notation. BPMN proposes the use of two sets of modeling constructs, one oriented to stakeholders who require a high-level view of the business process, and another one oriented to those stakeholders who require details of the business process workflow. EPC as part of the ARIS architecture offers a set of different and consistent integrated perspectives; however, it has been noted that recently ARIS is also supporting BPMN and UML. UML provides a set of mechanisms to formally extend the notation. Therefore, UML eases the elaboration of new modeling constructs to support the different perspectives. Petri Nets is a formal notation, and it presents some difficulties representing other than the workflow perspective of a business process. From these four BPM notations, BPMN has been selected as one of the BPMN notations for the executions of this research because: 1) its growing popularity; 2) it is considered as a standard by the OMG; and 3) it has a high degree of completeness.

Evidence shows that some of the characteristics that organizations often look for in a BPM notation are simplicity and ease of use (Eikebrokk *et al.*, 2011; Mendling, Reijers and Cardoso, 2007; Recker, 2010); however, BPMN is considered as a complex BPM notation. To take into account and address this concern, the management-oriented notation Qualigram (Berger and Guillard, 2000) has also been selected for this thesis. Besides being management-oriented, Qualigram notation has been selected for the execution of this research, because: 1) its modeling tool is based on Microsoft's Visio, which at the moment of executing the research was the most popular in the industry (Ami and Sommer, 2007; Harmon and Wolf, 2010); 2) its structure, in terms of levels of abstraction, has shown the potential to be well accepted by an organization's various stakeholders (Berger and Guillard, 2000); 3) it is based on the ISO 9000 (2008; 2010) quality management family of standards (Berger and Guillard, 2000); and 4) it is simple. The next subsection (subsection 1.3.3) reviews Qualigram notation in more detail.

1.3.3 Qualigram notation

Qualigram is a management-oriented BPM notation intended for the documentation and communication of business processes. It is based on the results of an international research project (Antonellis and Zonta, 1990; Dumas and Charbonnel, 1990). Qualigram's modeling tool, developed by Globalliance, is currently based on Microsoft's Visio.

Qualigram proposes three levels of abstraction for representing business processes. The top level of abstraction (i.e. the process description level) models the core business processes and their main objectives at a high level, aiming to represent *why* the organization needs to perform the business processes modeled, and *where to* go from the organization's strategic point of view. In short, this level of abstraction deals with the mission, objectives and policies of the organization. The intermediate level of abstraction (i.e. the procedure description level) models procedures and aims to represent *who* is responsible for what activity in the organization, and *what* is accomplished, describing how to achieve the objectives of the organization. Finally, the lowest level of abstraction (i.e. the instruction representation level) models the work instructions, aiming to represent *how* somebody in the organization performs a specific activity and *what* that person uses to do so. This level also deals with the control of some specific tasks. In summary, "a process is constituted by a set of procedures; a procedure is constituted by a set of work instructions; and an instruction is constituted by a set of elementary operations" (Berger and Guillard, 2000, p. 40). These concepts are depicted in Figure 1.3.

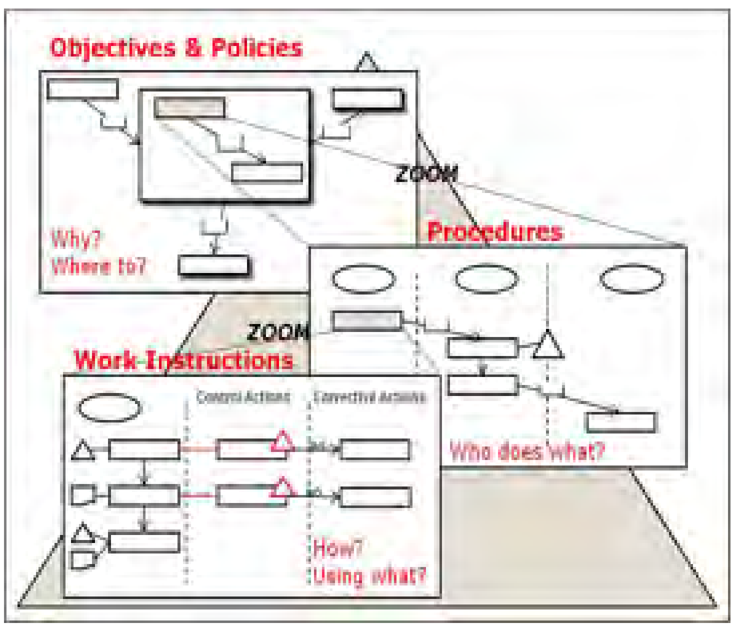

Figure 1.3 Qualigram pyramid
Adapted from Berger and Guillard (2000, p. 41), Copyright © 2000 by AFNOR, with permission from AFNOR

It is important to point out that the Qualigram concept of a process model (i.e. top-level model) is somewhat different from the mainstream concept of a business process model. Actually, a Qualigram procedure model (i.e. intermediate-level model) is closer to what is typically understood as a business process model. In this thesis, the term "business process model" is generic, and encompasses the variations and levels of detail that each BPM notation or author may prefer to use to represent an organization's business process. Therefore, within the scope of this thesis, both Qualigram process models and Qualigram procedure models are considered as business process models but with different levels of abstraction.

Qualigram was designed to satisfy the requirements of the ISO 9000 family of standards of the International Organization for Standardization for describing business processes (ISO, 2010). Another characteristic of Qualigram is its simplicity. The modeling constructs for each level of abstraction are based on a set of four basic concepts, along with their corresponding graphical forms: 1) action; 2) entity; 3) tool; and 4) information (See Figure 1.4). Variations of the *action* form are used to represent processes, procedures, work instructions, and elementary operations. Variations of the *entity* form are used to represent roles (internal and external), departmental units, and external entities. The *tool* form is used to represent any kind of physical tool or equipment, as well as any kind of document produced or used by an action. The *information* form is used to represent the input and output flows of information between the various types of elements modeled. Qualigram's simplicity makes it clear enough to be understood by any stakeholder in the organization (Berger and Guillard, 2000).

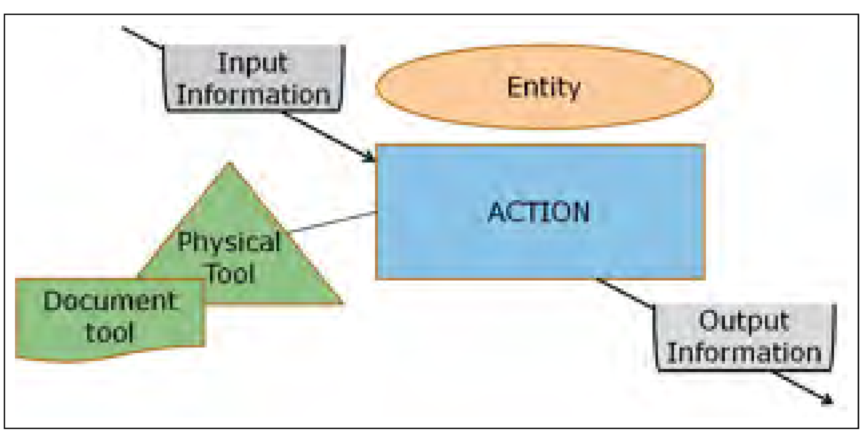

Figure 1.4 Graphical forms of the Qualigram notation
Adapted from Berger and Guillard (2000, p. 34), Copyright © 2000 by AFNOR, with permission from AFNOR

To summarize, the Qualigram notation has been selected for this thesis because: 1) it is management oriented; 2) its tool is based on Microsoft's Visio, which is very popular in industry; 3) its simplicity; 4) its ISO 9000 (i.e. quality management) based approach; and 5) the presence of a hierarchy of abstraction levels.

We can conclude so far that modeling a business process is not a trivial task. A business process involves many types of elements (e.g. activities, roles, events, etc.). Each stakeholder presents particular needs of the types of elements that should be modeled in a business process model. Many BPM notations have been proposed in the literature as an attempt to provide a response to all these modeling needs. Some of these modeling notations are more IT-oriented; others are more management-oriented. Despite of these efforts, the literature recognizes the difficulty to represent all the process' elements into only one business process model. The next section (section 1.4) presents what types of elements of a business process should be represented in a business process model.

1.4 What elements of a business process to represent?

Section 1.1 presented a description of business process that includes: actors, roles, activities, events, inputs, outputs, resources, objects to be transformed, customers, and goals to be achieved (Figure 1.1). From the definition of business process model (refer to subsection 1.3.1) it is understood that different authors have different criteria for selecting the elements when modeling a business process.

Medina-Mora *et al.* (1992) propose that the most important elements of a business process are the actors playing roles and having interactions. For Curtis, Kellner and Over (1992) any coordination activity should be represented, even if it is manually performed. Eriksson and Penker (2000) place more importance on an holistic representation of a business process, including its goals and how the activities help to reach the goals. Berger and Guillard (2000) stipulates that a "good" business process model should include the logical sequence of the actions, the roles of individuals responsible of the execution of the process, and the flow of information between these different roles. White (2004, p. 1) proposes that the important elements are the activities “and the flow control that define their order of performance”. Finally, Wohed *et al.* (2006) highlight the importance of representing the resources associated to the business process to be modeled.

This literature review confirms the presence of a number of important elements to be represented in a business process model. Each author places the emphasis on different types of elements of the business process, which leads to the need of various perspectives of a given business process. In the next subsections we further review the need of various BPM perspectives in a typical BPM project (subsection 1.4.1), and then we identify the most common BPM perspectives found in the BPM literature (subsection 1.4.2).

1.4.1 The need of BPM perspectives in a BPM project

From the criteria of the authors presented in subsection 1.4, we conclude that a business process model depends on the modeler's point of view. Thus various BPM perspectives can be created to represent each of these unique points of views. In this subsection we identify some findings from the BPM literature that support the need of modeling various perspectives in a typical BPM project.

Many authors (Becker, Rosemann and von Uthmann, 2000; Georgakopoulos, Hornick and Sheth, 1995; Kettinger, Teng and Guha, 1997; Smith and Fingar, 2007; Spanyi, 2003) report that BPM projects involve many stakeholders. Business process models should allow each of them to clearly understand the business processes, sharing a common vision of them, to ease the communication among them. Generally, two groups of stakeholders are often present in these projects: management and IT. Each group has different needs. Typically, management stakeholders are more interested on the design and diagnosis phases of the BPM project lifecycle. On the other hand, IT stakeholders are usually more interested on the implementation and enactment phases of the BPM project lifecycle (See Figure 1.5).

Moreover, each group of stakeholders involves their employees in these projects. The employees of each group typically share many of the needs; however, each employee presents his own particular needs, creating a number of perspectives when representing the business processes. Thus, the perspectives of a group of stakeholders might present similarities between them; nevertheless, every perspective might also present proper characteristics that differentiate it from the others.

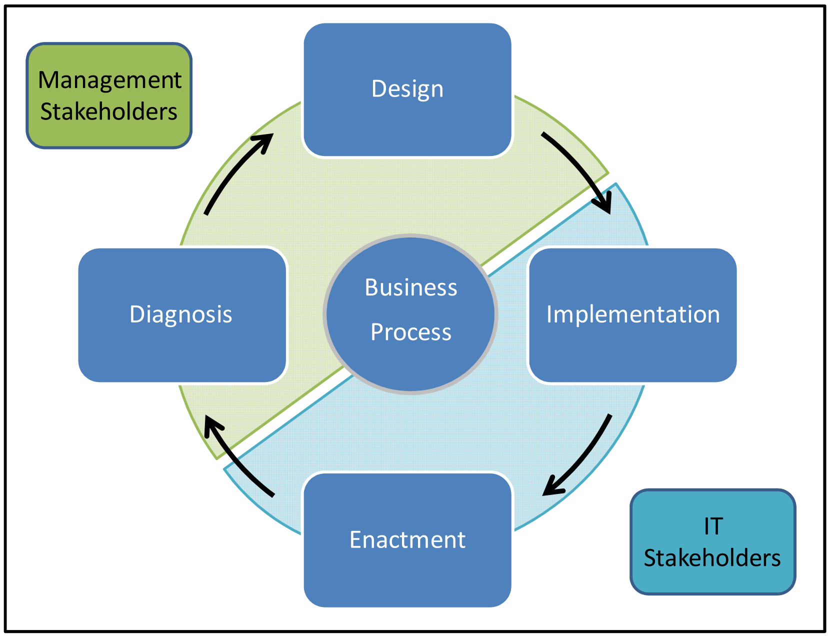

Figure 1.5 BPM lifecycle and organization's stakeholders

Many authors agree that each stakeholder requires different information from the same business process (Aguilar-Saven, 2004; Becker, Rosemann and von Uthmann, 2000; Curtis, Kellner and Over, 1992; Georgakopoulos, Hornick and Sheth, 1995; Green and Rosemann, 2000; Krogstie, 2003; Nysetvold and Krogstie, 2005; zur Muehlen, 2004). Therefore, a typical BPM project will have to deal with different perspectives of the same business process. These perspectives should not be exclusive: they should complement each other, and all of them together should contain most of the valuable elements and information from the business process. Each of the perspectives is a layer of the same business process. Each stakeholder can find helpful to use one or more of these layers at a given moment in time (refer to Figure 1.6). In the next subsection (1.4.2) we present the most common perspectives proposed by the BPM literature.

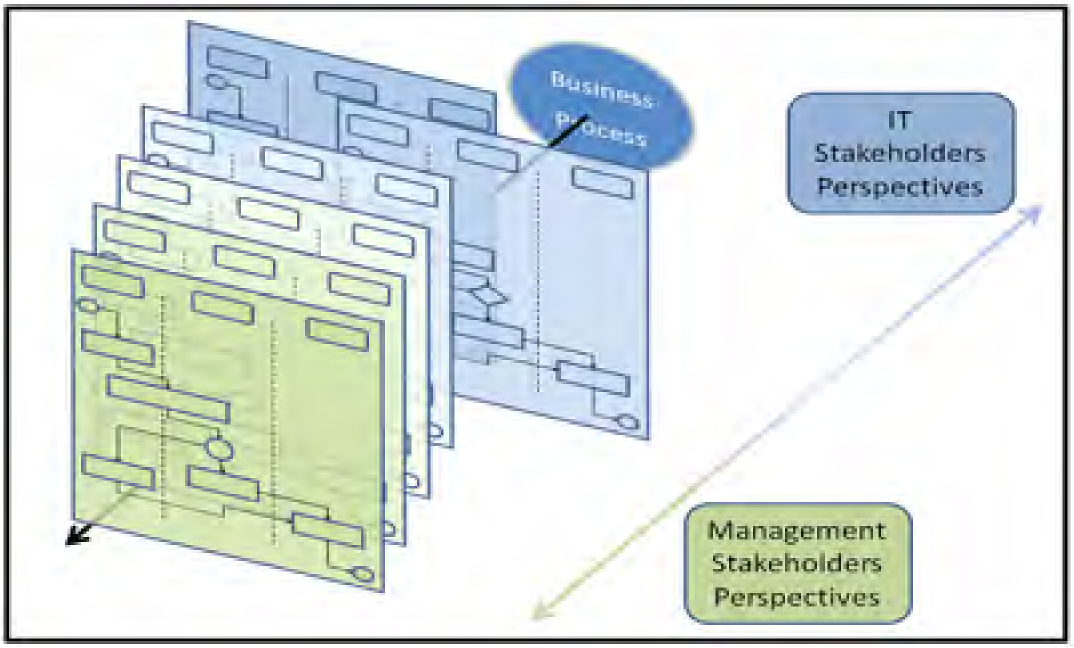

Figure 1.6 Perspectives as layers of a business process

1.4.2 Business process perspectives presented in the literature

In subsection 1.4.1 it was concluded that a BPM project should include different perspectives of a given business process to satisfy the various modeling needs of IT and management. This subsection identifies the most common BPM perspectives proposed by the literature. Table 1.3 summarizes the findings of this specific literature review. It is difficult to categorize all the BPM perspectives published by each author. We have found that perspectives may have the same (or a similar) name but do not necessary have the same meaning; however, the classification of perspectives presented in Table 1.3 is helpful in creating some categories.

Curtis, Kellner and Over (1992) stipulate that there are four different perspectives in a business process: 1) functional: that represents the *what*; 2) behavioral: that models the *when*

Table 1.3 Perspectives of a business process model

Authors	**Perspectives**					
(Curtis, Kellner and Over, 1992)	Functional	Behavioral	Informational	Organizational		
(Eriksson and Penker, 2000)	Process + Behavioral			Structural	Vision	
(van der Aalst, ter Hofstede and Dumas, 2005)	Operation	Process	Data/Information	Organization		Integration
(Engels *et al.*, 2005)	Actions & Control Flow + Interaction		Data & Object Flow + System Specific	Organizational Structure		
(Scheer, Thomas and Adam, 2005)	Output + Function	Control	Information	Organization		Control
(List and Korherr, 2006)	Functional	Behavioral	Informational	Organizational	Business Process Context	

and the *how*; 3) organizational: that abstracts the *where* and the *whom*; and 4) informational: that models the *information* entities. Eriksson and Penker (2000) present four different perspectives: 1) vision (vision, goals and problems); 2) process (activities, resources, value added); 3) structural (organizational structure, products and services structure); and 4) behavioral (states, transitions). Van der Aalst, ter Hofstede and Dumas (2005) propose five perspectives: 1) process (control flow); 2) organization (structure of resources); 3) data/information; 4) operation (atomic elements); and 5) integration. Engels *et al.* (2005) describe also five perspectives but they differ: 1) actions and control flow; 2) data and object flow; 3) organizational structure; 4) interaction centric; and 5) system specific. Scheer, Thomas and Adam (2005) also propose five business process perspectives to be represented: 1) organization (organizational units, their interactions and resources); 2) function (activities and their interactions); 3) information (information services, information objects, and data flow); 4) output (materials, products and services); and 5) control[7] (integration of all the perspectives, control flow). Finally, List and Korherr (2006) add to Curtis proposition a "business process context" perspective.

Each perspective responds to the needs of a stakeholder who wants to use a business process model for a specific purpose. Therefore, it is not possible to select a group of perspectives that will work for every BPM project and, even worse, to expect to model all the stakeholders' needs into only one business process model.

Moreover, many authors report on the difficulty of choosing a single BPM notation to represent all the required perspectives to allow an effective communication and participation of all the stakeholders during a BPM project (Curtis, Kellner and Over, 1992; Lankhorst, 2005; Lind and Seigerroth, 2010; Van Nuffel and De Backer, 2012). It has been also reported that to address this difficulty, particular conventions and techniques have been designed, or multiple BPM notations are used in a BPM project (Curtis, Kellner and Over, 1992; Indulska, zur Muehlen and Recker, 2009; Lankhorst, 2005; Smith and Fingar, 2007; Vara, Sánchez and Pastor, 2008; White, 2004; zur Muehlen and Ho, 2008). However, having many notations can cause miscommunications, rework and can also be a cause of software project delays, costs overruns and failure.

Other authors refer to current BPM notations as highly complex, in an attempt to satisfy the different modeling perspectives required by different stakeholders (Indulska, zur Muehlen and Recker, 2009; Recker *et al.*, 2009). This complexity has been reported as one of the key reasons why a modeling notation might not produce effective models (Wand and Weber, 2002); this presumption has been corroborated empirically (Indulska, zur Muehlen and Recker, 2009; Mendling, Reijers and Cardoso, 2007).

[7] Scheer, Thomas and Adam (2005) control's perspective presents characteristics of the behavioural perspective described by Curtis, Kellner and Over (1992), but also of the integration perspective described by van der Aalst, ter Hofstede and Dumas (2005).

In summary, a typical BPM project requires the participation of IT-oriented and management-oriented stakeholders. These two groups might have conflicting requirements from the business process models; nevertheless, the active participation of both groups is critical for the project. It is very complex to model all the perspectives into one single business process model, and it has been proven to be difficult to select a BPM notation capable of representing all these perspectives. It has been observed that difficulties to facilitate the active participation of all stakeholders create inefficiencies and duplications, resulting in numerous communication problems, causing rework, software project delays, costs overruns, and projects failure.

To address these issues many authors have argued that BPM at multiple levels of abstraction (MLA) helps to represent the information to be provided to various types of stakeholders (Berger and Guillard, 2000; Bhat and Deshmukh, 2005; Dreiling *et al.*, 2008; Gulla and Brasethvik, 2000; Haque, Pawar and Barson, 2003). The next section (section 1.5) presents a survey of the use of MLA in recent BPM publications

1.5 BPM at multiple levels of abstraction (MLA)

This section surveys the use of multiple levels of abstraction (MLA) in the BPM literature. First we introduce the theoretical foundations for using a MLA approach in BPM (subsection 1.5.1). Then, the use of MLA in various business process-oriented approaches is reviewed (subsection 1.5.2). All the approaches reviewed in this section recommend BPM at three levels of abstraction. However, depending on the author, the content of each abstraction level varies from one proposal to another. Additionally, we identify those research propositions that have been empirically tested using real case studies where participants are present both in IT and in management.

1.5.1 Theoretical foundations

Since its origins, business process management has been related to various management efforts proposed to bring competitiveness to the organizations, (Elzinga *et al.*, 1995; Zairi and Sinclair, 1995b). We have observed in subsection 1.2.1 that business process management is neither a technology nor a type of information system: it is a management approach, and this shapes the orientation followed by this thesis. It has been pointed out (Smith and Fingar, 2007) that one of the main reasons organizations were not doing business process management, is that it has been considered only as an IT initiative; therefore, management has had difficulties in identifying BPM's real value for the whole organization.

The motivation to include a MLA approach for BPM goes beyond the realization that management-oriented and IT-oriented groups of stakeholders must actively participate in a BPM project. Each of these groups involves their employees, and for each purpose of modeling it might be necessary to provide the means to represent the particular characteristics that differentiate the modeling needs of one employee from the needs of others. Many authors (Berger and Guillard, 2000; Curtis, Kellner and Over, 1992) have argued that the use of MLA helps to select the effective information to be provided to various types of users with various types of needs. Other authors (Burton-Jones and Meso, 2006) argue that research aimed at an organization should not be performed on an individual level basis, but on a multilevel approach. This multilevel approach does not necessarily consist of summing up the results of individual level studies, because the relationships between the various levels might be missing in the final result. To conduct a multilevel research, it is necessary to study and analyze various levels at the same time.

Therefore, a MLA approach should consider in a holistic way the needs of the various managerial activities of the organization. Therefore, the foundations for a MLA analysis are drawn upon theories of organizational management, management information systems, and decision support systems disciplines, where managerial activities have been extensively studied and explained. One long used approach for classifying the managerial activities is described by Anthony's model (Anthony, 1965) which defines three levels of activities: strategic planning, management control, and operational control. Each type of activity has enough particular properties to demand various details of information. Strategic planning covers all those activities related to the goals, objectives and policies of the organization. Management control deals with the attainment and efficient use of the resources of the organization. Operational control activities procure the efficient and effective execution of the specific tasks of the organization. The term "tactical planning" has been proposed instead of management control (Gorry and Morton, 1989, p. 59), arguing that "there is a need of planning and control" at all the three levels. Therefore, for this thesis the terms to be used for Anthony's model levels are: strategic, tactical, and operational. Nowadays it is possible to find several kinds of organizational structures that do not follow the traditional organizational pyramid. Even then, it can be argued that organizations host actors with various levels of information needs that respond to the various types of activities being performed at a specific moment in time.

Some authors (Gorry and Morton, 1989) have also pointed out that problems are getting more complex over time, and as a consequence organizations look for solutions that are not a responsibility of an individual but of a team; therefore, it is critical for organizations to provide the means for the coordination of the various members of the team. This vision is stressed by Courtney (2001), mentioning that the focus of information support systems has moved from the individual level to the organizational level, passing through the group level. It is required to consider the organizational knowledge as a "collective mind" that demands "communications between individuals", "sharing of knowledge", and "coordinating actions" (p. 24). To achieve these levels of cooperation, the points of view of all the stakeholders must be considered for the solution of a problem. Finally, it has been noted that the three levels defined by Courtney (i.e. individual, group, and organizational) should be studied concurrently (Burton-Jones and Meso, 2006).

1.5.2 MLA and its use in business process-oriented approaches

The use of MLA is common for various business process-oriented approaches. Many of them recognize that within any organization various types of stakeholders demand for different granularities of information. This subsection surveys the use of MLA in: 1) management-oriented approaches; 2) BPM notations; and 3) recent BPM research proposals.

1.5.2.1 MLA in management-oriented approaches

The Balanced Score Card (BSC) defines four process perspectives: financial, customer, internal process, and innovation/learning (Kaplan and Norton, 2007). In the internal process perspective, the focus is on the core business processes of the organization. Each of the process perspectives is typically represented by a three-layered structure: mission, objectives and measures.

The ISO 9000 family of standards follows a [business] process approach and recommends, besides the quality policy of the organization, three levels of documentation: the quality manual, the description of the processes, and the support records (ISO, 2008; 2010). The ISO standards reflect the three levels of managerial activities: strategic, tactical, and operational (Berger and Guillard, 2000). The close relationship between BPM and the ISO 9000 standards is well documented in the literature (Berger and Guillard, 2000; Hoyle, 2006; ISO, 2008; Lankhorst, 2005; Zairi, 1997). Moreover, there is evidence that organizations with an ISO 9000 familiarity are more likely to better exploit BPM initiatives (Elzinga *et al.*, 1995; Harmon and Wolf, 2010).

The Supply Chain Operations Reference (SCOR) model (SCC, 2008) proposes a process reference model with three levels of process description based on the value chain: 1) The top level defines the scope and the types of processes to be used by the organization; 2) The configuration level allows to further describe and configure each process, the chosen configuration determines the strategy of the organization; and 3) The process element level deals with the details and decomposition of each process. Additionally, SCOR allows organizations to add more levels of abstraction for a further decomposition and implementation of the processes.

Note from these examples that management-oriented process approaches usually present three layers of abstraction as depicted in Table 1.4. Also, these approaches consider as important the inclusion of the customers and providers in the modeling of the business processes. Table 1.4 does not aim at showing inter-levels equivalences.

Table 1.4 MLA in business process-oriented management approaches
Reprinted from Monsalve, April and Abran (2011a, p. 239), Copyright © 2011 by IADIS Press (http://www.iadis.org), with permission from IADIS Press

Anthony's Model		BSC Process Perspective	ISO 9000	SCOR Process Reference Model	
Level	Content	Content	Content	Level	Content
Strategic	Goals, objectives	Mission	Quality Manual	Top level	Scope, types
Tactical	Resources	Objectives	Business Processes	Configuration level	Description and configuration of processes
Operational	Specific tasks	Measure	Support records	Process element level	Details of each process: inputs, outputs, information, metrics.

1.5.2.2 MLA in BPM notations and methods

Qualigram proposes three levels of abstraction (See subsection 1.3.3). The top level (i.e. process level) models the processes, answering the questions "why" and "where to", and deals with the mission, objectives, and policies of the organization. The intermediate level (i.e. procedure level) models the procedures, answering the questions "who" and "what", and describes how to achieve the objectives of the organization. Finally, the lowest level (i.e. instruction level) models the work instructions, answering the questions "how" and "using what", and deals with the control of specific tasks. Qualigram does not aim at building the basis for the implementation of an information system. Therefore, Qualigram is clearly a management-oriented BPM notation.

The Architecture of integrated information systems (ARIS) defines five enterprise perspectives: organization, data, flow, output and control. Each of these perspectives presents three description levels: requirements definition, design specification, and implementation description. From a BPM point of view, ARIS works at three levels of abstraction: 1) strategy level, 2) design, control and optimization level, and 3) execution level (Davis, 2008; Scheer and Schneider, 2006; Scheer, Thomas and Adam, 2005). In addition, ARIS suggests a BPM hierarchy composed of three abstraction levels: high level process, functions, and tasks (Davis, 2008). ARIS responds to the idea of offering an integral solution for implementing information systems from the business processes of the organization; clearly it is an IT-oriented approach.

BPMN does not present a clear-layered approach. However, since BPMN was designed to provide a unified notation, both for IT and management stakeholders, it offers the possibility to model at different levels of granularity. The various BPMN versions present variations in the terminology and possibilities of modeling at different levels of granularity; for the reasons already explained in subsection 1.3.2.1, the following description of some BPMN characteristics is in accordance with BPMN version 1.2 (OMG, 2009a). To model at different levels of granularity BPMN offers two modeling resources: 1) Three types of models; and 2) Two sets of modeling constructs. These modeling resources can be combined according to the modeler requirements. The next paragraphs describe these two modeling resources.

BPMN[8] allows elaborating three types of models according to the necessities of the users: 1) private business process[9]; 2) abstract process[10]; and 3) collaboration process[11] (OMG, 2009a). A private business process allows representing a single business process as a white box; it can be as much detailed as needed, but it always requires to at least depict the main activities and their sequence flow. An abstract process allows representing the interactions between a private business process and an external entity (i.e. external business process or external participant) that is represented as a black box. Therefore, the representation of the external entity does not require depicting its activities and their sequence flow. Finally, a collaboration process allows representing the interactions between two or more private business processes. Therefore, this latter type of BPMN model requires each of the business processes to be represented as a white box.

In addition, to ease the interaction between various types of stakeholders, BPMN proposes the use of two sets of modeling constructs that were described in subsection 1.3.2.1: the BPD Core Element Set and the BPD Extended Set.

Based on these BPMN modeling resources, Silver (2009) proposes to model at three "BPMN levels of use"[12]. Each of the three levels of use includes a modeling methodology to ensure compliance to the BPMN specifications and consistency in the semantics of the modeling constructs used throughout a BPM initiative. The top level (i.e. Level 1) is a descriptive level[13] mainly intended for business and management users; it is based on the use of BPMN's Core Element Set plus collapsed processes. Level 1 aims at representing in a simple way the

---

[8] The process modeling conformance of BPMN version 2.0 (see subsection 1.3.2.1) defines three types of BPM diagrams: 1) process diagrams; 2) collaboration diagrams; and 3) conversation diagrams. The latter type of BPM diagram is out of the scope of this thesis. In addition, BPMN version 2.0 defines three types of "sub-models within an end-to-end BPMN model" (OMG, 2011, p. 22): 1) process sub-models (private and public); 2) choreography sub-models; and 3) collaboration sub-models. The choreography type of sub-model is out of the scope of this thesis.

[9] BPMN 2.0 defines two types of private business processes: non-executable and executable. The private business process defined in BPMN 1.2 corresponds to the non-executable version. The executable type of private process defined in BPMN 2.0 is out of the scope of this thesis.

[10] It is equivalent to the public type of business process defined in BPMN version 2.0.

[11] BPMN 2.0 also defines a collaboration type of process.

[12] Silver's work is based on BPMN 2.0 but the style and method he proposes is also applicable to BPMN 1.2.

[13] Corresponds to the Descriptive conformance sub-class defined by BPMN 2.0.

main workflow of a business process. The intermediate level (i.e. Level 2) is an analytical level[14] intended to detail the events and exceptions of the business processes, allowing representing all possible workflows of a business process but avoiding representing technical details that are intended to the execution of the business process. Level 2 targets at business analysts, business architects and IT personnel. At Level 2 the modeler may use all the modeling constructs offered by BPMN. Finally, the lowest level (i.e. Level 3) is an executable level intended to exploit the underlying Extensible Markup Language (XML) characteristics of BPMN. Level 3 targets at software developers that want to develop a software application governed by a business process model[15].

Table 1.5 summarizes the reviewed notations and methods. The table does not aim at showing inter-level equivalences.

Table 1.5 MLA in BPM notations
Adapted from Monsalve, April and Abran (2011a, p. 240), Copyright © 2011 by IADIS Press (http://www.iadis.org), with permission from IADIS Press

Qualigram		ARIS			BPMN		
Level	**Content**	**Perspective views**	**BPM point of view**	**Process model hierarchy**	**BPMN types of models**	**BPMN levels of use**	**Content**
Level 1: process level.	Processes, sub processes, objectives.	Requirements definition.	Strategy level.	High level processes.	Private business process.	Level 1: descriptive.	Basic set of modeling constructs.
Level 2: procedure level.	Procedures, instructions, roles.	Design specification.	Design, control and optimization level.	Functions.	Abstract process.	Level 2: analytical.	Detailed modeling constructs, exception handling, transactions.
Level 3: instruction level.	Instructions, operations, tools, documents.	Implementation description.	Execution level.	Tasks.	Collaboration process.	Level 3: execution.	XML description of executable processes.

[14] Corresponds to the Analytic conformance sub-class defined by BPMN 2.0.

[15] The specifications of BPMN 2.0 (OMG, 2011) define a “Common Executable” conformance sub-class which is related to Silver’s Level 3. This conformance sub-class, and therefore Silver’s Level 3 are out of the scope of this thesis.

1.5.2.3 MLA in BPM research proposals

Bhat and Deshmukh (2005) argue that in order to achieve flexibility it is necessary to model business processes at MLA. The stakeholders of the organization need to share a common vision of the business processes, and this is achieved by modeling at a top level (i.e. business process level) the core processes of the organization with a very low degree of detail. Then the proposed hierarchy includes two more levels of abstraction to introduce the individual requirements of the various stakeholders. The intermediate level (i.e. process workflow level) depicts the workflow for each business process, and includes sub-processes if it is necessary. The lowest level (i.e. business procedures or tasks level) depicts the details of each of the sub-processes, including the responsibilities of the human actors and the information system.

Haque, Pawar and Barson (2003) argue that experiences from both industry and academia have obtained better results when also considering organizational issues rather than only considering technological issues. Complex problems are better tackled by multi functional teams that demand mechanisms to share and collaborate. Their empirical work identified the following issues among the main factors that inhibit the process of integrated product development: 1) A functional orientation of the organization; 2) The "lack of ... communication and collaboration between internal functions and external partners"; and 3) A "weak process understanding across functions" (pp. 148-149). They conclude that it is critical to have a cross-departmental vision in the organization, facilitating the cooperation and sharing of knowledge, and providing detailed description of the processes. Therefore, they propose to model and analyze at the following levels of abstraction: 1) Level 3: "company strategy level", deals with the strategy and goals of the organization; 2) Level 2: "functional and process phase level", models the processes at a high level and explains how the various departments collaborate to achieve the goals; and 3) Level 1: "operating team and detailed operational process", details the core processes and the processes of the external partners.

Lin, Yang and Pai (2002) analyze various BPM notations to identify "essential components" of a business process model and the "modeling perspectives [that] cover the essence of" business processes (p. 29). Based on those results, they construct a framework for BPM and a generic method for modeling business processes. The generic method proposes three levels of abstraction: 1) gross grained level; 2) medium grained level; and 3) fine grained level. Each level corresponds to a different granularity of the information to be represented. They provide an example of how to apply the proposed method, modeling a supply chain network at the highest abstraction level (i.e. gross grained level), the core business processes of the organization at the intermediate level (i.e. medium grained level), and the functionality of each business process at the lowest level (i.e. fine grained level).

Gulla and Brasethvik (2000, p. 17) find a "lack of coordination of modeling activities" that causes the existence of multiple and inconsistent models of the same business process within the organization. Therefore, they see as critical to provide stakeholders with systematic means to facilitate the communication and cooperation when modeling business processes. Their methodology proposal analyzes the business processes at three levels of abstraction: 1) Functional tier: models a business process from an ERP functionality point of view; 2) Workflow tier: models how the various roles interact, the tools and applications they use; and 3) Business tier: deals with the goals and strategies of the organization. Gulla and Brasethvik (p. 22) conclude that "in practice, the process models are usually combinations of all these three tiers". This later conclusion is stressed as critical, pointing out that for a large project any modeling tool should provide the ability to model business processes at MLA.

Dreiling *et al.* (2008) argue that when a business process has been modeled for one specific purpose then it probably will not be easily reused for another purpose. To provide value to any BPM effort as part of an enterprise system project, they propose an approach that: 1) integrates the various modeling notations used by the stakeholders; and 2) provides the means to adapt existing business process models to the specific needs of the project. They propose three levels of abstraction: management oriented, business analyst oriented, and technical oriented. At the management-oriented level there is a need of modeling business processes at a high level of abstraction, depicting a simple and big picture of the business processes and their main inter-relations. At the business analyst oriented level business processes are modeled for multiple purposes (e.g. system requirements and process improvement) demanding for more detailed models; this level aims at the communication between business analysts and business users. At the technical-oriented level there is the need of a high degree of rigor in the models, having to depict all the business process information required for the implementation of the enterprise system. To integrate the various modeling notations used at each of the levels of abstraction, a mapping at a meta-level is proposed. The meta-level constructs that result from the mappings are used as intermediate layers. The approach includes an intermediate layer between the management-oriented and the business analyst oriented levels; and another one between the latter level and the technical-oriented level.

Table 1.6 summarizes the various proposals presented. The table does not aim at showing inter-level equivalences but a general overview. All these proposals share the use of three levels of abstraction. However, the characteristics of the levels of abstraction vary greatly from one proposal to another. Those proposals that present a *management orientation* privilege the representation of the goals and objectives of the core business processes, their metrics, and the details of the various activities, roles, resources and information required for the execution of the business processes. Those proposals that present an *IT orientation* privilege the representation of the details of the business processes; going from a 'big picture' perspective to detailed and rigorous description of each workflow. Unifying these approaches appears to be critical for achieving a shared vision of the business processes within an organization; however, it is a challenging task.

In summary, sections 1.4 and 1.5 presented the multiple efforts found in the literature trying to represent all the modeling needs and constraints demanded by the BPM stakeholders either by representing multiple BPM perspectives or by modeling the business processes at multiple levels of abstraction. Once again, the difficulties appear as a consequence that each stakeholder presents specific modeling needs according to the activity where the business

Table 1.6 MLA in BPM research proposals
Reprinted from Monsalve, April and Abran (2011a, p. 240), Copyright © 2011 by IADIS Press (http://www.iadis.org), with permission from IADIS Press

Bhat & Deshmukh	**Level**	Business processes	Process workflow	Business procedures
	Content	Core processes	Workflow and sub processes	Procedures, tasks, system info, details
Haque, Pawar & Barson	**Level**	Level 3. Company strategy	Level 2. Functional & Process phase	Level 1. Operating team
	Content	Strategy, goals.	High level business processes, functions.	Details of organization &providers processes.
Lin, Yan and Pai	**Level**	Gross grained	Medium grained	Fine grained
	Content	Supply chain network	Core processes	Functionality of each process
Gulla & Brasethvik	**Level**	Business	Workflow	Functional
	Content	Goals, strategy	Workflow, roles, tools, resources	ERP point of view of business processes
Dreiling, Rosemann, van der Aalst & Sadiq	**Level**	Management	Business analyst	Technical
	Content	High level business processes, inter-relations	Rich detail (workflow), some rigor, intuitive notation	Information required for the implementation of the systems

process model is planned to be used. Therefore, a first step aiming at a solution is to select a perspective or a group of perspectives to be represented. In the next section (section 1.6) the main approaches presented in the literature for representing the BPM perspectives required by software engineers and business analysts for software requirements elicitation are reviewed.

1.6 BPM and software requirements elicitation

This section focuses on the specific BPM perspectives demanded by two different types of stakeholders: software engineers and business analysts, performing one specific activity: software requirements elicitation. First, subsection 1.6.1 presents the references used in this thesis for delimiting the needs and constraints of software requirements elicitation. The emphasis of this part of the literature review is to show the relevance, according to the references identified, of using BPM as a technique for software requirements elicitation. Next, subsection 1.6.2 presents the main approaches described in the literature for representing the BPM perspectives required by software engineers and business analysts during software requirements elicitation.

1.6.1 The references for requirements elicitation

The references used in this thesis for delimiting the needs and constraints of software requirements elicitation are: 1) the guide to the Software Engineering Body of Knowledge (SWEBOK); and 2) the guide to the Business Analysis Body of Knowledge (BABOK).

The SWEBOK (Abran *et al.*, 2004) has two main purposes: 1) "to provide a consensually validated characterization of the bounds of the software engineering discipline;" and 2) "to provide a topical access to the Body of Knowledge supporting that discipline" (p. xvii). One of the "knowledge areas" (KA) presented in the SWEBOK is Software Requirements, and requirements elicitation is one of the key topics of this KA.

The SWEBOK defines a requirement as "a property that must be exhibited in order to solve some real-world problem" (Abran *et al.*, 2004, p. 1.3), and makes a clear difference between "system requirements" and "software requirements" (p. 2.3). System requirements involve not only software components but also other components such as hardware, firmware, and people. Software requirements express the needs and constraints placed on a software product that contribute to the solution of some real-world problem and "are derived from system requirements" (p. 2.3). According to the SWEBOK, software engineers mainly deal with the software requirements specifications; they are "concerned with the elicitation, analysis, specification, and validation of software requirements" (p. 2.1). The more general system requirements are recognized as a responsibility of other professionals.

The BABOK (IIBA, 2009) adopts a more general view of requirements. Business analysts are concerned with the understanding of “the structure, policies, and operations of an organization” (p. 266). Hence, a requirement is defined by the BABOK (pp. 230-231) as: 1) “A condition or capability needed by a stakeholder to solve a problem”; 2) “A condition or capability that must be met … by a solution … to satisfy a … formally imposed document”; and 3) “A documented representation of a condition or capability as in 1) or 2)”. Since not all of the requirements are found in an explicit way, an important objective for business analysts is to document and communicate them by means that are “understood by all stakeholders”.

According to the SWEBOK, functional requirements “describe the functions that the software is to execute” and non functional requirements “act to constrain the solution” (Abran *et al.*, 2004, p. 2.2). The SWEBOK also points out the importance of understanding the business processes of the organization during the requirements activities, something that is considered a complex task.

Conceptual modeling is considered by the SWEBOK, as well by other authors, as one of the main techniques for eliciting requirements (to depict scenarios), and for requirements analysis (to model and understand the problem, its context and its interface, its entities and their structure) (Abran *et al.*, 2004; Burton-Jones and Meso, 2006; Davies *et al.*, 2006; Indulska, zur Muehlen and Recker, 2009; Recker *et al.*, 2006; Recker *et al.*, 2007; Wand and Weber, 2002). Empirical results (Davies *et al.*, 2006) show that business process modeling is commonly used by practitioners to perform conceptual modeling. As a result, BPM has been considered as “an important task during requirements engineering” (Albani and Dietz, 2006; Mayr, Kop and Esberger, 2007, Abstract; Mili *et al.*, 2009).

According to the BABOK, BPM is a recommended technique for business analysis planning and monitoring, enterprise analysis, and requirements analysis. Both, the SWEBOK and the BABOK suggest complementary techniques for requirements elicitation; some examples of these techniques are: observation, prototyping, and interviews.

1.6.2 Proposed approaches to BPM for software requirements elicitation

Despite of their growing complexity, the most commonly used BPM notations still lack the appropriate modeling constructs to represent all kinds of requirements that software engineers and business analysts need to represent (Lapouchnian, Yu and Mylopoulos, 2007; List and Korherr, 2006; Pavlovski and Zou, 2008; Vara, Sánchez and Pastor, 2008). There are, in the literature, different proposals to improve the use of BPM for the requirements activities. Some of the proposed solutions have developed modeling notations and methodologies for representing different kinds of requirements with a business process orientation (e.g. i* based approaches) (Decreus and Poels, 2009; Kazhamiakin, Pistore and Roveri, 2004; Lapouchnian, Yu and Mylopoulos, 2007; Yu and Mylopoulos, 1994a; Yu and Mylopoulos, 1994b). Some others propose to extend UML (Engels *et al.*, 2005; Eriksson and Penker, 2000; List and Korherr, 2005; Vasconcelos *et al.*, 2001) or BPMN (Pavlovski and Zou, 2008; Vara, Sánchez and Pastor, 2008) to support requirements modeling. The use of an inter-lingua notation (Mayr, Kop and Esberger, 2007) for bridging BPM and the requirements activities has also been proposed. Other proposals describe a methodology for combining BPM with requirements elicitation without providing a solution for modeling the requirements (Demirors, Gencel and Tarhan, 2003; Kueng and Kawalek, 1997; Pichler and Rumetshofer, 2006). The next subsections (1.6.2.1 to 1.6.2.5) present a summary of five main types of proposals.

1.6.2.1 i* based approaches

According to the SWEBOK, one key source of software requirements are the goals (or objectives) of the software to be developed. From a BPM point of view, these goals are related to the goals of the underlying business processes. Some authors have suggested that one of the weaknesses of current BPM notations is their scarce support for representing goals (Kueng and Kawalek, 1997; Lapouchnian, Yu and Mylopoulos, 2007; List and Korherr, 2006; Yu and Mylopoulos, 1994a).

The early works of Yu and Mylopoulos (1994a; 1994b) proposed a new framework for modeling business processes: the i* framework, which includes two models: the strategic-dependency model (initially called actor-dependency model) and the strategic-rationale model. Their proposal is that business process models typically represent only the *what?* (e.g. what activity has to be executed by a role?); however, to represent the non-functional requirements a business process model also has to answer the *whys?* (e.g. why does the role needs to execute this activity?), as well as the *what-ifs*? (e.g. what if this activity is eliminated?).

Recently, Lapouchnian, Yu and Mylopoulos (2007) proposed a notation for developing goal-oriented models. This newer proposal is based on the strategic-rationale model of the i* framework. The goal-oriented models are more concerned with the goals of the business processes than with their control flow or behavior. The goal-oriented models are intended to depict all feasible and optional ways for achieving one main goal. The goal-oriented models also include the representation of functional goals (also known as hard goals) and qualitative goals (also known as soft goals), as well as a punctuation notation for describing the control flow and the impact of the functional goals over the soft goals. This last characteristic is useful for the dynamic configuration of the business processes to fit the users and customers requirements: *i.e.* business process variability management. This proposal includes a method for going from the preliminary goal models to code generation. Prototype tools for supporting the business process variability, and the generation of the corresponding code are available. The proposed notation is relatively easy to understand from the goals perspective; however, the control flow notation is not intuitive, and the roles involved in the business process are not represented. It is not clear, in the publications, if the proposed method has been validated.

Another i* based proposal, is the Tropos software development methodology, which is a requirements-oriented methodology (Castro, Kolp and Mylopoulos, 2002). Decreus and Poels (2009) combine the i* strategic-rationale model with the formal specification language of Tropos (Formal Tropos or FT), and the Semantic Business Process Management (SBPM) ontology. They propose a method for going from the requirements elicitation to coding the associated business processes in Business Process Modeling Ontology (BPMO) which is a superset of BPMN and EPC. The goals are first elicited using the i* model. Decreus and Poels propose that, for a detailed requirements analysis, a more formal description is needed. Thus, they translate the graphical model into FT language. This language has an analysis, verification and validation tool called the T-Tool, which is used for analyzing the requirements expressed in FT. Then, Decreus and Poels use SBPM to enrich the description in FT. They argue that this step eases the translation of the FT description into BPMO, which is the last step of their proposed method. No evidence of an empirical validation of the proposed method is provided.

Finally, Kazhamiakin, Pistore and Roveri (2004) propose a framework also based on Tropos, for going from a high-level requirements model to a skeleton coding. This proposal is oriented towards the requirements orchestration and choreography for organizations that might interconnect their business processes to the ones of other organizations. First, the different entities (internal and external) that will interact in the business process, their goals, and the main dependencies among them, are modeled. From this high-level model, an i* model is generated with the particularity that the interfaces of the external entities are also shown. The i* model includes the messages to be transmitted/received to/from the external entities. The model is then translated into FT language, and the T-Tool is used for a set of verifications. Finally, a skeleton code is generated from the FT description. Kazhamiakin, Pistore and Roveri claim that this method allows an early verification of the requirements and the business processes (before the final coding).

1.6.2.2 UML based approaches

Other approaches use UML and its extension capabilities (refer to subsection 1.3.2.4) as a foundation. Eriksson and Penker (2000) using the standard extensions mechanisms of UML proposed a modeling approach consisting of four different views they refer to as "Eriksson-Penker Business Extensions". The views not only include the traditional aspects represented by a business process model (behavioral and process views), but also the functional structure of an organization (structural view) and the vision view (refer to subsection 1.4.2 and Table 1.3). The latter, the vision view, was designed for depicting the goals of the organization and of its business processes. Some vendors have used the Ericksonn-Penker extensions in their products and services (CEPHAS, 2009; SPARKX, 2009), and an open source project also has used them (StarUML).

List and Korherr (2005) proposal creates another UML profile extending the meta-classes actor, property and class from the UML meta-model. The specifications for each of the created stereotypes are provided, as well as an example of their use. The profile includes extensions for representing goals, quantitative and qualitative measures, and the relationships of the business processes with their owners and the customers, and a classification of the business processes as core, management and support business processes. There is no evidence of empirical validation of this profile.

Vasconcelos *et al.* (2001) propose a three-tier UML profile for modeling business processes; it includes the strategic goals associated to them, and the information system components that support the business processes. The top tier, business strategy, models the goals and it is inspired from the Balanced Score Card approach (Kaplan and Norton, 2007); thus, goals are classified as strategic and operational, and each of these classes are sub-classified as qualitative and quantitative. When modeling a goal it is suggested to describe its specific perspective: financial, customer, internal business processes, or learning and growth (see subsection 1.5.2.1). The business process tier is the intermediate layer; its modeling is inspired from the value chain model (Porter and Millar, 1985). Business processes are classified as core (inbound, operations, outbound, and sales) and support business processes (management, human resources, technology, and procurement). Resources used or produced by the business processes are also modeled as part of the middle tier. Finally, the information systems tier models the "functional building blocks of the system" (extended components) (Vasconcelos *et al.*, 2001, p. 75). The authors state that this UML profile has been used in industrial projects.

1.6.2.3 BPMN based approaches

Pavlovski and Zou (2008) propose to extend BPMN to enhance it with the capabilities of representing the "operational behavior and the associated process constraints" (p. 103). They argue that representing these characteristics at an early stage of the software development process will allow the software development team to determine all the detailed non-functional requirements. They propose two extensions to the current BPMN notation. One, named "operating condition" for representing the business process constraints; and a second one, named "control case", for describing the risks associated to the constraints and the mechanisms to reduce their impact. The former one, operating condition, is graphically attached to any activity that presents a constraint. The control case is considered as optional, more likely to be used at a lower level of abstraction, and besides the graphical symbol suggested to represent it, it includes a text table with the description of the associated constraint, its risks, and means to reduce the impact. No publication could be identified indicating that these extensions have been validated empirically.

Vara, Sánchez, and Pastor (2008) combine BPMN and a goal-oriented notation called *Map* to propose a method for requirements analysis. They are also concerned with business process variability management. BPMN is used for modeling the business processes, and *Map* is used for modeling the goals and strategies that lead to fulfill the users' requirements. Initial BPMN models (i.e. as-is business process models) are updated by the results of the analysis of the *Map* model to get the to-be business process models. The BPMN notation is extended with a set of three labels that can be attached to different elements of the BPMN models to indicate if the elements are not part of the information system, are executed by a human actor, or are under the control of the information system. For every activity to be included in the information system a text table that describes the requirements associated to the activity has to be created. The method has been empirically validated with a single software development company. However, not all the software development team agreed that the approach was worth using.

1.6.2.4 Approaches based on other BPM notations

Mayr, Kop and Esberger (2007) argue that "the same basic [business process] modeling notions should be used" for requirements modeling. The basic business process modeling notions are: 1) tasks are executed in a workflow; 2) tasks are executed by roles; 3) resources are used by tasks; and 4) tasks have pre and post conditions. The use of an inter-lingua language named "Klagenfurt conceptual pre-design model" (KCPM) is proposed considering that stakeholders prefer to use natural language to specify their requirements. The KCPM language and its use for depicting the business processes and their requirements are described. Finally, the mapping of the KCPM generated models into available BPM notations is explained. The authors claim that KCPM has been empirically tested in one case study, and that the models were well accepted by the stakeholders.

Simon (2005) considers that the real value of BPM is the enactment of information systems based on the business processes' workflow. To achieve that, an iterative process is needed. According to Simon, the process should begin with the representation of the requirements which later needs to be transformed into the information system. Simon argues that this process is not possible to be handled by current BPM notations; therefore, a new BPM notation is introduced which makes use of "textual specification and graphical visualization" (p. 226). The textual specification is expressed using a formal language named Logic of Actions (LoA), and the graphical visualization provides symbols for representing the operations of the formal language and the resources that are created or used by the operations. An example is used to compare this approach with other current BPM notations; however, there are no evidences of an empirical validation.

1.6.2.5 Methodological approaches

Kueng and Kawalek (1997) point out the importance of considering goals when modeling business processes. They propose a methodology for modeling business processes (activities, roles and objects) having as a starting point the goals of the software to be developed. As Yu and Mylopoulos (1994a), Kueng and Kawalek believe that modeling business processes should consider the *whys?*. They also propose to use the requirements derived from their methodology to evaluate BPM; however, they do not provide a solution for the graphical representation of the goals and software requirements.

BPM has also been proposed as an effective mechanism for identifying system requirements from a software acquirer point of view. Demirors, Gencel, and Tarhan (2003) argued that for contracting the acquisition of a "software intensive system", it is not enough to consider the customer needs, but it is imperative to clearly "understand the concept, the domain as a whole, the technology to be utilized and technical and management constraints" (p. 409); furthermore, they argue that the process of going from the concept to the detailed system requirements might be done using the "notations and tools developed primarily for business process reengineering" (p. 409). The authors propose a methodology to be followed by means of describing a software acquisition case. The methodology follows four stages. The first one defines the concept of the system to be acquired, then the as-is business processes are modeled and analyzed; EPC is used among other modeling notations. The third stage consists of modeling the to-be business processes. Finally, the system requirements are defined. The methodology also includes a quality assurance activity, which performs a verification and validation of all the requirements, looking for their consistency and traceability to the modeled business processes.

Finally, Pichler and Rumetshofer (2006) describe the importance of using adequate visualizing tools for describing the requirements and the business processes along the life cycle of a software development project. Their work explains the experience of using different notations, tools and techniques, ranging from post-it papers to professional requirements tools, in a three year project.

In summary, the literature shows several examples of attempts to represent the business process perspectives demanded by software engineers and business analysts in order to use BPM for software requirements elicitation. Some of these attempts try to complement or improve what already exists; others propose the implementation of new BPM notations and methodologies. It is argued in this thesis that before proposing new solutions, a formal identification of the modeling constructs required for representing the needs and constraints of each group of stakeholders (i.e. business analysts and software engineers) for this specific task (i.e. software requirements elicitation) is needed as a first step. For this purpose, an ontological analysis could help to systematically identify "the basic things in the real world" that should be modeled (Wand and Weber, 1990). This thesis follows an ontological analysis based on a theory of the representational capabilities of a modeling notation (Wand and Weber, 1990; 1993; 1995) that has been extensively used to assess many of the most popular BPM notations (Recker *et al.*, 2009). The next section (section 1.7) introduces the underlying theoretical principles of such an ontological analysis.

1.7 An ontology based theory of representation

Ontology studies the real world, its structure, and the relations between the things that conform to the real world (Easterbrook *et al.*, 2008; Recker *et al.*, 2007; Wand and Weber, 1990). Since this thesis aims at identifying the required modeling constructs to represent the needs and constraints of a specific stakeholder performing a specific task, then an ontological basis is necessary.

Wand and Weber (1990; 1995) have proposed a set of ontological models that initially were focused on identifying the basic concepts that allow the description of any kind of information system, their structure, and their behavior. They point out that, from these models, it is the "representation model" that is useful to evaluate the expressiveness (i.e. completeness and clarity) of any modeling notation (Wand and Weber, 1993). The original definitions of the various concepts of the representation model can be found in (Wand and Weber, 1993; 1995). These original definitions have been adapted over the years. The version of the definitions used in this thesis is the adaptation proposed by (Green and Rosemann, 2000), and it can be found in Appendix X. A *thing* is the elementary concept:

according to the representation model "the real world is made up of things" that "might be of interest to users of information systems" (Green and Rosemann, p. 75).

Basically, the representation model is used to determine if a notation is complete (i.e. it has all the constructs needed to represent the real world) and clear (i.e. it has the sufficient constructs to allow an unambiguous interpretation of the generated models). Wand and Weber (1993) argue that a modeling notation is ontologically complete only if it is possible to map each of the concepts of the representation model into the modeling notation constructs (see (1) and (4) in Figure 1.7). They also argue that a modeling notation is ontologically clear only if each of its modeling constructs has a one-to-one mapping with the concepts of the representation model. Any deviation from this one-to-one mapping constitutes an ontological deficiency of the modeling notation, affecting its clarity. Wand and Weber present three types of ontological deficiencies: construct overload, construct redundancy, and construct excess. Construct overload (see (2) in Figure 1.7) exists when one specific construct of the

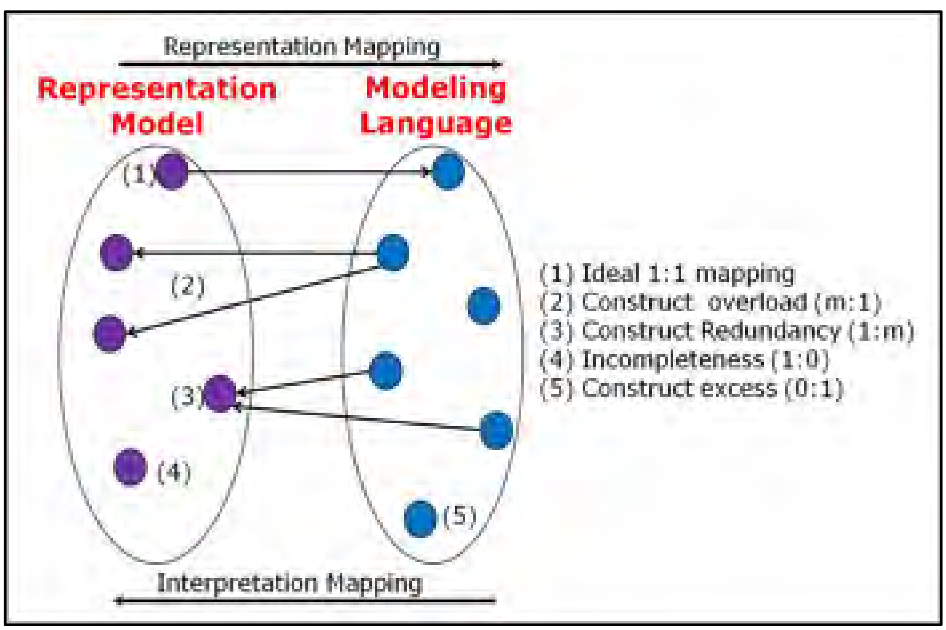

Figure 1.7 Representational analysis
Adapted from the adaptation of Weber (1997) made by Recker *et al.*(2009, p. 337), Copyright © 2009 by the Association for Information Systems (AIS), with permission from AIS

modeling notation can be mapped into two or more concepts of the representation model. Construct redundancy (see (3) in Figure 1.7) exists when one specific concept of the representation model can be mapped into two or more constructs of the modeling notation. Finally, there is construct excess (see (5) in Figure 1.7) when the modeling notation includes constructs that are not possible to be mapped to any of the concepts of the representation model. This representation model has been identified in the literature as the Bunge-Wand-Weber (BWW) representation model (Davies *et al.*, 2006; Green and Rosemann, 2000; Opdahl and Henderson-Sellers, 2002; Recker *et al.*, 2009); and the ontological assessment of modeling notations based on this representation model has been identified in the literature as the BWW "representational analysis" (Green *et al.*, 2006).

In addition, Wand and Weber (1993) propose two types of mappings between the representation model and the notation being assessed: 1) the representation mapping from the representation model to the modeling notation; and 2) the interpretation mapping from the modeling notation to the representation model (see top and bottom of Figure 1.7). The representation mapping requires mapping each ontological concept to its corresponding modeling notation constructs, and provides useful information for identifying the degree of completeness of the modeling notation assessed, as well as potential redundancy deficiencies of the modeling notation. The interpretation mapping requires mapping each modeling notation construct to its corresponding ontological concepts, and provides useful information for identifying the potential degree of overload of a modeling notation, as well as its constructs in excess.

In today's literature, it is possible to find proposals of other similar ontological assessment techniques (Albani and Dietz, 2006; Grüninger, Atefi and Fox, 2000; Guizzardi, Herre and Wagner, 2002; Hepp and Roman, 2007; Milton, 2007), but most of these have not achieved the popularity of representational analysis for the assessment of BPM notations (Gehlert and Esswein, 2007; Rosemann *et al.*, 2009). The BWW representational analysis has been considered as "the most popular reference ontology used for representational analyses" (Green *et al.*, 2006, Related Work section). The BWW representational analysis has been used to evaluate BPMN (Recker *et al.*, 2009), EPC (Green and Rosemann, 2000), UML (Opdahl and Henderson-Sellers, 2002), Petri Nets (Recker and Indulska, 2007), among others notations used for BPM (Recker *et al.*, 2009; Rosemann *et al.*, 2009), which shows that this analysis technique has reached a “high maturity” level (Gehlert and Esswein, 2007, p. 122). In addition, the representation model is considered to be “very well understood” (Gehlert and Esswein, 2007, p. 122). For these reasons, the BWW representational analysis and its representation model are adopted for the ontological analysis component of this thesis.

In summary, the BWW representational analysis is used in this thesis to formally identify the modeling concepts that should be represented in a business process model during a software requirements elicitation task. However, a business process model may be also used as a valuable source of infirmation for measuring the functional size of the software it represents. Therefore, this thesis studies the feasibility of using a business process model for this purpose using COSMIC ISO 19761 as the functional size measurement method. The next section (section 1.8) introduces the COSMIC FSM method, and then reviews previous related work where conceptual modeling and BPM have been used as a source for FSM.

1.8 BPM and functional size measurement

Software requirements specifications are typically used by software engineers as the source of information for measuring the functional size of the software to be developed. Since BPM can be used to elicit the software requirements, then a business process model may be a valuable source of information for functional size measurement (FSM). Several methods have been proposed for FSM. This thesis follows the COSMIC FSM method (COSMIC, 2009). It was not until recently that the use of BPM for COSMIC FSM has been studied (Kaya, 2010). This section first introduces the COSMIC FSM method (subsection 1.8.1), and then reviews related work (subsection 1.8.2).

1.8.1 The COSMIC FSM method

There are currently five FSM methods approved by the International Organization for Standardization (ISO/IEC, 2006; 2010). From these methods approved by the ISO, this thesis uses the one proposed by the Common Software Measurement International Consortium (COSMIC): the COSMIC FSM method (COSMIC, 2009). COSMIC has been accepted since 2003 as international standard ISO/IEC 19761:2011 “Software engineering – COSMIC: A functional size measurement method” (ISO/IEC, 2011). COSMIC was designed to be applied in various functional domains: 1) business application software; 2) real-time software; and 3) a combination of the two. It is completely open and available in multiple languages, and it has been reported to be easy to learn and use (COSMIC, 2009).

According to COSMIC (2009), a “functional user” of a software application might be a human, another software application, or a hardware device. A “boundary” acts as “a conceptual interface between the software … and its functional users” (p. 26). The interaction between the software application and its functional users is produced “via data movements”. A data movement is “a base functional component which moves a single data group type” (p. 44) where a data group is defined by COSMIC as “a distinct, non empty, non ordered and non redundant set of data attributes where each included data attribute describes a complementary aspect of the same object of interest” (p. 39). COSMIC defines four types of data movements: Entry (E), Exit (X), Write (W), and Read (R). An Entry occurs when a data group is moved from a functional user into the software. An Exit occurs when a data group is moved from the software to a functional user. A Write occurs when a data group is moved from the software into a persistent storage. Finally, a Read occurs when a data group is moved from a persistent storage into the software. The COSMIC measurement unit is a COSMIC function point (CFP), which represents one data movement of one data group. The functional size of software is obtained by adding the data movements identified. These concepts are illustrated in Figure 1.8.

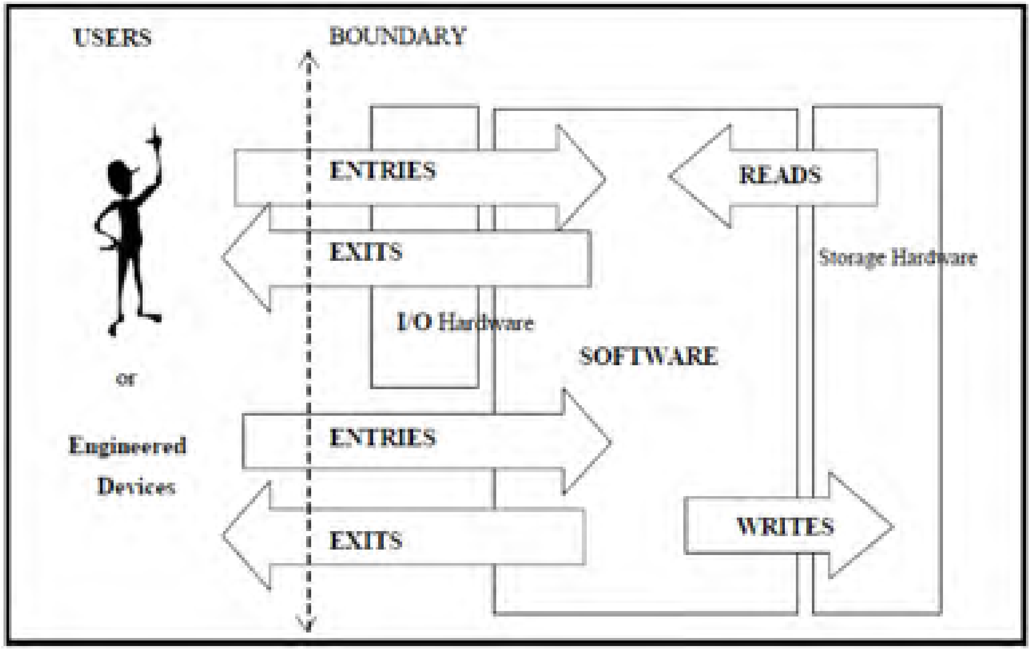

Figure 1.8 Generic flow of data attributes from a functional perspective
Reprinted from COSMIC (2003, p. 19), Abran *et al.*, Copyright © 2003 by the Common Software Measurement International Consortium (COSMIC), with permission from COSMIC

1.8.2 Functional size measurement from business process models using COSMIC

The use of conceptual models for FSM has been studied and analyzed in the research literature, and Marín, Giachetti and Pastor (2008) offer a survey of related works, including their own. In addition to the publications reported in that survey, other works have also studied the use of conceptual models for FSM. Lavazza and Del Bianco (2009) studied the use of UML diagrams (use case, component, and sequence diagrams) for modeling real-time software to be measured using the COSMIC FSM method. Sellami and Ben-Abdallah (2009) studied the potential relationships between the measurements obtained from UML use case diagrams and those obtained from other UML diagrams. From all these works, only two have included the use of some kind of BPM: Demirors and Gencel (2004) used an extension of EPC to model a military application. In this case, the EPC diagrams were used as part of the requirements elicitation methodology. The second work (van den Berg, Dekkers and Oudshoorn, 2005), proposes the use of UML activity diagrams as one of the possible options for representing the behavioral aspects of the software being modeled. However, neither of these two latter works presents a rule for mapping the business process models and the concepts of COSMIC FSM method. Moreover, the emphasis of these two works is not related to the feasibility of using only BPM for FSM.

At the moment of performing this research, only the work of Daneva (1999) could be identified in using BPM for FSM. Daneva's work is based on the use of the Function Point Analysis method (FPA), release 4.0 (as cited in Daneva, 1999, p. 145), endorsed by the International Function Point Users Group (IFPUG). It proposed an approach for measuring reuse "in the requirements conceptualization phase" of an ERP software project. To achieve its goal, the approach proposed to map the various modeling concepts of a commercial ERP software development tool to the "base logical components" of the IFPUG FPA method. The FSM method was used to measure "the size of the reusable requirements and the size of the total requirements" as indirect measurements of requirements reuse in an ERP "implementation project" (p. 144). The development tool used EPC diagrams to represent business requirements as business process models. Therefore, the mapping rules included the mapping of various EPC modeling concepts to the "base logical components" of the IFPUG FPA method. One of the conclusions of Daneva's work is that the "application of the counting model" from business process models requires validation (p. 149). Moreover, the FSM method used in Daneva's work is not the COSMIC FSM method.

1.9 Summary

The relevance of business process modeling both in the industry and in the academia has been pointed out, as well as its potential impact on software requirement elicitation activities. The main concepts related to business process management and business process modeling have been reviewed and defined according to the scope of this thesis. It has been pointed out the importance of generating high-quality business process models if these are meant to be used as part of a software requirements elicitation activity. The various efforts, found in the literature aiming at achieving this, have been analyzed and reported. It has been shown that in spite of these multiple efforts, the BPM notations and methods currently available still lack the capability of representing in a consistent way all the needs and constraints demanded by software engineers and business analysts when using a business process model as part of the software requirement specifications. It has also been pointed out that the use of a business process model for measuring the functional size of a software application has not been well explored, especially when the measurement method is meant to be the COSMIC FSM method.

Motivated by these findings, the main expected research deliverables of the research reported here are:

1. A novel BPM approach, referred to here on as BPM^+, that: a) takes into consideration the needs and constraints from the various stakeholders involved in a software requirements elicitation activity; b) represents in a consistent way those needs and constraints; c) easily communicate the requirements to the various stakeholders; and d) generates business process models that: i) can be shared among the various stakeholders; and ii) are rigorous enough to be used in the software development project; and
2. A procedure to measure the functional size of the software that supports (or might support) the business process represented in a business process model. This procedure is based on the COSMIC FSM method and defines a set of guidelines to be followed by practitioners in order to produce business process models that are feasible to be used for FSM purposes. The procedure includes a set of notation-independent guidelines that allow practitioners to apply the procedure to various BPM notations.

CHAPTER 2

RESEARCH METHODOLOGY, ACTIVITIES AND EXPECTED RESULTS

2.1 Research methodology and expected results

The research methodology to be used to tackle the proposed research question and research sub-questions (see page 6 of the Introduction) is presented using an adapted Basili's framework (Basili, Selby and Hutchens, 1986; Basili and Selby, 1991) for empirical research in software engineering as presented by Abran, Laframboise and Bourque (2003). The framework is composed of four phases: definition, planning, operation and interpretation.

2.1.1 Definition phase

The definition phase, depicted in Table 2.1, is composed of the following activities:

- The problem is identified and understood;
- A research topic where an original contribution to knowledge can be made within the timetable of a doctoral program is defined;
- The research question and research sub-questions are formulated;
- A potential solution to the problem is envisioned; and
- The beneficiaries of the research results are identified.

2.1.2 Planning phase

During the planning phase a literature review was conducted, covering:

1. The definitions and terminology used in the BPM domain;
2. The state of the art of business-process-modeling perspectives;
3. The state of the art in modeling notations used to represent business processes;
4. The state of the art in the use of BPM in software requirement elicitation activities;

Table 2.1 Definition phase

Motivation	Objective	Proposal	Research Users
How can a business process be represented to better suit the needs and constraints of the various stakeholders involved in software requirements elicitation activities?	To contribute to the representation of business processes for software requirements elicitation.	• A novel BPM approach (BPM^+) to consistently represent the needs and constraints of the various stakeholders. • A procedure to measure the functional size of a software application from a business process model.	• BPM vendors, practitioners and researchers. • Software engineers, business analysts, software measurers, information systems and IT professionals. • COSMIC community. • Organizations.

5. The different approaches proposed to represent the various needs and constraints of the stakeholders;
6. The state of the art in the assessment of the capacity of a BPM notation to represent a set of modeling needs; and
7. The state of the art in functional size measurement from business process models.

For the first research objective, the focus of the literature review was on identifying what already has been done, the efforts to standardize BPM, and to represent the various perspectives demanded by management and IT stakeholders. It was also identified what issues have not been tackled neither by the industry, nor by other research efforts.

For the second research objective, the focus of the literature review was on identifying if business process models have been used as a source for FSM, particularly when the functional size of software is measured based on the COSMIC FSM method. The efforts to perform FSM from conceptual modeling were studied in order to identify the accepted techniques, by industry and academia, to map the concepts of a FSM method to the concepts of a modeling technique.

During this phase, the research methodology for achieving the research objectives and answering the research sub-questions is planned and documented. This includes the research methods, and validation activities.

Table 2.2 Planning phase: BPM$^+$ modeling approach

Project steps	Inputs	Deliverables
Design of an *a priori* version of BPM$^+$.	Literature review of: 1) Business process management; 2) BPM; 3) Most common BPM notations; 4) BPM and SRE; 5) Business process perspectives; and 6) BPM at MLA.	• Number and scope of the BPM$^+$ levels of abstraction. • List of elements to be modeled at each level of abstraction • BPM notations selected for the research. • BPM$^+$ *a priori* version. • Research protocol for the evaluation of the *a priori* version of BPM$^+$ to be presented to the CÉR for its ethical acceptability
Evaluation of the *a priori* version of BPM$^+$ through: 1) a pilot case study (action research approach); 2) representational analyses; and 3) a survey with practitioners.	• BPM$^+$ *a priori* version. • BPMN and Qualigram specifications. • BWW representation model. • BPMN representational analysis. • SWEBOK and BABOK.	• Propositions to be tested by a survey. • Perceptions from the participants. • Focused representational analysis of BPMN and Qualigram. • Own reflections of the research team. • A set of BPM elements specifically selected for supporting SRE according to the SWEBOK and the BABOK.
Design of a reviewed version of BPM$^+$.	• Results from the evaluation of the *a priori* version of BPM$^+$.	• Modeling preferences at each of the BPM$^+$ levels of abstraction. • Preferences of the BPM notations. • Reviewed version of BPM$^+$. • Research protocol for the validation of the reviewed version of BPM$^+$ to be presented to the CÉR for its ethical acceptability
Validation of the reviewed version of BPM$^+$ through a case study (action research approach).	• Reviewed version of BPM$^+$. • Participants from a Canadian organization. • Own reflections of the research team.	• Validated scope and elements of the three levels of abstraction of BPM$^+$. • Validated preferences of the BPM notations. • Final version of BPM$^+$.

The summary of this phase is presented in two parts:

1. Table 2.2 for the BPM$^+$ modeling approach; and
2. Table 2.3 for the procedure proposed to measure the functional size of a software application from a business process model.

Table 2.3 Planning phase: FSM from BPM procedure

Project steps	Inputs	Deliverables
Design of the FSM procedure based on the Qualigram notation: business application domain.	• Literature review of: 1) FSM from BPM; and 2) FSM from conceptual modeling. • COSMIC FSM method. • Qualigram specifications. • COSMIC "C-registration system" case study (GELOG-ETS, 2008).	• Modeling guidelines for Qualigram notation. • Mapping rules between Qualigram notation and COSMIC FSM method.
Generalization and evaluation of the FSM procedure through its application to the real-time software domain and to a second BPM notation: BPMN.	• Lessons from the procedure designed for Qualigram. • BPMN specifications. • Control case study for the real-time software domain: "the rice cooker system" (Lavazza and Bianco, 2009).	• Modeling guidelines for BPMN. • Mapping rules between BPMN and COSMIC FSM method. • Notation-independent guidelines. • Procedure for the real-time domain.

The design process of BPM$^+$ requires the participation of human beings. Therefore, a research protocol for the research activities to be conducted in order to evaluate the *a priori* version of BPM$^+$ was elaborated and presented to the ÉTS Ethics Committee for Research (CÉR) for its evaluation within the ethics domain (See Appendix I). In the same line, a research protocol for the case study to validate the reviewed version of BPM$^+$ was elaborated and presented to the CÉR for its evaluation within the ethics domain (See Appendix III).

2.1.3 Operation phase

During the operation phase the research plan is executed. This phase includes three parts, one for each main deliverable:

1. the development of an *a priori* version of the BPM$^+$ approach;
2. the development of a reviewed version of BPM$^+$; and
3. the development of a procedure to measure the functional size of a software application from its business process models in accordance with the COSMIC FSM method.

Each of these parts includes two steps: 1) a validation of the deliverables; and 2) an analysis of the results (refer to Table 2.4 and Table 2.5).

The development of BPM$^+$ is based on a focused ontological analysis (Rosemann and Green, 2000) complemented by a multiple levels of abstraction analysis. In this way, the development of BPM$^+$: 1) would avoid proposing a solution that is only based on the "practical wisdom" of the authors "rather than on a scientific theory" (Recker *et al.*, 2007, p. 96); 2) systematically identifies "the basic things in the real world" that should be modeled based on the results of the focused ontological analysis (Wand and Weber, 1990, p. 1282); and 3) effectively selects the information to be provided to the various types of stakeholders, which present various types of needs, based on the analysis of various levels of abstraction (Berger and Guillard, 2000; Curtis, Kellner and Over, 1992).

The *a priori* version of BPM$^+$ is developed based on the results of the literature review. This *a priori* version includes the definition of three levels of abstraction: 1) strategic (top level); 2) tactical (intermediate level); and 3) operational (lower level); as well as their scope and a preliminary list of BPM elements that should be modeled at each level of abstraction. Then, the *a priori* version of BPM$^+$ is evaluated and improved taking into consideration the results of: 1) a pilot case study conducted at a small Canadian software development company; 2) the results of the focused representational analyses of BPMN and Qualigram notations; and 3) a survey with practitioners.

Table 2.4 Operation phase: development of BPM^+

Development of BPM^+	Validation	Analysis
Development of the *a priori* version of BPM^+	The *a priori* version of BPM^+ is evaluated by conducting: 1) a case study with a Canadian company willing to participate in the project. The perceptions of the participants of the company are recorded; as well as the own reflections of the research team. The case study is conducted following an action research methodology. The focus of the case study is to evaluate the BPM^+ levels of abstraction; 2) a survey with practitioners with experience in BPM and SRE. The survey is administered to the participants following a semi-supervised format. The questionnaire of the survey is based on a set of propositions derived from the pilot case study and from the representational analysis.	• Reviewed version of the scope and content of BPM^+ levels of abstraction. • Preferences of BPM notations according to the various types of stakeholders. • Propositions to improve the expressiveness of BPM notations for SRE. • Propositions to be tested with the survey. • Publications
Development of a reviewed and validated version of BPM^+	The reviewed version of BPM^+ is validated by a case study with a Canadian company willing to participate in the project. The perceptions of various participants of the company, including its top executives, are recorded. The case study is conducted following an action research methodology.	• Validated version of the scope and content of BPM^+ levels of abstraction. • Validated preferences of the BPM notations used in this research. • Enhancement proposals that are communicated to Qualigram developers. • Publications

The pilot case study helps to verify the scope and the content of the top and intermediate levels of abstraction of BPM^+. The representational analyses help to identify the specific elements that should be modeled in accordance to the needs presented by software engineers and business analysts when performing software requirements elicitation activities. The focused representational analyses are performed using the SWEBOK (Abran *et al.*, 2004) and the BABOK (IIBA, 2009) as the references for identifying the relevant concepts for software requirements elicitation. The survey is conducted following a semi-supervised format: that is, it is conducted as a workshop. The participants of the survey are practitioners with experience in BPM and in software requirements elicitation. The results of the survey further evaluate the scope and content of the levels of abstraction, and the modeling and BPM notation preferences according to the various types of stakeholders. The survey also helps to test a set of propositions derived from the focused representational analyses regarding the need to improve BPM notations to represent the relevant elements for software requirements elicitation. As a result a reviewed version of BPM^+ is developed.

The reviewed version of BPM^+ maintains the three levels of abstraction from the *a priori* version. However, in the reviewed version of BPM^+ the scope and content of each level of abstraction have been refined. In addition, a preliminary set of target types of stakeholders for each level of abstraction and the preferences of BPM notations for each type of stakeholder have been added. To validate the reviewed version of BPM^+ a second case study is conducted at a Canadian company. This second case study aims at further evaluating the three levels of abstraction, their scope and content, and the modeling preferences according to the various types of stakeholders involved in the project. As a result a reviewed and validated version of BPM^+ is developed, and a set of propositions for improving BPM notations are formulated.

A summary of these two first parts of the operation phase is presented in Table 2.4. The details of the design of each of the case studies are presented in section 2.2; the design of the representational analyses is presented in section 2.3; and the design of the survey is presented in section 2.4.

Table 2.5 Operation phase: development of FSM procedure

Development of FSM procedure	Validation	Analysis
Development of a procedure to measure the functional size of a software application from its BPM in accordance to the COSMIC FSM method.	• The procedure guidelines for the business application software domain are evaluated by comparing the results obtained using the FSM procedure to those obtained by the "C-Registration system" case study (GELOG-ETS, 2008). • The procedure guidelines for the real-time software domain are evaluated by comparing the results obtained using the FSM procedure to those obtained in the "Rice cooker" case study (Lavazza and Bianco, 2009). • The guidelines of the procedure are generalized by comparing the results obtained using Qualigram notation to those obtained using BPMN. A set of notation-independent guidelines are developed.	• Modeling guidelines and mapping rules for Qualigram notation (business application and real-time software domains). • Modeling guidelines and mapping rules for BPMN notation (business application software domain). • Notation-independent guidelines (business application software domain). • Publications of the results.

Finally, the development of a procedure to measure the functional size of a software application from its business process models in accordance with the COSMIC FSM method covers both the business application software domain and the real-time software domain. However the emphasis is on the business application software domain. In order to be able to use a business process model for FSM purposes it is necessary to model the business processes in accordance to some specific guidelines. Therefore, for each of the two software domains, besides developing a set of mapping rules between the BPM notation and the COSMIC FSM method, it is necessary to also develop a set of BPM guidelines. For the business application software domain the procedure proposes a set of BPM guidelines for the

Qualigram notation, as well as the corresponding mapping rules. For this domain the procedure also proposes a set of BPM guidelines for BPMN, as well as it corresponding mapping rules. For the real-time domain the procedure is studied only with the Qualigram notation. In order to generalize the results obtained for the business application software domain, a set of notation-independent BPM guidelines are proposed. The results obtained after using each set of guidelines are evaluated by comparing them to the results obtained by well known COSMIC reference case studies.

A summary of the operation phase for the development of the FSM from BPM procedure in is presented in Table 2.5. The details of the design of the methodology used for developing this procedure are presented in section 2.5.

2.1.4 Interpretation phase

Finally, an interpretation based on the analysis of all the findings and feedbacks is performed (see Table 2.6). The BPM^+ approach proposed aims at contributing to the efforts to close the gap between the business process representations demanded by various groups of stakeholders by offering a formal procedure to identify the specific set of BPM constructs that satisfies the needs of a stakeholder when performing a specific task. BPM^+ also identifies the appropriate level of abstraction of the information required by the various groups of stakeholders, and allows representing these levels of abstraction in a consistent and structured way. The FSM procedure proposed is a novel approach to measure the functional size of software from the business process models that represent the business processes supported (or that might be supported) by that software. At this phase, each of the research sub-questions is reviewed, the findings are documented, and future work is identified.

2.2 Design of the case studies

To increase the validity of the results, both case studies reported in this research have been designed and conducted following many of the principles recommended by Yin (2002) as

Table 2.6 Interpretation phase

Context	Extrapolation	Future Work
• BPM$^+$ contributes to the efforts to close the gap between management and IT when modeling business processes. • BPM$^+$ identifies the appropriate levels of abstraction for representing the modeling needs required by the various stakeholders. • BPM$^+$ allows representing in a consistent and structured way the modeling needs required by the various stakeholders. • The FSM from BPM procedure developed in this research constitutes a novel approach to measure the functional size of software that supports or might support business processes.	• At least two Canadian organizations and a group of practitioners with experience in BPM and software requirements elicitation collaborated with the project. • The participant organizations found useful for their interests the BPM approach proposed (BPM$^+$). • Several ÉTS master degree students have collaborated with the project. • The FSM from BPM procedure proposed in this research has been well received by an international community of FSM experts. • The notation-independent guidelines are applied by the COSMIC community to other BPM notations.	• Further publications of the results in journals. • Test BPM$^+$ with other BPM perspectives (i.e. other target stakeholders, other purposes of modeling). • Replicate the survey with experts on a world-wide basis. • Conduct further case studies both for the BPM$^+$ approach and for the FSM from BPM procedure. • Test the notation-independent guidelines of the FSM from BPM procedure with other BPM notations.

well as those recommended by Runeson and Höster (2009). Based on those recommendations a protocol for each case study was elaborated, as described next.

Three researchers (the author of this thesis and two ÉTS master degree students) were involved in performing the two case studies reported in this thesis. Two research groups of two researchers were formed. One of the researchers (i.e. the author of this thesis) participated in both groups. Each of the two research groups worked independently on only one of the two case studies.

Both case studies followed an action research methodology (Baskerville and Myers, 2004): that is, members of the research groups collaborated in the BPM projects together with the members of the participant companies. Due to the involvement of human beings, the execution of the case studies required the CÉR approval within the ethics domain. The first case study was included as part of the methodology required for elaborating the research propositions to be tested in the survey that was conducted as part of the SYS869 course in 2011 summer (See Appendix I). The CÉR representative issued a statement indicating that none of the research activities that were presented as part of the SYS869 course as a supervised academic activity needed an ethics certificate (See Appendix II). The second case study research protocol was presented to CÉR for its evaluation within the ethics domain (See Appendix III), and the CÉR issued its corresponding approval for all the case study activities and results described in this thesis (See Appendix IV).

The action research methodology aims at acquiring new knowledge but providing, at the same time, value to the participant company (Sjoberg, Dyba and Jorgensen, 2007). The action research approach has been extensively used for information systems research (Baskerville and Myers, 2004; Davison, Martinsons and Kock, 2004), and it is considered a valid case study methodology for empirical software engineering research (Runeson and Höster, 2009; Sjoberg, Dyba and Jorgensen, 2007). The guidelines proposed by Runeson and Höst (2009) for “reporting case study research in software engineering” have been considered to report both case studies conducted as part of this thesis.

Both case studies were conducted following the principles of canonical action research (Davison, Martinsons and Kock, 2004): 1) a verbal agreement between the research group and the participant company is established; 2) a theoretical framework is used as a basis for the research process; 3) a iterative model governs the research process; 4) at each iteration the outcomes are analyzed to learn from them; and 5) actions are taken based on the interpretation of the outcomes.

For each case study, during a first meeting-activity: the main contacts of the participant company were introduced to the research group, the methodology proposed to conduct the case study was discussed, important issues regarding the organizational culture of the participant company were discussed, and the access to the sources of information was demanded and granted. The theoretical framework for the pilot case study was built based on the results of the literature review. For the second case study the framework was improved by the results of: 1) the pilot case study; 2) the focused representational analyses; and 3) the survey of practitioners with experience.

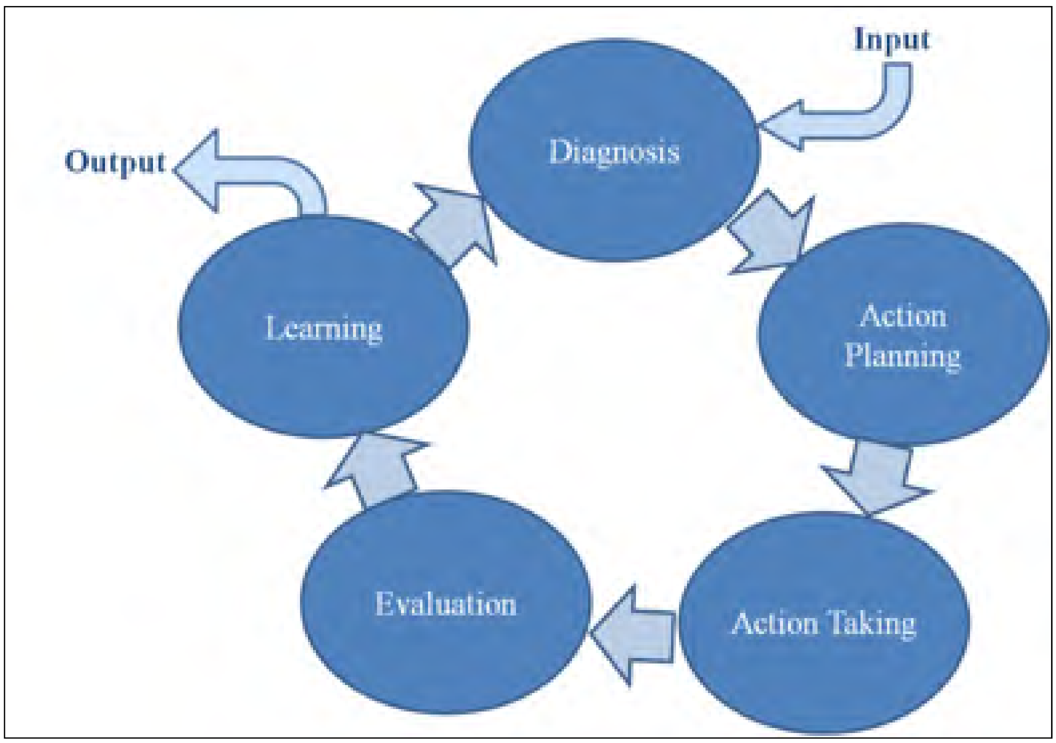

Figure 2.1 Action research: cyclical process model
Adapted from Davison, Martinsons and Kock (2004, p. 72), Copyright © 2004 by Blackwell Publishing Ltd, with permission from John Wiley and Sons

The iterative model mentioned in the third principle of canonical action research corresponds to the cyclical process model proposed by Susmand and Evered (1978) (refer to Figure 2.1). During the diagnosis phase the problem is understood, and next during the action planning phase a work plan to tackle the problem is agreed which is executed during the action taking phase. The goodness of the actions executed is assessed during the evaluation phase, and the results are interpreted to improve the BPM$^+$ approach.

The research objectives of the pilot case study are: 1) to test the strategic and tactical levels of abstraction of the *a priori* version of BPM$^+$; and 2) to assess the use of Qualigram and BPMN notations in representing business processes at these two levels of abstraction. The research objectives of the second case study are: 1) to empirically evaluate the refined version of the BPM$^+$ approach in a real BPM initiative; and 2) to empirically compare Qualigram and BPMN notations in terms of expressiveness and comprehension within a BPM initiative in industry. Based on these objectives, a set of research questions were formulated for each case study in order to guide their execution.

The unit of analysis for both case studies is the process of modeling a set of business processes of the participant company, business processes selected by mutual agreement at the beginning of the first iteration of the cyclical process model (See Figure 2.1). The unit of analysis includes both the evaluation of the corresponding BPM$^+$ version, and the assessment of the two BPM notations selected.

Evidence was collected through the analysis of: 1) existing archival data from the participant companies (e.g. documents, web site, etc.); 2) discussions and feedbacks from the members of the participant companies; 3) observation of how the work is performed within the participant companies; and 4) observation of the modeling process as an "observing participant". After the first meeting-activity a list of specific expected sources of evidence was elaborated for each phase of each case study. All relevant data were recorded and maintained (Runeson and Höster, 2009). The data collection and their analysis were iterative. The knowledge that emerged from the iterative process was documented. The members of the participant companies were allowed to review the results and were iteratively informed of the findings. Insights from the theoretical framework were used to contribute to the analysis of the data and their interpretation for further iterations.

Four types of meeting-activities were identified for conducting the case studies: discussion meetings, interview meetings, evaluation meetings, and research-group meetings. Data were collected during all of these types of meeting-activities. A calendar for all the meeting-activities required for each of the phases (i.e. diagnosis, action planning, action taking, evaluation, and learning) of each case study was prepared after the diagnosis phase of the first iteration. The calendar included the estimated amount of time required for each meeting, as well as its expected participants. The calendar was refined during the action planning phase. A description of each type of meeting-activity was provided to the members of each research group.

When necessary, research instruments for guiding the various meeting-activities were elaborated. For instance, interview guidelines were prepared for some of the interview meetings. Each research group created a structured file of all the data collected from its case study, including the written notes, documentation gathered from the participant company, project documentation, the various versions of the business process models, etc. All this information is what constitutes the case study database. The data collected was anonymized when it was considered necessary to protect the confidentiality of the participants. The case studies databases are kept under the supervision of the director of this thesis, and only the research groups have access to them.

Four main types of threats to the validity of the case studies may be identified: 1) construct validity; 2) internal validity; 3) external validity; and 4) reliability (Runeson and Höster, 2009; Yin, 2002). Construct validity was improved using multiple sources of evidence (e.g. archival data, opinions from the participants, observations) and having the members of the participant companies informed of the findings. In addition, for the second case study, many of the knowledge generated is supported by a chain of evidence that starts with the theoretical framework and that it is built up with the intermediate results of the pilot case study, the representational analyses, and the survey. That is, the interpretation of the results of the second case study considers a triangulation of evidences from various sources (i.e. theoretical framework, pilot case study, representational analysis and survey). Internal validity was improved both by having more than one member of the research team working in parallel within each group, and by addressing various rival explanations to the findings suggested by the various participants of each case study. External validity was improved by using the theoretical framework both as a basis for each case study and for the data analysis and interpretation. Moreover, most of the conclusions are supported by both case studies. Finally, reliability was improved by elaborating a research protocol and maintaining case studies databases.

One limitation of both case studies is related to their significance regarding the nature of the phenomenon being studied. Because the case studies were selected based on the availability and willingness of the participant organizations; none of the two case studies reports an experience in the use of BPM during a software requirements elicitation activity. However, the case studies were useful to evaluate the various concepts of BPM^+ with different types of stakeholders; and the obtained results have been complemented with the results of other research activities conducted as part of this thesis.

The details for each case study are described in sections 3.2 and 3.6.

2.3 Research methodology for the representational analyses

Wand and Weber point out that several language transformations might be required to go from the real world to its final representation (e.g. from the real world to a conceptual model, from the conceptual model to a high-level programming language, and from the high-level programming language to a machine language), and that each of these transformations involves a specific mapping (Wand and Weber, 1993; 1995). The research work reported in this thesis goes from a very specific domain from the real world (i.e. software requirements elicitation as set out in the SWEBOK and the BABOK) to a BPM notation, having the BWW representation model as an intermediate step (see Figure 2.2). Therefore, two sets of mappings are required: 1) the mappings between the specific real world domain and the

BWW representation model; and, 2) the mappings between the BWW representation model and the BPM notation.

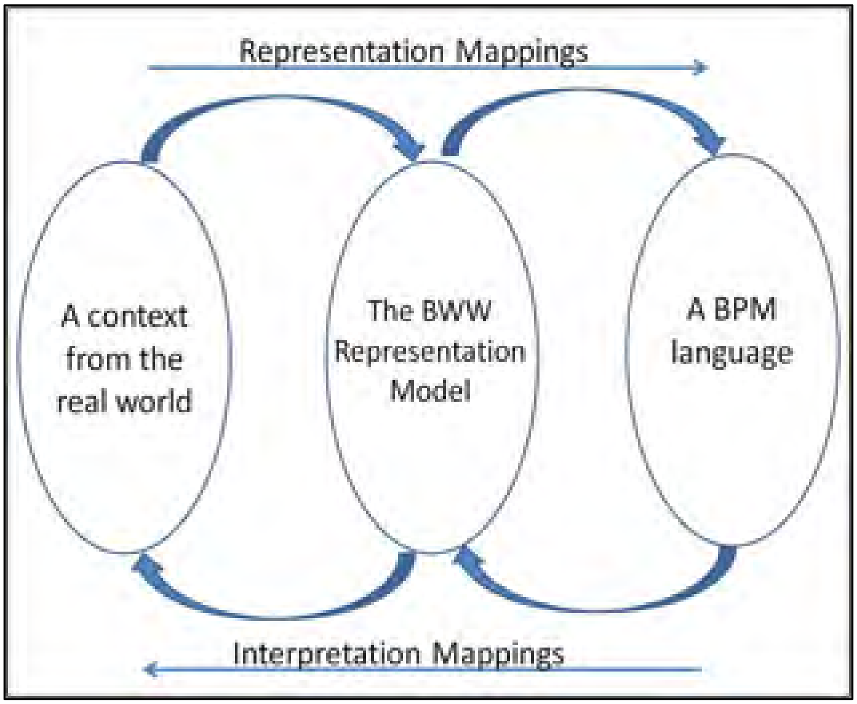

Figure 2.2 From a real-world domain to its representation in a BPM notation
Adapted from Wand and Weber (1993, pp. 220-223), Copyright © 2008 by John Wiley and Sons, with permission from John Wiley and Sons

Although popular, representational analysis has also been subject to criticism: lack of understandability, lack of objectivity, and lack of formalism (Gehlert and Esswein, 2007; Green *et al.*, 2006). To address some of these criticisms, Green *et al.* (2006) have proposed: 1) to produce the mappings with the aid of the meta-models of the ontology and the BPM notations being compared; 2) to involve more than one researcher in the mapping process, where each researcher would produce a draft of the mapping; and 3) to require multiple iterations leading to a consensus of the mapping results among the various researchers. In addition, Gehlert and Esswein (2007) have proposed to address some of these weaknesses by: 1) performing language comparisons based on the same set of ontological concepts; 2) specifying the version of the BPM notations being compared; 3) being explicit about whether the result of a mapping is an equivalence, a similarity, or a difference; and 4) providing the induced criteria to infer a similarity.

All these improvement recommendations have been taken into consideration and included in our research methodology. To improve understandability: 1) the mappings were produced with the aid of the meta-model of the BWW representation model (Rosemann and Green, 2002); 2) the meta-model of the Qualigram notation was produced and used during the mappings; and 3) it was not necessary to produce the meta-model of BPMN, as the results of the BPMN mappings performed by Recker *et al.* (2005; 2006) were already available for this thesis. To increase the objectivity in the results: 1) two researchers were involved in each of the mapping activities; 2) each researcher produced a draft of its mapping independently; and 3) multiple iterations followed by reviews were performed to reach a consensus on the mapping results among the various researchers. To increase formality in the mapping processes: 1) the BPM notations were compared based on the same set of the BWW representation model concepts; 2) the versions of the BPM notations and the representation model being compared are specified; 3) all the results of the mappings used or performed during this research constitute either an equivalence or a difference between the constructs being compared; and, finally, 4) as a consequence of the last point, no criteria are provided to infer a similarity for any of the mapping results.

Based on the aforementioned considerations, a research methodology for performing the focused representational analyses was designed. The SWEBOK (Abran *et al.*, 2004) and the BABOK (IIBA, 2009) were selected as the key references containing generally accepted knowledge of the software requirements elicitation domain. Figure 2.3 depicts the methodology for the representational analyses having the SWEBOK as the key reference for determining the relevant concepts of the software requirements elicitation domain.

The versions of BPMN and the BWW representation model used in all the research activities presented here are the same versions used by the representational analysis performed by

Recker *et al.* (2005; 2006)[16]. Three researchers (the author of this thesis and two ÉTS master degree students) were involved in performing the activities described in Figure 2.3. Two research groups of two researchers were formed. One of the researchers (i.e. the author of this thesis) participated in both groups. This allowed the research team to provide continuity in the research work, and at the same time reduced subjectivity in the analysis of the documents, elaboration of the mappings, and interpretation of the results. The first research group worked on the SWEBOK-BWW mappings (activity A1 in Figure 2.3), and the second one worked on the Qualigram-BWW mappings (activities A2 and A3 in Figure 2.3).

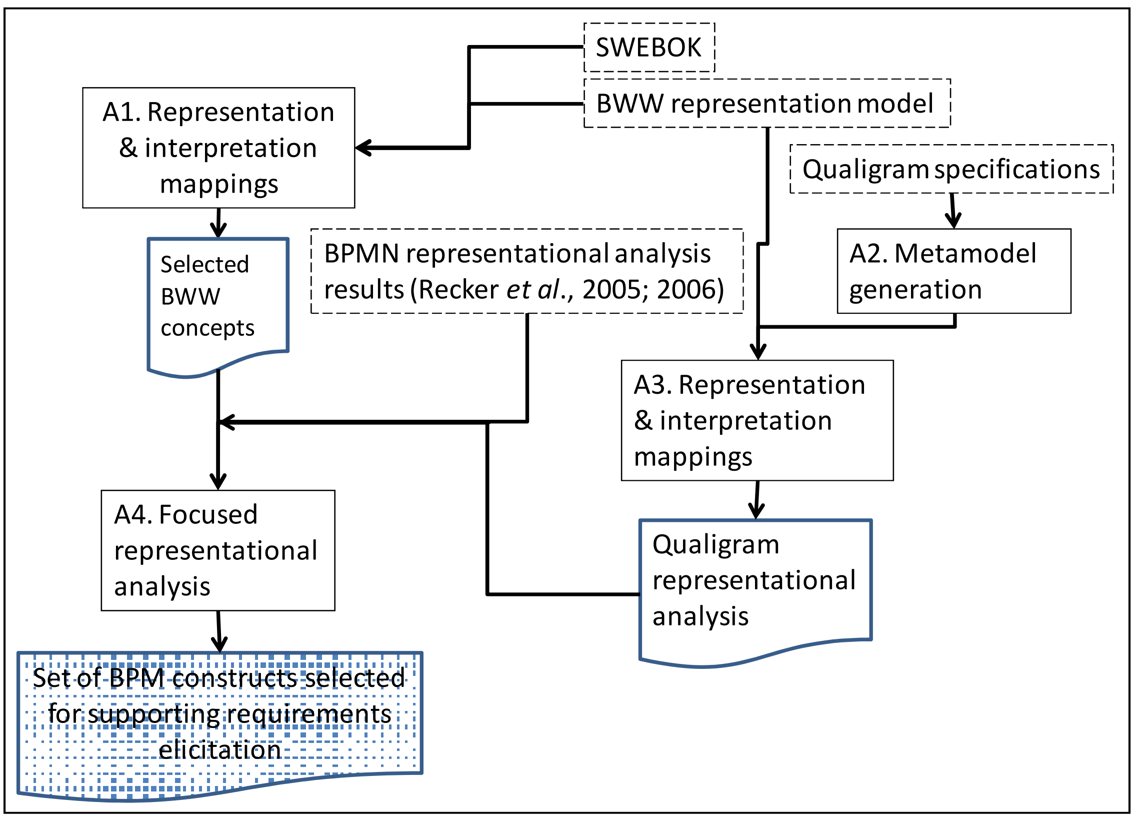

Figure 2.3 Methodology for the SWEBOK focused representational analyses

[16] Recker *et al.* conducted the representational analysis of BPMN version 1.0. Prezel, Gašević and Milanović (2010) have updated the representational analysis to BPMN version 2.0. The results of the representational analysis of BPMN 2.0 do not present an impact in the representational analyses executed as part of this thesis (see subsection 3.3.1.4).

The first research activity (activity A1 in Figure 2.3) consisted of performing the representation and interpretation mappings of the BWW representation model using the SWEBOK (Abran *et al.*, 2004) as the key reference for the mappings. For achieving this, the corresponding research group was asked to review its key reference (i.e. the SWEBOK) to identify the relevant concepts associated with requirements elicitation. Most of the concepts were found in the SWEBOK chapter devoted to the software requirements KA. However, the research group took the precaution of reviewing carefully all the SWEBOK text to ensure that the list of relevant concepts would be comprehensive. The research group required several review iterations to reach a consensus on its final candidate list of relevant concepts. Then, the same research group was asked to classify each item on its list of concepts into one of two sets: 1) those that should be represented in a business process model; and 2) those that should be supported by a modeling tool (e.g. software requirements traceability). Again, several iterations were required to reach a consensus on a common classification. From the two resulting sets of concepts (i.e. those attributed to a business process model and those attributed to a modeling tool), the set of concepts that should be represented in a business process model was taken as the reference for the next step. The final step consisted on mapping the BWW representation model with the set of concepts the research group selected. Once again, several iterations were needed to reach a consensus. The resulting mappings allowed us to identify a subset of BWW concepts which, according to the SWEBOK, represents concepts that are relevant to the domain of software requirements elicitation.

The specifications of the Qualigram notation are textual (Berger and Guillard, 2000). Therefore, the second research activity (refer to activity A2 in Figure 2.3) consisted on the elaboration of a meta-model of Qualigram notation (see Appendix V). The meta-model was validated with the members of the technical staff of Globalliance who developed the Qualigram toolset that implements Qualigram's grammar.

The third research activity (activity A3 in Figure 2.3) consisted of performing the representation and interpretation mappings of Qualigram notation. These mappings allow assessing the capability of the Qualigram modeling constructs to represent the concepts of the BWW representation model. The results of the Qualigram representation mappings were subjected to a final review after the mapping results from activity A1 had been obtained. The final results are presented in Appendix VI.

It was not necessary to perform the mappings of BPMN, as the results of the BPMN representational analysis performed by Recker *et al.* were already available for this thesis (Recker *et al.*, 2005; 2006). The results of the representational analysis performed by Recker *et al.* are presented in Appendix VII.

The last step (activity A4 in Figure 2.3) was to compare the two representational analyses, Qualigram and BPMN, with the BWW subset of concepts found during activity A1. These comparisons enabled to answer the following question: How well do Qualigram and BPMN represent the specific software requirements elicitation concepts in a software engineering context?

A similar methodology to the one depicted in Figure 2.3 was followed for the focused representational analyses based on the BABOK, the main difference being that the BABOK was the key reference for the process rather than the SWEBOK. Therefore, the BWW concepts selected in this case were those that according to the BABOK represent concepts that are relevant to software requirements elicitation. For the BABOK-BWW mappings a new research group of two researchers was formed. Again, one of the researchers was the author of this thesis. The second researcher was an ÉTS master degree student who did not participate during the research activities based on the SWEBOK.

2.4 Research design of the survey

To increase the validity of the results, the survey has been designed and conducted following many of the principles recommended by Kitchenham and Pfleeger (2008) as well as those recommended by Salant and Dillman (1994). Based on those recommendations a protocol was elaborated as described next.

The design objective of this survey is to test a set of research propositions formulated based on the findings from both the pilot case study and the focused representational analyses.

A questionnaire (See Appendix VIII) was designed based on the structure proposed by Davies, Rosemann and Green (2004) (see Figure 2.4). The structure of this questionnaire has been used and validated by previous studies conducted by other authors (Davies, Rosemann and Green, 2004; Recker *et al.*, 2005; 2006; Schlauderer and Overhage, 2011). The questionnaire was pre-tested three times with the help of IT professionals with more than 5 years of experience in software development projects. The pre-tests were planned to: 1) improve the quality of the questions; and 2) ensure an appropriate timeframe for answering the questionnaire. All the questions were of the closed type to facilitate their answering and coding.

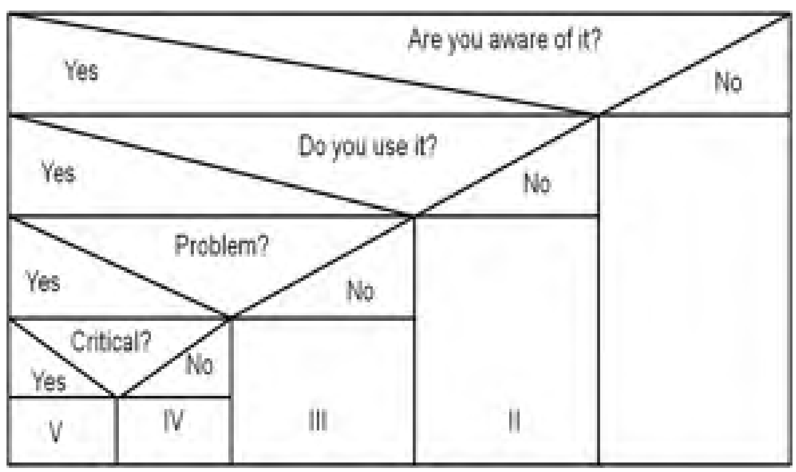

Figure 2.4 Questionnaire structure and severity levels
Based on Davies, Rosemann and Green (2004), Copyright © 2004 by Davies, Rosemann and Green, with permission from Islay Davies

The survey was planned to be conducted following a semi-supervised format: that is, as a workshop within the framework of the 2nd International Symposium in Software Engineering Management held in 2011 in Montreal, Canada. The total time planned for the execution of this workshop-survey was 50 minutes distributed as follows: 1) presentation of the motivation, background and research objective of the survey-workshop (10 minutes); 2) description of the activity and its instructions (5 minutes); and 3) execution of the survey: that is, filling the questionnaires (35 minutes). The first two activities were executed by the author of this thesis, who was available on site during all the workshop-survey to answer any question from the participants.

The target audience of the survey is practitioners with experience in BPM and Software Requirements Elicitation (SRE). The practitioners might be software engineers, business analysts, or professionals from related backgrounds. Knowledge or experience with BPMN is an optional, but desirable, attribute of the target audience.

Because the target audience is very specialized, the true population is difficult to determine. Therefore, a non-probabilistic sample (i.e. purposive sampling) was chosen for this survey (Kitchenham and Pfleeger, 2008; Salant and Dillman, 1994). To ensure a representative number of participants, besides the regular promotion channels used by the international symposium, personal invitations to participate in this survey were sent to members of the Montreal chapter of the International Institute of Business Analysis (IIBA), and to the members of the social network of Montreal Business Analysts (See Appendix IX).

The participants were volunteers and had the opportunity to withdraw from the survey at any time. Nineteen (19) participants were present at the beginning of the survey-workshop; seventeen (17) of them finished the activity and returned their answers. Similar previous studies have been conducted by other authors (Davies, Rosemann and Green, 2004; Green, Rosemann and Ipswich, 2001; Green and Rosemann, 2002; Recker *et al.*, 2005; 2006) with groups of 4 to 21 participants (refer to Table 2.7). The actual profile of the participants is described in subsection 3.4.2. The survey research protocol was presented to the CÉR for its evaluation within the ethics domain (See Appendices I and II).

A discussion of the validity threats of this survey follows. To increase construct validity, the questionnaire design followed well accepted guidelines found in the literature, and used a structure (refer to Figure 2.4) that has been already validated in similar studies. The

Table 2.7 Previous similar empirical research work

Reference	**(Green, Rosemann and Ipswich, 2001)**	**(Green and Rosemann, 2002)**	**(Davies, Rosemann and Green, 2004)**	**(Recker *et al.*, 2005)**	**(Recker *et al.*, 2006)**
Method-Instrument	Pilot survey	Pilot survey & Structured interview	Interview	Semi-structured interview	Semi-structured interview
Target Population	16 graduate students	12 graduate students (pilot) & 4 practitioners from 2 organizations (interview)	21 modelers from 8 organizations	11 participants with different levels of experience from 6 organizations	19 participants with different levels of experience from 3 organizations

formulation of the questions was based on the propositions to be tested. Moreover, the questionnaire was pre-tested and discussed three times with professionals that fit the target-audience profile.

Regarding internal validity, a great effort was made to ensure participants' experience both in BPM and SRE; even though they did not present the same levels of experience, most of them confirmed to have experience in both fields of knowledge (See subsection 3.4.2). One Ph.D. researcher, who might be acceptable as a proxy participant since he presents a similar profile to the target professionals, participated in the survey. In addition, since the survey was semi-supervised it allowed to clear doubts from the participants.

Regarding external validity, the main threat is the sample size used in this research. However, the results are strengthened in terms of generalization by the fact that the propositions being tested were derived from previous theoretical and empirical research work (i.e. representational analyses and pilot case study). Therefore, if a proposition is supported, the results of the survey converge with the results of our literature review and our pilot case study, establishing in this way a chain of evidence that supports the final results (Runeson and Höster, 2009; Yin, 2002). In addition, the sample size of this survey is similar to the sample size of previous studies (refer to Table 2.7).

Finally, to increase reliability a survey protocol was elaborated (See Appendix I), the questionnaire was retested with one of the professionals who volunteered for the pre-tests, and closed questions were preferred to reduce the bias of the researcher when coding the responses.

2.5 Methodology for developing the FSM from BPM procedure

The methodology used in this part of the research is twofold: 1) The steps to be followed for the business application software domain, as explained in subsection 2.5.1; and 2) The steps to be followed for the real-time software domain, as explained in subsection 2.5.2.

2.5.1 Business application software domain

Figure 2.5 depicts the methodology for the business application software domain. The same methodology is followed for each of the selected BPM notations (i.e. Qualigram and BPMN). To test the feasibility of the proposed approach, the version of the C-Registration System case study dated February 23, 2008, and published by the COSMIC Group is used (GELOG-ETS, 2008). Based on the definitions of the various modeling constructs offered by the modeling notation, and the definitions of the various COSMIC concepts, a mapping table between the COSMIC concepts and the modeling constructs is generated (first result of activity A1 in Figure 2.5). Also, as a result of the comparison, a set of specific modeling guidelines is identified (second result of activity A1 in Figure 2.5) to allow the business process models to be used for FSM. The C-Registration System is modeled following these modeling guidelines (activity A2 in Figure 2.5). The mapping rules and the business process models generated are used to measure the functional size of the software application (activity A3 in Figure 2.5). Finally, the measurement results are compared with those presented in the C-Registration System case study (activity A4 in Figure 2.5).

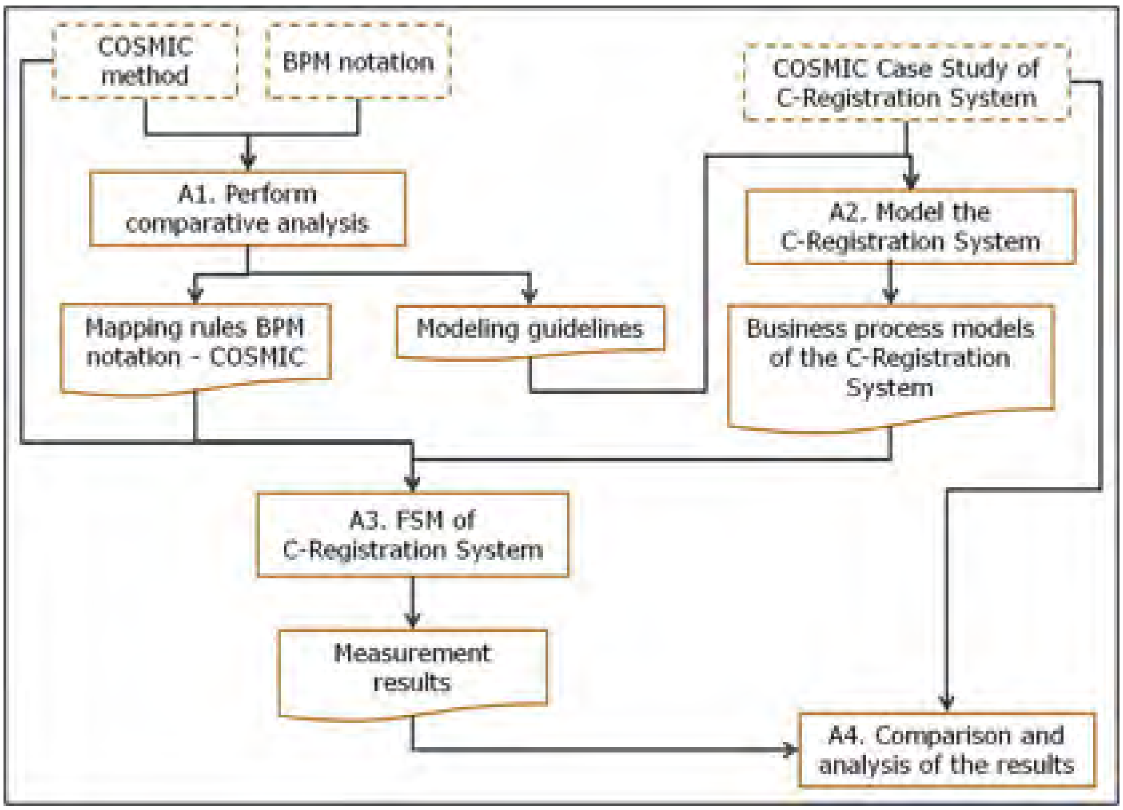

Figure 2.5 Methodology for the business application software domain
Adapted from Monsalve, Abran and April (2011, p. 317),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

In addition, the results obtained using each of the BPM notations (i.e. Qualigram and BPMN) are analytically compared, in order to generate a set of notation-independent BP modeling guidelines for FSM purposes.

2.5.2 Real-time software domain

The methodology for the real-time software domain is similar to that depicted in Figure 2.5. The main differences are the case study to be analyzed and the way the feasibility of the proposed approach is tested. The May 22, 2008, version of the Rice Cooker case study (COSMIC, 2008) is used to illustrate the approach. To verify the value of the approach, the results obtained are compared with those obtained by Lavazza and Bianco (2009) for the same case study. From the two selected BPM notations (i.e. BPMN and Qualigram), only Qualigram is used for analyzing the real-time software domain. Since BPM notations are typically intended to model the business processes of an organization, and the case study corresponds to a real-time software controller, it is very likely that some specific modeling guidelines for FSM purposes have to be derived. Finally, as in the previous case, a set of mapping rules is elaborated and used to measure the functional size of the software components of the Rice Cooker system. This methodology is depicted in Figure 2.6.

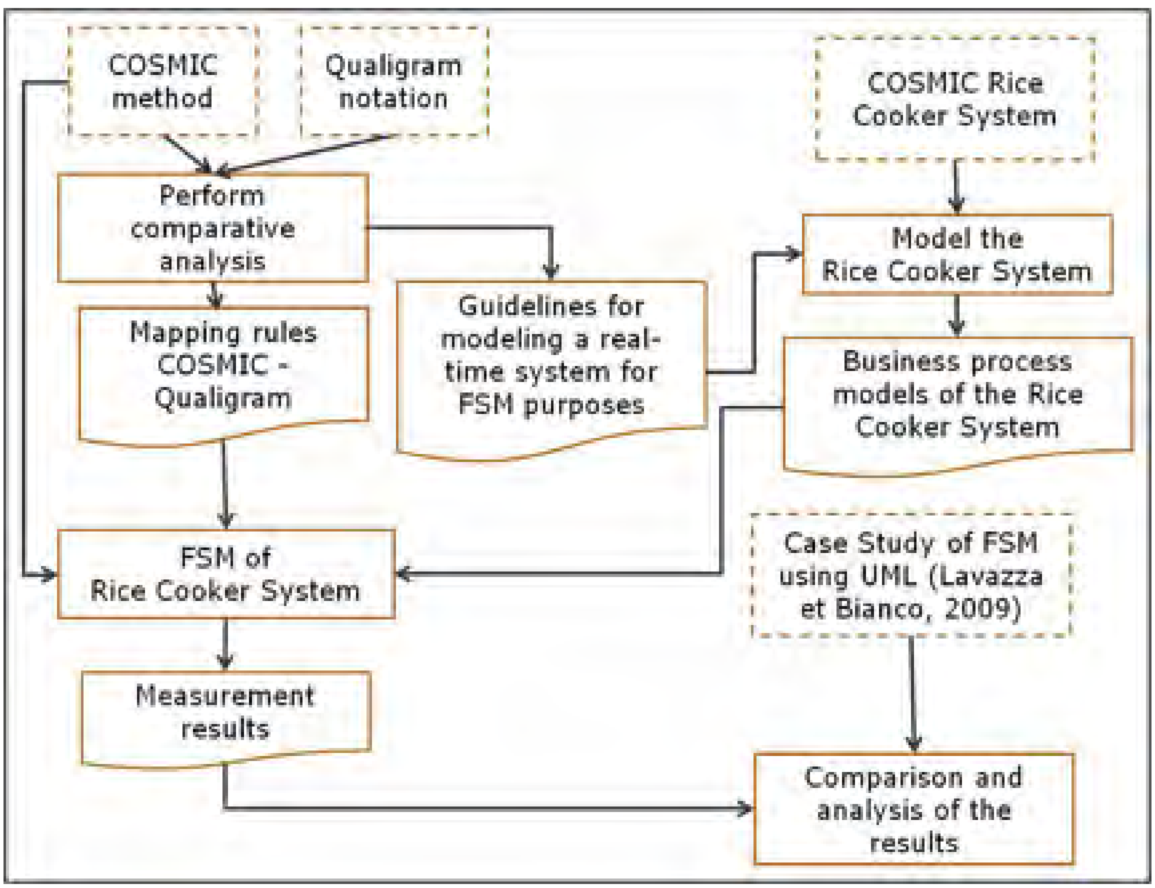

Figure 2.6 Methodology for the real-time software domain
Adapted from Monsalve, Abran and April (2011, p. 318),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

2.6 Summary of the research methodology

In summary, the research methodology used in this thesis is twofold: 1) The methodology followed for the development of the BPM$^+$ approach; and 2) The methodology followed for the development of the FSM from BPM procedure based on the COSMIC FSM method. The former is depicted in Figure 2.7, and the latter in Figure 2.8. Both figures represent the corresponding methodologies in terms of their inputs, research methods, deliverables and outcomes.

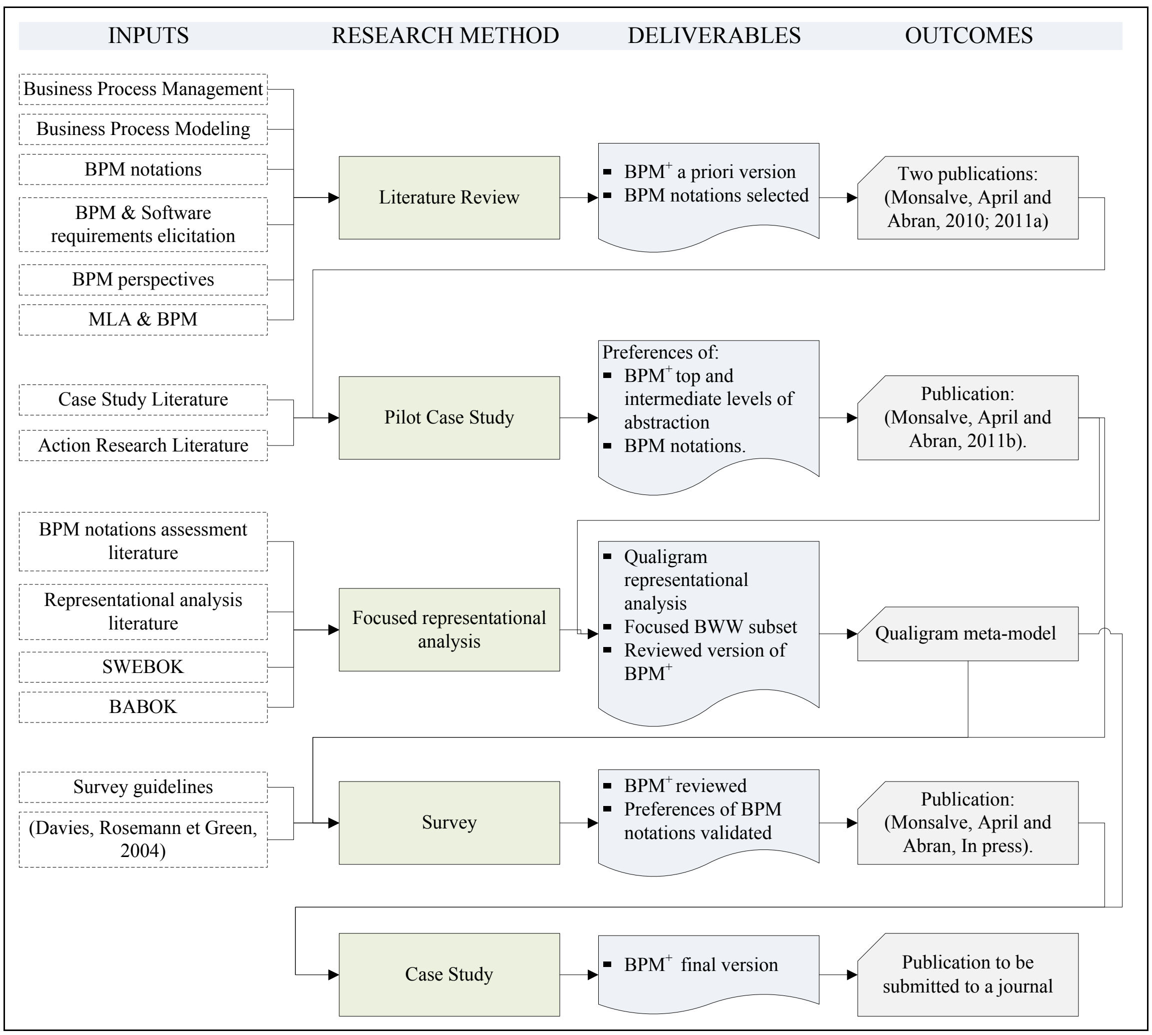

Figure 2.7 Research methodology summary: development of the BPM$^+$ approach

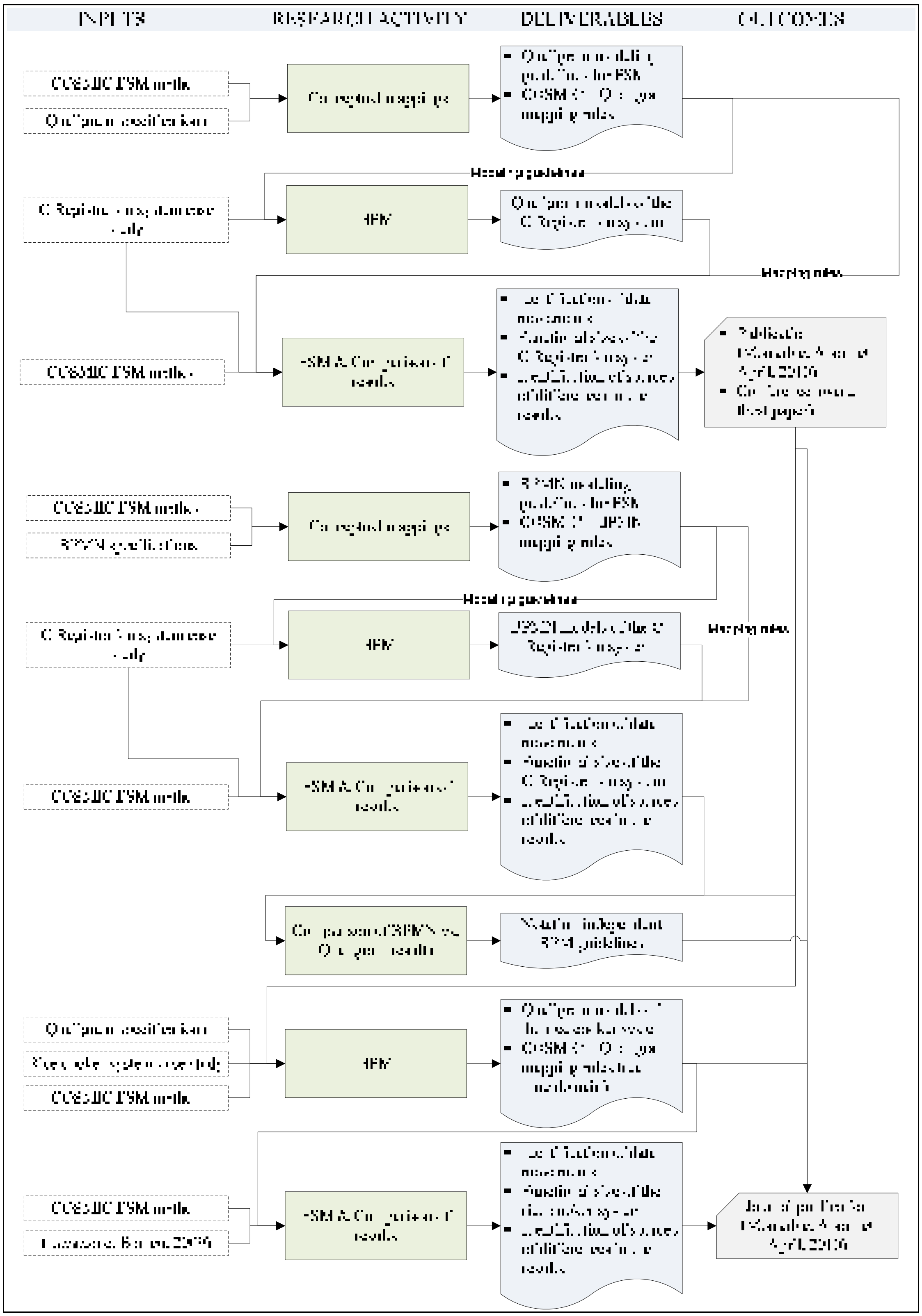

Figure 2.8 Research methodology summary: development of the FSM procedure

Finally, some limitations and validity threats of this research have been identified and described throughout this Chapter. An overview of these limitations and validity threats is presented in Table 2.8.

Table 2.8 Overview of the limitations and validity threats

Research method or research activity	Threat or limitation	Discussion	Section in the thesis
Case studies	Construct validity	Multiple sources of evidence were used, and the participants were informed of the findings.	2.2
	Internal validity	More than one researcher worked in parallel, and rival explanations were addressed.	
	External validity	A theoretical framework was used both as a basis for each case study and for the data analysis and interpretation.	
	Reliability	A research protocol was elaborated and case studies databases were maintained.	
Representational analysis	Lack of understandability	Qualigram's meta-model was developed. During the mappings processes this meta-model and BPMN's meta-model were used.	2.3
	Lack of objectivity	More than one researcher was involved in the mapping processes. Multiple iterations were needed to reach consensus among the various researchers.	
	Lack of formalism	The same set of ontological concepts and BPM notations were used during all the mappings. All the mapping results constitute either an equivalence or a difference between the concepts compared.	
Survey	Construct validity	The questionnaire design followed well accepted guidelines, and used a structure that has been validated in similar studies. The questionnaire was pre-tested.	2.4
	Internal validity	A great effort was made to ensure participants' experience both in BPM and SRE. In addition, since the survey was semi-supervised it allowed to clear doubts from the participants.	
	External validity	The generalization of the results is strengthened by the fact that the tested propositions were derived from previous research work. A chain of evidence that supports the results was established.	
	Reliability	A survey protocol was elaborated, the questionnaire was retested, and closed questions were preferred to reduce any coding bias.	
Development of FSM procedure.	Generalization	Notation-independent modeling guidelines should be tested with other popular BPM notations. Other measurement case studies should be conducted.	2.5

CHAPTER 3

BUILDING THE BPM$^+$ APPROACH

3.1 The BPM$^+$ *a priori* version

As described in the Introduction chapter of this thesis, one key factor for a successful BPM project is the active participation of all the stakeholders and sharing a common vision of business processes. This translates into modeling business processes that can be successfully shared among all the involved stakeholders. To achieve this, a typical BPM project will have to deal with different perspectives of the same business process (refer to section 1.4). These perspectives cannot be exclusive: they must complement each other, and all of them together should contain most of the valuable elements and information from the business process. If only a technical point of view is considered when modeling the business processes then it is likely that not all the stakeholders will find valuable the resulting business process models (Smith and Fingar, 2007; Van Nuffel and De Backer, 2012). Each of the perspectives should be seen as a layer of the same business process. In this way, each stakeholder can find helpful to use one or more of these perspectives at a given moment in time (refer to Figure 1.6).

From the literature we learned that the various groups of stakeholders might have conflicting requirements from a business process model, and that it is not possible to represent all the BPM perspectives into one single model (refer to section 1.4). For instance, if a business process model is going to be used as a source of information for developing software then it needs formality (Becker, Rosemann and von Uthmann, 2000). The SWEBOK (Abran *et al.*, 2004, p. 2.3) mentions that "software requirements should be stated as clearly and as unambiguously as possible". However, there might be an evident conflict here, because the more formal the BPM notation is, the more difficult to understand it likely is, especially by non-IT stakeholders. To address these problems, many authors have argued that BPM at multiple levels of abstraction (MLA) helps to represent the appropriate information to be provided to various types of stakeholders (refer to section 1.5).

BPM^+ aims at providing the means for a consistent and structured way of modeling various business process perspectives. There are many modeling notations that have been proposed and used for modeling business processes (refer to subsection 1.3.2); therefore, the rationale behind the design of BPM^+ is not to design yet another BPM notation, but to develop a BPM approach that incorporates MLA, and that selects an appropriate set of BPM constructs for each of its levels of abstraction. The solution proposed by BPM^+ needs to be simple and should not significantly increase the complexity of the BPM notations selected, thereby allowing business process models to be easily understood by different stakeholders.

BPM^+ levels of abstraction cannot be designed and studied separately: they must be designed and studied concurrently, considering the needs and constraints of: each actor (i.e. individual), each unit or department (i.e. team or group), and the organization. The result of using BPM^+ within an organization should be a set of models that allows the understanding and the coordination of the activities performed by the various stakeholders of the organization (refer to subsection 1.5.1).

Traditionally BPM has been mainly focused around the behavioral aspects of a business process (refer to Table 1.3). By designing the MLA structure concurrently, BPM^+ adds other complementary aspects to the traditional BPM. For instance, a business process element represented at a high-level model should be able to be traced to an element represented at a lower-level model; and a low-level representation should be able to be traced back to a high-level representation. This traceability capability of BPM^+ helps to improve the "scope management" and the "change impact analysis" of its MLA approach (IIBA, 2009, p. 130). That is, it should be possible in BPM^+ to trace to a lower-level activity or dependency the impact caused by a change in a high-level business process representation.

As a consequence of the afore-discussed considerations, the design of BPM^+ is based on a focused ontological analysis (refer to section 2.3) complemented by the use of MLA in BPM (refer to section 1.5). It is argued here that following this design approach:

1. BPM^+ is not a solution that is only based on the "practical wisdom" of the authors "rather than on a scientific theory" (Recker *et al.*, 2007).
2. The BPM constructs proposed by BPM^+: a) are systematically identified based on an ontological analysis; and b) are necessary to represent the concepts from the real world that have been found to be relevant to be modeled (refer to section 2.3).
3. BPM^+ provides information and formality at the appropriate level of abstraction according to what is required by the different types of stakeholders.

We have already reviewed (refer to subsection 1.5.2) several approaches to incorporate a MLA structure in BPM. Anthony's model (refer to subsection 1.5.1) has been chosen as the basis of BPM^+ MLA structure because:

1. Anthony's model has influenced: the design of commercial BPM notations such as Qualigram (Berger and Guillard, 2000), and recent BPM research (Bhat and Deshmukh, 2005; Haque, Pawar and Barson, 2003).
2. The recommendations of the ISO for documenting business processes (ISO, 2010) reflect the three levels of activities found in organizations as described by Anthony's model.
3. Anthony's model has been used in the literature as a basis for the classification of the added value of IT investment in the organization (Shang and Seddon, 2002).

In addition, we observe that all BPM research proposals reviewed in subsection 1.5.2.3 recommend the use of three levels of abstraction, even though, depending on the author, the content of each level of abstraction varies from one proposal to another.

Therefore, BPM^+ is developed based on an abstraction hierarchy (see Figure 3.1) that includes three levels of abstraction: strategic, tactical, and operational. The design of each of these three levels of abstraction is inspired from Anthony's model (Anthony, 1965; Gorry and Morton, 1989), the ISO recommendations for documenting business processes (ISO, 2010), and Qualigram's pyramid (refer to Figure 1.3). Each abstraction level represents a particular detail of information such as:

Figure 3.1 BPM$^+$ levels of abstraction

- The strategic level describes the core processes, goals and policies of the organization, aiming at representing "why" the organization needs to perform the business process modeled.
- The tactical level deals with the attainment and efficient use of the resources of the organization: it describes the procedures, aiming at representing "what" activities are done in the organization as part of a business process and "who" in the organization is responsible for each of the activities.
- The operational level procures the efficient and effective execution of atomic tasks: it describes specific activities of the organization, aiming at representing "how" to perform each activity.

In BPM$^+$ a core process is composed of a set of procedures, a procedure is composed of a set of activities, and an activity is composed of a set of atomic tasks. When necessary, a process can be decomposed into sub-processes, and a procedure into sub-procedures. All the operational-level models are integrated through the tactical-level models, the tactical level-models are integrated through the strategic-level models, and the entire business process models are integrated through a main strategic-level model. These characteristics of the BPM$^+$ abstraction hierarchy are depicted in Figure 3.1.

The next subsections describe in more detail each of the three BPM$^+$ levels of abstraction (*a priori* version). For each level of abstraction, a description indicates the purpose or objective of the level of abstraction; then its scope is delimited and its vocabulary (i.e. modeling concepts) identified. Finally, some syntax considerations are discussed.

3.1.1 The BPM$^+$ strategic level of abstraction

At the top level of abstraction (i.e. strategic level) BPM$^+$ models a high-level view of the core processes of the organization and their main relationships; it also represents the external stakeholders who are relevant to the organization (e.g. customers and providers). The strategic level of abstraction also serves to communicate the goals of the organization, thereby answering the question “why”.

Scope

The strategic level of abstraction describes the processes and sub-processes of the organization. It includes the representation of procedures to better describe the processes or sub-processes; however, the procedures are described at the intermediate level of abstraction (i.e. tactical level).

Modeling concepts and semantics considerations

To identify the modeling concepts we follow, where possible, the classification depicted in Figure 1.4:

- Actions: processes, sub-processes, procedures;
- Entities: relevant external stakeholders;

- Information (relationships or dependencies): relationships between actions, relationships between actions and entities; and
- Tools: none.

In addition, at the top level of abstraction there is the need to represent the goals of the core business process.

Table 3.1 shows the recommended representations of the strategic-level BPM^+ modeling concepts in both Qualigram and in BPMN notations. Notice that in the case of BPMN there is more than one possible modeling construct to represent the following BPM^+ modeling concepts: sub-process, procedure, and all types of relationships. Therefore, a goal of the empirical activities planned as part of this thesis' research methodology is to assess the best way to represent these concepts in BPMN notation.

Table 3.1 Representation of strategic-level BPM^+ modeling concepts in Qualigram and BPMN notations

BPM^+ modeling concepts	**Qualigram**	**BPMN**
Process.	Process.	Pool.
Sub-process.	Sub-process.	Lane, collapsed sub-process.
Procedure.	Procedure.	Collapsed sub-process, task.
External stakeholder.	External entity.	Pool.
Relationship between actions.	Information.	Sequence flow, message flow.
Relationship between actions and entities.		
Goal.	Performance indicator.	Not available; use text annotation attached with an association.

Syntax considerations

Any BPM^+ top-level model is elaborated aiming at the description of a core process of the organization. When the modeler considers it necessary, a core process can be decomposed into one or more sub-processes. Therefore, a procedure might be represented either as part of a process or as part of a sub-process. All the procedures that are intended to be further described at BPM$^+$ intermediate level of abstraction (i.e. tactical level) must be depicted at a BPM$^+$ top-level model.

Since there might be various possible perspectives to be modeled at the strategic level of abstraction, then it is possible to have more than one BPM$^+$ top-level model. For instance, when the organization has many core processes to be represented, the modeler might choose to elaborate several strategic-level models to represent the various relationships between the core processes; each strategic-level model representing the relationships focused on only one of the core processes at a time.

Qualigram offers four types of models at the strategic level of abstraction: that is, the macroscopic type of model, the relational type of model, the detailed type of model, and the transversal type of model (Berger and Guillard, 2000):

1. The macroscopic type of model depicts a general high-level view of the core business processes of the organization;
2. The relational type of model adds to the macroscopic type of model the interactions between the core processes and the interactions between the relevant external entities and the core processes;
3. The detailed type of model depicts a detailed view of each core business process represented at the macroscopic type of model, showing its main sub-processes, procedures and their relationships; and
4. The transversal type of model depicts how each of the core business processes depicted in a macroscopic type of model traverses the various organizational units or departments to achieve its goal.

From these four types of models, BPM$^+$ proposes to only use the macroscopic, the relational and the detailed types of models. The transversal type of model is discarded because the semantics and the syntax of this type of model conflicts with those of the models proposed by Qualigram at the intermediate level of abstraction (i.e. tactical level). This last consideration was reviewed with the members of the technical staff of Globalliance who developed the Qualigram toolset that implements Qualigram's grammar. They confirmed that for the same reasons explained above, the transversal type of model is rarely used among their Qualigram's customers.

Only the BPMN modeling constructs included in the "descriptive level of use" (i.e. Level 1) proposed by Silver (refer to subsection 1.5.2.2) should be used when a strategic-level model is elaborated using BPMN. Only the "private business process" and the "abstract process" types of BPMN models (refer to subsection 1.5.2.2) should be used. If BPMN lanes are used to represent BPM^+ sub-processes, then the use of BPMN collapsed sub-processes should be reserved for representing BPM^+ procedures. If BPMN collapsed sub-processes are used to represent BPM^+ sub-processes then BPMN tasks should be used to represent BPM^+ procedures.

3.1.2 The BPM^+ tactical level of abstraction

At the intermediate level of abstraction (i.e. tactical level), BPM^+ describes the procedures of the organization; depicting how the various roles and departments of the organization interact performing the various organizational activities, as well as the resources required for the execution of the procedures. That is, the tactical level models the workflow of the procedures of the organization, answering the questions "who" and "what". This level of abstraction should also identify the critical activities to achieve the goals of the organization, assigning specific objectives to those activities if it is considered necessary. Typically, the goals are translated into the satisfaction of the needs of the external stakeholders (e.g. customers) of the organization.

Scope

The tactical level of abstraction describes the procedures and sub-procedures of the organization. It includes the representation of activities to better describe the procedures or sub-procedures; however, the activities are described at the lowest level of abstraction (i.e. operational level).

Modeling concepts and semantics considerations

The modeling concepts to be represented at the tactical level of abstraction are:

- Actions: sub-procedures, activities;
- Entities: external stakeholders that interact with the procedure, roles (i.e. internal actors), departments (i.e. organizational units), owner or responsible of a procedure;
- Information (relationships or dependencies): relationships between actions, relationships between actions and entities; and
- Tools: physical tools (e.g. computers, software tools, machinery), documents (i.e. documents that are used or produced by the procedure).

In addition, at the tactical level of abstraction there is the need to represent:

- The objectives of the critical activities;
- The events that trigger each procedure;
- The events that indicate the end of the workflow of each procedure; and
- Control flow patterns (Russell *et al.*, 2006a): decisions, merges, splits, synchronizations.

Table 3.2 shows the recommended representations of the tactical-level BPM^+ modeling concepts both in Qualigram and in BPMN notations. Notice that in the case of Qualigram, there is more than one possible modeling construct to represent triggering and end events. Therefore, a goal of the empirical activities planned as part of this thesis' research methodology is to assess the best way to represent these concepts in Qualigram notation.

In addition, notice that in the case of BPMN, the relationships between actions and entities can be represented by two different modeling constructs: lane and message flow. We will explain the difference of use of these two modeling constructs during the "syntax considerations" part of this subsection.

Table 3.2 Representation of tactical-level BPM+ modeling concepts in Qualigram and BPMN notations

BPM+ modeling concepts	Qualigram	BPMN
Sub-procedure.	Sub-procedure.	Collapsed sub-process.
Activity.	Work instruction.	Task.
External stakeholder.	External role.	Pool.
Role.	Role.	Lane.
Department.	Unit.	Lane.
Procedure owner.	Responsibility.	Not available; use text annotation attached with an association.
Relationship between actions.	Information.	Sequence flow.
Relationship between actions and entities.	Swim-lane.	Lane, message flow.
Physical tool.	Physical tool.	Not available; use text annotation attached with an association.
Document.	Document.	Data object.
Objective.	Constraint indicator, performance indicator.	Not available; use text annotation attached with an association.
Triggering event.	Start event, up-stream action.	Start event.
End event.	End event, down-stream action.	End event.
Control flow pattern.	And operator, Or operator.	Gateway.

Syntax considerations

Any BPM+ tactical-level model is elaborated aiming at the description of the procedures of the organization. When the modeler considers it necessary a procedure can be decomposed into one or more sub-procedures. Therefore, activities might be represented either as part of a procedure or as part of a sub-procedure. All the activities that are intended to be further described at BPM+ lowest level of abstraction (i.e. operational level) must be depicted at a BPM+ tactical-level model.

Qualigram notation offers at the tactical level of abstraction other modeling constructs than those listed in Table 3.2: that is, the macro-instruction, the collaborative instruction, the target role, the source role, the control indicator and the corrective indicator. The a priori version of BPM^+ proposes not to use these additional modeling constructs offered by Qualigram notation. The various activities planned as part of this thesis' research methodology will assess the necessity or not to include these modeling constructs.

At BPM^+ tactical level of abstraction it is possible to use the BPMN modeling constructs included in the "descriptive" (i.e. Level 1) and the "analytical" (i.e. Level 2) levels of BPMN use, as proposed by Silver (refer to subsection 1.5.2.2). Only the "private business process" and the "collaboration process" types of BPMN models (refer to subsection 1.5.2.2) should be used.

A BPMN lane is used to represent a relationship between actions and entities if the entity is a role or a department. If the entity is an external stakeholder then a BPMN message flow should be used instead.

3.1.3 The BPM^+ operational level of abstraction

At the lowest level of abstraction (i.e. operational level), BPM^+ describes specific activities of the organization according to the needs of the target stakeholders, answering the question "how". The operational level of abstraction is very challenging. It can present multiplicity of forms that depend on the specific needs of each stakeholder at the operational level. For example, when the stakeholder is dealing with the implementation of a software application, then all the additional information required to implement the application should be modeled at this level of abstraction in a formal way. On the other hand, when the stakeholder is responsible of formalizing the business processes of the organization to comply with an external regulation, then the tasks for each critical activity, their control criteria and their corrective actions should be modeled at this level of abstraction (Ouanouki and April, 2008).

Scope

The operational level of abstraction describes the critical activities of the organization. It includes the representation of the atomic tasks that compose each of the critical activities.

Modeling concepts and semantics considerations

Depending on the purpose of modeling, the BPM^+ operational models might vary. This translates into a variable set of modeling concepts that should be selected according to the needs of each purpose of modeling. However, some modeling concepts should be common to any variation of a BPM^+ operational model:

- Actions: tasks;
- Entities: roles (i.e. internal actors);
- Information (relationships or dependencies): relationships between tasks; and
- Tools: physical tools (e.g. computers, software tools, machinery), documents (i.e. documents that are used or produced by the activity).

In addition, at the operational level of abstraction there is the need to represent:

- The events that trigger each critical activity;
- The events that indicate the end of the workflow of each critical activity; and
- Control flow patterns (Russell *et al.*, 2006a): decisions, merges, splits, synchronizations.

A goal of the research activities planned as part of this thesis' methodology is to improve the basic list of modeling concepts described above based on the modeling needs encountered during the execution of the empirical research activities.

Table 3.3 shows the recommended representations of the operational-level BPM^+ modeling concepts listed above in both Qualigram and in BPMN notations.

Table 3.3 Representation of operational-level BPM+ modeling concepts in Qualigram and BPMN notations

BPM+ modeling concepts	Qualigram	BPMN
Task.	Operation.	Task.
Role.	Role.	Lane.
Relationship between actions.	Information.	Sequence flow.
Physical tool.	Physical tool.	Not available; use text annotation attached with an association.
Document.	Document.	Data object.
Triggering event.	Start event, up-stream action.	Start event.
End event.	End event, down-stream action.	End event.
Control flow pattern.	And operator, Or operator.	Gateway.

Syntax considerations

Any BPM+ operational-level model is elaborated aiming at the description of the critical activities of the organization. Each critical activity is described through its component atomic tasks that should be performed by only one role. That is, in BPM+ a task cannot be further decomposed either functionally or organizationally. The details of each operational-level model will depend on the needs of the purpose of modeling given by the target stakeholder.

At the operational level of abstraction, Qualigram notation, besides allowing the representation of atomic operations (e.g. BPM+ tasks), allows the representation of control operations and corrective operations. Providing these additional modeling constructs, Qualigram notation allows the modeler to describe how a specific role of the organization has to perform a specific activity, what controls have to be performed for some critical component tasks of the activity, and what to do in case one or more of those controls are not complied. That is, Qualigram's operational level of abstraction is clearly oriented to the documentation of process activities as recommended by the ISO 9000 family of standards (i.e. quality management) (ISO, 2008; 2010) that asks to describe what actions the organization should take in order to remove an existing nonconformity. This way of modeling the operational level might be very useful for stakeholders willing to model the business processes, for purposes such as: quality control, ISO 9000 compliance, external-regulation compliance, etc. However, the various activities planned as part of this thesis research methodology should explore other possible ways of modeling BPM^+ operational level in order to satisfy other purposes of modeling.

At BPM^+ operational level of abstraction the BPMN modeling constructs included in the “analytical” level (i.e. Level 2) of BPMN use as proposed by Silver (refer to subsection 1.5.2.2) should be used. Silver’s “executable” level of use (i.e. Level 3) should not be used because it is aimed at the implementation and enactment phases of the BPMS lifecycle (refer to Figure 1.2) which are out of the scope of this thesis. Only the “private business process” type of BPMN model (refer to subsection 1.5.2.2) should be used.

In summary, the three levels of abstraction of the *a priori* version of BPM^+ have been described. This version of the BPM approach proposed requires to be validated. The next sections (sections 3.2 to 3.4) present the results of the various research iterations that have been performed to refine BPM^+ according to the planned research methodology (refer to CHAPTER 2). Section 3.5 presents the reviewed version of BPM^+ after considering the findings obtained from these research iterations.

3.2 The pilot case study

This section presents the results of the pilot case study where the usefulness and acceptance of the BPM^+ strategic and tactical levels of abstraction were tested. The case study was conducted at a small Canadian software development company. The case study aimed at evaluating not only the BPM^+ approach, but also the BPM notations selected for this thesis (both: their fitness to BPM^+ and their perceived participant’s acceptance).

The main product offered by the participant company is an ERP system. The participant company was selected for this case study due to: 1) its willingness to initiate a BPM initiative; 2) its accessibility; and 3) its interest in the project. The company was willing to model the business processes supported or affected by the ERP in order to:

1. document them;
2. show the customers how the ERP interacts with the various end-users and business processes of the organization; and
3. communicate to their new employees the ERP functional characteristics.

3.2.1 Details of the research design

The principles and generalities of the research design of this case study have been already reported in section 2.2. The planned duration of the case study was of 4 months, and it required the participation of a research team of two members (i.e. the author of this thesis and one ÉTS master degree student) together with two members of the participant company. Both members of the research team had and adequate level of knowledge of the BPM notations used in the case study, and one of them (i.e. the author of this thesis) has had a previous experience defining and modeling at a high level the business processes of an academic organization. Regarding the members of the participant company, the first member was its owner and top-executive. His participation ensured considering not only a technical point of view but also the commercial and organizational points of view of the business processes to be modeled. His participation was complemented with a member of the technical staff who supports the development of the ERP system. Neither of the two members of the participant company had previous experience with BPM and neither of them had knowledge of the BPM notations used in this case study.

The strategic and tactical levels of abstraction of BPM^+ were applied to three business processes selected by agreement with the participant company at the beginning of the project: procurement, sales at the counter, and sales by contract.

Besides collecting evidence from the sources of information already described in section 2.2, evidence was also collected in this case study through the observation of the use of the ERP system.

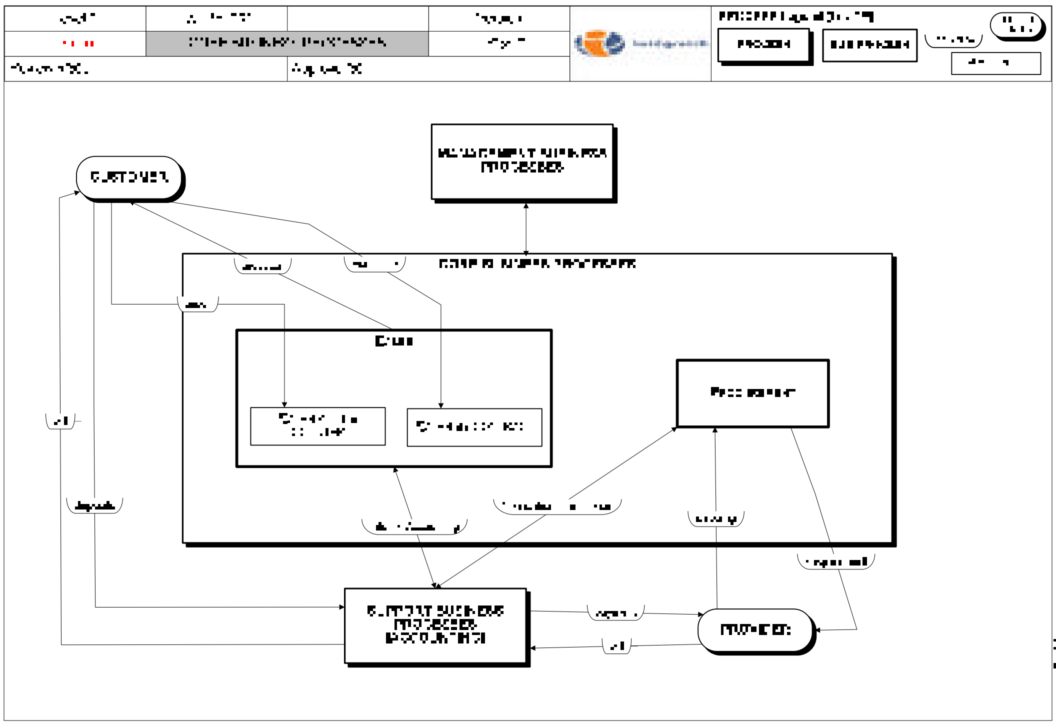

Figure 3.2 Pilot case study: Qualigram's relational model

3.2.2 Results

3.2.2.1 Results related to Qualigram notation

As part of the case study, the research team aimed at identifying the perceived value of each of the four types of models offered by Qualigram at the strategic level of abstraction (refer to subsection 3.1.1). From the four types of models, it was decided to only use the macroscopic and the detailed types of models. The relational type of model (See Figure 3.2) was discarded because the members of the participant company considered that it does not add relevant information to the information already provided by the macroscopic and detailed models.

The members of the participant company confirmed the reasons indicated in subsection 3.1.1 to discard the transversal type of model.

The macroscopic type of model (see Figure 3.3) was considered relevant because it:

1. identifies the main external stakeholders of the organization (e.g. customers and providers);
2. identifies the core business processes that interact with the ERP; and
3. allows classifying the business processes in a structured way (e.g. management processes, core processes, and support processes as it was the case for this participant company).

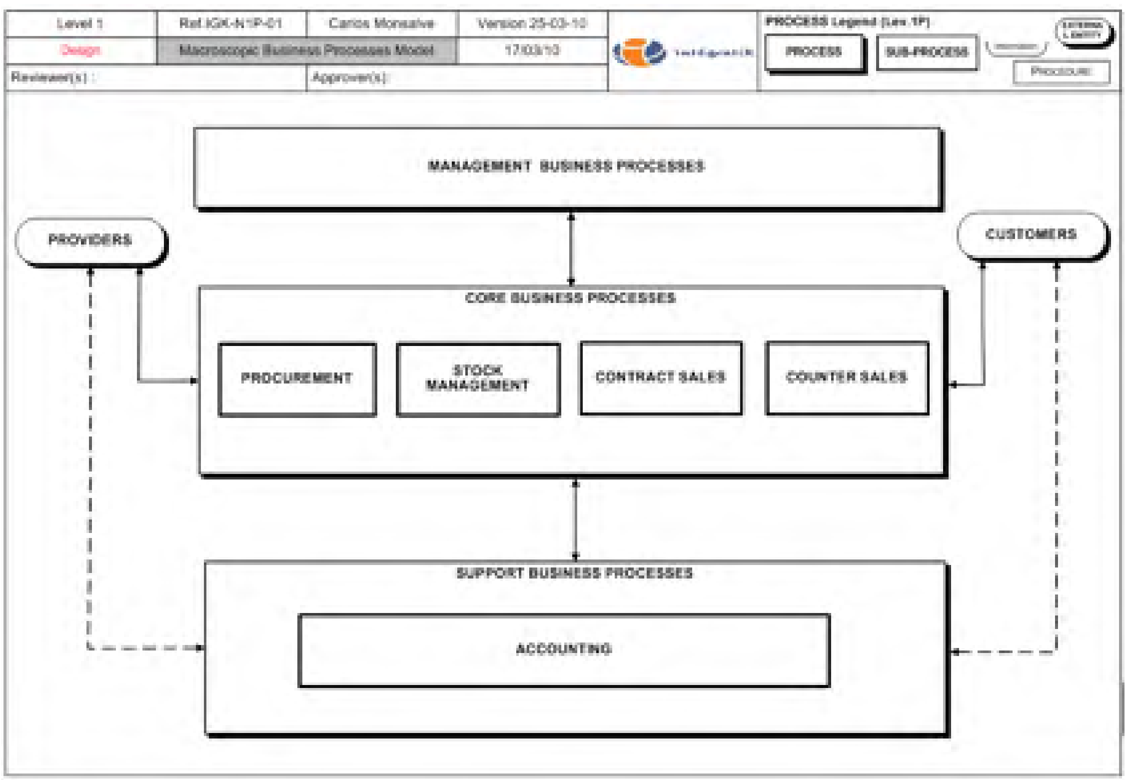

Figure 3.3 Pilot case study: Qualigram's macroscopic model
Reprinted from Monsalve, April and Abran (2011b, p. 238),
ISBN: 978-960-474-281-3, with permission from WSEAS Press

A detailed model was developed for each of the core business processes depicted in the macroscopic model. For instance, Figure 3.4 presents the detailed model for the procurement

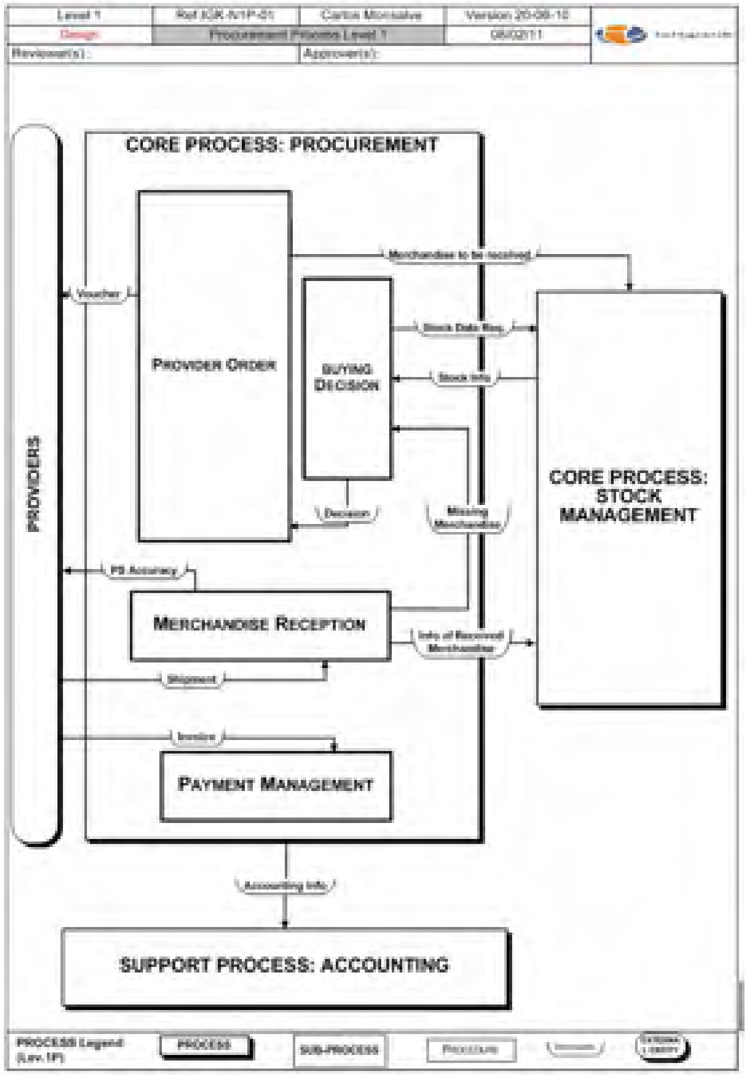

Figure 3.4 Qualigram's detailed model of the Procurement business process
Reprinted from Monsalve, April and Abran (2011b, p. 238), ISBN: 978-960-474-281-3, with permission from WSEAS Press

business process. The detailed type of model was considered relevant by the members of the participant company because it provides:

1. a high-level model for each core business process;
2. the relationships between each businesses process and the main external stakeholders (e.g. customers and providers);
3. the position of each business process in relation to its own context; and

4. a logical link between the macroscopic model and the tactical level models.

The strategic level of abstraction does not aim at representing any kind of workflow (i.e. chronological flow of activities) of the business processes. However, the participant organization found it useful to always model as close as possible to the workflow of the business process. For instance, the detailed model showed in Figure 3.4 is depicted trying to resemble the workflow of the procurement business process.

Each procedure depicted in a detailed model was then modeled at the intermediate level of abstraction (i.e. tactical level), representing its workflow and the roles and departments responsible of the execution of each activity. For instance, Figure 3.5 presents the intermediate level model for the “Payment Management” procedure of the procurement business process depicted in Figure 3.4.

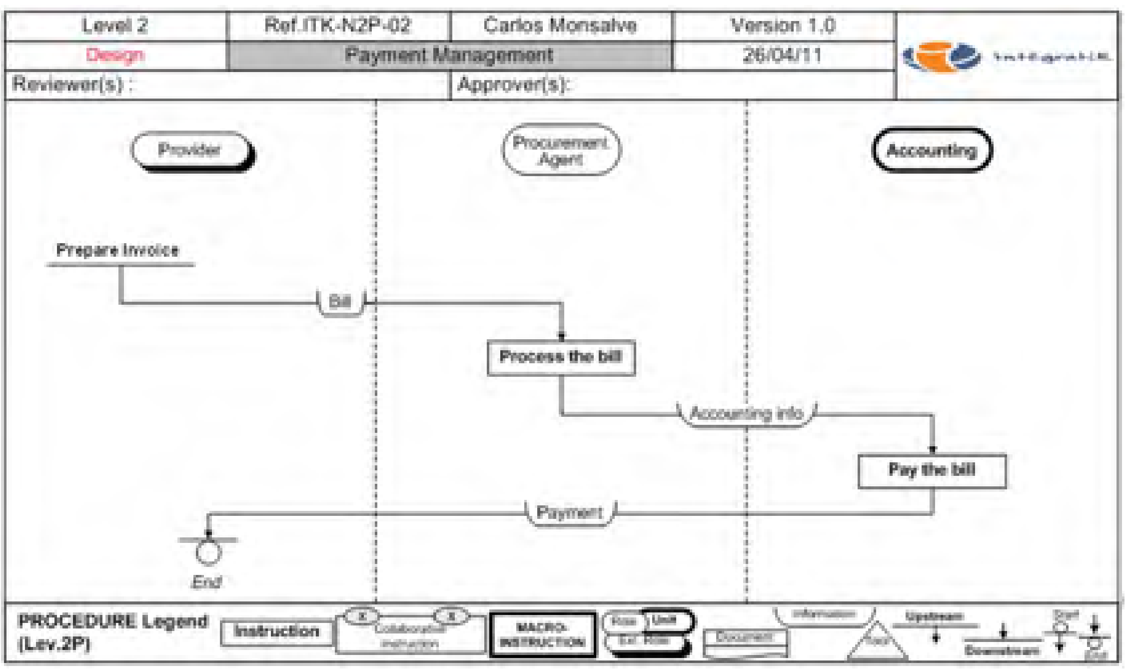

Figure 3.5 Qualigram's tactical model of the Payment Management procedure

Notice from Figure 3.2 to Figure 3.4 that at the strategic level Qualigram notation does not represent the workflow of a business process. For instance, even in Figure 3.4 it is not possible to accurately identify the triggering event and the end event of the business process depicted. Also, notice that at the strategic level the Qualigram notation does not represent the various roles and departments (i.e. units) of the organization. It is at the tactical level of abstraction where Qualigram notation represents both the workflow of the business processes and the various roles and departments involved in the execution of the business process as it can be observed in Figure 3.5.

3.2.2.2 Results related to BPMN

As recommended by the *a priori* version of BPM^+, the research team used Levels 1 and 2 of BPMN use (refer to subsection 1.5.2.2) to model at the strategic and tactical levels of abstraction. The aim was to produce BPMN models similar to those generated with Qualigram notation. However, the results of the case study showed that this was not always possible. For instance, since BPMN requires the modeler to always represent the workflow of a business process, then the Level 1 of BPMN requires an explicit representation of the starting and ending events as well as a chronological ordered set of BPMN activities of the business process being modeled, as it is depicted in Figure 3.6. Therefore, this intrinsic characteristic of BPMN[17] alone does not allow elaborating a business process model without entering into the details of a workflow, as it is desired at the BPM^+ strategic level of abstraction.

Since BPMN always requires modeling the workflow of a business process, it was not possible to generate a BPMN model with the characteristics of Qualigram's macroscopic type of model (refer to Figure 3.3). Moreover, in BPMN each business process requires its

[17] The specifications of BPMN 2.0 (OMG, 2011) establishes that based on the three types of process sub-models (i.e. private process, public process and collaboration process) the modeler can elaborate many types of process diagrams. However, each BPMN vendor is free to decide how to use the three types of sub-models to allow its customers to create various types of process diagrams. There is no restriction to represent all the business processes as black boxes (i.e. empty pools) and therefore, it is not mandatory to represent the workflows of the business processes modeled. This fact presents an impact on the results of our pilot case study as it is discussed later in this subsection (see footnotes 18 and 19).

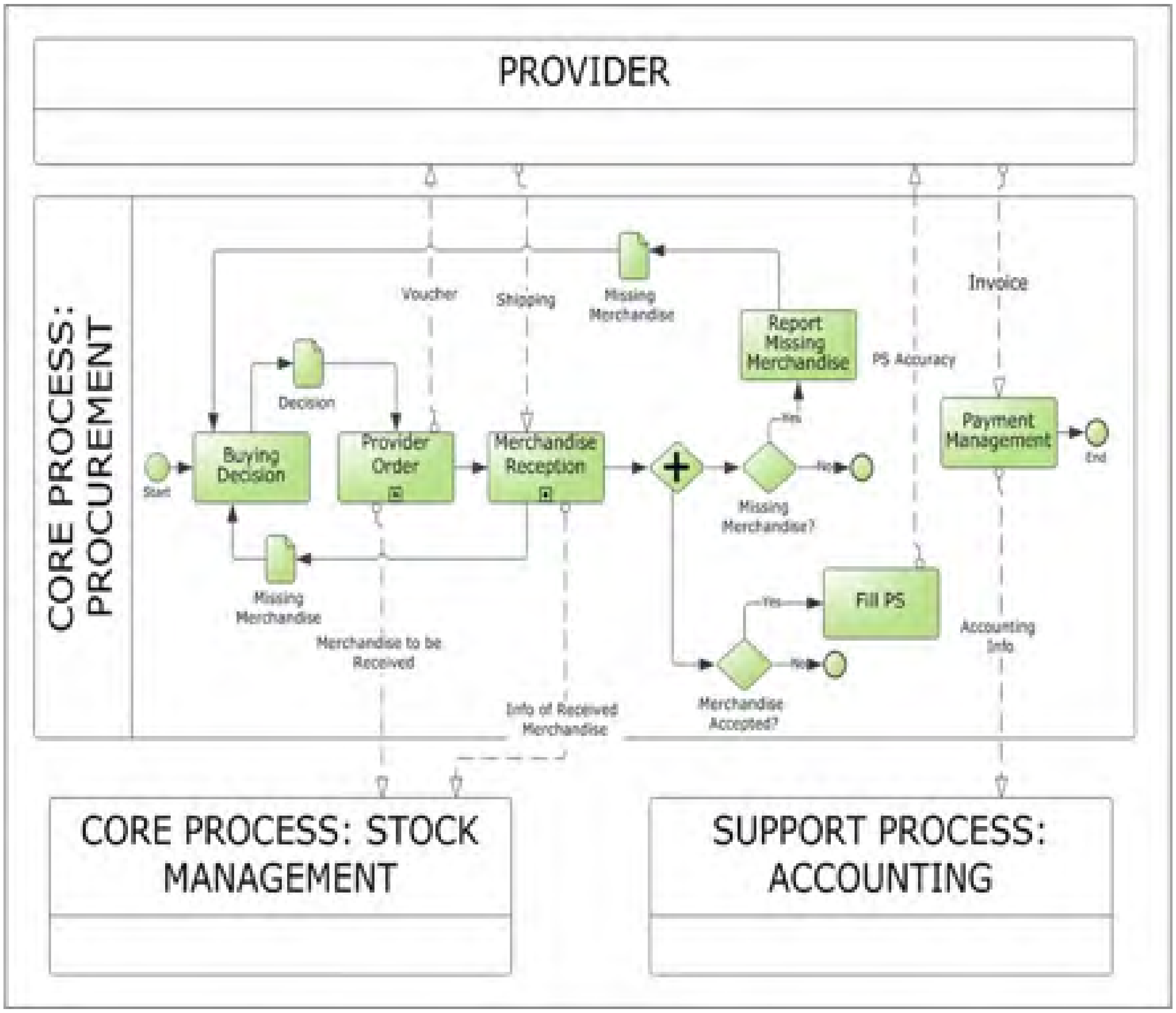

Figure 3.6 BPMN Level 1 model without lanes of the Procurement business process
Reprinted from Monsalve, April and Abran (2011b, p. 238),
ISBN: 978-960-474-281-3, with permission from WSEAS Press

own model (either a Level 1 model or a Level 2 model). Therefore, BPMN cannot represent a big picture of all the core business processes in only one model, as Qualigram does with the macroscopic type of model[18].

[18] Even though BPMN 2.0 allows representing any business process as a black box (i.e. empty pool), it is not possible to create in BPMN 2.0 a high-level model similar to Qualigram's macroscopic type of model.

The closest BPMN's scenario to the strategic level of abstraction[19] is a model based on the Level 1 of BPMN use but restricting the use of lanes. BPMN lanes can be used to represent roles in the workflow of the business process, something that is not required at BPM$^+$ strategic level of abstraction. A model of this type (i.e. Level 1 without lanes) was generated for each of the core business processes depicted in Qualigram's macroscopic model (refer to Figure 3.3). For instance, Figure 3.6 presents the Level 1 model without lanes for the procurement business process. This type of BPMN model might be compared to Qualigram's detailed type of model, with the main difference that the BPMN version includes the representation of the main workflow of the business process (compare Figure 3.4 and Figure 3.6). Because of that, the BPMN version of the model includes the use of BPMN gateways and events. The BPMN Level 1 model with lanes restricted was considered as a relevant modeling reference by the members of the participant company because it helps to generate high-level BPMN models for each core business process.

At the tactical level of abstraction, each BPM$^+$ model should represent the workflow and the various responsible roles of the business process modeled. Therefore, a first approach to the tactical level of abstraction using BPMN was to model following the recommendations of Level 1 of BPMN use but without the restriction of using lanes. A model of this type was generated for each core business process depicted in Figure 3.3. For instance, Figure 3.7 presents the model for the procurement business process. As it can be noticed from the comparison between Figure 3.6 and Figure 3.7, the main difference is that in Figure 3.7 the BPMN activities have been distributed among the roles (i.e. warehouse and procurement agent) responsible of the procurement business process.

Finally, Level 2 of BPMN use was used as a reference when it was considered necessary to resort to the modeling constructs of the BPMN Extended Set. For instance, Figure 3.8 depicts the details of the "Management of a registered customer payment" procedure that is part of

[19] BPMN 2.0 allows creating a true strategic-level model: a collaboration diagram where all the pools are empty (i.e. black boxes). Each of the empty pools might represent a different core business process. The diagram depicts the collaborations between all the processes represented. This BPMN diagram is comparable to Qualigram's relational type of model.

the “Sales at the Counter” business process. This procedure requires the use of a BPMN or-gateway that is a modeling construct unavailable at the Level 1 of BPMN use.

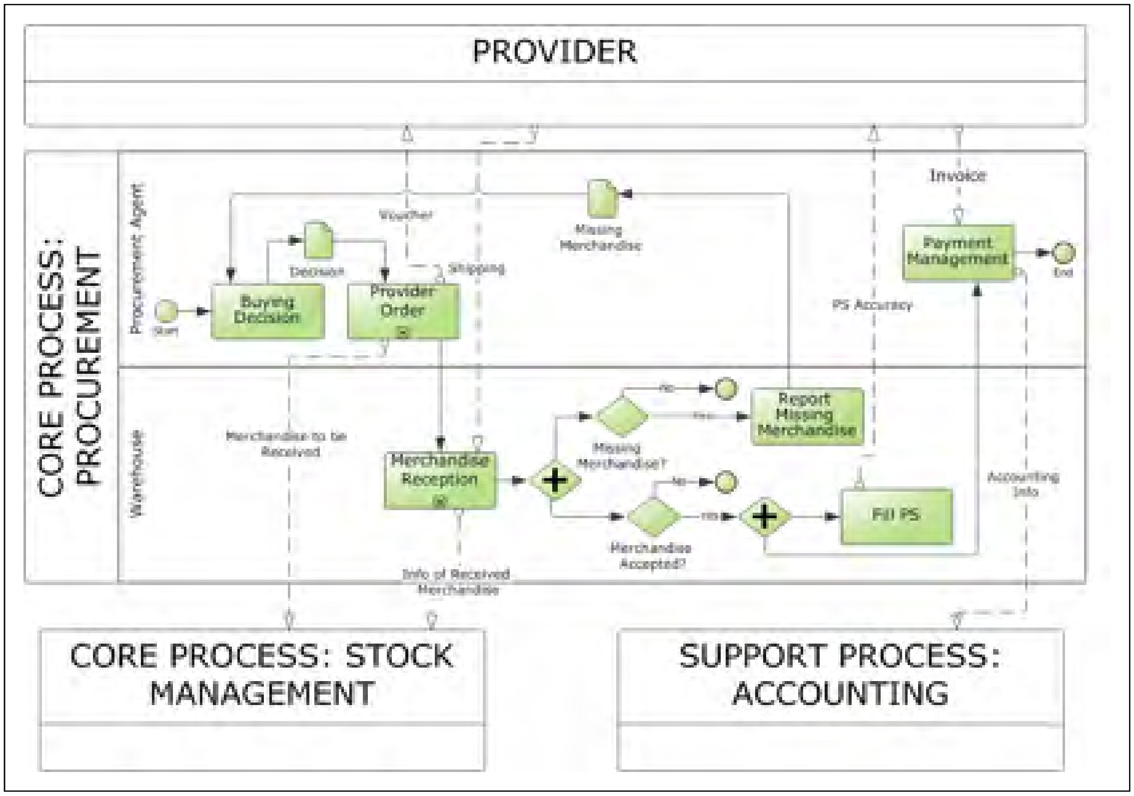

Figure 3.7 BPMN Level 1 model with lanes of the Procurement business process

3.2.3 Comparison of the BPM notations

The research team experimented with the ease of understanding of the two BPM notations (i.e. Qualigram notation and BPMN) by the members of the participant company. First, the research team provided the participants various Qualigram models, and they were able to interpret them mostly correctly. The experience was quite different with BPMN due to the diversity of modeling constructs used. The members of the participant company mentioned that they would require training before starting a BPM initiative using BPMN. Therefore, the members of the participant company found Qualigram notation easier to understand than

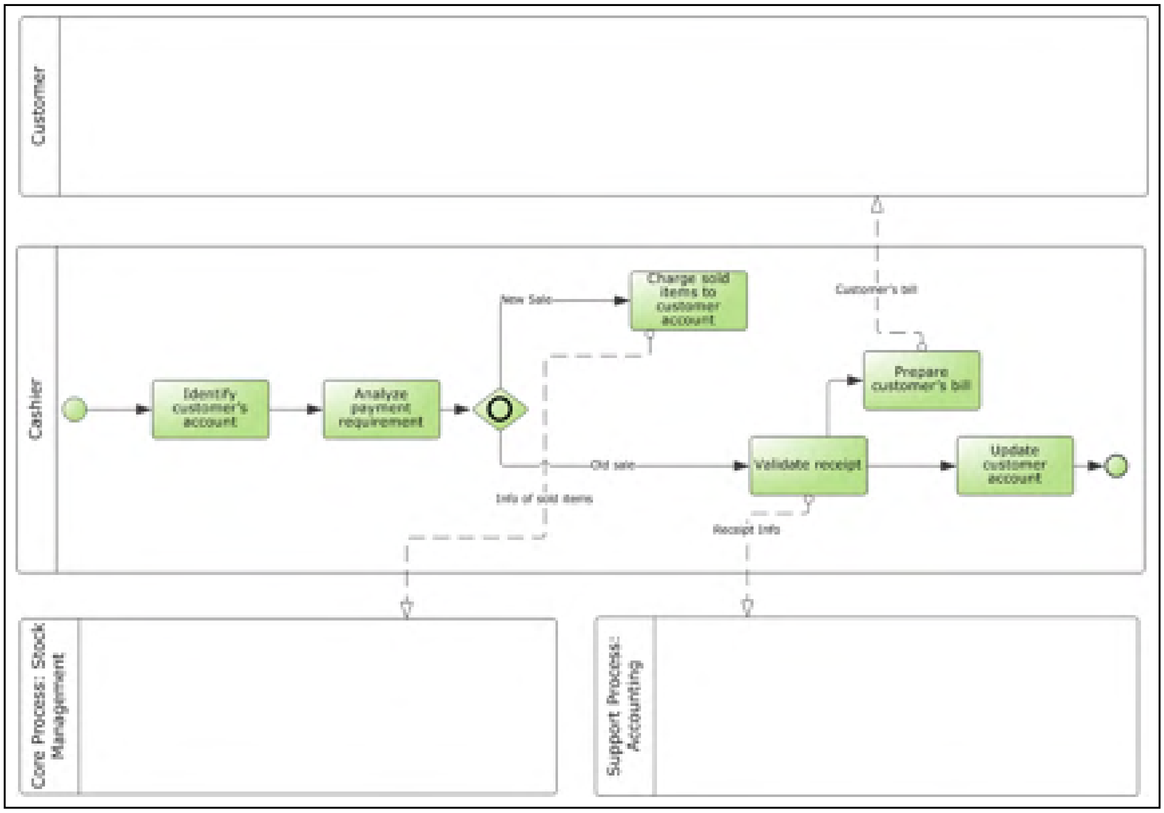

Figure 3.8 BPMN Level 2 model of the Management of a registered customer payment procedure

BPMN. Qualigram models were found to be more suitable for introducing business processes to customers, to administrative staff, and to new members of the IT staff.

The members of the participant company found that BPMN is more formal and detailed than Qualigram notation for modeling business processes: that is, BPMN models were considered more precise than their Qualigram counterparts. Therefore, BPMN models were found to be more suitable as an input for the software development team.

3.2.4 Interpretation and summary of the results

The previous subsections have reported the results of a pilot case study conducted to test the BPM$^+$ strategic and tactical levels of abstraction in a BPM initiative. The methodology used for this case study (i.e. action research) showed to be adequate for empirically testing BPM$^+$.

However, it is critical to the success of this type of methodology to ensure the collaboration of the leader of the participant organization.

One of the main objectives of using levels of abstraction is to ease the active participation of the various stakeholders in the BPM initiative. The use of the BPM^+ strategic and tactical levels of abstraction was well accepted by the members of the participant company: they considered relevant to have various levels of abstraction in order to reach various types of stakeholders. Moreover, they indicated that BPM^+ eases using BPM for various types of activities both inside the organization and with their customers.

The members of the participant company identified the value of the strategic level of abstraction as the ability to expose their customers and their new employees to the business processes supported by the ERP system. The value of the tactical level of abstraction was identified as the ability to deliver the details of the business processes to both the IT personnel in charge of the maintenance of the software applications that support the business processes, and the employees of the organization who are responsible for the execution of the business processes modeled.

Two different BPM notations: Qualigram notation (more management oriented) and BPMN (more IT oriented) were compared in terms of their appropriateness for modeling at the strategic and tactical levels of abstraction. Qualigram notation was found more suitable for customers, new employees, and management; while BPMN was considered more suitable for the technical staff. Probably, the best approach is to use a management oriented BPM notation as Qualigram for modeling at the strategic level of abstraction, and to allow the stakeholder to choose at the tactical level of abstraction between a more formal and detailed model generated in BPMN and a more simple model generated in Qualigram notation.

From the four types of models offered by Qualigram notation at its top level of abstraction, two have been confirmed as valuable for the BPM^+ strategic level of abstraction: the macroscopic type of model, and the detailed type of model. The perceived value of each of these two types of model has been identified. It was also confirmed the lack of necessity of elaborating a transversal type of model at the strategic level of abstraction.

The case study showed that BPMN does not allow generating a model that fulfils all BPM$^+$ criteria at the strategic level of abstraction[20]. For instance, it is not possible to generate a BPMN model similar to the macroscopic type of model generated with Qualigram notation. The highest-level BPMN model was obtained based on the Level 1 of BPMN use but restricting the use of lanes. This type of model resembles the detailed type of model offered by Qualigram notation. However, the BPMN model violates one of the characteristics proposed by BPM$^+$: do not represent a business process workflow at the strategic level of abstraction.

According to Table 3.1, a sub-process might be represented in a BPMN strategic-level model either by a lane or by a collapsed sub-process. The case study showed that the latter is the most appropriate BPMN modeling construct to represent a sub-process. According to the same table, a procedure might be represented in a BPMN strategic-level model either by a collapsed sub-process or by a task. The collapsed sub-process showed to be the most appropriate BPMN modeling construct to represent procedures at the strategic level of abstraction. Notice that if sub-processes and procedures are represented in a BPMN strategic-level model as indicated before, then the same BPMN modeling construct (i.e. collapsed sub-process) has two possible meanings in a strategic-level model.

A BPMN strategic-level model might use two BPMN modeling constructs to represent relationships: 1) the "sequence flow" modeling construct that is used to represent relationships between actions contained within the pool that represents the business process being modeled; and 2) the "message flow" modeling construct that is used to represent two types of relationships:

---

[20] It has been explained in subsection 3.2.2.2 that BPMN 2.0 allows creating a BPMN diagram similar to Qualigram's relational type of model. Therefore, BPMN 2.0 allows generating a model that fulfils the BPM$^+$ criteria at the strategic level of abstraction. However, it is not possible to generate a model similar to Qualigram's macroscopic type of model.

- relationships between external stakeholders and the business process being modeled (e.g. the relationships between the provider and the procurement business process in Figure 3.7); and
- relationships between the business process being modeled and other business processes (e.g. the relationships between the procurement business process and the stock management and accounting business processes in Figure 3.7).

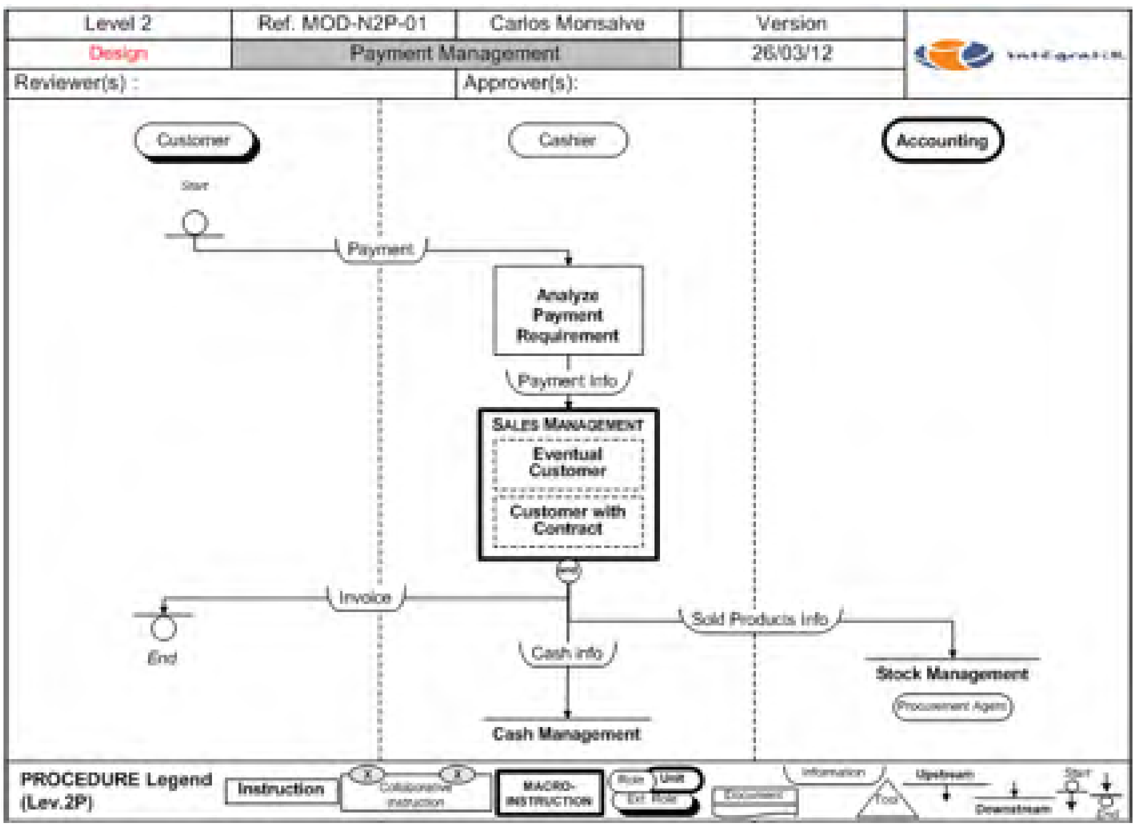

Figure 3.9 Qualigram's tactical level of Payment Management procedure (sales at the counter business process)

Qualigram notation offers at the tactical level of abstraction two possible modeling constructs (i.e. start event and up-stream action) to represent a triggering event, and two modeling constructs (i.e. end event and down-stream action) to represent an end event. The case study showed that all these modeling constructs are useful. For instance, Figure 3.5 depicts the use of an up-stream action (i.e. “Prepare invoice”), and an end event; while Figure 3.9 depicts the use of a start event, two down-stream actions (i.e. “Cash Management” and “Stock Management”), and one end event.

The pilot case study confirmed the relevance of representing most of the modeling concepts proposed by BPM+ at the strategic and tactical levels of abstraction. For instance, the model depicted in Figure 3.10 makes use, besides others already observed in previous figures, of three Qualigram modeling constructs: document (i.e. “Cashiers report”), physical tool (i.e. “System”), and responsibility (i.e. “R” under “Accounting officer” swim-lane). However, during this case study it was not possible to test the relevance of representing goals and objectives as part of the strategic-level and tactical-level models respectively. The relevance of these two latter BPM+ modeling concepts will be further tested by the survey that is part of this thesis methodology (see section 3.4).

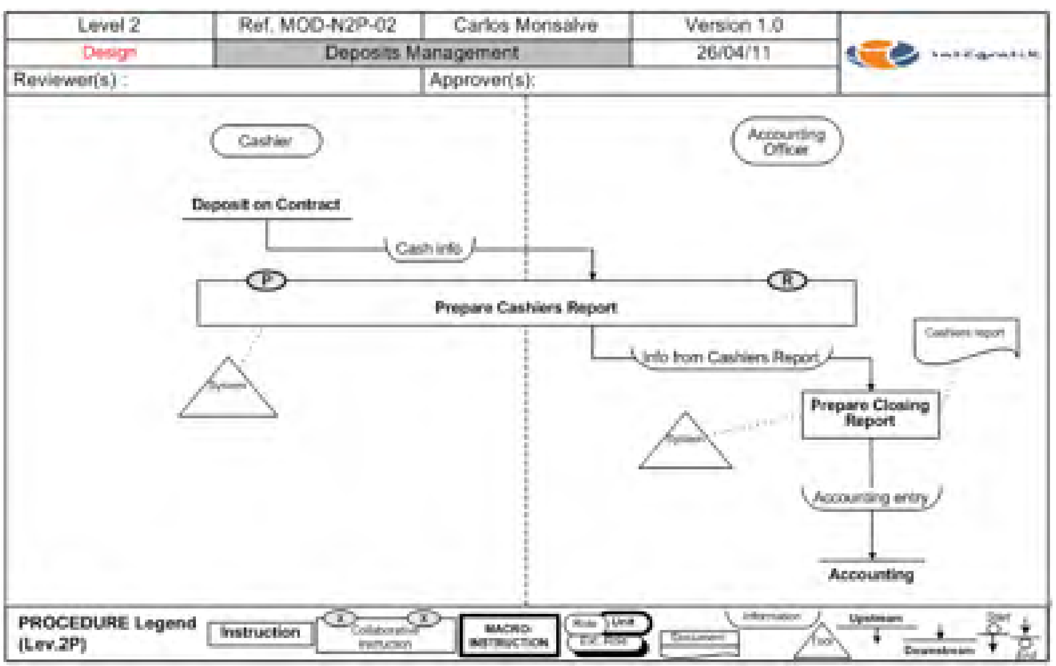

Figure 3.10 Qualigram's tactical model of the Deposits Management procedure (sales at the counter business process)

Finally, Qualigram notation offers at the tactical level of abstraction some modeling constructs that are not included in BPM^+ (refer to Table 3.2): macro-instruction, collaborative instruction, target role, source role, control indicator and corrective indicator. From these modeling constructs, the target role and the source role sometimes must be used together with the down-stream action and up-stream action respectively. The source role identifies the role that has triggered the up-stream action, and the target role identifies the role that will be the responsible of the down-stream action; see for example the target role “Procurement Agent” below the “Stock Management” down-stream action in Figure 3.9. The macro-instruction (e.g. “Sales Management” in Figure 3.9) and the collaborative instruction (e.g. “Prepare Cashiers Report” in Figure 3.10) demonstrated to be useful; however, they can be replaced by alternative ways of modeling if necessary. The control indicator and the corrective indicator modeling constructs demonstrated to be not necessary to be included in a tactical-level model.

3.3 Refining the set of BPM^+ modeling concepts

In the Introduction chapter of this thesis it has been illustrated from the literature that a software development project is highly dependent on the quality of the software requirements process. A high quality software requirements elicitation (SRE) task depends, among other things, on good communication between software engineers and end-users (Abran *et al.*, 2004; Wand and Weber, 2002). Conceptual modeling supports this level of participation during SRE as it helps to understand the subject matter it represents (Abran *et al.*, 2004; Wand and Weber, 2002): one of the most popular conceptual modeling representation approaches is BPM (Davies *et al.*, 2006). To ensure the effectiveness of BPM, it is important that the appropriate BPM notation be selected. A good BPM notation needs to be complete and clear, that is, capable of expressing all the relevant concepts in the domain under study (Wand and Weber, 2002). Which are the relevant concepts for software requirements elicitation? How can we assess if our BPM notation is complete and clear?

One analysis technique frequently used for assessing the expressiveness in terms of completeness and clarity of a BPM notation (Gehlert and Esswein, 2007; Rosemann *et al.*, 2009) is the Bunge-Wand-Weber (BWW) representational analysis (Wand and Weber, 1995), which is based on an ontological model (i.e. the BWW representation model). Both, the BWW representational analysis and the BWW representation model have been already introduced in section 1.7.

To improve the relevance of an assessment based on an ontological model, some authors have recommended to select the ontology, or a subset of it, that best fits the specific context that encompasses the modeling process (Rosemann and Green, 2000; Rosemann *et al.*, 2009). No previous research was found in which the fitness of the BWW representation model for the specific context of SRE was studied previously. This section presents our research work to analyze the BWW representation model within the context of SRE. The findings of this analysis are used for:

1. determining the set of relevant concepts for SRE that should be considered when elaborating a business process model;
2. selecting a subset of the BWW representation model that better fits the SRE context; and
3. assessing the expressiveness of the BPMN and Qualigram notations in terms of their completeness and clarity to represent the BWW subset selected in this section.

The methodology to be followed in order to perform the three research activities mentioned above has been presented in section 2.3. The starting point is to consider that software engineers and business analysts are both professionals trained to perform requirements elicitation tasks, and that each of these two professions can rely on a guide to its body of knowledge (refer to subsection 1.6.1):

1. the Guide to the Software Engineering Body of Knowledge (SWEBOK); and
2. the Guide to the Business Analysis Body of Knowledge (BABOK).

Therefore, the SWEBOK and the BABOK are used here as the key references for analyzing the BWW representation model within the specific context of SRE. The next subsections present the results obtained after each step of the methodology (refer to Figure 2.3).

3.3.1 Mapping results and analysis

Before performing the analysis of the BWW representation model, it is necessary to identify in the SWEBOK and the BABOK the relevant concepts associated with requirements elicitation. Table 3.4 shows the concepts that:

1. are the result of the consensus reached by the research groups involved in this part of the research (refer to section 2.3);
2. were selected by the research groups because they are considered as pertinent when generating a business process model; and
3. are common to the SWEBOK and the BABOK.

The various concepts found in the SWEBOK and the BABOK have been classified in Table 3.4 according to their semantics relationships. In some cases the same concept appears more than once since it is semantically related to more than one concept of its counterpart reference. For instance, the *workflows* SWEBOK concept appears twice in Table 3.4 because it can be related to the *dependencies between tasks and activities* and the *process* BABOK concepts.

To facilitate the traceability of the concepts presented in Table 3.4, this table includes for each set of concepts the reference to the sections of the SWEBOK or the BABOK.

Table 3.4 SRE relevant concepts found in the SWEBOK and BABOK

SWEBOK		BABOK	
Concepts	**References**[21]	**Concepts**	**References**
Users, roles, third party, devices, software components, entities from the problem domain, interfaces with the environment	Sections 1.1, 1.2, 1.6, 2.2, 3.2, and 4.2	Stakeholders, users, roles, classes, objects, business units, departments	Sections 1.5.6, 2.2.3, 2.2.5, 5.4.4, 6.2.4 and 6.2.5
Conceptual modeling, state models, object models, event traces, usage scenarios, workflows	Section 4.2 and 5.1	Behavioral characteristics of a solution (features and functions), interactions people-solution-system, IT responses, dependencies between tasks and activities (action flows)	Sections 1.3.3, 2.3.4, 4.4, 5.4.4 and 6.2.4
How are tasks done, what the software product is not expected to do, functional requirements (capabilities), non-functional requirements (constraints, quality requirements) emergent properties, industry practices, product parameters	Sections 1.2, 1.3, 1.4, 3.2, and 5.3. Chapter 9, section 2.4	Rules, business constraints, responsibilities, constraints	Sections 2.2.3, 6.2.4 and 6.4.2
Event traces	Section 4.2	External events, internal events, scheduled events, triggers, milestones	Sections 2.3.4, 6.2.4 and 6.5.4
Business processes, workflows, activities, tasks	Sections introduction, 1.1 and 5.1	Outcomes, actions or tasks, activities (may be broken into tasks), process, sub-process	Sections 1.2, 1.5.2, 2.3.4, 6.2.4, 6.5.4 and 9.21.3
Interaction between components, relationships and dependencies between entities, interaction between users, interaction between users and their software	Sections 3.2, 4.2 and 4.3	Interactions between roles and stakeholders	Sections 2.2.5 and 6.2.5
Levels of organization, views (high-level), software components	Sections 1.1, 1.6 and 4	Business domain, sub-domain, high-level requirements, specific stakeholder requirements	Sections 1.3.1, 2.1.4, 3.3.7 and 6.3.4
Levels of organization, views (high-level)	Sections 1.1 and 4	Levels of abstraction, requirements structure	Sections 6.2.4 and 6.3.3
Goals and objectives	Section 3.1	Objectives	Sections 1.2 and 1.3.3

[21] Unless otherwise stated, all section references are to SWEBOK's chapter 2.

3.3.1.1 A SWEBOK insight into the BWW representation model

This subsection presents the mappings of the SWEBOK software requirements concepts (listed in Table 3.4) with the BWW representation model[22]. The outcome of these mappings is a subset of concepts from the BWW representation model[23] that has been selected based on its capabilities to represent the relevant SRE concepts found in the SWEBOK.

Table 3.5 Representation mapping of the BWW representation model based on selected SWEBOK concepts

SWEBOK Concepts	BWW Concepts
Business processes	History, transformation
What the software product is not expected to do	State law
Workflows	History, transformation, lawful transformation, lawful event space
Activities	Transformation
Tasks	Transformation
Users / roles	System composition, thing
Third party	System composition, thing
Devices	System composition, thing
Interfaces with the environment	Thing, system composition
Levels of organization	Subsystem, level structure
View (high-level)	Subsystem, level structure
Emergent properties	State law
Product parameters	State law
Functional requirements (capabilities)	State law
Non-functional requirements (constraints, quality requirements)	State law
Industry practices	State law
Software components	Thing, subsystem, system composition
Interaction between components	Coupling, acts on
Entities from the problem domain	Thing, system composition
Relationships and dependencies between entities	Coupling, acts on
Interaction between users	Coupling, acts on
Interaction between users and their software	Coupling, acts on
Conceptual modeling	Conceivable event space, conceivable state space, history
Object models	Conceivable state space
State models	Conceivable state space, history
How are tasks done	State law
Event traces	Conceivable event space, event
Usage scenarios	Conceivable event space
Goals, objectives	

[22] The set of BWW representation model concepts used in this thesis is the same set of concepts used in Table 2 by (Recker *et al.*, 2005).

[23] The full set of BWW representation model concepts used in this thesis can be found in Appendix X.

The representation (see Table 3.5) and interpretation (see Table 3.6) mappings were obtained using the methodology described in section 2.3 (refer to activity A1 in Figure 2.3). Notice in Table 3.5 that the SWEBOK concepts *goals* and *objectives* do not have a corresponding BWW representation model concept.

Table 3.6 Interpretation mapping of the BWW representation model based on selected SWEBOK concepts

BWW Concepts	SWEBOK Concepts
Thing	Users, roles, third party, devices, software components, entities from the problem domain, interfaces with the environment
Class	
Kind	
Property	
State	Information
Conceivable state space	Conceptual modeling, state models, object models
Lawful state space	
State law	What the software product is not expected to do, emergent properties, product parameters, functional requirements, non-functional requirements, industry practices, how are tasks done
Stable state	
Unstable state	
History	Business processes, workflows, conceptual modeling, state models
Event	Event traces
External event	
Internal event	
Well-defined event	
Poorly defined event	
Conceivable event space	Conceptual modeling, event traces, usage scenarios
Lawful event space	Workflow
Transformation	Business processes, workflows, activities, tasks
Lawful transformation	Workflow
Acts on	Interaction between components, relationships between entities, interaction between users, interaction between users and their software
Coupling	Interaction between components, relationships between entities, interaction between users, interaction between users and their software
System	
System environment	Environment (organizational, operational)
System composition	Users, roles, third party, devices, software components, entities from the problem domain , interfaces with the environment (all together)
System decomposition	
System structure	
Subsystem	Levels of organization, views, software components
Level structure	Levels of organization, views

Notice that Table 3.6 includes those BWW concepts that could not be mapped to any of the SRE concepts found in the SWEBOK (e.g. *class* and *kind*). Also notice that Table 3.6 includes two SWEBOK concepts that do not appear in Table 3.4 (i.e. *information* and *environment*). These two latter SWEBOK concepts are not included in Table 3.4 because they lack semantically related concepts in the BABOK. However, these two concepts appear in the interpretation mapping (Table 3.6) to show that the BWW representation model concepts *state* and *system environment* are useful to represent some SRE concepts found in the SWEBOK.

Table 3.7 Set of BWW representation model concepts that based on the SWEBOK better fits the SRE context

Cluster	BWW concepts
Things including properties and types of things	Thing
States assumed by things	State
	Conceivable state space
	State law
	History
Events and transformations occurring on things	Event
	Conceivable event space
	Lawful event space
	Transformation
	Lawful transformation
	Acts on
	Coupling
Systems structured around things	Subsystem
	System environment
	System composition
	Level structure

Based on the mappings presented in Table 3.5 and Table 3.6, the subset of BWW representation model concepts that better fits the SRE context is obtained. The resulting subset includes only those concepts that, according to the SWEBOK Guide, are relevant to the specific context of software requirements elicitation. This subset is presented in Table 3.7 grouping the various BWW representation model concepts into the four clusters proposed by

(Recker et al., 2005, second page): “1) things, including properties and types of things; 2) the states assumed by things; 3) the events and transformations occurring on things; and 4) the systems structured around things”.

3.3.1.2 A BABOK insight into the BWW representation model

This subsection presents the mappings of the BABOK software requirements concepts (listed in Table 3.4) with the BWW representation model. The outcome of these mappings is a

Table 3.8 Representation mapping of the BWW representation model based on selected BABOK concepts

BABOK concepts	BWW concepts
Objectives	
Requirements structure	Level structure
Stakeholders, users and roles	Thing
Classes	Thing, class
Objects	Thing
Business domain	System
Sub-domain	Subsystem
Boundaries for business domain and sub-domains	Lawful state space
External events	Event
Internal events	Event
Scheduled events	Event
Rules	State law
Business constraints	State law
Triggers	Event
Outcomes	Transformation
Responsibilities	State law
Interactions between roles and stakeholders	Acts on, coupling
High-level requirements	Subsystem
Specific stakeholder requirements	Subsystem
Levels of abstraction: from high to low level	System decomposition, level structure
Actions or Tasks	Transformation
Activities (may be broken into tasks)	Transformation
Dependencies between tasks and activities (action flows)	Lawful transformation
Milestones	Event
Process	Transformation
Sub-process	Transformation
Behavioral characteristics of a solution (features and functions)	Conceivable state space
Business units, departments	Thing
IT responses, interaction people-solution-systems	Conceivable event space, lawful event space
Constraints	State law

subset of concepts from the BWW representation model that has been selected based on its capabilities to represent the relevant SRE concepts found in the BABOK.

The representation (see Table 3.8) and interpretation (see Table 3.9) mappings were obtained using the methodology described in section 2.3. Notice in Table 3.8 that the BABOK concept *objectives* does not have a corresponding BWW representation model concept.

Table 3.9 Interpretation mapping based on selected BABOK concepts

BWW concepts	BABOK concepts
Thing	Objects, business units, departments, stakeholders, classes, users and roles
Class	Classes, business units, departments
Kind	
Property	
State	
Conceivable state space	Behavioral characteristics of a solution (features and functions)
Lawful state space	Boundaries for business domain and sub-domains
State law	Constraints, rules
Stable state	
Unstable state	
History	
Event	External events, triggers, milestones, internal events, scheduled events
Internal event	Internal events, triggers, milestones
External event	External events, triggers, milestones
Well-defined event	Scheduled events, triggers, milestones
Poorly defined event	Internal events, external events, triggers
Conceivable event space	Interactions between roles and stakeholders, IT responses, interaction people-solution-system
Lawful event space	Interactions between roles and stakeholders, IT responses, interaction people-solution-system
Transformation	Outcomes, actions or tasks, activities (may be broken into tasks), process, sub-process
Lawful transformation	Dependencies between tasks and activities (action flows)
Acts on	Interactions between roles and stakeholders
Coupling	Interactions between roles and stakeholders
System	Business domain
System environment	
System composition	Objects, business units, departments, stakeholders, classes, users and roles (all together)
System decomposition	Levels of abstraction: from high to low level, high-level requirements, specific stakeholder requirements
System structure	
Subsystem	Specific stakeholder requirements, high-level requirements, sub-domain
Level structure	Requirement structure, levels of abstraction: from high to low level

Table 3.9 includes those BWW concepts that cannot be mapped to any of the SRE concepts found in the BABOK (e.g. *kind* and *property*). Also notice that Table 3.9 includes a BABOK concept that does not appear in Table 3.4 (i.e. *boundaries for business domain and sub-domains*). This BABOK concept is not included in Table 3.4 because it lacks a semantycally related concept in the SWEBOK. However, this concept appears in the interpretation mapping (Table 3.9) to show that the BWW representation model concept *lawful state space* is useful to represent a SRE concept found in the BABOK.

Based on the mappings presented in Table 3.8 and Table 3.9, the subset of BWW representation model concepts based on the BABOK that better fits the SRE context is obtained. The resulting subset includes only those concepts that, according to the BABOK Guide, are relevant to the specific context of SRE. This subset is presented in Table 3.10.

Table 3.10 Set of BWW representation model concepts that based on the BABOK better fits the SRE context

Cluster	BWW concepts
Things including properties and types of things	Thing
	Class
States assumed by things	Conceivable state space
	State law
	Lawful state space
Events and transformations occurring on things	Event
	Internal event
	External event
	Well-defined event
	Poorly defined event
	Conceivable event space
	Lawful event space
	Transformation
	Lawful transformation
	Acts on
	Coupling
Systems structured around things	System
	Subsystem
	System decomposition
	System composition
	Level structure

3.3.1.3 A BWW representation model subset for SRE

Based on the results obtained from Table 3.5 to Table 3.10, a final subset of BWW representation model concepts that better fits the SRE context according to both the SWEBOK and the BABOK is obtained and it is presented in presented in Table 3.11.

Table 3.11 Mappings between the BWW representation model, selected SWEBOK concepts and selected BABOK concepts

SWEBOK Concepts	BWW Concepts	BABOK Concepts
Users, roles, third party, devices, software components, entities from the problem domain, interfaces with the environment	Thing	Stakeholders, users, roles, classes, objects, business units, departments
Conceptual modeling, state models, object models	Conceivable state space	Behavioral characteristics of a solution (features and functions)
How are tasks done, what the software product is not expected to do, functional requirements, non-functional requirements, emergent properties, industry practices, product parameters	State law	Rules, business constraints, responsibilities, constraints
Event traces	Event	External events, internal events, scheduled events, triggers, milestones
Conceptual modeling, event traces, usage scenarios	Conceivable event space	Interactions people-solution-system, IT responses
Workflows	Lawful event space	Interactions people-solution-system, IT responses
Business processes, workflows, activities, tasks	Transformation	Outcomes, actions or tasks, activities (may be broken into tasks), process, sub-process
Workflows	Lawful transformation	Dependencies between tasks and activities (action flows)
Interaction between components, relationships between entities, interaction between users, interaction between users and their software	Acts on	Interactions between roles and stakeholders
Interaction between components, relationships between entities, interaction between users, interaction between users and their software	Coupling	Interactions between roles and stakeholders
Levels of organization, views, software components	Subsystem	Business sub-domain, high-level requirements, specific stakeholder requirements
Levels of organization, views	Level structure	Levels of abstraction, requirements structure

Table 3.11 shows the associated SWEBOK and BABOK concepts for each of the selected BWW representation model concepts. The subset of BWW concepts presented in Table 3.11 allows to perform a representational analysis of BPMN and Qualigram notations but focused in the SRE context. The results of these representational analyses are presented in the next subsection.

3.3.1.4 Qualigram and BPMN mappings

This subsection presents the results of the research activity A4 of the research methodology (refer to Figure 2.3). For performing activity A4 it was necessary to use the results of activity A3 of Figure 2.3: that is the representational analysis of Qualigram notation. The results of activity A3 can be found in Appendix VI. In addition, for performing activity A4 it is also neccesary to use the representational analysis of BPMN. This thesis uses the results of the representational analyis performed by Recker *et al.* (2005; 2006) – see Appendix VII.

From the BPMN representational analysis performed by Recker *et al.*, only those results that correspond to the subset of BWW representation model concepts listed in Table 3.11 have been selected to be used in this section. Consequently, we use in this section only the results of the representational analysis of Qualigram notation that correspond to the same subset of BWW representation model concepts.

Table 3.12 presents the resulting mappings identified for BPMN[24] and Qualigram notation. Since the specific set of BWW concepts presented in Table 3.11 has been used for these mappings, then, according to the theoretical framework discussed in section 3.3, the mappings presented in Table 3.12 are more relevant to the software requirements elicitation context than the full mappings presented in Appendix VI and Appendix VII.

---

[24] The results of the representation mapping of BPMN are obtained from the following research work: (Recker *et al.*, 2005). The BPMN version analyzed by Recker *et al.* is version 1.0. Prezel, Gašević and Milanović (2010) have updated the analysis to BPMN version 2.0; which maintains all the completeness deficiencies depicted in Table 3.12. That is, BPMN 2.0 lacks of modeling constructs to represent the following BWW representation model concepts: *conceivable state space, state law, conceivable event space and lawful event space*. Therefore, the use of BPMN 2.0 does not change the results presented in Table 3.12.

Table 3.12 Comparison of the completeness of BPMN and Qualigram to represent the subset of the BWW representation model concepts selected for the SRE context

Cluster	BWW concepts	BPMN	Qualigram
Things including properties and types of things	Thing	v	v
States assumed by things	Conceivable state space		
	State law		
Events and transformations occurring on things	Event	v	v
	Conceivable event space		
	Lawful event space		
	Transformation	v	v
	Lawful transformation	v	v
	Acts on	v	v
	Coupling	v	v
Systems structured around things	Subsystem	v	v
	Level structure	v	v

Observe in Table 3.12 that BPMN and Qualigram notations present the same incompleteness deficiencies to represent the selected set of BWW representation model concepts. This means that both notations lack modeling constructs to represent the following BWW representation model concepts: *conceivable state space, state law, conceivable event space, and lawful event space*. Recent studies have shown that most of the popular BPM notations lack modeling constructs to represent these BWW concepts (Recker *et al*., 2009; Rosemann *et al.*, 2009). All these BWW representation model concepts have been interpreted, and confirmed through empirical research, as concepts that allow the business rules of an organization to be described (Recker *et al*., 2005; 2006). The inability of a BPM notation to represent these concepts explains the difficulty that modelers have when trying to represent business rules in a business process model (Green, Rosemann and Ipswich, 2001; Recker *et al*., 2006; Rosemann *et al*., 2009). As a consequence, several researchers are currently working on the representation of business rules in business process models (Goedertier and Vanthienen, 2007; Milanovic, Gasevic and Rocha, 2011; zur Muehlen and Indulska, 2010), and therefore we do not cover in this thesis this deficiency.

From Appendix VI and Appendix VII it is also possible to find out which BPMN and Qualigram notation modeling constructs can be used to respresent each of the BWW representation model concepts for which there is a tick on Table 3.12. These findings are presented in Table 3.13.

Table 3.13 Focused representation mappings of BPMN and Qualigram based on the subset of the BWW representation model concepts selected for the SRE context

BWW concept	BPMN modeling constructs	Qualigram modeling constructs
Thing	Lane, pool	External entity, external role, internal role, unit, document, physical tool, target role, source role
Event	Start event, intermediate event, end event, message, timer, error, cancel, compensation, terminate	Start event, down-stream action, end event, up-stream action
Transformation	Activity, task, collapsed sub-process, expanded sub-process, nested sub-process, transaction	Process, sub-process, procedure, work instruction, sub-procedure, macro-instruction, collaborative instruction, alternative instruction, operation, macro-operation, alternative operation, control operation, corrective operation
Lawful transformation	Default flow, uncontrolled flow, exception flow	Information, up-stream action, down-stream action, control line
Acts on	Message flow	Information, up-stream action, down-stream action, control line
Coupling	Message flow	Information, up-stream action, down-stream action, control line
Subsystem	Pool, lane	Process, sub-process, procedure, action, instruction, sub-procedure, macro-instruction, alternative instruction, macro-operation, alternative operation.
Level structure	Pool, lane	Process, sub-process, procedure, action, instruction, sub-procedure, macro-instruction, alternative instruction, macro-operation, alternative operation.

The results presented in Table 3.13 are useful for refining the set of BPM+ modeling concepts. Observe that the BWW concepts *thing* and *transformation* present several redundancies both in BPMN and Qualigram notation. The *a priori* version of BPM+ has also introduced the same type of redundancies because BPM+:

1. calls for differentiating between external stakeholder, role, procedure owner, department, physical tool and document; all these concepts corresponding to the BWW concept *thing*; and
2. requires a structured decomposition of a business process in order to support the MLA hierarchy; thus, the *transformation* BWW concept corresponds to the process, sub-process, procedure, sub-procedure, activity, and task BPM^+ modeling concepts.

The *subsystem* and *level of structure* BWW concepts also present redundancies especialy in the Qualigram notation. Again, these redundancies have been also introduced in the *a priori* version of BPM^+ to support the MLA hierarchy of BPM^+.

3.3.2 Discussion of the results

Some relevant SRE concepts selected from the SWEBOK or the BABOK do not appear in Table 3.11 as they could not be mapped to the BWW representation model. One candidate explanation for this could be that the BWW representation model is incomplete in terms of fully describing the knowledge of the software requirements elicitation context. Among these concepts, *goals* and *objectives* were the only concepts to belong to both SWEBOK and BABOK (refer to Table 3.4).

In terms of overloading, it can be observed from Table 3.11 that the most overloaded BWW representation model concepts are: *thing*, *state law*, *transformation*, *acts on*, and *coupling*. Regarding the overload of *thing* and *transformation*, we have already suggested in the previuos subsection that there might be a need in having specializations for these two BWW concepts in order to support the design of BPM^+.

Concerning the overload of *thing*, Recker *et al*. (2005; 2006) have stated that BPMN presents a number of redundant modeling constructs (i.e. *lane* and *pool*) to represent things (see Table 3.13). They argue that this might cause confusion among modelers. Their empirical study confirmed that some confusion does result when those modeling BPMN constructs are used.

However, this confusion might originate from other factors, such as: 1) the apparent similarity between the modeling constructs *lane* and *pool*; and 2) the poor definitions of these two modeling constructs. The latter has been confirmed as a potential cause of confusion by the empirical results of Recker *et al*. Table 3.13 reveals that Qualigram notation has a richer – and well differentiated – set of modeling constructs than those of BPMN for representing different types of *things*. The initial findings of our pilot case study seem to suggest that there is some use in having such a specialization for *things*.

Regarding the overload of *transformation* found in Table 3.11, this coincides with the observed results already published by previous theoretical and empirical studies that evaluated various BPM notations using the BWW representation model (Recker *et al.*, 2009; Rosemann *et al.*, 2009). These previous results have also suggested a need to specialize the BWW concept *transformation*, since various BPM notations have been designed with specialized modeling constructs to represent *transformations*, and some participants in their empirical studies have confirmed the usefulness of such specialization.

Concerning the overload of *state law*, this has been observed, and confirmed through empirical research, as a difficulty for modelers attempting to represent business rules in business process models (Green, Rosemann and Ipswich, 2001; Recker *et al.*, 2006; Rosemann *et al.*, 2009). As it was indicated in previous subsection, this thesis does not cover this type of overload, since there are several researchers currently working on the representation of business rules in business process models.

Finally, the overload of *acts on* and *coupling* is generated by the explicitness of the SWEBOK and the BABOK in defining various types of interactions, dependencies, or relationships between users, stakeholders, roles, entities, software components, and the environment of the system represented. According to the BWW representational analysis theoretical framework, the stakeholder interpreting a business process model may be confused by the various meanings adopted by the BPM notation to represent these two BWW concepts (i.e. *acts on* and *coupling*). The BWW concepts *acts on* and *coupling* also contribute to a redundancy in the BWW concepts. Any type of interaction or dependency (e.g. interaction between users) may be described either by the *coupling* concept or by the *acts on* concept. This choice hinders the modeling of a business process because, for instance, a modeler has to decide whether to choose a modeling construct that represents the *coupling* concept or a modeling construct that represents the *acts on* concept to describe an interaction between users. Notice that BPM^+ calls for two types of interactions: 1) relationships between actions; and 2) relationships between actions and entities (refer from Table 3.1 to Table 3.3). Therefore, the *a priori* version of BPM^+ also introducres redundancies related to the BWW concepts *acts on* and *coupling*.

All these findings, as well as those from the pilot case study, will serve as a basis for the formultation of a set of propositions to be tested through the survey that is described in the next section.

3.4 A survey of practitioners with experience

This section presents the findings from a survey conducted with experienced practitioners in order to test a set of propositions formulated with the intention to support:

1. some criteria in the design of the *a priori* version of BPM^+; and
2. some refinements in the design of BPM^+.

These propositions are based on the results of the representational analyses presented in the previous section, and on the results of the pilot case study presented in section 3.2. The research design of the survey has already been discussed in section 2.4. The survey questionnaire can be found in Appendix VIII.

3.4.1 The research propositions

Based on the findings of the initial case study reported in section 3.2, a set of five propositions is formulated:

P1: Modeling business processes at the strategic, tactical and operational levels of abstraction contributes to generating consistent business process models that can be shared by the various groups of stakeholders.

P2: A business process model at the strategic level of abstraction eases the communication to customers, non-IT employees, and new employees of the business processes represented.

P3: The macroscopic and detailed types of models are the most useful types of models from the four types of models offered by Qualigram at the strategic level of abstraction.

P4: Qualigram notation is preferred over BPMN notation by practitioners to model business processes when the target user is a customer, a non-IT employee, a new employee, or a management-oriented stakeholder.

P5: BPMN notation is preferred over Qualigram notation by practitioners to model business processes when the target user is an IT-oriented stakeholder or a business analyst.

Based on the findings of the represenational analyses reported in section 3.3, an additional set of four propositions is formulated:

P6: Practitioners require representing *goals* and *objectives* in a business process model intended to be used for software requirements elicitation.

P7: Practitioners require specialized modeling constructs to represent each of the following concepts: roles (i.e. internal users), external stakeholders (e.g. customers, providers), devices, objects, departments (i.e. business units), software interfaces and software components.

P8: Practitioners require specialized modeling constructs to represent each of the following concepts[25,26]: actions (i.e. tasks), activities, sub-processes, and processes.

P9: Practitioners require specialized modeling constructs to represent each of the following types of interactions: interactions between roles, interactions between roles and external stakeholders, interactions between software components, interactions between roles and software components, interactions between external stakeholders and software components, interactions between objects, interactions between business units.

3.4.2 Data analysis

The three major variables that define the demographics of the survey participants are:

1. their profession or job function;
2. their years of experience in BPM; and
3. their years of experience in SRE.

Table 3.14 Participants BPM experience

BPM experience	Frequency	Percentage	Cumulative Percentage (CP)
15 years or more	1	6%	6%
6 to 15 years	5	29%	35%
2 to 5 years	3	18%	53%
Less than 2 years	8	47%	100%
Grand Total	17	100%	

Table 3.14 shows the participants BPM experience, and Figure 3.11 depicts the distribution of the 17 participants according to their profession and BPM experience. It can be noticed that over half of the participants (53%) have more than 2 years of experience in BPM.

---

[25] This proposition was formulated based on the terminology proposed by the BABOK (refer to Table 3.9) where a business process is decomposed into sub-processes, a sub-process is decomposed into activities, and an activity is decomposed into actions or tasks. Using SWEBOK terminology, then a business process is decomposed into activities and an activity is decomposed into tasks. BABOK terminology was preferred because it is closer to BPM^{+}'s proposal of how to decompose a business process.

[26] The survey participants were explained about the hierarchy of the concepts that are part of proposition P8.

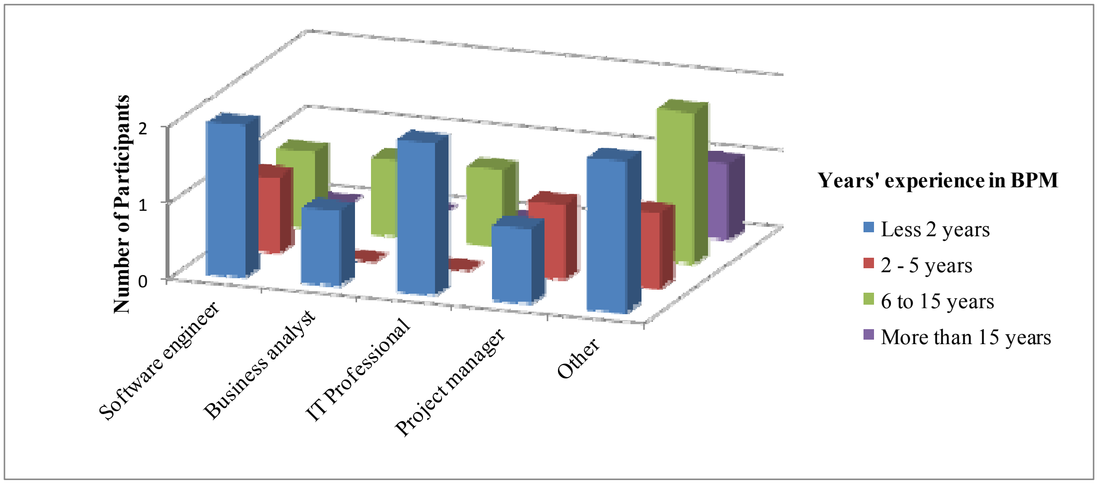

Figure 3.11 Participants' demographics (profession & BPM experience)

Regarding the profession or job function, 6 out of 17 have been classified under "other", of whom:

1. three indicated to perform several professions, even though the corresponding question explicitly solicited to choose the answer that best describes their profession or job function;
2. one is a Ph.D. researcher; and
3. the other two indicated to be respectively a measurement consultant and a process improvement specialist.

Table 3.15 Participants SRE experience

SRE experience	Frequency	Percentage	CP
15 years or more	1	6%	6%
6 to 15 years	7	41%	47%
2 to 5 years	5	29%	76%
Less than 2 years	4	24%	100%
Grand Total	17	100%	

Table 3.15 shows the participants SRE experience, and Figure 3.12 depicts the distribution of the participants according to the profession or job function and the years of experience in

SRE. It can be noticed in the comulative percentage (CP) colum that 47% of the participants have more than 6 years of experience in SRE.

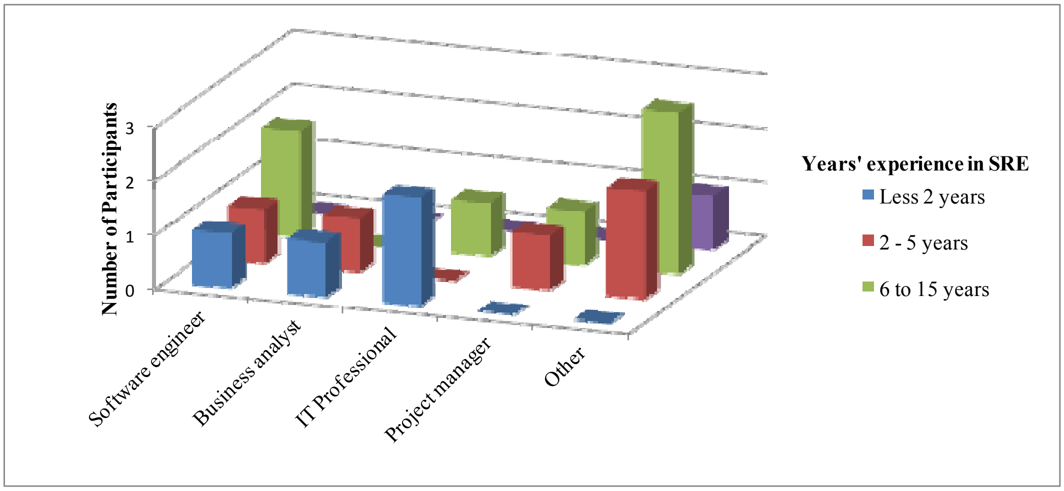

Figure 3.12 Participants' demographics (profession & SRE experience)

Regarding the use of levels of abstraction in BPM (i.e. **proposition P1**), 82% of the participants expressed the need to represent business processes at the strategic, tactical and operational levels of abstraction: 65% of the participants need to model at these 3 levels of abstraction to be able to represent a general view of a business process (see Table 3.16), in addition to a detailed view of it. The next two most common reasons are: a) to ease the sharing of the business process between various stakeholders (53%) and b) to be able to

Table 3.16 Most important benefits of modeling at multiple levels of abstraction

Perceived benefit	Not perceived as a benefit		Perceived as a benefit		No answer		Grand Total	
	Freq.	Percent	Freq.	Percent	Freq.	Percent	Freq.	Percent
Various levels of detail	10	59%	4	24%	3	18%	17	100%
General and detailed views	3	18%	11	65%	3	18%	17	100%
Consistent and structured	7	41%	7	41%	3	18%	17	100%
Organizational pyramid	13	76%	1	6%	3	18%	17	100%
Ease the sharing of BPM	5	29%	9	53%	3	18%	17	100%

Number of respondents (N) = 17

represent in a consistent and structured way the business processes (41%). These findings support proposition P1.

Table 3.17 presents the summary of the ranking performed by the survey's participants to the level of importance (i.e. High = 3; Medium = 2; and Low = 1) of modeling at various levels of abstraction (i.e. Higher; Medium; and Lower) according to the type of stakeholder. The

Table 3.17 Level of abstraction according to type of stakeholder

Type of stakeholder	Level of abstraction	Frequency of importance rank			Total	Scaled total	Median	Mode
		High (3)	Med. (2)	Low (1)				
Customers	Higher	10	2	2	14	36	High	High
	Medium	3	10	0	13	29	Med	Med
	Lower	3	1	10	14	21	Low	Low
Providers	Higher	5	3	6	14	27	Med	Low
	Medium	3	9	0	12	27	Med	Med
	Lower	8	0	6	14	30	High	High
Non-It employees	Higher	11	2	2	15	39	High	High
	Medium	3	7	2	12	25	Med	Med
	Lower	1	3	8	12	17	Low	Low
New employees	Higher	10	1	3	14	35	High	High
	Medium	3	9	1	13	28	Med	Med
	Lower	2	2	8	12	18	Low	Low
Managers	Higher	14	1	1	16	45	High	High
	Medium	2	12	1	15	31	Med	Med
	Lower	1	2	11	14	18	Low	Low
IT employees	Higher	1	2	9	12	16	Low	Low
	Medium	5	9	1	15	34	Med	Med
	Lower	10	3	2	15	38	High	High
IT consultants	Higher	3	1	9	13	20	Low	Low
	Medium	2	10	2	14	28	Med	Med
	Lower	9	2	2	13	33	High	High
Business analysts	Higher	7	2	5	14	30	H-M	High
	Medium	2	10	3	15	29	Med	Med
	Lower	8	2	5	15	33	High	High

scaled total presented in Table 3.17 has been calculated as a weighted sum of the frequencies for the High, Medium and Low ranks of importance, where the weights are 3, 2 and 1 respectively. Notice in column "Total" of Table 3.17 that not all the respondents (N=17) ranked the importance of the levels of abstraction for all the types of stakeholders: that is the reason why the column "Total" presents values that range from 12 to 16.

Notice in Table 3.17 that the four types of stakeholders that present the highest scaled total for the higher level of abstraction are: customers, non-IT employees, new employees and managers (36, 39, 35 and 45 respectively). In addition, these are the only four types of stakeholders whose median and mode for the higher level of abstraction is *High*. Cross tabulation and Pearson Chi-square test of independence were performed to examine whether a relationship exists between the types of stakeholders and the importance of modeling at a higher level of abstraction. To prevent cells from having an expected frequency lower than five wich, otherwise, would violate the acceptability requirements for this statistical technique, we classified the types of stakeholders into two groups (see Table 3.18): Group A (i.e. customers, non-IT employees, new employees and managers) and Group B (i.e. providers, IT employees, IT consultants and business analysts). Each cell in Table 3.18 corresponds to the sum of frequencies (from Table 3.17) of a given rank of importance of modeling at a higher level of abstraction for each of the types of stakeholders included in a given Group.

Table 3.18 Contingency table for the higher level of abstraction

Groups of stakeholders	**Group added frequencies of the importance rank**			**Total**
	High (3)	**Med. (2)**	**Low (1)**	
Group A	45	6	8	59
Group B	16	8	29	53
Total	61	14	37	112
p-value<0.01 N=17	χ^2=25.7432			*df*=2

The null hypothesis is that the two groups of stakeholders (i.e. Group A and Group B) and the importance of modeling at a higher level of abstraction are statistically independent. As the *p*-value of the Pearson Chi-square is below the 0.01 level (p=2.57x10^{-6}), statistical evidence was found to confirm that a relation exists between the groups of stakeholders and the importance of modeling at a higher level of abstraction. Therefore, customers, non-IT employees, new employees, and managers are the types of stakeholders to whom the survey participants found more important to communicate business processes using BPM at a higher level of abstraction (refer to Figure 3.13, where “h” stands for higher level of abstraction, “m” for medium level of abstraction, and “l” for lower level of abstraction). These findings support **proposition P2**.

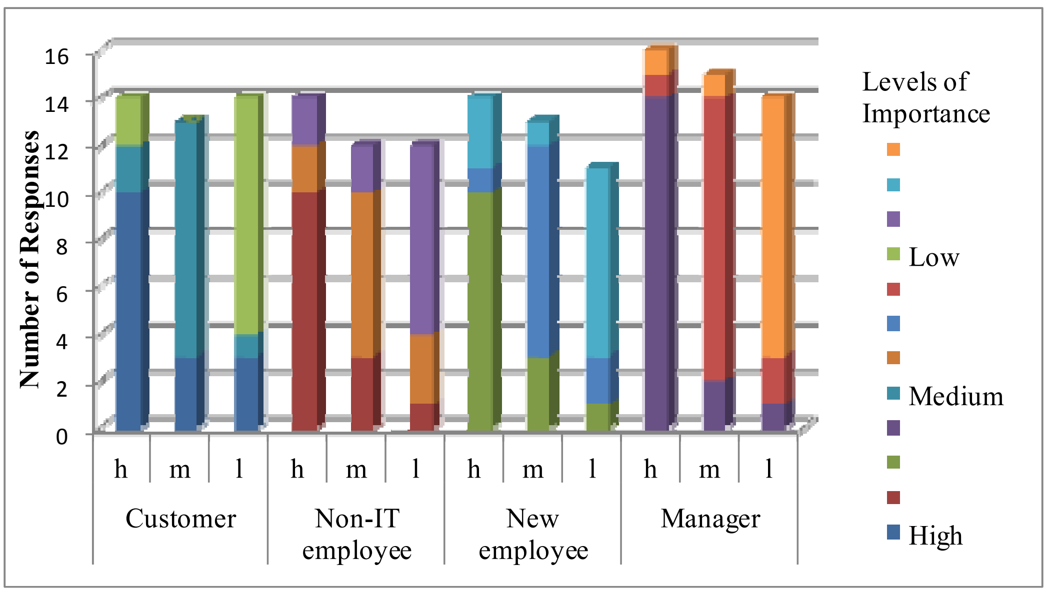

Figure 3.13 Types of stakeholders to whom it is more important to communicate BPM at a higher level of abstraction

Table 3.19 and Figure 3.14 depict the preferences regarding the usefulness of the various types of business process models that Qualigram notation allows to represent at the strategic level of abstraction. According to their frequencies, the macroscopic (56%) and the relational (29%) types of models are considered as the most useful. We decided not to perform a Pearon Chi-square test of independence to further examine the data presented in Table 3.19

because the table presents three cells with a frequency lower than five, wich violates the acceptability requirements for this statistical technique. However, our survey also included dichotomous questions (i.e. yes or no questions) that asked the participants to answer if they need these three types of models at the strategic level of abstraction. We analyze the results of these questions in the next paragraph.

Table 3.19 Usefulness of Qualigram's types of models at the strategic level of abstraction

Type of Model	Most useful		Second most useful		Less Useful		Total	
	Freq.	Percent	Freq.	Percent	Freq.	Percent	Freq.	Percent
Macroscopic	9	56%	3	19%	4	25%	16	100%
Relational	5	29%	8	47%	4	24%	17	100%
Detailed	3	19%	5	31%	8	50%	16	100%

N=17 respondents

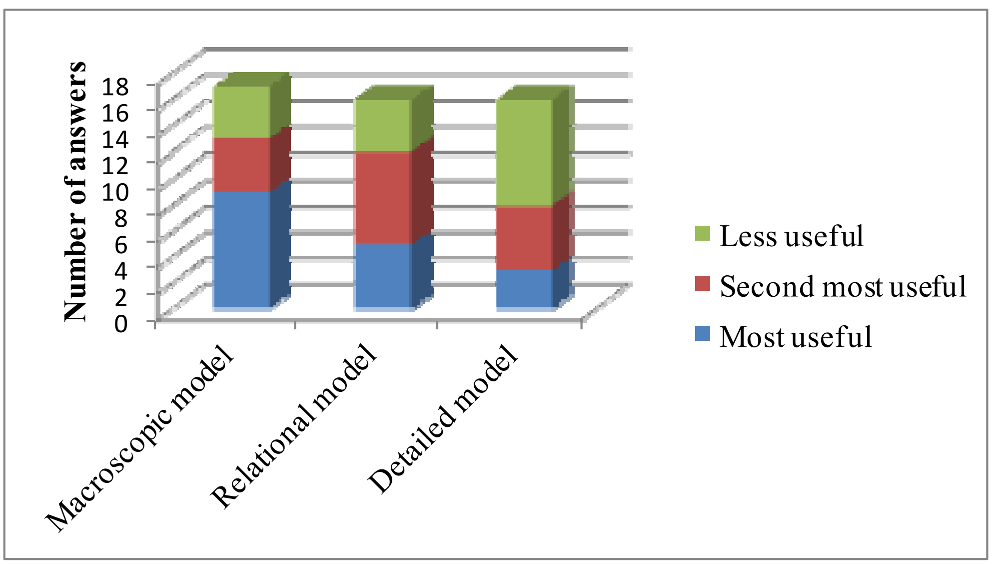

Figure 3.14 Usefulness of Qualigram's types of models at the strategic level of abstraction

Table 3.20 presents the frequencies of the answers of our 17 participants (N=17) when asked if they need each of the types of models offered by Qualigram notation at the strategic level of abstraction. Notice in Table 3.20 that the macroscopic and the relational types of models present the highest frequencies of need: 88% and 59% respectively. Therefore, these results concur with the results presented in Table 3.19 and Figure 3.14. These findings partially support **proposition P3**, which proposes the macroscopic and detailed types of models as the most useful.

Table 3.20 Need of the types of models at the strategic level of abstraction

	Macroscopic		Relational		Detailed	
	Frequency	Percentage	Frequency	Percentage	Frequency	Percentage
Needed	15	88%	10	59%	7	41%
Not needed	2	12%	7	41%	10	59%
Total	17	100%	17	100%	17	100%

Table 3.21 shows the preferences of the participants regarding the use of Qualigram notation or BPMN to communicate business processes to the various types of stakeholders. The majority of participants (more than 59%) prefers Qualigram notation over BPMN notation to communicate business processes to the following types of stakeholders: customers, providers, non-IT employees, new employees, managers, and project managers. If we consider that the frequencies of "Any of the two" BPM notations can be added to both

Table 3.21 BPM notation preferences according to the type of stakeholder

Type of stakeholder	Qualigram		BPMN		Any of the two		Undecided		Total	
	Freq.	Percent	Freq.	Percent	Freq.	Percent		Freq.	Freq.	Percent
Customer	10	59%	4	24%	2	12%	1	6%	17	100%
Provider	10	59%	3	18%	2	12%	2	12%	17	100%
Non-IT employee	10	59%	3	18%	4	24%	0	0%	17	100%
New employee	10	59%	3	18%	3	18%	1	6%	17	100%
Manager	10	59%	4	24%	3	18%	0	0%	17	100%
Administrative	8	47%	4	24%	5	29%	0	0%	17	100%
IT employee	4	24%	7	41%	6	35%	0	0%	17	100%
IT consultant	4	24%	5	29%	8	47%	0	0%	17	100%
Business consultant	8	47%	5	29%	4	24%	0	0%	17	100%
Business analyst	8	47%	5	29%	4	24%	0	0%	17	100%
Project manager	11	65%	0	0%	6	35%	0	0%	17	100%
Quality assurance	7	41%	2	12%	7	41%	1	6%	17	100%

Qualigram and BPMN frequencies of preference (see Table 3.22), then we can add the administrative employees to the group of types of stakeholders to whom over 59% of the participants prefer Qualigram notation to communicate business processes. That is, this group of stakeholders include: customers, providers, non-IT employees, new employees, managers, administrative employees, and project managers.

Table 3.22 Aggregated BPM notation preferences according to the type of stakeholder

Type of stakeholder	Qualigram		BPMN		Undecided		Total	
	Freq.	Percent	Freq.	Percent	Freq.	Percent	Freq.	Percent
Customer	12	63%	6	32%	1	5%	19	100%
Provider	12	63%	5	26%	2	11%	19	100%
Non-IT employee	14	67%	7	33%	0	0%	21	100%
New employee	13	65%	6	30%	1	5%	20	100%
Manager	13	65%	7	35%	0	0%	20	100%
Administrative	13	59%	9	41%	0	0%	22	100%
IT employee	10	43%	13	57%	0	0%	23	100%
IT consultant	12	48%	13	52%	0	0%	25	100%
Business consultant	12	57%	9	43%	0	0%	21	100%
Business analyst	12	57%	9	43%	0	0%	21	100%
Project Manager	17	74%	6	26%	0	0%	23	100%
Quality assurance	14	58%	9	38%	1	4%	24	100%

If we name the group of stakeholders identified in the previous paragraph Group A, and we group the rest of the types of stakeholders (i.e. IT-oriented employees, IT consultants, business consultants, business analysts and quality assurance managers) into a second group (i.e. Group B), we can perform the Pearson Chi-square test of independence. By doing so, we can determine whether a relationship exists between these two groups of types of stakeholders and the preference of BPM notation (Qualigram vs. BPMN). Therefore, the null hypothesis is that these two groups of types of stakeholders and the preference of BPM notation are statistically independent. As the *p*-value (p=0.0229) of the Pearson Chi-square is below the 0.05 level, statistical evidence was found to confirm that a relation exists between the two groups of types of stakeholders and the preference of BPM notation (see Table 3.23).

Therefore, it is possible to conclude that Qualigram notation is prefered over BPMN notation to communicate business processes to the types of stakeholders that belong to Group A. These findings support **proposition P4**.

Table 3.23 Pearson Chi-square test of independence: groups of stakeholders vs. BPM notation preference

Groups of stakeholders	**Frequencies of preference of BPM notation according to the groups of stakeholders**		**Total**
	Qualigram	**BPMN**	
Group A	94	46	140
Group B	60	53	113
Total	154	99	253
p-value<0.05 N=17	χ^2=5.179		*df*=1

On the other hand, there is no such a clear preference for the use of BPMN notation when modeling business processes for the types of stakeholders included in **proposition P5** (see Figure 3.15 and Table 3.22). According to the participants, BPMN notation presents an

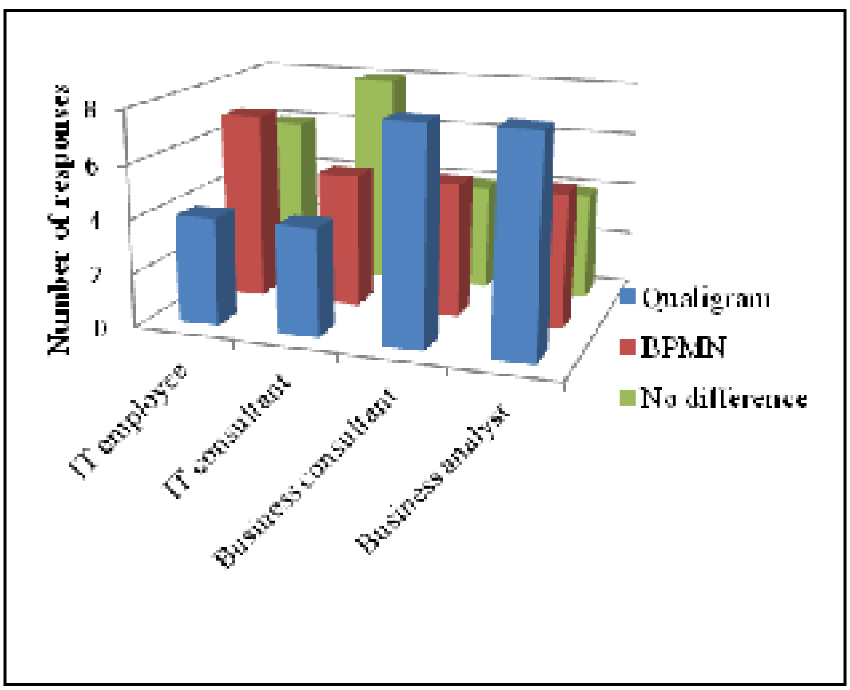

Figure 3.15 BPMN vs. Qualigram preferences

advantage over Qualigram notation when the target users are IT-employees (57%) or IT-consultants (52%); however, 57% of the participants perceive an advantage of Qualigram notation over BPMN notation when the target users are business analysts or business consultants. These findings present a partial support to **proposition P5**, which proposed the preference of BPMN by IT-oriented stakeholders and by business analysts.

65% of the participants agreed that *goals* and *objectives* are necessary to be represented in a business process model. Moreover, 70% of the participants who agreed on that need indicated that not being able to directly represent *goals* or *objectives* in a business process model constitutes a problem. These findings support **proposition P6**.

Table 3.24 Importance of the need of BPM constructs to represent *things*

Type of *thing*	**Essential**			**Not essential but use it**			**Not use it**			**Not understand it**			**Not aware of it**		
	F	**%**	**CP**	**F**	**%**	**CP**	**F**	**%**	**CP**	**F**	**%**	**CP**	**F**	**%**	**CP**
Internal role	8	47%	47%	6	35%	82%	2	12%	94%	1	6%	100%	0	0%	100%
External stakeholder	11	65%	65%	4	24%	88%	1	6%	94%	0	0%	94%	1	6%	100%
Business unit	11	65%	65%	5	29%	94%	1	6%	100%	0	0%	100%	0	0%	100%
Device	7	47%	47%	5	33%	80%	1	7%	87%	1	7%	93%	1	7%	100%
Object	6	38%	38%	5	31%	69%	2	13%	81%	2	13%	94%	1	6%	100%
Software interface	8	47%	47%	4	24%	71%	1	6%	76%	2	12%	88%	2	12%	100%
Software component	8	47%	47%	4	24%	71%	1	6%	76%	2	12%	88%	2	12%	100%

Legends: F=frequency; %=percentage; CP=cumulative percentage.

Table 3.24 shows that 65% of the participants have expressed the need of a BPM construct specifically designed to represent the following concepts: external stakeholders and business units (i.e. departments). This need was expressed by 47% of the participants for the following concepts: internal roles, devices, software interfaces and software components. If we observe the comulative percentage (CP) of the “Not essential but use it” column of Table 3.24, then we find that over 80% of the participants express the need to use the following concepts but not necessarily representing them with a BPM construct specifically designed for that purpose: internal roles, external stakeholders, business units and devices. These findings partially support **proposition P7**.

Table 3.25 shows that over 69% of the participants expressed the need of a BPM construct specifically designed to represent the following concepts: tasks, activities, sub-processes, and processes. This finding supports **proposition P8**.

Table 3.25 Importance of the need of BPM constructs to represent actions

Type of action	Essential			Not essential but use it			Not use it			Not understand it			Not aware of it		
	F	%	CP	F	%	CP	F	%	CP	F	%	CP	F	%	CP
Tasks	11	73%	73%	4	27%	100%	0	0%	100%	0	0%	100%	0	0%	100%
Activities	12	75%	75%	4	25%	100%	0	0%	100%	0	0%	100%	0	0%	100%
Sub-processes	11	69%	69%	4	25%	94%	0	0%	94%	1	6%	100%	0	0%	100%
Processes	12	75%	75%	3	19%	94%	0	0%	94%	1	6%	100%	0	0%	100%

Legends: F=frequency; %=percentage; CP=cumulative percentage.

Table 3.26 shows that over 53% of the participants expressed the need of a BPM construct specifically designed to represent the following concepts: interaction between roles, and interaction between roles and external stakeholders. This need was expressed by over 41% of the participants for the following concepts: interactions between software components, interactions between roles and software components, and interactions between business units. If we observe the comulative percentage (CP) of the "Not essential but use it" column of Table 3.26, then we find that over 80% of the participants express the need to use the

Table 3.26 Importance of the need of BPM constructs to represent relationships and dependencies

Type of interactions	Essential			Not essential but use it			Not use it			Not understand it			Not aware of it		
	F	%	CP	F	%	CP	F	%	CP	F	%	CP	F	%	CP
Between roles	9	56%	56%	7	44%	100%	0	0%	100%	0	0%	100%	0	0%	100%
Between roles and external stakeholders	9	53%	53%	6	35%	88%	1	6%	94%	0	0%	94%	1	6%	100%
Between software components	7	41%	41%	7	41%	82%	1	6%	88%	2	12%	100%	0	0%	100%
Between roles and software components	7	44%	44%	7	44%	88%	0	0%	88%	2	13%	100%	0	0%	100%
Between external stakeholders & software components	6	38%	38%	8	50%	88%	1	6%	94%	1	6%	100%	0	0%	100%
Between business units	7	41%	41%	7	41%	82%	2	12%	94%	1	6%	100%	0	0%	100%
Between objects	5	31%	31%	7	44%	75%	0	0%	75%	3	19%	94%	1	6%	100%

Legends: F=frequency; %=percentage; CP=cumulative percentage.

following concepts but not necessarily representing them with a BPM construct specifically designed for that purpose: interactions between roles, interactions between roles and external stakeholders, interactions between software components, interactions between roles and software components, interactions between external stakeholders and software components, and interactions between business units. These findings partially support **proposition P9**.

Finally, from the nine propositions originally formulated to guide this survey, five have been supported by the results of the survey, and four have been partially supported. Table 3.27 summarizes these survey results.

Table 3.27 Summary of survey results

	Propositions	Observations
Supported	P1, P2, P4, P6 and P8	
Partially supported	P3, P5, P7 and P9	Qualigram's detailed type of model was not found as useful as the relational type of model. BPMN presents an advantage over Qualigram notation for IT-oriented stakeholders. Qualigram notation is preferred for business analysts. Not all the types of *things* require a specifically designed modeling construct. Only some types of interactions require a specifically designed modeling construct.

3.4.3 Conclusion

The results confirm that BPM at the strategic, tactical and operational levels of abstraction contributes to generate consistent business process models that can be shared by the various groups of stakeholders. The strategic level of abstraction is particularly useful for communicating business processes to non-IT stakeholders and new employees. Moreover, practitioners perceive that for these types of stakeholders it is better to represent a business process using Qualigram notation.

The findings from this survey also confirm that practitioners perceive as very useful the macroscopic type of model offered by Qualigram notation at the strategic level of abstraction. Regarding the detailed and relational types of model, there is a difference between the results obtained in the pilot case study with those obtained from the survey. The pilot case study indicated that practitioners preferred the detailed type of model; however, the survey results show that practitioners prefer the relational type of model. Thus, further empirical research needs to be conducted in order to study practitioners' preferences regarding the relational and detailed types of models. Therefore, this will be one of the research goals for our second case study.

Regarding practitioners' preference of BPM notations, there is again a difference between the results obtained in the pilot case study with those obtained from the survey. Therefore, further empirical research should be conducted to study the preference of BPMN notation over Qualigram notation. Thus, this will be another research goal for our second case study

Representing *goals* and *objectives* in a business process model has been identified as a need if that model is intended to be used for software requirements elicitation. Not being able to represent *goals* and *objectives* is perceived as a problem.

The specialization of the BWW concept *thing* is perceived as necessary to differentiate between two concepts that are relevant to software requirements elicitation: external stakeholders and departments. In addition, the results of the survey show that such a specialization is not relevant for the concepts: objects and devices.

The specialization of the BWW concept *transformation* is perceived as necessary to differentiate between various concepts that are relevant to software requirements elicitation: processes, sub-processes, activities and tasks.

Some importance is reflected in the survey results regarding the ability to represent, with a BPM construct specifically designed for that purpose, the interactions between internal roles and the interactions between internal roles and external stakeholders. In addition, the survey results show that it is not relevant to represent in a distinctive way the interactions between external stakeholders and software components or the interactions between objects.

Taking these findings into account, as well as those from the pilot case study and the representational analyses, the design of BPM$^+$ can be refined. The next section presents a reviewed version of BPM$^+$. This new version will be empirically tested with our second case study (section 3.6).

3.5 The BPM$^+$ reviewed version

Table 3.28 and Table 3.29 present a summary of the findings obtained from the pilot case study, the representational analyses and the survey of practitioners with experience in BPM and SRE. All of these findings constitute the basis for refining our *a priori* version of BPM$^+$.

The BPM$^+$ levels of abstraction (i.e. strategic, tactical and operational levels of abstraction) have been supported by these findings (see Table 3.28) and therefore are kept in the refined version presented here. The need of decomposition of a business process has been also supported; therefore, the refined version presented in this section maintains the decomposition of a business process into procedures, a procedure is further decomposed into activities, and an activity is decomposed into a set of atomic tasks. When necessary, a process can be decomposed into sub-processes and a procedure into sub-procedures. In the following subsections we present the refined version of each of the levels of abstraction of BPM$^+$ following a similar structure than the one we used to present the *a priori* version in subsections 3.1.1 to 3.1.3. In the presentation structure of the reviewed version we add the subsection “Notes for users” after the “Scope” subsection of each level of abstraction to explain the details related to the modeling preferences according to the target audience of the models to be generated based on BPM$^+$. Where necessary, the change introduced to the *a* *priori* version will be noted and justified. Those parts from the *a priori* version that do not need to be changed are not repeated here.

Table 3.28 BPM$^+$ aspects reviewed during the research activities: Part A

Issue		Pilot case study	Representational analysis	Survey of experts
Usefulness of modeling at three levels of abstraction	Strategic level	Supported, and value identified	Supported: decomposition of a business process.	Supported, and value identified
	Tactical level	Supported, and value identified		
	Operational level			
Relevance of various types of strategic level models	Macroscopic	Supported, and reasons identified		Supported
	Relational	No relevant		Supported
	Detailed	Supported, reasons, and modeling recommendations identified		Less relevant
	Transversal	Discarded		Discarded
BPM$^+$ modeling constructs	Representation of various types of relationships between the different types of roles	Supported	Supported	Supported (interactions between roles, interactions between roles and external stakeholders)
	Representation of goals and objectives		Supported	Supported
	Representation of various types of roles	Supported	Supported	Supported (external stakeholders and departments)
	Representation of various types of actions	Supported	Supported	Supported (tasks, activities, sub-processes, processes)
	Representation of relationships between roles and software components, and between objects	Not proved to be necessary.		Not supported
	Representation of business rules		Need to be added	

Table 3.29 BPM+ aspects reviewed during the research activities: Part B

Issue		Pilot case study	Representational analysis	Survey of experts
BPM+ at the tactical level	Use of document, physical tool and responsibility	Supported	Representation of objects supported	Not necessary to have a specific modeling construct to represent them.
Qualigram at the tactical level	Use of events and up-stream/down-stream actions	Supported the use of all of them	Supported (events)	
	Use of macro-instruction and collaborative instruction	Useful but might be replaced by alternative ways of modeling		
	Target role and source role	Supported		
Use of BPMN at the strategic level	Use of Levels 1 and 2 of BPMN use.	Level 1 of BPMN use supported but with restricted use of lanes. Include use of events and gateways. Considered relevant		
	Sub-process representation	Use collapsed sub-processes.		
	Procedure representation	Use collapsed sub-processes.		
	Relationships representation	Use sequence flows and message flows depending on the situation	Supported	Supported
Use of BPMN at the tactical level	Use of Level 1	Supported		
	Use of Level 2	Supported, when it is considered necessary		
Target users of the Qualigram models		More suitable for customers, administrative staff, new employees		More suitable for customers, providers, non-IT employees, new employees, managers, administrative employees and project managers.
Target users of the BPMN models		More suitable for the software development team		More suitable for IT employees and IT consultants.

3.5.1 The BPM+ strategic level of abstraction

The reviewed version of BPM+ keeps the original description and scope of the strategic level of abstraction of its *a priori* version (refer to subsection 3.1.1).

The value of the strategic level of abstraction lies in the ability to allow an organization to expose their customers and their new employees to its business processes.

Notes for users

Modeling at the strategic level of abstraction is particularly useful for customers, non-IT employees, new employees and managers of the organization.

Even though the strategic level of abstraction does not aim at representing any kind of workflow of a business process, in order to ease the understanding of the target users the modeler should consider generating strategic-level models that resemble as close as possible the workflow of the business processes.

Modeling concepts and semantic considerations

The sets of actions, entities, and information (i.e. relationships or dependencies) required for the strategic level of abstraction are the same as those described in the *a priori* version of BPM+. In addition, Table 3.28 shows the need to represent goals in the business process models. The reviewed version of BPM+ introduces some adjustments to Table 3.1 related to the representation of the BPM+ strategic-level modeling concepts by BPMN. Specifically, according to the findings depicted in Table 3.29, there is the need to refine the BPMN representation of a sub-process and a procedure: that is, both BPM+ concepts should be represented in BPMN by a collapsed sub-process. These two small adjustments to Table 3.1 are emphasized in red in Table 3.30.

Table 3.30 Reviewed version of strategic-level BPM$^+$ modeling concepts

BPM$^+$ modeling concepts	Qualigram	BPMN
Process.	Process.	Pool.
Sub-process.	Sub-process.	**Collapsed sub-process.**
Procedure.	Procedure.	**Collapsed sub-process.**
External stakeholder.	External entity.	Pool.
Relationship between actions.	Information.	Sequence flow, message flow.
Relationship between actions and entities.		
Goal.	Performance indicator.	Not available; use text annotation attached with an association.

Notice in Table 3.30 that two BPMN modeling constructs might be used to represent relationships: 1) "sequence flows" that should be used when representing relationships between actions contained within the pool of the business process being modeled; and 2) "message flows" that should be used when representing any of two types of relationships:

- relationships between external stakeholders and the business process being modeled; and
- relationships between the business process being modeled and other business processes.

Syntax considerations

Since there might be various possible perspectives to be modeled at the strategic level of abstraction, then it is possible to have more than one BPM$^+$ top-level model. From all the possible perspectives, there is one that BPM$^+$ proposes to always model as a starting point of any BPM initiative: the macroscopic type of model.

The macroscopic type of model depicts a general high-level view of the core business processes of the organization. This type of model is relevant because it:

1. identifies the main external stakeholders of the organization; and
2. identifies the core business processes of the organization.

The modeler can generate the macroscopic type of model following any approach expected by the organization. For instance, the macroscopic type of model might be represented following the traditional classification of the business processes as core, management and support business processes; another example of an option might be to generate a macroscopic type of model resembling a value chain. Notice that it is not possible to generate a macroscopic type of model using BPMN[27].

Besides the macroscopic type of model, two other perspectives might be generated at the strategic level of abstraction: the relational type of model and the detailed type of model (refer to subsection 3.1.1).

The relational type of model is relevant because it provides:

1. the relationships between each business process and the main external stakeholders; and
2. the relationships between each business process and the other core business processes.

The detailed type of model is relevant because it provides:

1. a high-level model for each core business process;
2. the position of each business process in relation to its own context; and
3. a logical link between the strategic level of abstraction and the tactical level models.

If BPMN is used by the organization to represent its business processes, then the modeler should generate for each business process a strategic-level model based on the "descriptive level of use" (i.e. Level 1) of BPMN (refer to subsection 1.5.2.2). Only the "private business process" and the "abstract process" types of BPMN models (refer to subsection 1.5.2.2) should be used; in any case the model must not include the use of lanes.

[27] If BPMN 2.0 is used for modeling the business processes, then the first BPMN model to generate should be a BPMN collaboration model representing all the core business processes as black boxes (i.e. empty pools) and depicting all the relationships between the core business processes.

3.5.2 The BPM$^+$ tactical level of abstraction

The reviewed version of BPM$^+$ keeps the original description and scope of the tactical level of abstraction of its *a priori* version (refer to subsection 3.1.2).

The value of the tactical level of abstraction lies in the ability to allow an organization to deliver the details of a business process to all the employees of the organization regardless of their professional profiles.

Notes for users

Qualigram models are recommended if the target users are management-oriented professionals, project managers, administrative staff, customers, providers or new employees of the organization. BPMN models are recommended if the target users are IT-consultants, IT-oriented employees or users who are part of a software development team.

Modeling concepts and semantics considerations

The sets of actions, entities, and information (i.e. relationships or dependencies) required for the tactical level of abstraction are the same as those described in the *a priori* version of BPM$^+$.

Based on the findings presented in Table 3.29, BPM$^+$ allows representing a triggering event in Qualigram notation with either a start event or with an up-stream action. In the same way, an end event can be represented in Qualigram notation with either a down-stream action or with an end event. In addition, the findings presented in Table 3.29 confirm the necessity of modeling objectives at the tactical level of abstraction. This table also shows that Qualigram's macro-instruction and collaborative instruction are useful but both of them can be replaced by alternative modeling constructs. However, Table 3.29 shows that Qualigram's source role and target role are sometimes necessary when modeling a triggering event or an end event. Finally, Qualigram's control indicator and corrective indicator are not necessary at the tactical level of abstraction. Therefore, the reviewed version of BPM^+ keeps the list of modeling concepts and their representation in Qualigram notation and BPMN as presented in Table 3.2, but adding Qualigram's target role and source role as possible representations of a BPM^+ role. These small adjustments are emphasized in red in Table 3.31.

Table 3.31 Reviewed version of tactical-level BPM^+ modeling concepts

BPM^+ modeling concepts	**Qualigram**	**BPMN**
Sub-procedure.	Sub-procedure.	Collapsed sub-process.
Activity.	Work instruction.	Task.
External stakeholder.	External role.	Pool.
Role.	Role, **target role, source role**.	Lane.
Department.	Unit.	Lane.
Procedure owner.	Responsibility.	Not available; use text annotation attached with an association.
Relationship between actions.	Information.	Sequence flow.
Relationship between actions and entities.	Swim-lane.	Lane, message flow.
Physical tool.	Physical tool.	Not available; use text annotation attached with an association.
Document.	Document.	Data object.
Objective.	Constraint indicator, performance indicator.	Not available; use text annotation attached with an association.
Triggering event.	Start event, up-stream action.	Start event.
End event.	End event, down-stream action.	End event.
Control flow pattern.	And operator, Or operator.	Gateways.

Syntax considerations

If BPMN is used to represent the business processes, then the first tactical-level model that the modeler should generate for each business process is the same decriptive-level model generated at the strategic level of abstraction but adding the use of lanes to represent all the roles involved in the execution of the business process. Then, the modeler can elaborate as many refinements as necessary to achieve the level of detail expected by the target users. These models might be based on the “descriptive level of use” (i.e. Level 1) or the “analytical level of use” (i.e. Level 2) of BPMN (refer to subsection 1.5.2.2). Both the “private business process” and the “abstract process” types of BPMN models (refer to subsection 1.5.2.2) might be used.

3.5.3 The BPM$^+$ operational level of abstraction

The reviewed version of BPM$^+$ keeps the original description and scope of the operational level of abstraction of its *a priori* version (refer to subsection 3.1.3).

Notes for users

Qualigram models are recommended if the target users are management-oriented professionals, project managers, administrative staff, customers, providers or new employees of the organization. BPMN models are recommended if the target users are IT-consultants, IT-oriented employees or users that are part of a software development team.

Modeling concepts and semantics considerations

The sets of actions, entities, and information (i.e. relationships or dependencies) required for the operational level of abstraction are the same as those described in the *a priori* version of BPM$^+$. The representational analyses revealed the need to represent business rules in a business process model. Therefore, the reviewed version of BPM$^+$ keeps the list of modeling concepts and their representation in Qualigram notation and BPMN as presented in Table 3.3 but adding business rules as one of the modeling concepts (see Table 3.32).

Table 3.32 Reviewed version of operational-level BPM+ modeling concepts

BPM$^+$ modeling concepts	Qualigram	BPMN
Task.	Operation.	Task.
Role.	Role.	Lane.
Relationship between actions.	Information.	Sequence flow.
Physical tool.	Physical tool.	Not available; use text annotation attached with an association.
Document.	Document.	Data object.
Triggering event.	Start event, up-stream action.	Start event.
End event.	End event, down-stream action.	End event.
Control flow pattern.	And operator, Or operator.	Gateways.
Business rules	**Control operations, corrective operations.**	**Not available; use text annotation attached with an association.**

3.6 Case study with a Canadian forensic engineering company

This section presents the results of a case study where the usefulness and acceptance of the reviewed version of BPM$^+$, as well as the scope and content of each of its levels of abstraction were tested. The case study was conducted at a Canadian forensic engineering company that offers its services throughout the Québec-Windsor corridor. The participant company has 56 employees distributed between its four branch offices (i.e. Laval, Québec, Ottawa and Toronto). The case study aimed at evaluating not only the BPM$^+$ approach, but also the BPM notations selected for this thesis, and the modeling preferences of the various types of stakeholders involved in a BPM project.

The participant company was selected for this case study due to: 1) it was in the process to start a BPM initiative; 2) its accessibility; and 3) its interest in the project. The company was willing to start the BPM initiative in order to:

1. improve the quality of the services offered;
2. better coordinate the business processes throughout the organization; and
3. increase its efficacy and ultimately its profitability.

3.6.1 Details of the research design

The principles and generalities of the research design of this case study have been already reported in section 2.2. The planned duration of the case study was of 6 months, and it required the participation of a research team of two members (i.e. the author of this thesis and one ÉTS master degree student). Both members of the research team had and adequate level of knowledge of the BPM notations used in the case study.

Fourteen (14) staff members of the participant company voluntarily collaborated with the execution of this case study, including some of its top executives. The staff members came from various business units and various levels of the organization (e.g. IT, management, human resources, etc.) For a more efficient execution of the BPM initiative the participant company decided to create an internal BPM committee. This committee was multidisciplinary and it was responsible of:

- the global coordination of the BPM initiative;
- planning the activities established in the project calendar;
- ensuring the resources for the execution of the various activities;
- selecting the business process to be modeled and the participants of the company;
- evaluating the tools to be used in the BPM initiative;
- ensuring that the results obtained by the BPM project correspond to the objectives envisioned by the company;
- taking the necessary measures to ensure the integrity of the BPM initiative; and
- acting as a coach within the company.

Table 3.33 presents the profile of each of the 14 company participants, including the members of the BPM committee. For each participant the table describes: the job function within the company, the professional background (i.e. professional training), the number of

Table 3.33 Profiles of the company participants

No.	Job function	Professional background	Experience (years)	BPM experience (years)	BPM notations	Member of the BPM committee
1	Top executive	Bachelor in electrical engineering	36	None	None	Yes
2	Middle management	Bachelor in chemical engineering MBA	20	10	Flow charts	Yes
3	Middle management	Bachelor in IT engineering Msc IT engineering	18	5	UML BPMN Flow charts Qualigram	Yes
4	Forensics engineer	Bachelor in electrical engineering	16	8	Flow charts	Yes
5	Branch Director	Bachelor in electrical engineering	39	15	Flow charts	No
6	Human resources	Bachelor in industrial engineering Msc Professional counseling	16	0.5	Flow charts	No
7	Branch Director	Bachelor in electrical engineering	22	None	None	No
8	Branch Director	Bachelor in electrical engineering	23	None	None	No
9	Branch Director	Bachelor in electrical engineering	25	None	None	No
10	Forensics group leader	Bachelor in electrical engineering	17	None	None	No
11	Forensics group leader	Bachelor in materials engineering Msc Materials engineering Ph.D. materials engineering	12	None	None	No
12	Forensics group leader	Bachelor in civil engineering and applied mechanics	20	None	None	No
13	Middle management	Administrative assistant	34	None	None	No
14	Accounting	Bachelor in marketing research and finance	28	None	None	No

years of experience modeling business processes, the BPM notations that has been used before, and whether or not the participant is a member of the company's BPM committee.

Notice that some of the participants have had previous experience modeling business processes; all of them using simple flow-charts for this purpose. In addition, one of the members of the BPM committee has had previous experience modeling business processes with UML, BPMN and Qualigram. All the participants received training in the basics of BPMN and Qualigram modeling notations.

Considering the professional backgrounds, job positions, and BPM experience of the participants it is possible to conclude that Table 3.33 presents an adequate blend of case study participants according to the purpose of this thesis.

The case study design includes the possibility to conduct four types of meeting activities with the participants: 1) discussion meeting activities; 2) interview meeting activities; 3) evaluation meeting activities; and 4) research-team meeting activities. These meeting activities might be conducted during any of the five phases (i.e. diagnosis, action-planning, action-taking, evaluation and learning) of the cyclical process model (CPM) of the action-research approach (refer to Figure 2.1). Table 3.34 shows which types of meeting activities were conducted at each phase of this case study; who the participants were; and whether the meeting was conducted as a group or individual activity.

Table 3.34 Meeting activities conducted during the case study

Type of meeting activity	CPM phases	Participants	Group / Individual
Discussion meeting	All 5 phases	BPM committee	Group
Interview meeting	Diagnosis, action-taking, evaluation	All the participants	Individual
Evaluation meeting	Evaluation	All the participants	Group / individual
Research-team meeting	All 5 phases	Research-team	Group

None of the business processes of the participant company were previously documented. Therefore, the participant company began the BPM initiative by creating an inventory of its business processes and categorizing them according to the needs of the organization. After this preparatory process, the participant company defined the following criteria to select the business processes to be modeled as part of this research work:

1. the information of the business process is considered relatively easy to gather;
2. the business process directly adds value to the organization; and
3. the business process is rich in details, easing its representation at different levels of abstraction.

In the same way, the participant company defined the following two criteria to exclude a business process from the group of business processes to be modeled as part of this research work:

1. the business process is not fully implemented in the company; and
2. the business process is categorized as a management or as an operational business process.

Based on the categorization of the business processes and the criteria to include/exclude them, the participant company proposed to model the Budgeting business process. This proposal was agreed by the research team. According to the participant company, the Budgeting business process has high-priority and impacts the whole organization.

The BPM committee of the company validated and agreed to the methodology (i.e. action-research approach) to be followed during this research project. In the same way, the committee:

1. validated and approved the resulting business process models;
2. was informed of the findings of each iteration of the research; and
3. was allowed to review and comment the data collected and the results.

3.6.2 Results

3.6.2.1 Results related to the number and scope of the levels of abstraction

The hierarchical levels of abstraction of BPM$^+$ were well accepted by the members of the participant company. They considered that by implementing the various levels of abstraction the various members of the company might be adequately and consistently reached. Specifically, for the group of participants of this research project, they considered that each of the levels of abstraction would be particularly aimed at specific participants as shown in Figure 3.16. The participants considered that modeling in this way it is possible to consistently provide the adequate level of information to each of the members of the company.

Figure 3.16 Participants and levels of abstraction

The participants considered useful the strategic level of abstraction because:

1. it provides a common and global vision of the core business processes;
2. it allows a better understanding of the business processes;

3. it ensures uniformity in the management of the business processes within the organization; and
4. it allows to identify the main sub-processes and procedures that require to be further detailed.

The participants considered useful the tactical level of abstraction because it describes the knowledge of the organization. Specifically, the tactical level of abstraction:
1. allows to describe each procedure in detail including its stages;
2. ensures uniformity in the activities to be executed;
3. allows to better train the new employees of the company;
4. allows a quick identification of the organization problems and their solution;
5. allows to identify all the roles associated to each procedure; and
6. allows each member of the company to understand his or her role in the execution of a procedure.

The participants considered useful the operational level of abstraction but not to be applied to every business process in the organization. Most of the participants considered that only complex business processes, and those that present a high level of operative tasks, should be modeled at this level of abstraction. According to the participants, the main reasons to model a business process at the operational level of abstraction are:
1. to accurately describe the tasks to be performed by a specific role;
2. to describe in detail a specific activity; and
3. to ensure uniformity in the execution of specific tasks.

Some participants considered not necessary to model a business process at the operational level of abstraction because they perceived that the benefits of elaborating and maintaining such a model would not compensate the amount of effort required.

3.6.2.2 Results related to the modeling requirements for each level of abstraction

Since the participant company was willing to experiment with modeling a business process at all the three levels of abstraction (i.e. strategic, tactical and operational), and due to the fact that BPMN does not allow to model at all these levels of abstraction as defined in the reviewed version of BPM$^+$, it was decided to start the modeling process using the Qualigram notation.

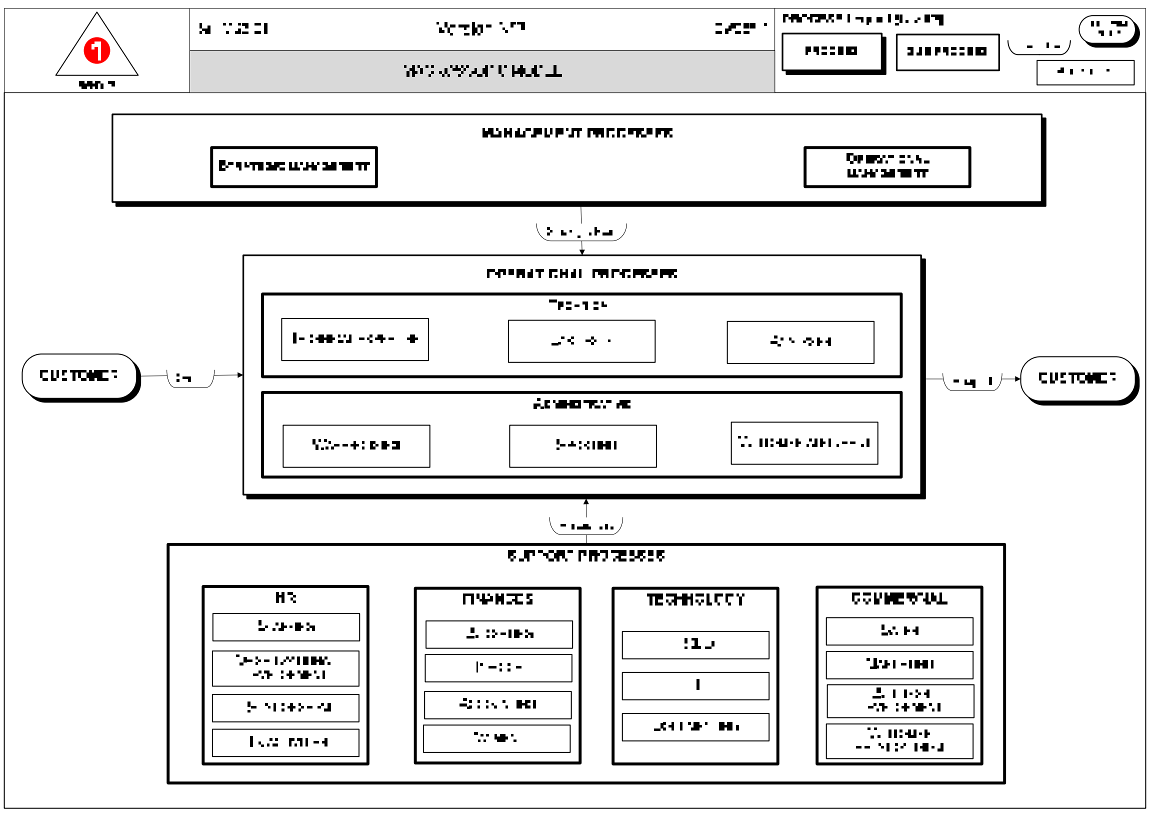

Figure 3.17 Macroscopic model: Forensics Company

At the strategic level of abstraction, the macroscopic type of model (see Figure 3.17) was useful for depicting an inventory and categorization (i.e. management, operational and support business processes) of the business processes of the organization. Doing this turned to be very useful during the process of selecting the first business process to be modeled. However, from the business process management point of view, the participants considered that the macroscopic type of model was not completely useful because it is a type of model that does not include relevant dependencies between the business processes to ease their understandability.

This case study shows that the usefulness found in a relational type of model depends on the type of stakeholder. The higher the position of a participant in the organizational pyramid of the company is (see Figure 3.16), the more useful the participant will find the relational type of model. The participant company decided not to use this type of model because it was not considered useful enough for all the stakeholders.

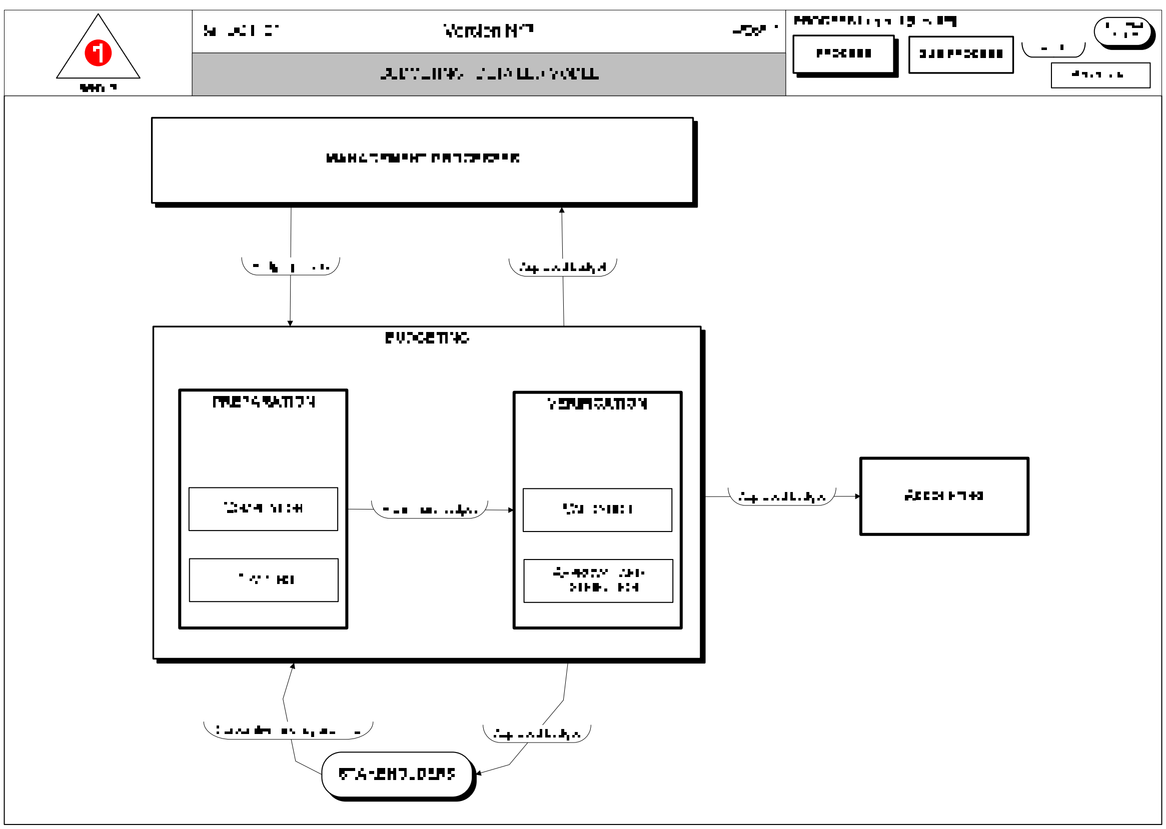

Figure 3.18 Strategic level of abstraction: detailed model of the budgeting business process

Regarding the detailed type of model (see Figure 3.18), most of the participants found it useful because:

1. it identifies all the sub-processes and procedures of each of the business processes; and
2. it depicts the dependencies and the information flow between the sub-processes and the procedures.

In addition, the participants considered that it would be useful to be able to represent at the strategic level of abstraction some key internal roles of the company (BPM^+ does not include internal roles at the strategic level of abstraction). This suggestion was found particularly useful for the detailed type of model.

Regarding the tactical level of abstraction, all the participants found all their modeling needs satisfied with the modeling constructs included in BPM^+. However, they also found useful two modeling constructs offered by Qualigram at the tactical level of abstraction: the macro-instruction (e.g. “Implement corrective actions” in Figure 3.19) and the collaborative instruction (e.g. “Prepare the regional and corporative budget” in Figure 3.19).

Finally, some of the participants considered that the modeling constructs offered by Qualigram at the operational level of abstraction are good enough if:

1. the organization is aiming at complying with the ISO 9000 family of standards (i.e. quality management);
2. the activity is considered by the organization as problematic; and
3. the business process and the organization are mainly concentrated on the execution of operational tasks.

Some participants mentioned that if the organization does not aim at an ISO 9001 certification then it would be better to model the operational level of abstraction with other BPM notation that presents a different approach. However, these participants were not able to specify what other types of details they perceive as necessary to model at this level of abstraction.

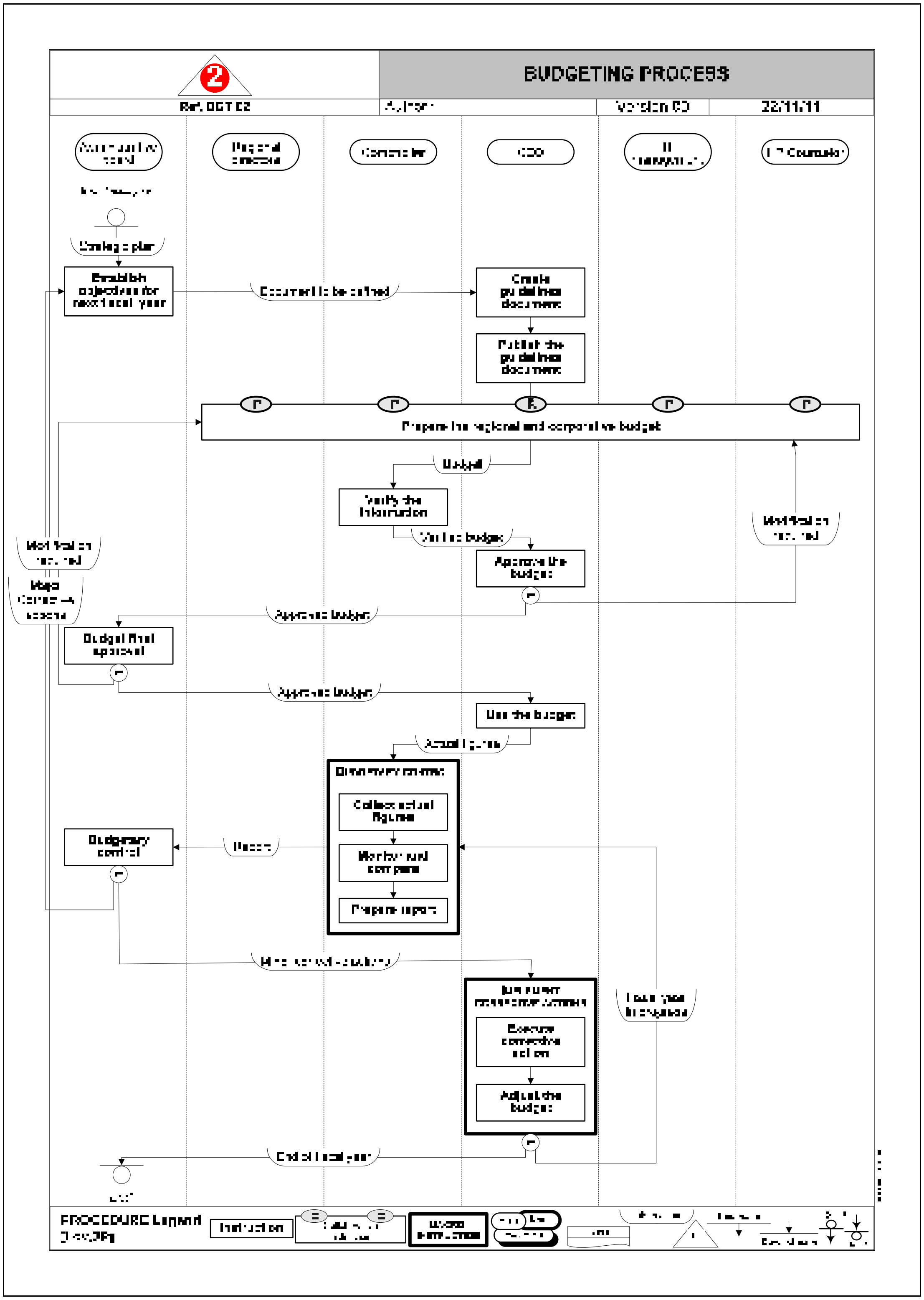

Figure 3.19 Tactical model of the budgeting business process

3.6.2.3 Results related to the BPM notations

One of the already discussed characteristics of BPMN[28] is its lack of support for modeling business processes at the strategic level of abstraction, as in BPM^{+}. This characteristic hinders a comparison of BPMN with the Qualigram notation. For this case study the participant company decided to model at the three levels of abstraction; therefore Qualigram was their first choice of modeling notation. However, the participant company was opened to evaluate the results obtained with BPMN.

Once the Qualigram models were generated, validated and approved by the participant company, the research team converted the tactical level model (Figure 3.19) to its BPMN equivalent (see Figure 3.20). This new model constitutes the BPMN tactical level model of the budgeting process. This BPMN model was then presented to the participants in order to obtain their perceptions.

Since the BPMN version of the budgeting process is relatively simple, the participants did not have mayor difficulties understanding it. Therefore, they found no major differences between using BPMN or using Qualigram notation to model the tactical level of the budgeting business process. However, when the participants were exposed to all the modeling constructs that BPMN offers for further detailing a business process, they expressed that several of those modeling constructs were too complex and that the company did not require reaching such a level of detail. In addition, considering the financial point of view of the organization, the participants indicated that the company should adopt only one modeling notation, and if that is the case the preference was clearly oriented towards Qualigram notation because of its simplicity and its support of the three levels of abstraction.

---

[28] If BPMN version 2.0 were used then it would be possible to model at the strategic level of abstraction. However, it would be possible to generate only a relational type of model.

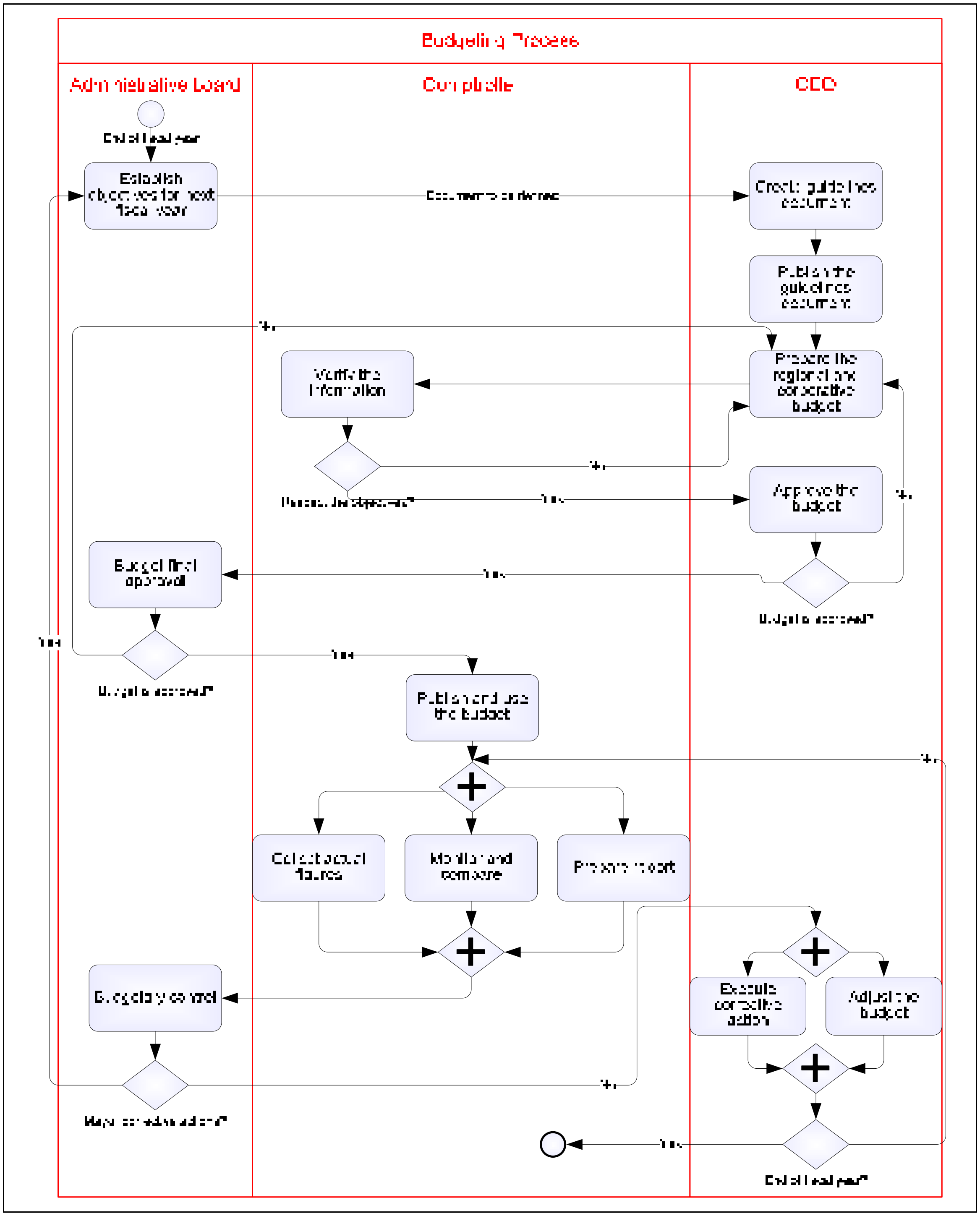

Figure 3.20 BPMN tactical model of the budgeting process

3.6.3 Interpretation and summary of the results

This section has reported the results of a case study conducted to test the reviewed version of the BPM$^+$ approach. The BPM$^+$ proposal of modeling business processes at three levels of abstraction (i.e. strategic, tactical and operational) was well accepted by the participants of the company, who considered that this way of modeling business processes facilitates the participation of the various members of the organization in a BPM initiative, and allows to achieve a common vision of the business processes modeled.

This case study once again shows the value of first generating a macroscopic type of model. This type of model does not only represent a common global vision of the core business processes of the organization, but helps to categorize and inventory them. Regarding the other types of models at the strategic level of abstraction (i.e. relational and detailed types of model), the results of this case study allow a triangulation with those from the pilot case study where the detailed type of model was found more useful than the relational type of model. Despite this result, the final version of BPM$^+$ still will allow the modeler to choose between the relational type of model (supported by our survey results) and the detailed type of model. In addition, this case study has provided some additional specific reasons for modeling at the strategic level of abstraction, and it has confirmed that the set of modeling concepts included at the strategic level of BPM$^+$ would satisfy the modeling needs of a heterogeneous group of stakeholders. The only additional modeling need of the participants at the strategic level of abstraction has been the need to represent internal roles at the detailed type of model. This additional need requires further empirical work that should be conducted after this thesis.

This case study confirms the value of modeling at the tactical level of abstraction, providing some additional specific reasons for modeling at this level of abstraction. In addition, the set of modeling concepts included at the tactical level of BPM$^+$ showed to satisfy the modeling needs of a heterogeneous group of stakeholders. However, the results of this case study permit a triangulation with those from the pilot case study where two additional Qualigram’s modeling constructs were used: the macro-instruction and the collaborative-instruction. Therefore, the final version of BPM^+ will include these two modeling constructs as optional modeling concepts at the tactical level of abstraction.

Regarding the operational level of abstraction, its usefulness has been confirmed and some specific reasons for adopting it have been identified. In addition, this case study confirms the optional status given by BPM^+ to the operational level of abstraction. Therefore, not all the business processes must be modeled at this level of abstraction. At each BPM initiative the participants will have to decide when and where the operational level of abstraction is applicable. According to this case study, BPM^+ should add complex business processes with a high level of operative tasks as possible candidates to be modeled at the operational level of abstraction. The set of modeling concepts proposed by BPM^+ for the operational level of abstraction has been encountered as valid by the participants of this case study; however, the participants have also mentioned that the operational level of abstraction proposed by Qualigram is not always the best choice to follow. Business processes should be modeled (at the operational level of abstraction) following Qualigram's approach if the company is interested on being granted or in maintaining an ISO 9001 (i.e. quality management) certification. If that is not the case, the approach proposed by Qualigram might not be the most recommended to follow. However, the participants of this case study were not able to provide additional information to: 1) develop other scenarios where Qualigram's approach should be used; or 2) identify other scenarios where other BPM notation (e.g. BPMN) would be a better option at the operational level of abstraction. Therefore, further work after this thesis is needed to tackle these issues.

BPMN and Qualigram notations were compared in terms of the participants' preferences for modeling at the tactical level of abstraction. In general terms, the participants did not find major differences between them. However, this result was influenced by the level of detail that the participant company wanted to represent at the tactical level of abstraction. If the level of detail would be increased then the participants would have considered the BPMN models as more difficult to be understood than their Qualigram equivalents. In addition, the participants clearly mentioned that if the company has to choose only one BPM notation, then it should be Qualigram because of its flexibility to model at the three levels of abstraction. This undermines the scenario proposed by BPM^+ to use Qualigram notation at the strategic level of abstraction and then to allow the modeler to choose between Qualigram and BPMN at the tactical and operational levels of abstraction.

Finally, this case study used BPMN version 1.2 for the reasons already explained in this thesis. For the scope and purpose of this thesis, the main difference introduced by BPMN version 2.0 is the possibility to model a collaboration diagram where all the pools (i.e. business processes) are empty (i.e. black boxes). This type of diagram depicts the collaborations between all the processes represented as black boxes. The result is a BPMN model comparable to a Qualigram's relational type of model. However, with BPMN 2.0 it is still not possible to generate models comparable to Qualigram's macroscopic and detailed types of model. Therefore, the use of BPMN version 2.0 would not significantly change the results of this case study. However, the final version of BPM^+ will reflect the fact that it is possible to generate a relational type of model with BPMN.

All these findings have been considered in the final version of BPM^+ which can be found in Appendix XI.

CHAPTER 4

MEASURING FUNCTIONAL SIZE FROM BUSINESS PROCESS MODELS WITH COSMIC FSM METHOD

4.1 Introduction

The ISO 14143-1 standard (i.e. Functional Size Measurement) (ISO/IEC, 2007) specifies that a functional size measurement (FSM) method must provide measurement procedures to quantify the functional user requirements (FUR) of software. Such quantitative information, functional size, is typically used, for instance, in software estimation. One of the international standards for FSM is the COSMIC method — ISO/IEC 19761 — which was designed to be applied both to the business application software domain and to the real-time software domain (ISO/IEC, 2011). A recurrent problem in FSM is the availability and quality of the inputs required for measurement purposes; that is, well documented FUR. Business process models, as they are commonly used to gather requirements from a project, could be a valuable source of information for FSM. In this chapter, the feasibility of such an approach for the business application software domain is analyzed using both: the Qualigram notation (subsection 4.2.1); and the BPMN notation (subsection 4.2.2).

In addition, this chapter also:

1. presents notation-independent guidelines for the business application software domain (subsection 4.2.3); and
2. analyzes the possibility of using BPM to perform FSM in the real-time software domain (section 4.3).

To evaluate the FSM procedure proposed, the measurement results obtained from BPM are compared with those of previous FSM case studies. The results are discussed in section 4.4.

4.2 The business application software domain

The purpose of this section is to measure the functional size of the C-Registration System (GELOG-ETS, 2008), based on a set of models generated using both Qualigram notation (subsection 4.2.1) and BPMN notation (subsection 4.2.2), in order to analyze the feasibility of using business process models as the source of information for FSM in the business application software domain.

4.2.1 Results obtained with Qualigram notation

The purpose of this subsection is to analyze the feasibility of using business process models generated with Qualigram notation as the source of information for FSM. The functional size of the C-Registration System is obtained based on a set of models generated using Qualigram notation. For achieving this: 1) a set of BPM guidelines for Qualigram; and 2) a set of mapping rules between the modeling constructs of Qualigram notation and the concepts of the COSMIC FSM method are derived.

Therefore, the **scope** of the measurement presented in this subsection is given by all the FUR of the C-Registration System as described in (GELOG-ETS, 2008). The C-Registration System is a business application software that belongs to the "application **layer**" of the "typical layered software architecture" (COSMIC, 2009). In the next subsection (4.2.1.1), the BPM guidelines for producing Qualigram models suitable to be used for FSM are identified. In addition, in order to be in agreement with the **level of granularity** expected by the COSMIC FSM method, the appropriate level of abstraction of these models is determined. In subsection 4.2.1.2, the mapping rules between COSMIC and Qualigram are defined to measure the functional size of the C-Registration System.

4.2.1.1 Modeling guidelines for Qualigram

At the top level of abstraction (i.e. strategic level), Qualigram notation models the strategy of the organization, asking the questions: "why" and "where to" (i.e. the main goals of the organization and the relevant external actors) (refer to subsection 1.3.3). At this level of abstraction, therefore, Qualigram notation represents those processes that are directly related to the goals and external actors of the organization. The external actors are either the destination of the products or services produced by the organization, or the important partners whose services or products are required to achieve the goals. It is also possible to represent the relations between the various processes, and between the processes and the external actors.

Modeling Guideline QBA1: At the top level of abstraction (i.e. strategic level), represent the software to be measured as a process.

Modeling Guideline QBA2: Following the COSMIC principles, consider any external software component that interacts with the measured software as an external actor.

Qualigram notation allows some of the processes to be more detailed at the top level of abstraction by representing their sub-processes and main procedures. The procedures constitute the elements that are further detailed at Qualigram's intermediate level of abstraction (i.e. tactical level). Any procedure represented at that level of abstraction must present at least one input and one output of information. Qualigram notation prohibits representation of any internal actor of the organization at the top level of abstraction.

Modeling Guideline QBA3: Consider any logical instruction-set that is worth detailing in more depth as a procedure.

Modeling Guideline QBA4: Represent any user of the software, who allows representation of a procedure inputs or outputs, as an external actor.

Figure 4.1 shows the top-level model of the C-Registration System, based on the requirements of the system (GELOG-ETS, 2008) and the annotated modeling guidelines.

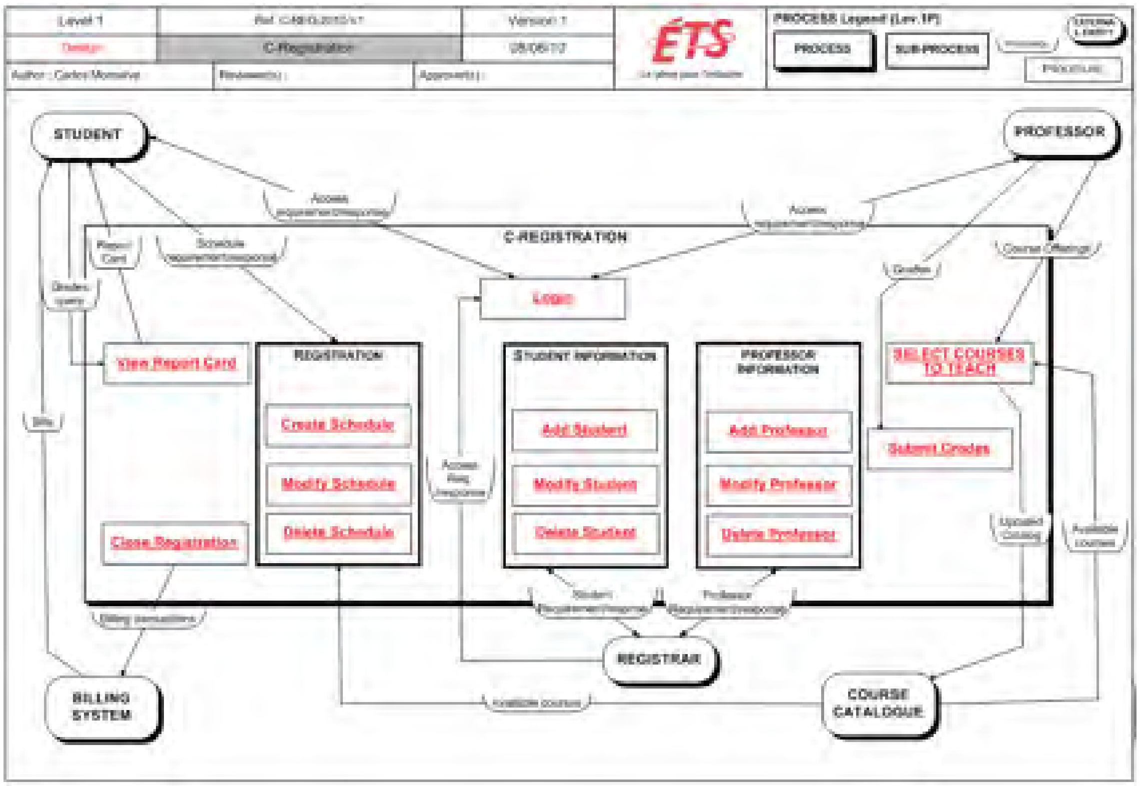

Figure 4.1 Qualigram's top level model of the C-Registration System
Reprinted from Monsalve, Abran and April (2011, p. 320),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

The requirements mention the registrar as an actor. From an organizational point of view, the registrar should be considered as an internal actor who would not appear in the top-level model. However, in order to represent the inputs and outputs of some of the procedures modeled, the registrar is represented as an external actor.

Qualigram notation uses more specialized terms at its intermediate level of abstraction (i.e. tactical level). The terms "internal role", "external role", and "unit" are used to identify specific types of actors. For modeling and analyzing the procedures at this level of abstraction, the modeler needs to answer the questions "who is doing what" and "what is being done" (i.e. the various instructions to be executed as part of the procedure, and the different actors involved in the procedure).

According to Qualigram rules, a procedure requires a minimum of two actors and five instructions. Moreover, the login procedure, or the "create student/professor" procedure, would not typically be considered as a candidate to be modeled using a BPM notation. From the organizational point of view, a typical business process crosses different functional departments of the organization, and that is not the case for this type of procedure. For example, the login procedure contains only an interaction between a user and the information system, and the login requirement does not cross any functional department of the organization. Something similar happens with the create, modify, update, or delete types of procedures.

Modeling Guideline QBA5: At the intermediate level of abstraction (i.e. tactical level), represent the software being measured as an internal role.

Modeling Guideline QBA6: At the intermediate level of abstraction (i.e. tactical level), represent any peer software component that interacts with the software being measured as an internal role.

COSMIC requires identifying those data movements that retrieve or write information from/to a persistent storage. Qualigram allows representation of the tools that are used or produced by an instruction, which could be of the document type or of the material type (refer to subsection 1.3.3). A material tool is used to represent any material resource, such as a piece of software, a machine, a software tool, office material, etc.

Modeling Guideline QBA7: Any instruction that requires retrieving or writing relevant data from/to a persistent storage should be associated with a material tool. That tool has to be labeled, indicating the type of operation to be applied to the persistent storage: **R** for retrieve, **W** for write.

According to Qualigram, “a procedure is never started spontaneously”, and always requires a “triggering element” (Berger and Guillard, 2000, p. 103). This triggering element might be any organizational event (i.e. a customer requirement), or a requirement coming from another procedure. COSMIC defines a triggering event as an event “that causes a functional user of the piece of software to initiate (‘trigger’) one or more functional processes” (COSMIC, 2009, p. 34). After identifying a triggering event, the functional user typically initiates the functional process sending a message to the software. This message constitutes a triggering Entry, which is considered by COSMIC as a data movement of the Entry type. Where a functional process has to wait for additional data from the functional user after having undergone the triggering event, only one Entry has to be considered; this is true even in the case where the functional user requires a prompt message after producing the triggering Entry for entering the additional data. Moreover, “in the business application domain, control commands shall be ignored” (COSMIC, 2009, p. 55).

Modeling Guideline QBA8: If the procedure being modeled requires, at its inception, information to be entered by the role that triggered it, then represent the initial submission of information as the triggering event.

COSMIC establishes that “all messages generated and output by software without user data shall be considered” as “a single Exit…within each functional process” (COSMIC, 2009, p. 70).

Modeling Guideline QBA9: All the error conditions identified by a role must be collected by a unique instruction executed by the same role before reporting them to another role.

COSMIC determines the Exits and Entries to a functional process by identifying those data movements that cross the boundary of the functional process. The boundary is defined as “a conceptual interface between the software being measured and its functional users” (COSMIC, 2009, p. 26). At Qualigram’s intermediate level of abstraction (i.e. tactical level), each role is confined to a swim-lane. If a role A needs to pass the control of the workflow to a role B, then role A needs to send a flow of information to role B crossing the swim-lane of role A.

Modeling Guideline QBA10: Avoid representing flows of information between roles when those flows are only aimed at indicating a possible end to the workflow.

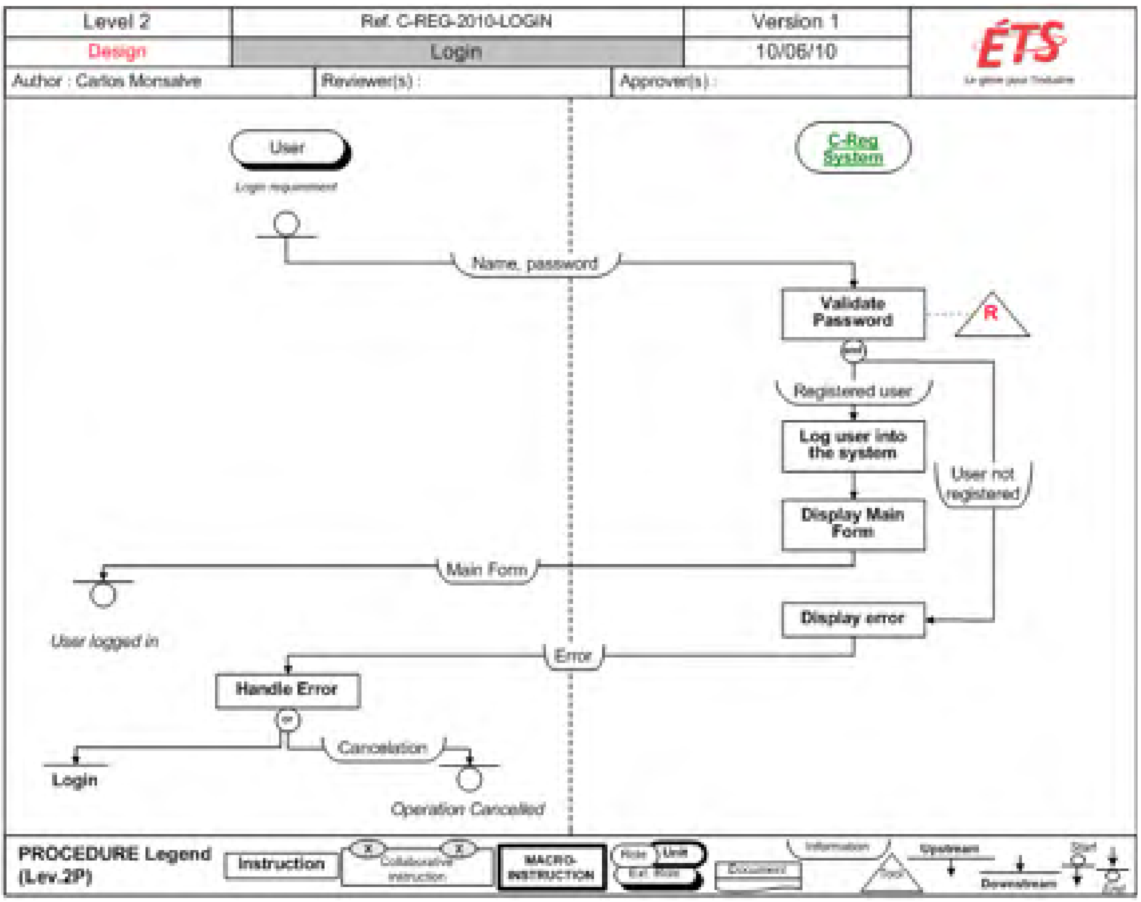

Figure 4.2 Qualigram model of the Login procedure
Reprinted from Monsalve, Abran and April (2010, p. 284), © copyright Sharker Verlag

Based on the requirements of the software application (GELOG-ETS, 2008) and the annotated modeling guidelines, a second-level model for the procedures depicted in Figure 4.1 has been produced. For instance, Figure 4.2 and Figure 4.3 present the models for the Login and Add Professor procedures respectively.

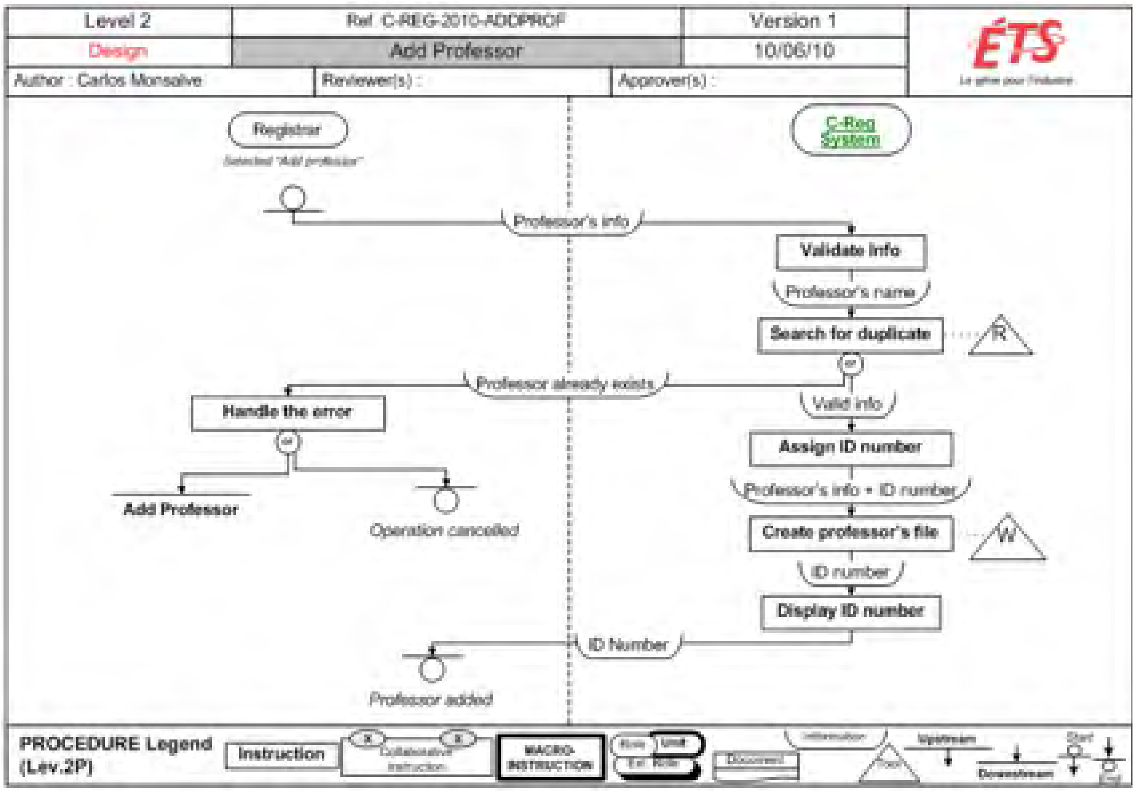

Figure 4.3 Qualigram model of the Add Professor procedure
Reprinted from Monsalve, Abran and April (2011, p. 321),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

According to COSMIC, the recommended level of granularity of the FUR is achieved when the functional users: 1) are individuals; and 2) "detect single occurrences of events" (COSMIC, 2009, p. 28). Looking at Figure 4.2 and Figure 4.3, it is possible to conclude that these conditions seem to be satisfied with the intermediate level of abstraction of Qualigram notation (i.e. tactical level). Therefore, this part of the research will not look into analyzing the bottom-level of abstraction (i.e. operational level).

4.2.1.2 Mapping and measuring based on Qualigram

Before measuring the functional size of the C-Registration System, the mapping rules between the COSMIC concepts (COSMIC, 2009) and the modeling constructs of Qualigram

(Berger and Guillard, 2000) must be defined. From the analysis of the Qualigram models generated in subsection 4.2.1.1, it is possible to derive some of these rules. Table 4.1 shows the mapping rules that have been defined based on that analysis and a comparison of the definitions of the COSMIC concepts and the Qualigram notation constructs.

Table 4.1 Rules for mapping between COSMIC and Qualigram
Reprinted from Monsalve, Abran and April (2011, p. 321),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

COSMIC FSM Method V.3.0.1	Qualigram Notation	Comments
Functional User	Role	Only those roles that interact with the software
Boundary	The process box that represents the software	Top-level model
	The swim-lane of the role that represents the software	Intermediate-level model
Functional Process	Procedure	The procedures included in the process box of the software
Triggering Event	Triggering element	
Data Group	May be provided as part of the information flow	Between roles
	May be provided for describing the material tool	For an instruction that requires access to a persistent storage
Entry	An incoming flow of information	
Exit	An outgoing flow of information	
Read	Description (R) given in a material tool	
Write	Description (W) given in a material tool	

According to COSMIC (2009, p. 39), “a data group is a distinct, non empty, non ordered and non redundant set of data attributes” that describes an “object of interest”, the latter being “anything that is identified from the point of view of the functional user requirements”. A data group may be represented in Qualigram notation by means of the flows of information between roles. For example, observe the “Add Professor” procedure in Figure 4.3: the first flow of information from the registrar to the C-Registration System includes the data group "professor's info". Also, a data group may be represented as part of the information describing a material tool that represents a persistent storage. For example, to the Login procedure in Figure 4.2, it is possible to add the "user's data" data group to the "R" description of the triangle that represents the persistent storage.

Table 4.2 Measurement results based on Qualigram: Business application — The C-Registration system (GELOG-ETS, 2008)
Adapted from Monsalve, Abran and April (2011, p. 314), DOI: 10.1142/S0218194011005359,
© copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

Procedures	**Data Movements**				**CFP**
	E	**X**	**R**	**W**	
Login	1	2	1		4
Add Professor	1	2	1	1	5
Modify Professor	2	2	1	1	6
Delete Professor	3	3	1	1	8
Select Courses to Teach	4	6			10
Add Student	1	1	1	1	4
Modify Student	2	2	1	1	6
Delete Student	3	3	1	1	8
Create Schedule	5	5	2	2	14
Modify Schedule	5	6	3	2	16
Delete Schedule	3	4	2	1	10
Close Registration	3	5	2	1	11
Submit Grades	4	4	2	1	11
View Report Card	1	2	2		5
Total	38	47	20	13	**118**

Figure 4.4 shows an example of how to apply the mapping rules to the "Delete Schedule" procedure. The measurement results are next obtained by simply adding the different data movements (Entries (E), Exits (X), Writes (W), and Reads (R)) that appear in the models representing the various procedures. Table 4.2 shows the measurement results obtained. A discussion of these results and the modeling rules is presented in subsection 4.4.1.

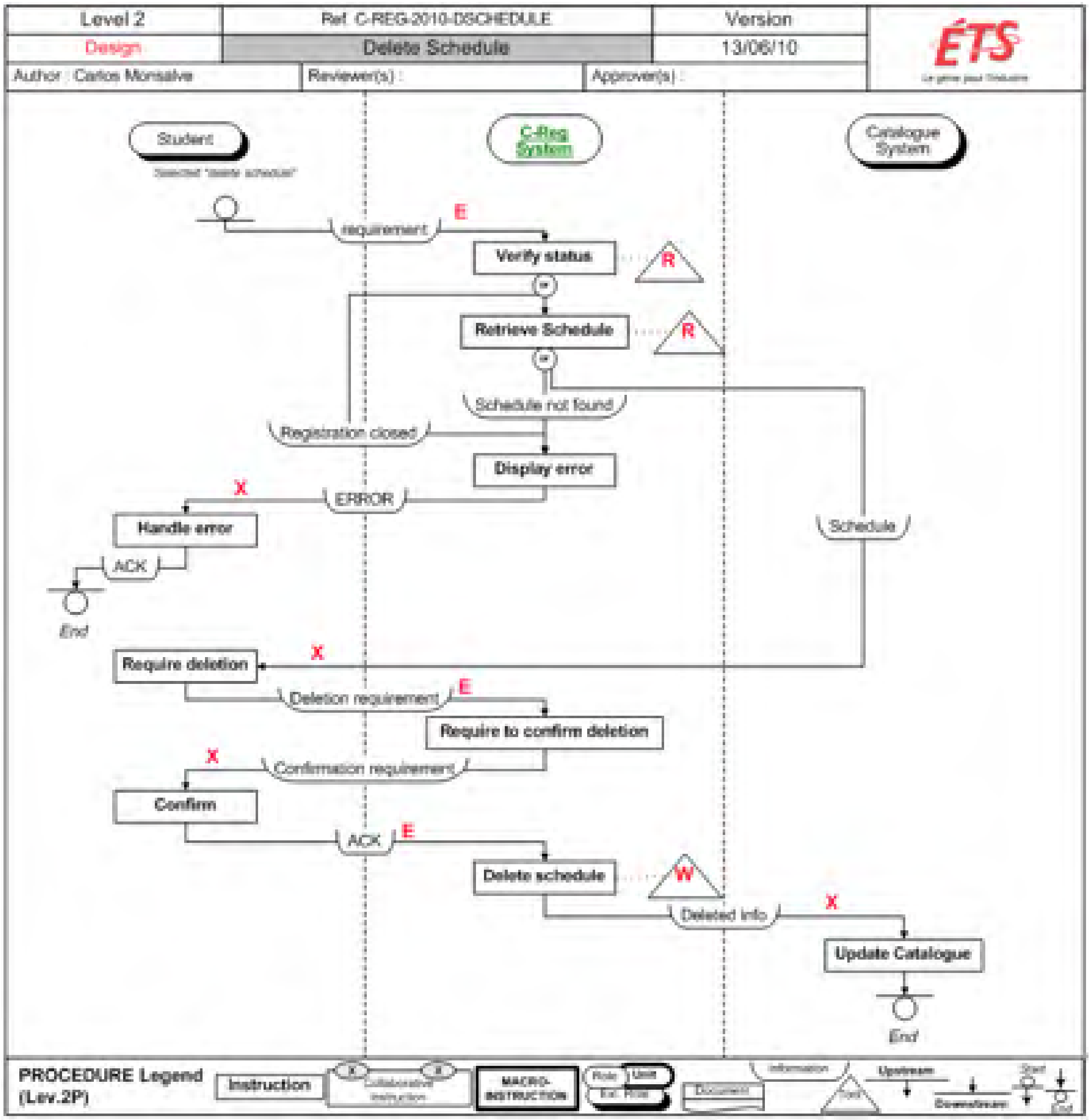

Figure 4.4 Qualigram model of the Delete Schedule procedure
Reprinted from Monsalve, Abran and April (2010, p. 286),
© copyright Sharker Verlag

4.2.2 Results obtained with BPMN notation

This section uses BPMN version 1.2 (OMG, 2009a) for modeling the specifications of the C-Registration system and for deriving the BPMN modeling guidelines for FSM purposes. There is a version 2.0 of BPMN (OMG, 2011), but it was still considered a Beta 2 version at the time of executing this research. When the term BPMN is used in this Chapter without any reference to either of the two versions, it has to be understood that, for understanding the thesis-author's argument, the BPMN version does not affect the meaning of his assertion.

4.2.2.1 Modeling guidelines for BPMN

BPMN does not offer the possibility of representing the C-Registration system by a model with similar characteristics to the one depicted in Figure 4.1. In BPMN, it is always necessary to represent the workflow of the business process; i.e. each business process has at least one clear start event that triggers the first activity (task or sub-process), after which a finite set of activities is executed following a predetermined flow that finishes at a clear end event (OMG, 2009a). A business process may have multiple end events.

In BPMN 1.2, a business process should be contained in a pool, and, even if it is not drawn, it is "implied by default" (Silver, 2009, p. 26). A business process can interact with any external participant (e.g. customer, provider, external actor, other business process) through sending and receiving messages (OMG, 2009a). In these cases, the external participant is considered as an external business process and may be represented as a pool in the BPMN diagram. In order to differentiate between the pool of the business process and the pool of any external participant, this research will refer to them as "main pool" and "secondary pool" respectively. A pool may be partitioned into lanes, which are used to represent any organization or categorization of activities (OMG, 2009a). Typically, lanes are used to represent "performer roles or organizational units" (Silver, 2009, p. 26).

Modeling Guideline BPMN1: Consider any logical instruction set that is worth detailing as a separate business process.

Modeling Guideline BPMN2: Represent the software to be measured as a lane in the main pool.

Modeling Guideline BPMN3: Represent any external software component that interacts with the measured software as a secondary pool.

Modeling Guideline BPMN4: Represent any user of the software as a secondary pool (external user) or as a lane in the main pool (internal user).

Modeling Guideline BPMN5: All the error conditions identified within the lane that represents the software to be measured must be collected by a unique event or a unique activity before reporting them to another lane or pool.

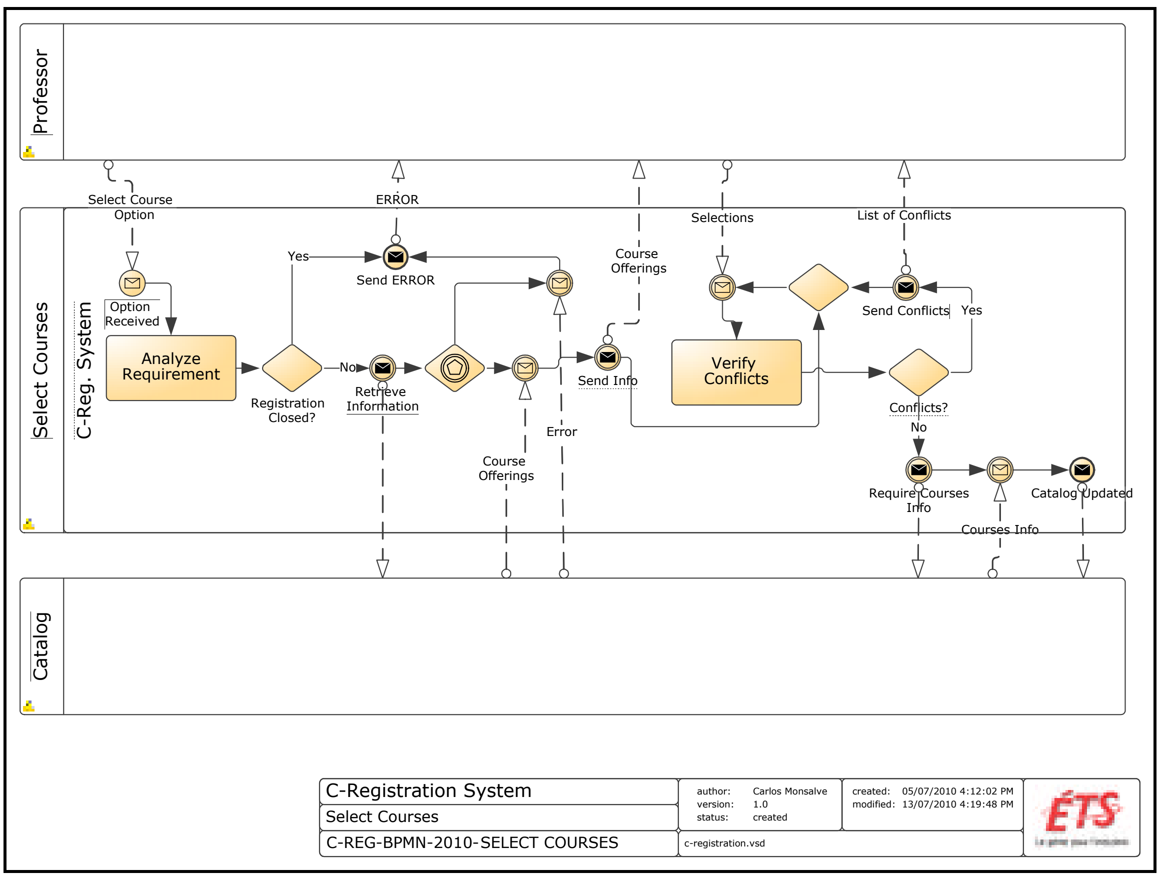

Figure 4.5 BPMN model of the "Select Courses" functional process
Reprinted from Monsalve, Abran and April (2011, p. 323),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

Figure 4.5 presents the application of these modeling guidelines for representing the "Select Courses" functional process based on the requirements of the system (GELOG-ETS, 2008).

Modeling Guideline BPMN6: Avoid representing a sequence flow between lanes or a message between pools when that flow or message is only aimed at indicating a possible end to the workflow.

Modeling Guideline BPMN7: Any modeling construct that requires retrieving or writing relevant data from/to persistent storage should be associated with a data object.

Modeling Guideline BPMN8: Use link events when the lane of the software to be measured must be crossed in order to return to an activity (e.g. for representing a feedback).

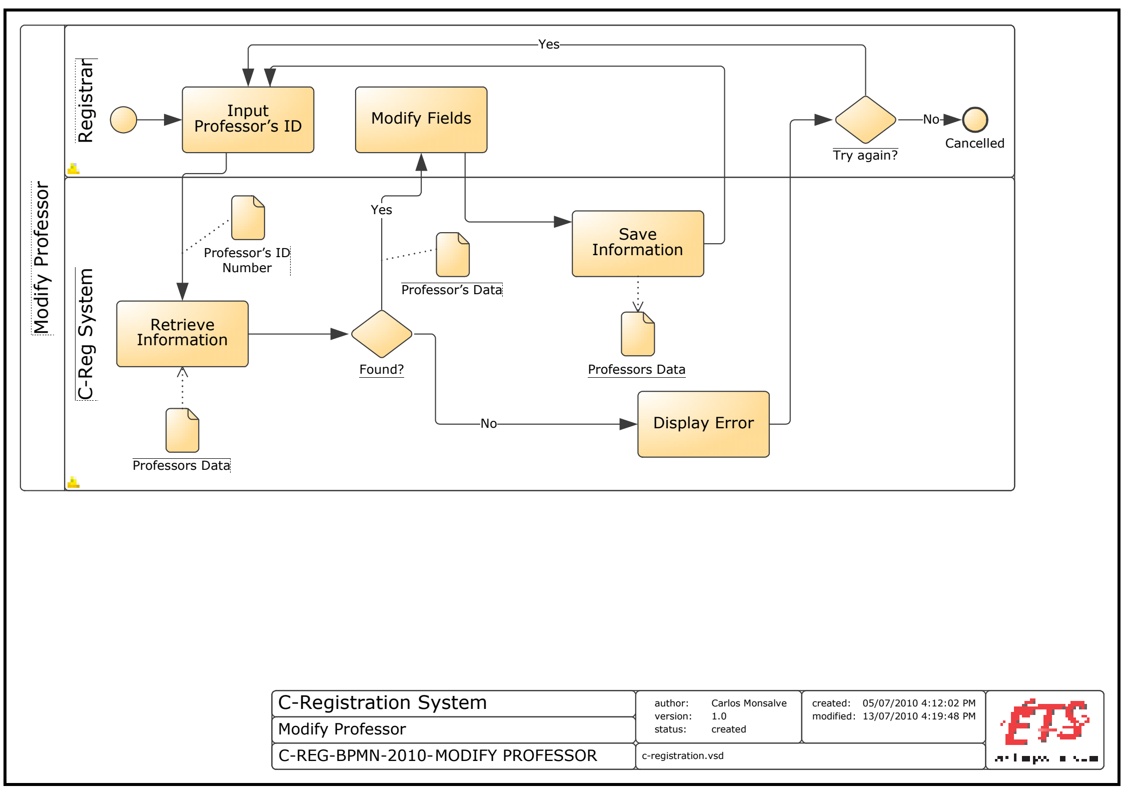

Figure 4.6 BPMN model of the "Modify Professor" functional process
Reprinted from Monsalve, Abran and April (2011, p. 324),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

Figure 4.6 depicts the application of these modeling guidelines for representing the "Modify Professor" functional process based on the requirements of the software application (GELOG-ETS, 2008). Looking at Figure 4.5 and Figure 4.6, it is possible to conclude that the COSMIC conditions (COSMIC, 2009) for the recommended level of granularity seem to be satisfied with the level of detail of the BPMN models.

4.2.2.2 Mapping and measuring based on BPMN

A data group may be represented in BPMN by means of the messages exchanged between pools. For example, observe the representation of the "Select Courses" functional process in Figure 4.5: the last message sent to the catalog pool by the C-Registration System includes the data group "Catalogue". Also, a data group may be represented in BPMN by the

Table 4.3 Mapping between COSMIC and BPMN version 1.2
Reprinted from Monsalve, Abran and April (2011, p. 325),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

COSMIC FSM Method V.3.0.1	BPMN 1.2	Comments
Functional User	Lane and pool	Those who interact with the lane of the software to be measured
Boundary	The lane that represents the software to be measured	
Functional Process	Pool	Those that contain the lane that represents the software to be measured
Triggering Event	Start Event	
Data Group	Name of a message	Between pools
	Data Object	When a persistent storage must be accessed
Entry	An incoming message or sequence flow	
Exit	An outgoing message or sequence flow	
Read	An upstream association with a data object	
Write	A downstream association with a data object	

information describing a data object that represents a persistent storage. For example, the first data object in the representation of the "Modify Professor" functional process (Figure 4.6) shows the "Professor's ID number" data group.

Before measuring the functional size of the C-Registration System, the rules for mapping between the COSMIC concepts (COSMIC, 2009) and the modeling constructs of BPMN 1.2 (OMG, 2009a) must be defined. From the analysis in Figure 4.5 and Figure 4.6, some of these rules can be derived. Table 4.3 shows the mapping rules that have been defined based on that analysis, as well as a comparison of the definitions of the COSMIC concepts and the BPMN constructs.

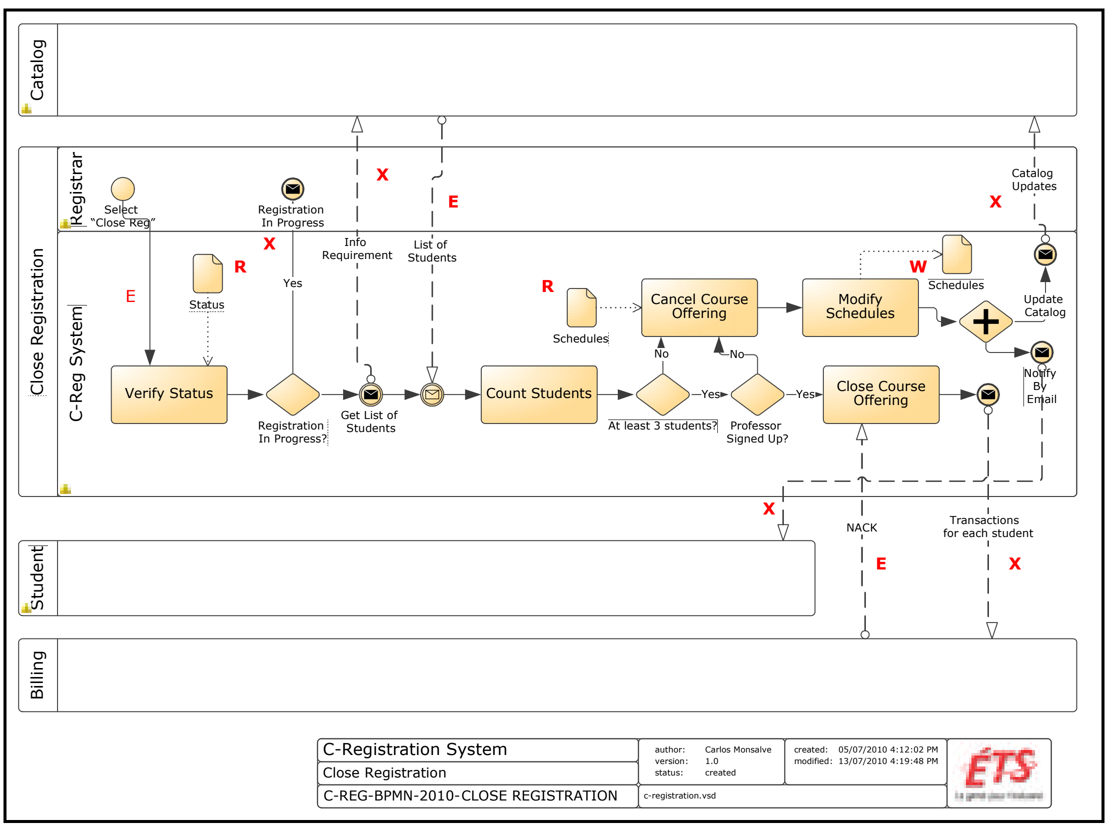

Figure 4.7 Application of the mapping rules to the "Close Registration" functional process
Reprinted from Monsalve, Abran and April (2011, p. 326),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

Figure 4.7 shows an example of how to apply the mapping rules to representing the "Close Registration" functional process. The measurement results are obtained by simply adding the various data movements (Entries (E), Exits (X), Writes (W), and Reads (R)) that appear in the BPMN models representing the various processes. Table 4.4 shows the measurement results obtained. A discussion of these results and the modeling rules is presented in subsection 4.4.1

Table 4.4 Measurement results based on BPMN version 1.2: Business application — The C-Registration system (GELOG-ETS, 2008)

Functional process	Data Movements				CFP
	E	X	R	W	
Login	1	2	1		4
Add Professor	1	2	1	1	5
Modify Professor	2	2	1	1	6
Delete Professor	3	3	1	1	8
Select Courses to Teach	4	6			10
Add Student	1	1	1	1	4
Modify Student	2	2	1	1	6
Delete Student	3	3	1	1	8
Create Schedule	5	5	2	2	14
Modify Schedule	5	6	3	2	16
Delete Schedule	3	4	2	1	10
Close Registration	3	5	2	1	11
Submit Grades	4	4	2	1	11
View Report Card	1	2	2		5
Total	38	47	20	13	**118**

4.2.3 Deriving notation-independent BPM guidelines and mapping rules

Based on the analytical comparison of the results obtained in subsections 4.2.1 and 4.2.2, a set of notation-independent modeling guidelines for FSM is derived first, and then a general set of mapping rules is proposed. Both the modeling guidelines and the mapping rules are intended for use in the business application software domain.

It is critical when performing FSM based on BPM to choose the correct level of abstraction for modeling the FUR of the software to be measured. Doing so will ensure that the specifications will be represented with the right level of granularity. For example, it would be extremely difficult to obtain all the required information for FSM from a high-level buisness process model, like the one depicted in Figure 4.1, with the Qualigram notation. From the business process models presented in subsections 4.2.1 and 4.2.2, it is possible to conclude that a good level of granularity is achieved when modeling at the tactical level of abstraction.

Guideline 1: If the selected BPM notation offers various modeling levels of abstraction, choose one that allows depiction of the business process workflow, including its activities, roles, events, and flow of information.

Modeling guidelines QBA2, QBA6, and BPMN3 are related to the same concepts and can be generalized as follows:
Guideline 2: Consider any peer software component that interacts with the measured software as an external participant (i.e. external role).

Modeling guidelines QBA3 and BPMN1 are very similar, and can be generalized as follows:
Guideline 3: Represent any logical instruction set that is worth detailing as a separate business process workflow.

Modeling guidelines QBA4 and BPMN4 share some concepts, and can be generalized as follows:
Guideline 4: Represent any user of the software, external to the organization, as an external participant (i.e. external role).

In addition, modeling guideline BPMN4 includes some relevant considerations that can be generalized as follows:
Guideline 5: Represent any user of the software, internal to the organization, as an internal participant (i.e. internal role).

Modeling guidelines QBA5 and BPMN2 can be generalized as follows:

Guideline 6: Represent the measured software as an internal participant (i.e. internal role).

Modeling guidelines QBA7 and BPMN7 present some concepts in common and can be generalized as follows:

Guideline 7: Anytime relevant data must be retrieved from or written to persistent storage, represent that type of action as a resource or as a data object used in the business process. Associate the resource or data object with the corresponding modeling construct, and then differentiate a retrieval action from a writing action in an appropriate way.

Modeling guideline QBA8 is relevant and should be generalized:

Guideline 8: If the business process being modeled requires, at its inception, that information be entered by the user triggering it, represent the initial submission of information as the triggering event.

Modeling guidelines QBA9 and BPMN5 are very similar, and can be generalized as follows:

Guideline 9: All the error conditions identified by the internal participant (i.e. internal role) representing the measured software must be collected by a single modeling construct associated with the same internal participant, before reporting those conditions to another participant (i.e. role).

Modeling guidelines QBA10, BPMN6, and BPMN8 share common concepts, and can be generalized as follows:

Guideline 10: Avoid representing flows of information between participants (i.e. roles), whether they are internal or external, when those flows are only aimed at indicating a possible end to the workflow, or a repetition of it.

Comparing Table 4.1 and Table 4.3, it is possible to generalize the mapping rules for the business application software domain, as presented in Table 4.5. The COSMIC data group concept presents two mapping options, as described in Table 4.5. The first option is to map a data group to the information provided as part of a flow. This option is valid for the data groups that are exchanged between the measured software and the functional users. The second option is to map a data group to the name of a resource or data object. This option is valid for the data groups that are retrieved from, or moved to, a persistent storage by the measured software.

Table 4.5 Rules for mapping between COSMIC and BPM notation
Reprinted from Monsalve, Abran and April (2011, p. 329),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

COSMIC FSM method V.3.0.1	BPM notation
Functional User	Construct that represents a role or participant
Boundary	The swim-lane of the measured software
Functional Process	Business process
Triggering Event	Start Event
Data Group	Information provided as part of a flow
	Name of a resource or data object
Entry	An incoming flow
Exit	An outgoing flow
Read	A resource or data object representing the retrieval of data
Write	A resource or data object representing the writing of data

4.3 The real-time software domain

The **purpose** of this section is to measure the functional size of the software components of the Rice Cooker Controller (COSMIC, 2008) based on a set of business process models, in order to analyze the feasibility of using them as the source of information for the FSM of real-time software. Therefore, the **scope** of this measurement is given by all the software requirements of the Rice Cooker Controller case study of the COSMIC Group, which is a real-time system. All its software components are at the same hierarchical level, and at a similar **level of decomposition**. Therefore, in this thesis, we consider that all the software components of the Rice Cooker Controller belong to a single **software layer**. In the next subsection (4.3.1), the specific modeling guidelines for producing Qualigram models of real-time software for FSM purposes are presented. In addition, the appropriate level of abstraction of the models generated is determined, in accordance with the **level of granularity** expected by the COSMIC FSM method. In subsection 4.3.2, the mapping rules between COSMIC and Qualigram notation for the real-time software domain are defined, in order to arrive at a measure of the functional size of the software components of the Rice Cooker Controller.

4.3.1 Modeling guidelines for the real-time software domain

The modeling guidelines presented in subsection 4.2.1.1 for the business application software domain can be adapted as follows:

Modeling Guideline QRT1: Represent the various software components of the real-time system as one process at the top level of abstraction (i.e. the strategic level).

Modeling Guideline QRT2: Consider any hardware interacting with the software as an external entity.

Modeling Guideline QRT3: Consider as a procedure any software requirement that: 1) presents an autonomous functionality (i.e. does not depend on other software components); and 2) can be detailed more deeply.

Figure 4.8 shows the top-level model of the software components of the Rice Cooker Controller.

Modeling Guideline QRT4: Represent the software being measured as an internal role at the intermediate level of abstraction (i.e. the tactical level).

Modeling Guideline QRT5: As any instruction requiring that persistent data be retrieved or written should be associated with a material tool, label every material tool, indicating the type of operation to be applied: R for retrieve, W for write.

Modeling Guideline QRT6: Collect all the error conditions identified by a role by means of a unique instruction executed by the same role before reporting them to another role.

Modeling Guideline QRT7: Avoid representing flows of information between roles when those flows are only aimed at indicating a possible end to the workflow.

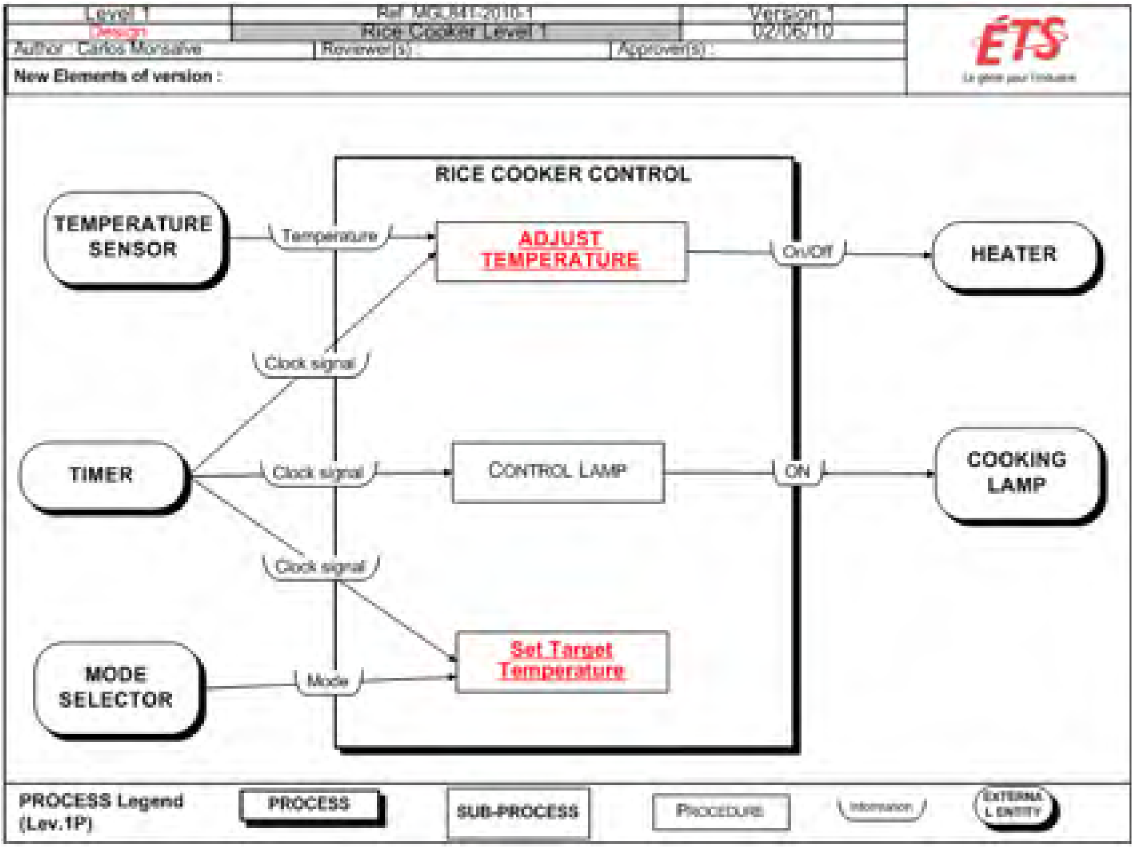

Figure 4.8 Top-level Qualigram model of the Rice Cooker Controller
Adapted from Monsalve, Abran and April (2011, p. 330),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

Based on the specifications of the software application (COSMIC, 2008) and the annotated modeling rules, an intermediate-level model for each procedure depicted in Figure 4.8 has been produced. For instance, Figure 4.9 presents the model for the "Adjust Temperature" procedure. Since every user of the software components modeled is individually represented at the intermediate level as a role, and every procedure responds to a single triggering event, we can conclude that the appropriate **level of granularity** seems to be satisfied with the intermediate level of abstraction (i.e. tactical level) of the Qualigram notation. We will not, therefore, look into the analysis of the bottom level of abstraction (i.e. the operational level) in this subsection of the thesis.

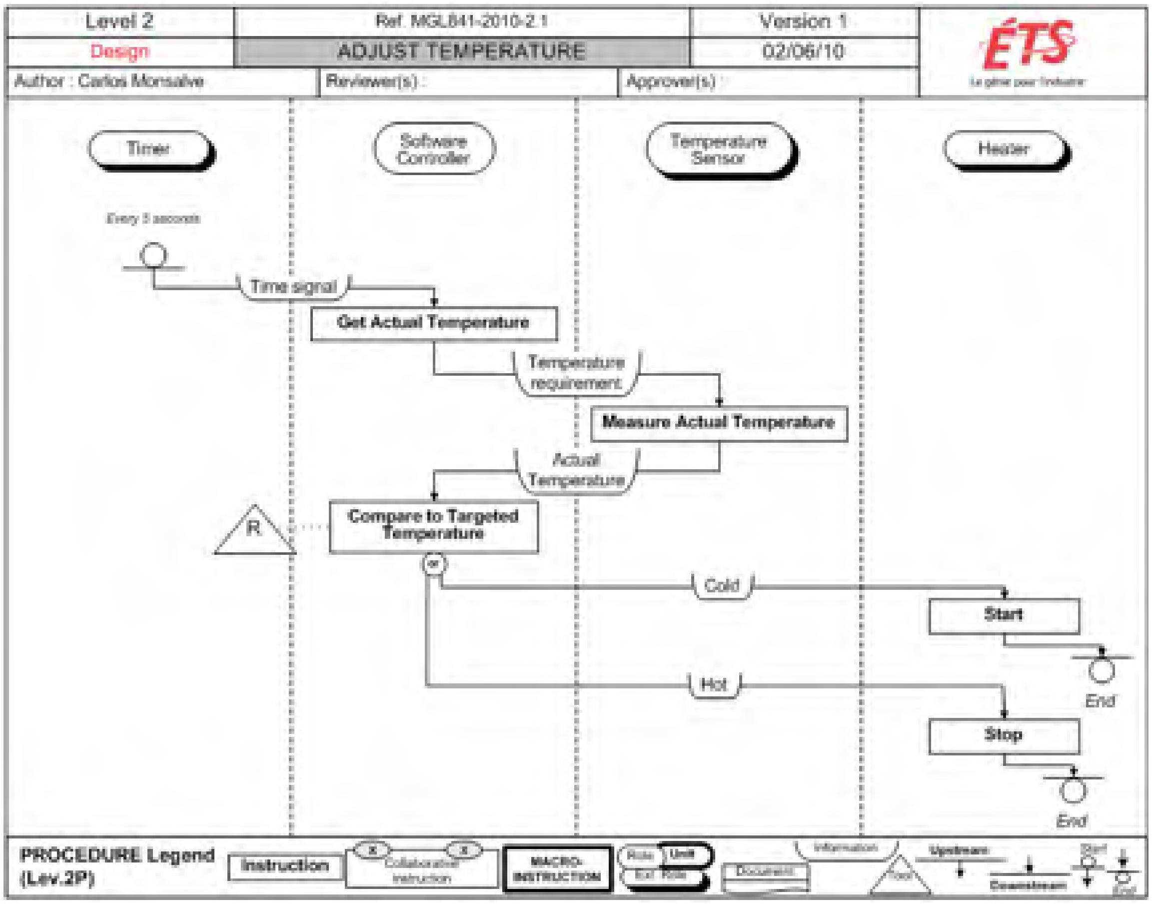

Figure 4.9 Qualigram model of the “Adjust Temperature” procedure
Reprinted from Monsalve, Abran and April (2011, p. 331),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

4.3.2 Mapping and measuring

After analyzing the models obtained for the Rice Cooker Controller and comparing the definitions of the COSMIC concepts with those of the Qualigram modeling constructs, we can conclude that the rules defined in Table 4.1 also apply to the real-time software domain. Figure 4.10 shows an example of how to apply the mapping rules to the “Set Target Temperature” procedure. The measurement results are obtained by simply adding the data movements (Entries (E), Exits (X), Writes (W), and Reads (R)) that appear in the models representing the various procedures.

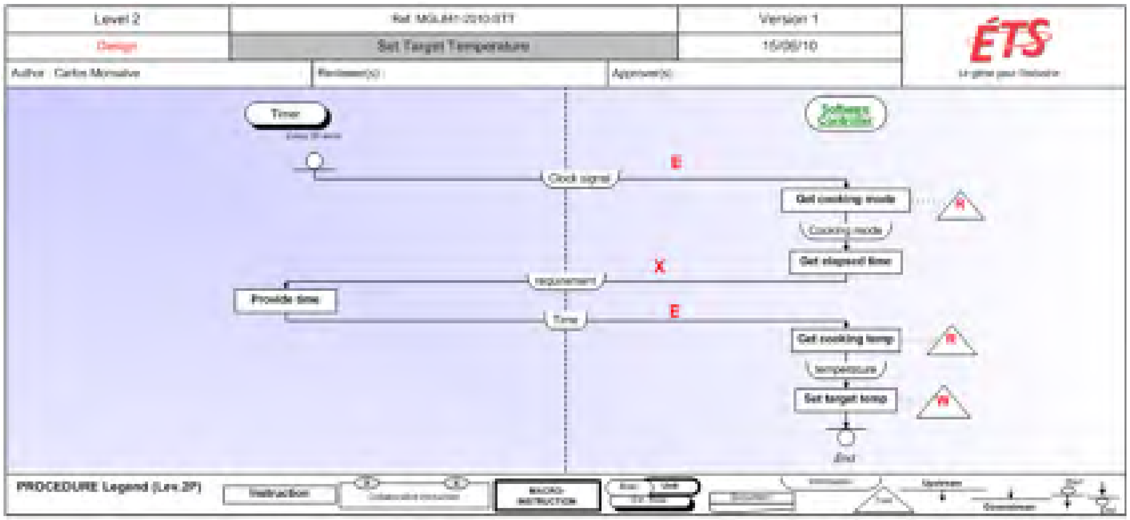

Figure 4.10 Applying the mapping rules to the "Set Target Temperature" procedure
Reprinted from Monsalve, Abran and April (2011, p. 332),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

The results are then compared with those obtained by Lavazza and Bianco (2009) for the same COSMIC case study. Table 4.6 shows this comparison. A discussion of the results is presented in subsection 4.4.2.

Table 4.6 Measurement results: Real-time domain
Adapted from Monsalve, Abran and April (2011, p. 332),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

FP	**Results obtained by Lavazza and Del Bianco (Lavazza and Bianco, 2009)**						**Measurement Results based on Qualigram Notation**					
	Process	Data Movements				CFP	Procedures	Data Movements				CFP
		E	X	R	W			E	X	R	W	
FP1	Tick (control lamp)	1	1			2	Control Lamp	1	1			2
FP2	5 sec. Signal management (control heater)	2	1	1		4	Adjust Temperature	2	3	1	0	6
FP3	30 sec. Signal management (set target temperature)	2		2	1	5	Set Target Temp	2	1	2	1	6
	Total	5	2	3	1	**11**	**Total**	5	5	3	1	**14**

Legend: FP=Functional Process

4.4 Discussion of results

4.4.1 Business application software domain

To evaluate the results of the FSM procedure proposed in this thesis, the results obtained are compared to those obtained in the C-Registration-System COSMIC case study (GELOG-ETS, 2008). The findings from the comparison are discussed next, and improvements to the FSM procedure are proposed if necessary in order to obtain results in conformity with all the rules of the COSMIC ISO 19761 standard.

The COSMIC case study (GELOG-ETS, 2008) presents two sets of results: “step 1” and “step 2”:

1. The first set is obtained after applying the COSMIC FSM method to the FUR “exactly as they are written” in the original specifications of the C-Registration System (p. 5).
2. The second set results from modifying the FUR in step 1 “by a further assumption” (p. 42).

This research has only considered the FUR as given in step 1 of the case study. In addition, Table 4.2 and Table 4.4 show that the measurement results obtained using Qualigram notation are exactly the same as those obtained using BPMN; this result not only supports the generalization of the approach proposed in this thesis, but it also allows to evaluate the results of the FSM procedure by comparing the results obtained using either Qualigram notation or BPMN with those obtained in the COSMIC case study. Table 4.7 shows the measurement results obtained with the FSM procedure proposed in this thesis compared with those obtained in step 1 of the COSMIC case study.

In order to help with the analysis and discussion of the comparison findings, besides tabulating the data movements, Table 4.7 also shows the absolute difference (Δ_i) found between the quantity of data-movements of type *i* (where *i* belongs to {E, X, R, W}) obtained in the COSMIC case study step 1 and its corresponding data-movement quantity obtained

Table 4.7 Comparison of the measurement results: C-Registration System

Functional Processes		Measurement results obtained in the COSMIC case study Step 1					Measurement results obtained with the FSM procedure proposed in this thesis									Δ_{Total}
		Data Movements				CFP	Data Movements								CFP	
		E	X	R	W		E	Δ_E	X	Δ_X	R	Δ_R	W	Δ_W		
FP1	Logon	1	1	1		3	1	0	2	1	1	0		0	4	1
FP2	Add a professor	1	2	1	1	5	1	0	2	0	1	0	1	0	5	0
FP3	Modify a professor	2	2	1	1	6	2	0	2	0	1	0	1	0	6	0
FP4	Delete a Professor	2	2	1	1	6	3	1	3	1	1	0	1	0	8	2
FP5	Select Courses to Teach	4	5			9	4	0	6	1		0		0	10	1
FP6	Add a student	1	1	1	1	4	1	0	1	0	1	0	1	0	4	0
FP7	Modify a student	2	2	1	1	6	2	0	2	0	1	0	1	0	6	0
FP8	Delete a Student	2	2	1	1	6	3	1	3	1	1	0	1	0	8	2
FP9	Create a schedule	5	5	1	2	13	5	0	5	0	2	1	2	0	14	1
FP10	Modify a schedule	5	6	2	2	15	5	0	6	0	3	1	2	0	16	1
FP11	Delete a schedule	2	3	1	1	7	3	1	4	1	2	1	1	0	10	3
FP12	Close registration	2	5	1	1	9	3	1	5	0	2	1	1	0	11	2
FP13	Submit Grades	4	5	2	1	12	4	0	4	1	2	0	1	0	11	1
FP14	View Report Card	1	3	2		6	1	0	2	1	2	0		0	5	1
	Total	34	44	16	13	**107**	38	4	47	7	20	4	13	0	**118**	**15**

with the FSM procedure proposed in this thesis. For instance, the values of the Δ_E column result from the absolute differences between the Entry (E) quantities. In addition, Δ_{TOTAL} results from adding the differences (Δ_i) found for all the four types of data movements: that is, the addition of Δ_E, Δ_X, Δ_R and Δ_W. Note in Table 4.7 that although the difference between the measurement obtained in the reference case study (107 CFP) and the measurement obtained with the FSM procedure (118 CFP) is of 11 Cosmic Function Points (Δ_{CFP} = 11 CPF that corresponds to an error percentage of 10.28%), the Δ_{TOTAL} equals 15 CFP that corresponds to an error percentage of 14.02%. Therefore, improvements to the FSM procedure need to be introduced in order to reduce these error percentages.

We begin the evaluation of the FSM procedure by discussing the results of using Qualigram notation in the business application domain, and then we discuss the results of using BPMN. These discussions aim to identify the source of the measurement differences Δ_i. At the end of each discussion (i.e. Qualigram and BPMN discussions) the findings are summarized and then a set of actions to be taken in order to improve the FSM procedure is proposed.

The modeling guidelines defined in subsection 4.2.1.1 embody a slight inconsistency. The same reality (i.e. the measured software) is represented as two different Qualigram concepts, depending on the BPM level of abstraction. At the top level of abstraction (i.e. strategic level), the measured software is represented as the main process; however, at the intermediate level of abstraction (i.e. tactical level), it is represented as an internal role. Qualigram is a management-oriented notation, and does not ask for representation of the information systems supporting the business processes. Therefore, for this research, it has been necessary to provide a modeling guideline at each of the two levels of abstraction to represent the software components for which modeling is required for performing FSM.

Table 4.7 shows that there are some differences in the results obtained based on the FSM procedure proposed in this thesis and those obtained in the C-Registration System case study. These differences can be summarized as follows:

1. there are differences because some modeled information should or should not be considered as a data group; and
2. there are differences because some modeled details are not considered (or are considereded in a different way) by the “C-Registration system” case study.

These differences are described in more detail in the following paragraphs.

The inclusion and analysis of the data groups as part of the flows of information between roles are critical for identifying the Entries (E) and the Exits (X) to be measured from a Qualigram business process model. Consider the Login functional process (FP1 in Table 4.7), as depicted in Figure 4.2: according to the C- Registration System specifications, the registration software has to send a form at the end of the Login procedure. This requirement has been represented as the information flow that starts after the “Display Main Form” instruction. According to the mapping rules, an outgoing flow of information is considered as an Exit (see Table 4.1). However, according to the COSMIC measurement rules, a form sent to a user for entering information cannot be considered as an Exit. Consequently, there is a difference of one Exit between the results of the reference case study and those obtained in this research (see Δ_X column in Table 4.7). To address this difference, the flows of information should include the data groups, and it must be determined during the measurement process whether or not each of the information flows corresponds to a data group.

The importance of the inclusion and analysis of the data groups as part of the of the flows of information is also critical for explaining the differences of Entries (Δ_E) and Exits (Δ_X) found in Table 4.7 for the “Delete a Professor” (FP4), “Delete a Student” (FP8) and “Delete a Schedule” (FP11) functional processes. For instance, consider the “Delete a Schedule” functional process: According to the C-Registration System specifications, the registration software has to send a confirmation requirement after receiving a requirement from the student to delete a schedule, and the student must confirm that deletion requirement to the registration software. This deletion handshake is represented in the Qualigram model of the “Delete a Schedule” functional process (see Figure 4.4). According to the mapping rules (see Table 4.1), an outgoing flow of information is considered as an Exit, and an incoming flow of information is considered as an Entry. However, according to the COSMIC measurement rules, this kind of deletion handshake cannot be considered as a source of data movements. Consequently, there is a difference of one Exit and one Entry between the results of the reference case study and those obtained in this thesis (see Table 4.7). Something similar is what causes the differences of Entries (Δ_E) and Exits (Δ_X) found in the “Delete a Professor”

(FP4) and "Delete a Student" (FP8) functional processes. To address these differences, the flows of information should include the data groups, and it must be determined during the measurement process whether or not each of the flows of information corresponds to a data group.

The difference of one Exit (see Δ_E column in Table 4.7) for the "Select Courses to Teach" (FP5) functional process is caused by the fact that the reference case study apparently considers that the course offerings that the professor wishes to teach for the upcoming semester is updated in the Catalogue System every time this system is consulted about the potential conflicts of the offerings selected by the professor. In this research, these two functions (i.e. updating the Catalogue System and consulting the Catalogue System about potential conflicts) have been disaggregated, because the course offerings should be updated only after the professor has resolved the conflicts. Therefore, we notice here a potential need of clarification in the reference case study.

There is one more Read in the "Create a schedule" (FP9), "Modify a schedule" (FP10) and "Delete a schedule" (FP11) functional processes than in the results of the reference case study (see Δ_R column in Table 4.7). According to the C-Registration system specifications, the registration software has to verify some conditions (e.g. if the registration for the current semester has been closed) before attemping to execute the Create, Modify or Delete requirement sent by the student. The reference case study does not consider the FUR associated with these verifications before meeting the student's requirements. The reason given by the case study is the poor quality of the requirements. Even though this may be true, the required verifications have been considered in this thesis because they have been modeled as instructions to be executed for these functional processes. For instance, in the "Delete a schedule" (FP11) functional process the FUR associated with verifying the status of the registration process (closed or not closed) before meeting the student's requirement (see the "Verify status" instruction in Figure 4.4). Something similar happens with a verification FUR at the beginning of the "Close registration" (FP12) functional process that also causes a difference of one Read (see Δ_R column in Table 4.7). Therefore, we notice here a potential need of improvement of the reference case study.

The specifications mention that during the “Close registration” functional process (FP12), it is possible that the billing system will not respond to the requirements of the registration system. If that is the case, the specifications ask that the requirement be retried an undetermined number of times. The reference case study has not considered this as a functionality to be measured, probably because there is no data group associated with it. However, it has been measured as an Entry here, because the registration system needs to receive a message from the billing system in order to retry the requirement. To address this difference, the flow of information caused by the retry requirement should include the data group, if any, and it must be determined during the measurement process whether or not this flow of information corresponds to a data group.

Finally, the impact of the data groups is again evident in the measurement difference that appears at the “Submit Grades” (FP13) functional process. After retrieving the list of students and retrieving the grades (two different data groups), the specifications ask for a display of the student’s grades. In the Qualigram model, this is represented by only one instruction, which displays the names of the students and their grades, and it counts as one Exit. However, the reference case study considers two Exits, because of the two different data groups. This produces a difference of one Exit (see Δ_X column in Table 4.7). Something very similar happens with the “View Report Card” (FP14) functional process. To address these differences, the flows of information should include the names of the data groups, and it must be determined during the measurement process the number of data groups associated to each of the flows of information.

In summary (see Table 4.8), the inclusion and analysis of the data groups as part of the Qualigram business process model showed to be critical for identifying the Entries (E) and Exits (X) to be measured. Other measurement differences were related to details of the functional processes that were required to be represented as part of the Qualigram business process model, even though they were not considered in the interpretation of the specifications in the case study. Next, we propose an additional Qualigram modeling guideline that should be considered during the measurement process in order to eliminate the differences that result from not including the data groups in the flows of information.

Table 4.8 Summary of measurement differences using Qualigram Notation: C-Registration System

Functional process		Type of difference	Source of differences	Actions to be taken
FP1	Login	Δ_X	Need to consider the data groups associated to the flows of information.	Add a new modeling guideline and consider it during the measurement process.
FP4	Delete a Professor	Δ_E and Δ_X		
FP5	Select Courses to Teach	Δ_X	Different assumptions in the interpretation of the specifications.	Potential need of clarification of the reference case study.
FP8	Delete a Student	Δ_E and Δ_X	Need to consider the data groups associated to the flows of information.	Add a new modeling guideline and consider it during the measurement process.
FP9	Create a schedule	Δ_R	Different assumptions in the interpretation of the specifications.	Potential need of improvement of the reference case study.
FP10	Modify a schedule			
FP11	Delete a schedule	Δ_E and Δ_X	Need to consider the data groups associated to the flows of information.	Add a new modeling guideline and consider it during the measurement process.
FP12	Close registration	Δ_R	Different assumptions in the interpretation of the specifications.	Potential need of improvement of the reference case study.
		Δ_E	Need to consider the data groups associated to the flows of information	Add a new modeling guideline and consider it during the measurement process
FP13	Submit Grades	Δ_X		
FP14	View Report Card			

Modeling Guideline QBA11: Any flow of information should be labeled indicating the name of each of the data groups associated with it. The various data-group names must be separated by commas. If an information flow is not associated with any data group then the label of this information flow must begin with the description **NDG** (i.e. no data group).

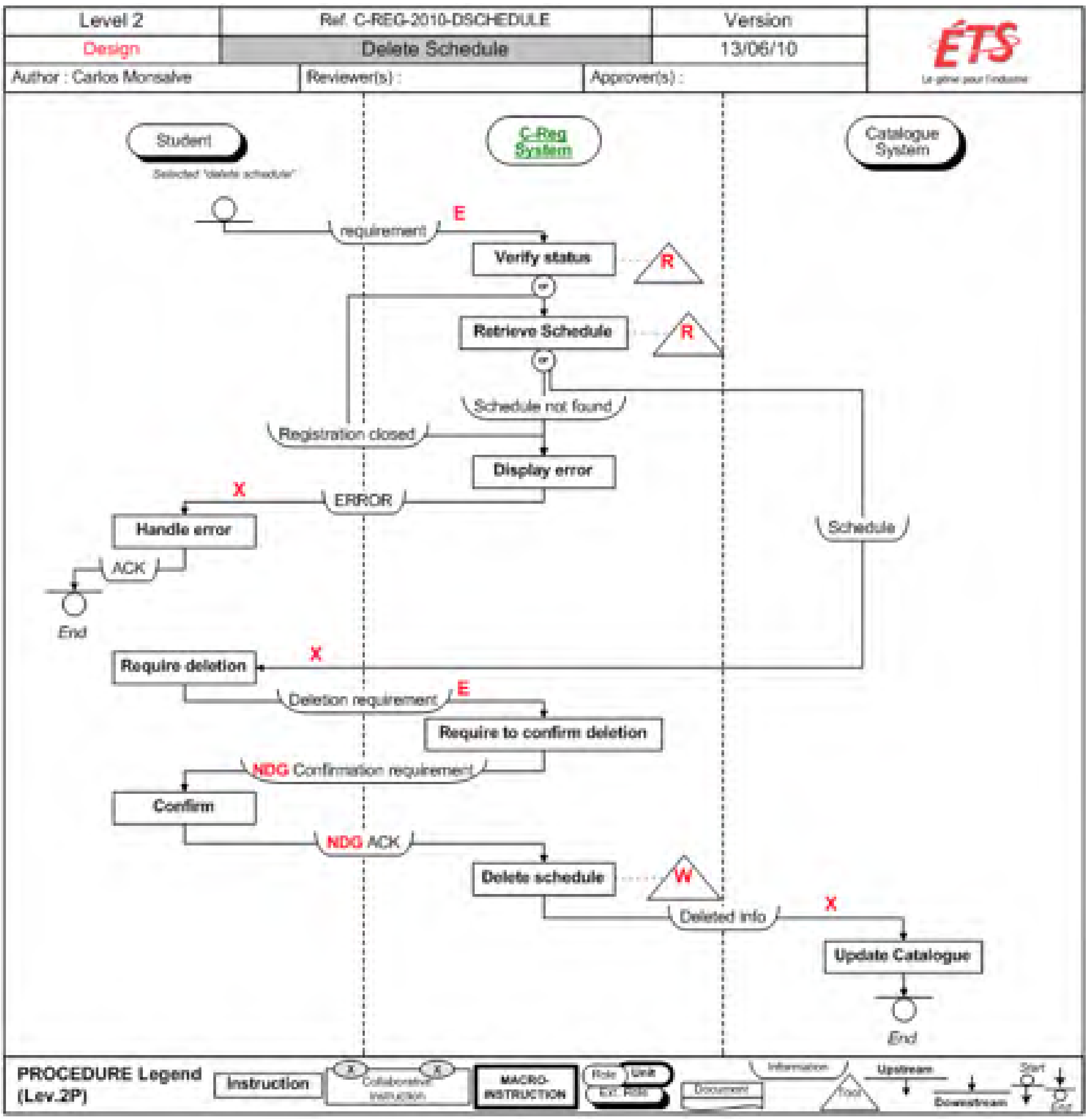

Figure 4.11 Qualigram model of the "Delete a schedule" functional process revisited
Adapted from Monsalve, Abran and April (2010, p. 286),
© copyright Sharker Verlag

The measurement process must incorporate this additional modeling guideline (i.e. QBA11). The number of Entries (E) and Exits (X) represented in the Qualigram models of the various functional processes has to be modulated by the number of data groups associated with each flow of information. For instance, if the label of an outgoing flow of information indicates two data-group names, then two Exits (2X) must be considered for that flow of information. If the label of a flow of information begins with the description **NDG** then that flow of information does not contribute with any data movement. Therefore, Table 4.1 needs to be modified in order to reflect these adjustments. Table 4.9 shows the final version of the mapping rules for Qualigram notation; the improvements are emphasized in red.

Table 4.9 Rules for mapping between COSMIC and Qualigram: final version
Adapted from Monsalve, Abran and April (2011, p. 321),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

COSMIC FSM Method V.3.0.1	Qualigram Notation	Comments
Functional User	Role	Only those roles that interact with the software
Boundary	The process box that represents the software	Top-level model
	The swim-lane of the role that represents the software	Intermediate-level model
Functional Process	Procedure	The procedures included in the process box of the software
Triggering Event	Triggering element	
Data Group	Must be provided as part of the information flow	Between roles. Data-group names separated by commas. **NDG** indicates no data group
	May be provided for describing the material tool	For an instruction that requires access to a persistent storage
Entry	An incoming flow of information	The number of Entries must be determined by the number of data groups associated with it
Exit	An outgoing flow of information	The number of Exits must be determined by the number of data groups associated with it
Read	Description (R) given in a material tool	
Write	Description (W) given in a material tool	

Figure 4.11 depicts an example of how to apply the additional Qualigram modeling guideline (QBA11) and the final version of the mapping rules (refer to Table 4.9) to the "Delete a schedule" functional processes. Observe in Figure 4.11 how the "Confirmation requirement" and the "ACK" flows of information have been affected.

Table 4.10 Comparison of the final measurement results: C-Registration System

Functional Processes		**COSMIC case study Step 1**					**Measurement results obtained with the FSM procedure**									Δ_{Total}
		Data Movements					Data Movements									
		E	X	R	W	CFP	E	Δ_E	X	Δ_X	R	Δ_R	W	Δ_W	CFP	
FP1	Logon	1	1	1		3	1	0	1	0	1	0		0	3	0
FP2	Add a professor	1	2	1	1	5	1	0	2	0	1	0	1	0	5	0
FP3	Modify a professor	2	2	1	1	6	2	0	2	0	1	0	1	0	6	0
FP4	Delete a Professor	2	2	1	1	6	2	0	2	0	1	0	1	0	6	0
FP5	Select Courses to Teach	4	5			9	4	0	6	1		0		0	10	1
FP6	Add a student	1	1	1	1	4	1	0	1	0	1	0	1	0	4	0
FP7	Modify a student	2	2	1	1	6	2	0	2	0	1	0	1	0	6	0
FP8	Delete a Student	2	2	1	1	6	2	0	2	0	1	0	1	0	6	0
FP9	Create a schedule	5	5	1	2	13	5	0	5	0	2	1	2	0	14	1
FP10	Modify a schedule	5	6	2	2	15	5	0	6	0	3	1	2	0	16	1
FP11	Delete a schedule	2	3	1	1	7	2	0	3	0	2	1	1	0	8	1
FP12	Close registration	2	5	1	1	9	2	0	5	0	2	1	1	0	10	1
FP13	Submit Grades	4	5	2	1	12	4	0	5	0	2	0	1	0	12	0
FP14	View Report Card	1	3	2		6	1	0	3	0	2	0		0	6	0
	Total	34	44	16	13	**107**	34	0	45	1	20	4	13	0	**112**	**5**

Table 4.10 shows the final measurement results after applying the additional modeling guideline (i.e. QBA11**)** and the final version of the Qualigram mapping rules (Table 4.9) to all the functional processes of the C-Registration System. Observe that the final measurement obtained with the FSM procedure is now 112 CFP, producing a ΔCFP of five CFP that corresponds to an error percentage of 4.67%. In addition, observe that the Δ_{TOTAL} also equals five CFP. All of these five measurement differences are caused by different assumptions in the interpretation of the C-Registration System specifications. Therefore, the FSM procedure using Qualigram notation does not introduce any error in the final measurement, producing results in conformity with all the rules of the COSMIC ISO 19761 standard.

In the following paragraphs a discussion regarding the results obtained based on BPMN is presented.

The inclusion and analysis of the data groups as part of the messages between pools, or of the sequence flows between lanes, are critical for identifying the Entries and Exits to be measured from the BPMN models of the “Login” (FP1), “Delete a Professor” (FP4), “Delete a Student” (FP8) and “Delete a schedule” (FP11) functional processes. For instance, consider the “Delete a Student” functional process: According to the C-Registration System specifications, the registration software has to send a confirmation requirement after receiving a requirement from the registrar to delete a professor’s record, and the registrar must confirm that deletion requirement to the registration software. This deletion handshake is represented in the BPMN model of the “Delete a Student” functional process (see Figure 4.12). According to the mapping rules (refer to Table 4.5), an outgoing sequence flow is considered as an Exit, and an incoming sequence flow is considered as an Entry. However, according to the COSMIC measurement rules, this kind of deletion handshake cannot be considered as a source of data movements. Consequently, there is a difference of one Exit and one Entry between the results of the reference case study (GELOG-ETS, 2008) and those obtained in this thesis (see Δ_E and Δ_X columns in Table 4.7). To address these differences, the messages and sequence flows should include the data groups, and it must be determined during the measurement process whether or not each of the messages or sequence flows corresponds to a data group.

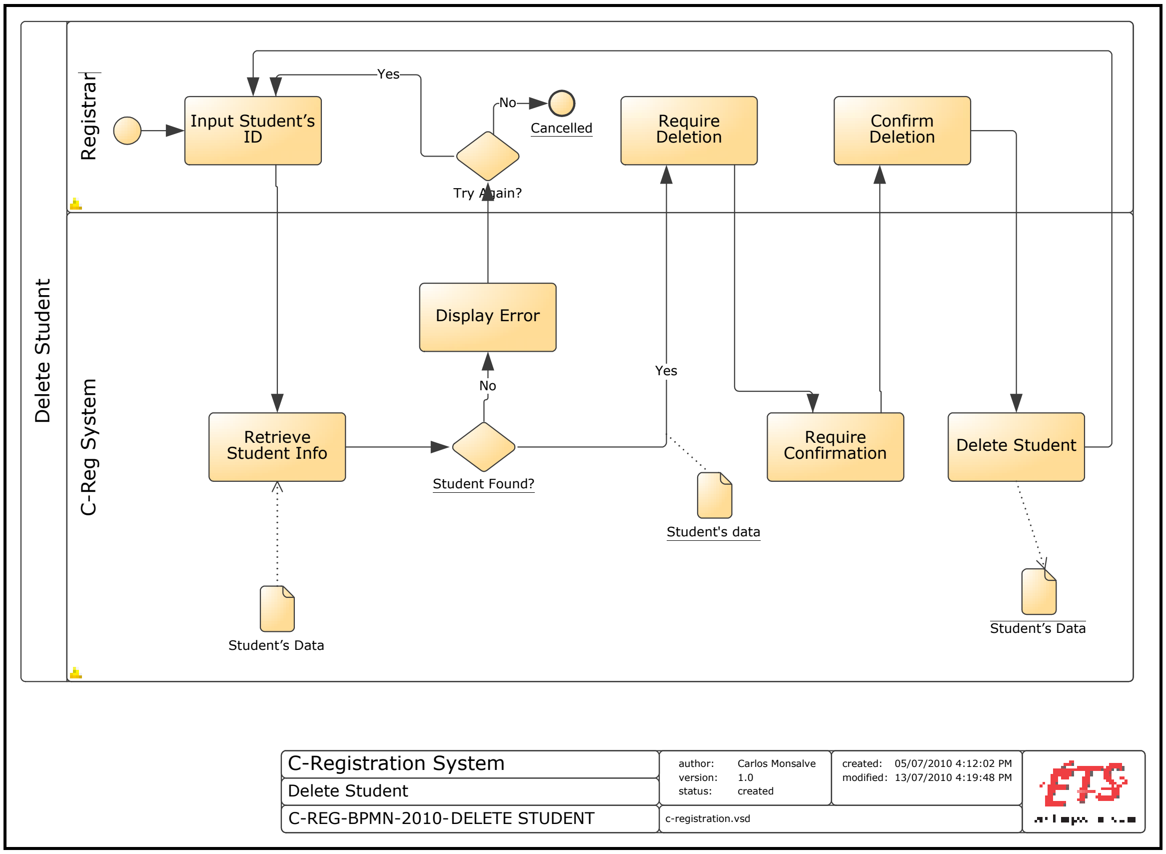

Figure 4.12 BPMN model of the "Delete Student" functional process
Reprinted from Monsalve, Abran and April (2011, p. 333),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

The difference of one Exit for the "Select Courses to Teach" functional process (FP5 depicted in Figure 4.5) is caused by the same reasons that caused the difference of one Exit in the measurement based on the Qualigram model of this functional process. That is, a difference in the interpretation of how the course offerings are updated in the Catalogue System. Therefore, we notice here a potential need of clarification in the reference case study.

There is a difference of one Read for the “Create a schedule” (FP9), “Modify a schedule” (FP10), “Delete a schedule” (FP11) and “Close registration” (FP12) functional processes. These differences are caused by the same reasons that caused the difference of one Read in the measurement based on the Qualigram models of the same functional processes. That is, the reference case study does not consider the FUR associated with some verifications required by the specifications of these functional processes because of the quality of the specifications. However, these verifications have been considered in this thesis because they have been modeled as activities and gateways in the BPMN models of these functional processes. For instance, in the “Close registration” functional process (FP12 depicted in Figure 4.7) the FUR associated with verifying the status of the registration process (in progress or not) before meeting the registrar’s requirement has been modeled as a sequence of the task “Verify Status” followed by the gateway “Registration in Progress?” (see Figure 4.7). Therefore, we notice here a potential need of improvement of the reference case study.

The “Close registration” (FP12) functional process also presents a difference of one Entry (see Δ_E column in Table 4.7). The specifications mention that during the “Close registration” functional process, it is possible that the billing system will not respond to the requirements of the registration system. If that is the case, the specifications ask that the requirement be retried an undetermined number of times. The reference case study has not considered this as a functionality to be measured, probably because there is no data group associated with it. However, it has been measured as an Entry here (see NACK message in Figure 4.7), because the registration system needs to receive a message from the billing system in order to retry the requirement. To address this difference, the message caused by the retry requirement should include the data group, if any, and it must be determined during the measurement process whether or not this message corresponds to a data group.

Finally, the impact of the data groups is again evident in the measurement differences that appear for the “Submit Grades” (FP13) and “View Report Card” (FP14) functional processes. These functional processes require to display to the functional user information that is composed by two data groups, counting therefore as two Exits. However, in the

BPMN models this is represented as only one message or sequence flow that is sent to the functional user, counting as only one Exit. To address these differences, the messages or sequence flows should include the names of the data groups, and it must be determined during the measurement process the number of data groups associated to each of the messages or sequence flows.

In summary (see Table 4.11), the inclusion and analysis of the data groups as part of the BPMN models shown to be critical for identifying the Entries (E) and Exits (X) to be

Table 4.11 Summary of measurement differences using BPMN: C-Registration System

FP	Functional process	Type of difference	Source of differences	Actions to be taken
FP1	Login	Δ_X	Need to consider the data groups associated to the messages and sequence flows.	Add a new modeling guideline and consider it during the measurement process.
FP4	Delete a Professor	Δ_E and Δ_X		
FP5	Select Courses to Teach	Δ_X	Different assumptions in the interpretation of the specifications.	Potential need of clarification of the reference case study.
FP8	Delete a Student	Δ_E and Δ_X	Need to consider the data groups associated to the messages and sequence flows.	Add a new modeling guideline and consider it during the measurement process.
FP9	Create a schedule	Δ_R	Different assumptions in the interpretation of the specifications.	Potential need of improvement of the reference case study.
FP10	Modify a schedule			
FP11	Delete a schedule	Δ_E and Δ_X	Need to consider the data groups associated to the messages and sequence flows.	Add a new modeling guideline and consider it during the measurement process.
FP12	Close registration	Δ_R	Different assumptions in the interpretation of the specifications.	Potential need of improvement of the reference case study.
		Δ_E	Need to consider the data groups associated to the messages and sequence flows.	Add a new modeling guideline and consider it during the measurement process
FP13	Submit Grades	Δ_X		
FP14	View Report Card			

measured. Other measurement differences were related to details of the functional processes that were required to be represented as part of the BPMN models, even though they were not considered in the interpretation of the specifications in the case study. Next, we propose an additional BPMN modeling guideline that should be considered during the measurement process in order to eliminate the differences that result from not including the data groups in the messages between pools or the sequence flows between lanes.

Modeling Guideline BPMN9: Any message or sequence flow should be labeled indicating the name of each of the data groups associated with it. The various data-group names must be separated by commas. If a message or sequence flow is not associated with any data group then the label of this message or sequence flow must begin with the description **NDG** (i.e. no data group).

The measurement process must incorporate this additional modeling guideline (i.e. BPMN9). The number of Entries (E) and Exits (X) represented in the BPMN models of the various functional processes has to be modulated by the number of data groups associated with each message or sequence flow. For instance, if the label of an outgoing message indicates two data-group names, then two Exits (2X) must be considered for that message. If the label of a message or sequence flow begins with the description **NDG** then that message or sequence flow does not contribute with any data movement. Therefore, Table 4.3 needs to be modified in order to reflect these adjustments. Table 4.12 shows the final version of the mapping rules for BPMN; the improvements are emphasized in red.

If we apply the additional modeling guideline (i.e. BPMN9) and the final version of the BPMN mapping rules (Table 4.12) to each of the functional processes of the C-Registration System, then the FSM procedure produces the results shown in Table 4.10: that is a final measurement of 112 CFP, producing a Δ_{CFP} of five CFP that corresponds to an error percentage of 4.67%. In addition, the Δ_{TOTAL} also equals five CFP. All of these five measurement differences are caused by different assumptions in the interpretation of the C-Registration System specifications. Therefore, the FSM procedure using BPMN does not introduce any error in the final measurement measurement, producing results in conformity with all the rules of the COSMIC ISO 19761 standard.

Table 4.12 Mapping between COSMIC and BPMN version 1.2: final version
Adapted from Monsalve, Abran and April (2011, p. 325),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

COSMIC FSM Method V.3.0.1	BPMN 1.2	Comments
Functional User	Lane and pool	Those who interact with the lane of the software to be measured
Boundary	The lane that represents the software to be measured	
Functional Process	Pool	Those that contain the lane that represents the software to be measured
Triggering Event	Start Event	
Data Group	Name of a message or sequence flow	Data group names separated by commas. **NDG** indicates no data group.
	Data Object	When a persistent storage must be accessed
Entry	An incoming message or sequence flow	The number of Entries must be determined by the number of data groups associated to the message or sequence flow.
Exit	An outgoing message or sequence flow	The number of Exits must be determined by the number of data groups associated to the message or sequence flow.
Read	An upstream association with a data object	
Write	A downstream association with a data object	

In the following paragraphs a discussion regarding the impact of the two addtitional modeling guidelines (i.e. QBA11and BPMN9) and the final versions of the mapping-rules tables (i.e. Table 4.9 and Table 4.12) on the notation-independent modeling guidelines is presented.

Modeling guidelines QBA11 and BPMN9 present concepts in common and can be generalized as follows:

Guideline 11: Any flow of information between participants (i.e. roles) should be labeled indicating the name of each of the data groups associated with it. The various data-group names must be separated by commas. If a flow of information between participants is not associated with any data group then the label of this flow of information must begin with the description **NDG** (i.e. no data group).

Table 4.5 needs to be modified in order to reflect this additional modeling guideline. Table 4.13 shows the final version of the generalized mapping rules for the business application software domain; the improvements are shown in the Comments column.

Table 4.13 Rules for mapping between COSMIC and BPM notations: final version
Adapted from Monsalve, Abran and April (2011, p. 329),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company, http://www.worldscinet.com/ijseke

COSMIC FSM method V.3.0.1	BPM notation	Comments
Functional User	Construct that represents a role or participant	
Boundary	The swim-lane of the measured software	
Functional Process	Business process	
Triggering Event	Start Event	
Data Group	Information provided as part of a flow	Data-group names are separated by commas. **NDG** indicates no data group.
	Name of a resource or data object	
Entry	An incoming flow	The number of Entries must be determined by the number of data groups associated with it
Exit	An outgoing flow	The number of Exits must be determined by the number of data groups associated with it
Read	A resource or data object representing the retrieval of data	
Write	A resource or data object representing the writing of data	

4.4.2 Real-time software domain

It is very likely that a BPM notation would not be used for modeling real-time software, as the actual purpose of this kind of modeling notation is to represent organizational business processes. However, it is possible to conclude from the results of this research that, following the correct modeling guidelines, clear and useful models representing real-time software components can be produced.

Table 4.6 shows that the difference between the measurement obtained in the reference case study (11 CFP) and the measurement obtained with the FSM procedure proposed in this thesis (14 CFP) is 3 CFP (Δ_{CFP} = 3 CFP) that corresponds to an error percentage of 27.27%. This measurement difference is caused by 3 extra Exits; that is $\Delta_{TOTAL}=\Delta_{X}$ = 3 CFP. Therefore, improvements to the FSM procedure need to be introduced in order to reduce the error percentage.

The first difference between the results obtained in this research and those obtained by Lavazza and Bianco (2009) is caused by the way in which the two possible signals to be sent to the heater are represented (see FP2 in Table 4.6). Both signals are represented as independent flows of information in the Qualigram model (see Figure 4.9), and are therefore considered as two Exits. However, Lavazza and Bianco considered these two signals as part of the same Exit. In a typical real-time system, an ON/OFF type of control is performed using binary signals that behave as ON/OFF switches. For our case study one binary signal is enough to start or to stop the heater. Therefore, Qualigram notation presents a limitation to represent these binary signals without considering them as a double Exit. To address this limitation, one of the flows of information caused by the binary Software Controller requirement should not be considered during the measurement process.

A second difference is caused by the representation of the way the Software Controller receives data from the Temperature Sensor. According to the specifications, the Temperature Sensor is an external component that interacts with the Software Controller. Therefore, the

Temperature Sensor is modeled in Qualigram notation as an external role (see Figure 4.9). As a consequence, the Software Controller requires to send a “Temperature requirement” to the Temperature Sensor each time the current temperature is needed: this requirement is considered as an Exit. Next, the Temperature Sensor answers the Software Controller sending back the current temperature: this response is considered as an Entry. Lavazza and Bianco (2009, p. 112) mention: “a message is sent to TemperatureSensor (ReadTemp) (an outgoing message), the TemperatureSensor sends back the temperature reading (an Entry)”. However, they only considered the Temperature Sensor response in the measurement (one Entry), causing this a difference of one Exit (see FP2 in Table 4.6). Therefore, we notice here a potential need of clarification in the reference case study (Lavazza and Bianco, 2009).

Finally, the third difference appears in the “Set Target Temperature” functional process (see Figure 4.10). The reference case study (Lavazza and Bianco, 2009) mentions that the elapsed time is proactively sent by the timer to the Software Controller. However, in the Qualigram model, the timer sends the time elapsed after receiving a requirement from the Software Controller. It is therefore considered as an additional Exit in this thesis (see FP3 in Table 4.6). Therefore, the Qualigram model depicted in Figure 4.10 presents a limitation to represent active components that proactively send signals. To address this limitation, an alternative and valid way of representing active components needs to be proposed.

In summary, the treatment and analysis of binary-control signals and active-component signals shown to be critical for identifying the Exits (X) to be measured. One additional measurement difference was related to the requirement signal sent to the Temperature Sensor that was not considered in the measurement performed by the reference case study. Next, we propose two additional real-time modeling guidelines that should be considered during the measurement process in order to eliminate the differences that result from the limitations of the Qualigram models.

Modeling Guideline QRT8: Represent any binary signal (e.g. Start/Stop signal) that should behave as an ON/OFF switch to control an external component, as two flows of information (one for each signal state). Each flow of information should be labeled indicating one of the two possible states of the binary signal. The lable of one of the two flows of information must begin with the description **NDG** (i.e. no data group).

Modeling Guideline QRT9: Represent any proactive signal (e.g. clock signal) sent by an external active component (e.g. timer) to the software being measured as a flow of information triggered by an upstream action assigned to the role representing the active component.

Figure 4.13 shows how to apply the modeling guideline QRT9 to the “Set Target Temperature” functional process.

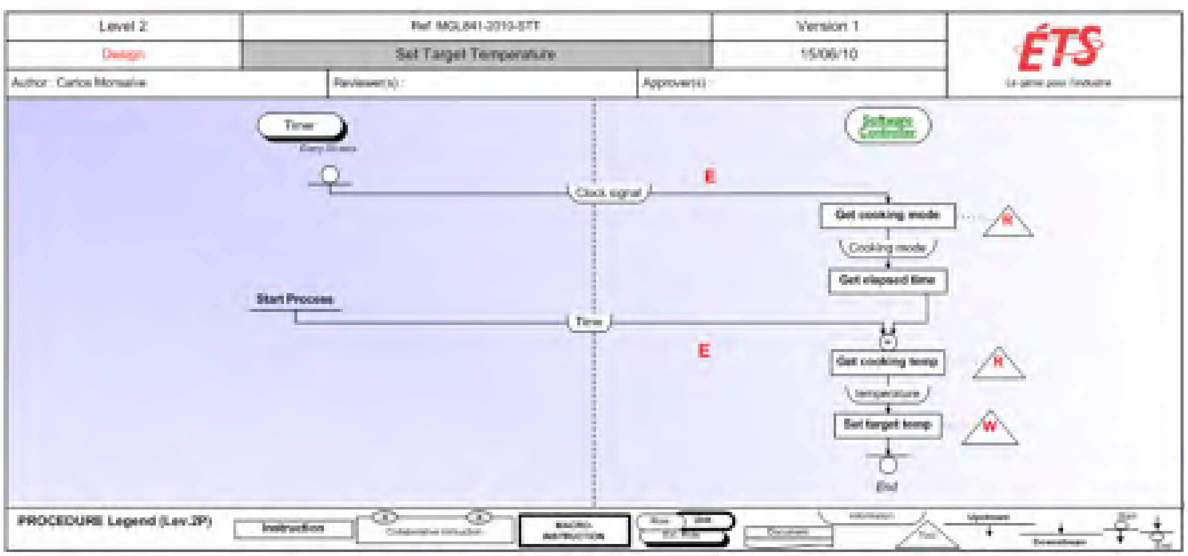

Figure 4.13 Applying modeling guideline QRT9 to the "Set Target Temperature" functional process

The measurement process must incorporate these two additional modeling guidelines. In the case of modeling guideline QRT8, the number of Exits (X) represented in the Qualigram models of the various functional processes has to be modulated by the number of data groups associated with each flow of information. If the label of a flow of information begins with the description **NDG** then that flow of information does not contribute with any data movement.

Therefore, the rules defined in Table 4.9 also apply to the real-time software domain. Figure 4.14 shows how to apply the modeling guideline QRT8 and the final version of the mapping rules (Table 4.9) to the “Adjust Tempertaure” functional process.

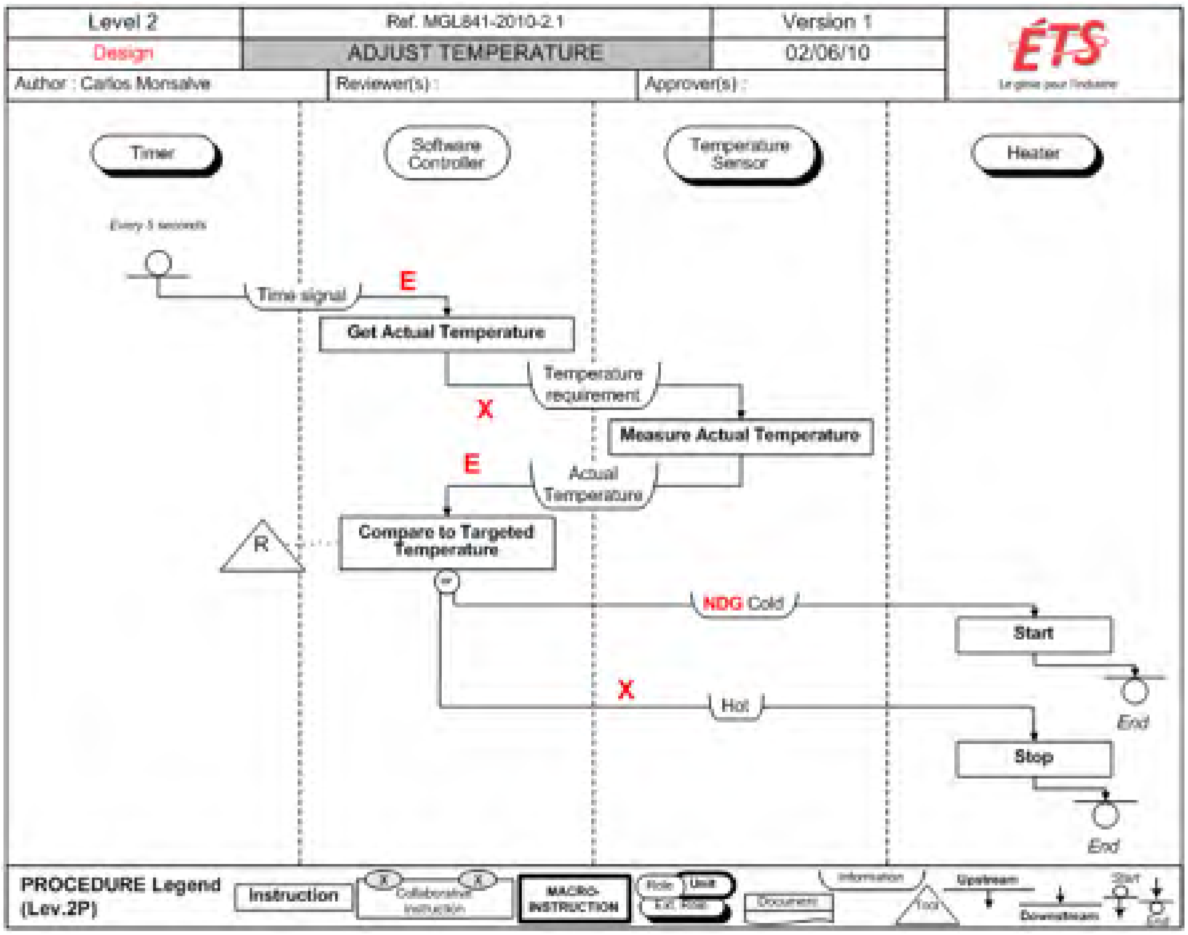

Figure 4.14 Measuring the "Adjust Temperature" functional process
Adapted from Monsalve, Abran and April (2011, p. 331),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

After applying the two additional modeling guidelines and the final version of the mapping rules (Table 4.9), then the FSM procedure produces the results shown in Table 4.14: that is a final measurement of 12 CFP, producing a Δ_{CFP} of one CFP that corresponds to an error percentage of 9.09%. In addition, the Δ_{TOTAL} also equals one CFP. The measurement difference of one CFP, and therefore the error percentage of 9.09%, is caused by the requirement signal sent to the Temperature Sensor that was not measured by the reference case study. Therefore, the FSM procedure for the real-time software domain does not introduce any error in the final measurement, producing results in conformity with all the rules of the COSMIC ISO 19761 standard.

Table 4.14 Final measurement results: Real-time domain
Adapted from Monsalve, Abran and April (2011, p. 332),
DOI: 10.1142/S0218194011005359, © copyright World Scientific Publishing Company,
http://www.worldscinet.com/ijseke

FP	**Results obtained by Lavazza and Del Bianco (Lavazza and Bianco, 2009)**						**Measurement Results based on Qualigram Notation**					
	Process	Data Movements				CFP	Procedures	Data Movements				CFP
		E	X	R	W			E	X	R	W	
FP1	Tick (control lamp)	1	1			2	Control Lamp	1	1			2
FP2	5 sec. Signal management (control heater)	2	1	1		4	Adjust Temperature	2	2	1	0	5
FP3	30 sec. Signal management (set target temperature)	2		2	1	5	Set Target Temp	2		2	1	5
	Total	5	2	3	1	**11**	**Total**	5	3	3	1	**12**

Legend: FP=Functional Process

4.5 Summary

This research has shown the technical feasibility of using BPM for FSM with the COSMIC measurement method (COSMIC FSM). A set of modeling guidelines to represent the software components to be measured using either of two modeling notations (Qualigram or the BPMN 1.2) has been defined for the business application software domain. The modeling guidelines for representing real-time software in Qualigram notation have also been defined. In addition, the rules for mapping between the COSMIC concepts and both the Qualigram and BPMN 1.2 modeling constructs have been derived. The modeling guidelines and mapping rules have been applied to two case studies, one for the business application software domain and the other for the real-time software domain. The results have been compared with those obtained in previous works for the same case studies.

The modeling guidelines for the business application software domain have been generalized, producing a set of notation-independent BPM guidelines for FSM purposes. However, the strengths of these guidelines should be further tested with other popular BPM notations[29]. Moreover, to increase the validity of the guidelines, they should be tested with other case studies rather than only testing with the C-Registration System.

The measurement results show that, following the modeling guidelines and using the mapping rules, BPM might be used successfully for FSM in both software domains. Moreover, there is evidence that the measurement results are not affected by the BPM notation selected. However, the strength of these results should be further tested with other case studies; preferably case studies where it is possible to cover business processes that are typically modeled in the industry. The results obtained using the mapping rules may be compared to the results obtained by expert COSMIC measurers, if that is the case.

In the business application software domain, just a small additional effort is foreseen for modeling the business processes for FSM purposes when using BPMN. The BPMN models generated in this research do not present important differences with those that are typically generated in industry. A different scenario is foreseen when using Qualigram notation. The Qualigram intermediate-level models (i.e. procedure models) generated in this research have required representing the software being measured as another role. Organizations using Qualigram typically do not represent any information system as a role in their intermediate-level models. Therefore, using the proposed modeling guidelines for Qualigram will probably require either a change in the modeling paradigm of organizations (for new business process models), or a rework of the intermediate level models (for already existent business process models). These conclusions are some of the issues that should be tested in the future case studies.

Considering that a common use of BPM is to gather requirements from the early stages of the project (Dumas, van der Aalst and Ter Hofstede, 2005; Indulska *et al.*, 2009; Mayr, Kop and

[29] The notation independent guidelines have been recently adapted in a Master thesis by (Erasmus, 2012) to measure the functional size of the software components of an ERP system modeled with ARIS – EPC.

Esberger, 2007), this raises the opportunity that the proposed FSM procedure might be very useful at the early stages of a software project. Therefore, it needs to be tested against other case studies that are based on high-level specifications typically used at an early stage of the software development process. It will also be necessary to perform more case studies, in order to:

1. evaluate the generalization of the modeling guidelines and mapping rules for the business application software domain;
2. study the additional effort required for the modeler and the organization for applying the modeling guidelines;
3. evaluate the stability of using business process models, which typically change in response to the dynamics of the organization, for FSM as a means for estimating effort; and
4. analyze the advantages and disadvantages of using FSM results as a vehicle to estimate effort based on business processes.

CONCLUSION

Summary

This chapter presents the conclusion of this thesis that has addressed two problems associated to the representation of business processes during the requirements elicitation stage of a software project. The first problem is related to the necessity of generating business process models that contribute to the elaboration of high-quality software requirement specifications; and the second problem is related to the use of a business process model for measuring the functional size of the software that it represents.

The research goal of this thesis has been to contribute to the representation of business processes for its use during the requirements elicitation stage of a software project. More specifically, this thesis has aimed at helping software engineers, business analysts, and BPM practitioners to better model business processes when those models are meant to be used: as part of a software requirements specification document; and for functional size measurement purposes.

To achieve this research goal the following two research objectives were formulated:

1. To propose a novel modeling approach that generates business process models intended to be used in a software requirements elicitation activity. A measure of the success of this proposal is that it should not significantly increase the complexity of the BPM notations used to represent the business processes; and it must allow the active participation of the various stakeholders involved in a typical software project in order to represent, in a consistent and structured way, their needs and constraints. The resulting models should be easily understood and shared by the various stakeholders; easing the communication between them.
2. To develop a procedure to measure the functional size of a software application from the business process models representing its underlying functional requirements. This measurement procedure should be compatible with the COSMIC ISO 19761 FSM

method; and it should be able to be used independently of the BPM notation used to represent the business process.

To achieve the first research objective a research methodology for building a novel BPM approach was designed (see Figure 2.7). An *a priori* version of this novel BPM approach (BPM^+) was designed based on the findings of our literature review (refer to section 3.1) which included the suggestion of modeling at three levels of abstraction. The *a priori* version was iteratively refined based on the results of a pilot case study (refer to section 3.2), a series of representational analyses (refer to section 3.3), and a survey of practitioners with experience in BPM and software requirements elicitation (refer to section 3.4). As a result a reviewed version of BPM^+ was developed (see section 3.5). The reviewed version was then evaluated through a second case study (see section 3.6). Therefore, the design of BPM^+ has been based on a triangulation of evidences (Dahlander, 2005; Miller, 2008; Paré, 2002; Runeson and Höster, 2009) obtained from various sources (i.e. pilot case study, representational analysis and survey).

The pilot case study helped to better define the levels of abstraction of the *a priori* version of BPM^+. The representational analyses helped to identify the specific concepts that should be modeled in accordance to the needs presented by software engineers and business analysts when performing software requirements elicitation activities. The survey helped to further evaluate the scope and content of the levels of abstraction of the *a priori* version of BPM^+. The survey also helped to identify modeling and BPM notation preferences according to the various types of stakeholders involved in a typical software project. The second case study aimed at evaluating the levels of abstraction of the reviewed version of BPM^+, their scope and content, and the modeling preferences according to the various types of stakeholders involved in the project.

To achieve the second research objective a research methodology for developing a FSM procedure based on a business process model was designed (see section 2.5). An analytical comparison between the specifications of COSMIC and the specifications of the BPM notations selected for this research (i.e. BPMN and Qualigram notation) was performed. This analytical comparison helped to define, for each of the two BPM notations, a set of modeling guidelines to allow FSM in the business aplication software domain (refer to subsections 4.2.1.1 and 4.2.2.1). The analytical comparison also allowed to define a set of mapping rules between the modeling constructs of the BPM notations and the COSMIC concepts (refer to subsections 4.2.1.2 and 4.2.2.2). In order to generalize the results obtained for the business application software domain, the set of modeling guidelines for BPMN was compared with the set of modeling guidelines for Qualigram, producing a set of notation independent modeling guidelines (see subsection 4.2.3). In addition, the results obtained using the Qualigram notation were adapted for their application to the real-time software domain, producing a set of Qualigram modeling guidelines for this domain (see section 4.3). Finally, the accuracy of the measurement procedure was evaluated by comparing its measurement results to those obtained in reference case studies published in the COSMIC literature; the measurement differences and their impact were discussed (refer to section 4.4) and improvements to the FSM procedure were added when necessary. These improvements included additional modeling guidelines and final versions of the mapping-rules tables (see Tables 4.9, 4.12 and 4.13). After applying the additional modeling guidelines and the final version of the mapping rules, the errors observed in the initial application of the FSM procedure were eliminated, producing accurate measurement results (see Table 4.10 and Table 4.14) in conformity with all the rules of the COSMIC ISO 19761 standard.

The research question revisited

One research question motivated this thesis: How can a business process be represented to better suit the needs and constraints of the various stakeholders involved in software requirements elicitation activities? To focus the research work, this research question was subdivided in seven sub-questions that are discussed next:

1. What are the needs and constraints of the various stakeholders that should be represented by specific business process modeling constructs when conducting modeling during the software requirements elicitation activity?

The starting point to answer this question was to consider that software engineers and business analysts are both professionals trained to perform requirements elicitation activities, and that each of these two groups of professionals can rely on a guide to its body of knowledge: the SWEBOK Guide, and the BABOK Guide. Therefore, to answer this question we proceeded to identify the needs and constraints of these two groups of professionals when conducting software requirement elicitation activities using the SWEBOK and the BABOK as the references. Table 3.4 showed the relevant software requirements elicitation concepts found in the SWEBOK and the BABOK. As a final step, the mapping of this set of concepts with the BWW representation model was performed to identify a subset of ontological concepts which, according to the SWEBOK and the BABOK, represents concepts that are relevant to the domain of software requirements elicitation. The selected subset of ontological concepts was presented in Table 3.11. Therefore, a business process model should incorporate these ontological concepts in order to represent the needs and constraints of the various stakeholders when conducting a software requirements elicitation activity. Ideally, each of these ontological concepts should be represented by a specific modeling construct of the BPM notation used to generate the business process models.

2. What is the appropriate level of abstraction to represent all these modeling constructs in a business process model? If more than one level of abstraction is required, then what modeling constructs should be represented at each level of abstraction?

The literature confirmed that the various groups of stakeholders might have conflicting requirements for the representation of the same business process, and that it is not possible to represent all the required business process elements into one single model (refer to section 1.4). To address this, many authors have argued that the use of multiple levels of abstraction helps to represent the appropriate information to be provided to various types of stakeholders (refer to section 1.5).

This thesis has reviewed several approaches to incorporate multiple levels of abstraction in BPM (refer to subsection 1.5.2). Moreover, all BPM research proposals reviewed recommend the use of three levels of abstraction, even though, depending on the authors, the content of each level of abstraction varies from one proposal to another. From all these approaches and proposals, this thesis has chosen Anthony's model (refer to subsection 1.5.1) as the basis to propose three levels of abstraction to model business processes: the strategic, tactical and operational levels of abstraction (refer to Figure 3.1). Tables 3.30, 3.31 and 3.32 presented the sets of modeling concepts that should be represented at the strategic, tactical and operational levels of abstraction respectively.

3. How well do current business process modeling notations represent these levels of abstraction and modeling constructs?

The literature has considered that BPMN is the most complete of all BPM graphical notations (refer to subsection 1.3.2.1). This is one of the reasons why BPMN has been selected in this thesis; it can be argued that BPMN is an appropriate benchmark to assess the capability of other BPM notations to represent the required modeling concepts and the three levels of abstraction. The literature also confirmed that some of the characteristics that organizations look for in a BPM notation are simplicity and ease of use; however, BPMN is considered as a highly complex BPM notation (refer to subsection 1.3.2). To address this concern, this thesis also selected Qualigram, which is considered as a simple and easy to understand notation.

To assess the capability of these two BPM notations to represent the selected set of modeling concepts, the modeling constructs of the two notations were mapped with the selected set of ontological concepts relevant to the domain of software requirements elicitation (see Table 3.11). The results of these mappings were presented in Table 3.12, showing that BPMN and Qualigram are equally capable to represent the selected set of concepts. The results also showed that neither BPMN nor Qualigram are capable of representing some ontological concepts that have been confirmed by the literature as concepts that allow describing the business rules of an organization (refer to subsection 3.3.1.4). Therefore, BPMN and Qualigram are equally capable of representing all the selected relevant concepts but business rules.

The capability of BPMN and Qualigram to support the three levels of abstraction (i.e. strategic, tactical and operational) was assessed by conducting two case studies (refer to sections 3.2 and 3.6). Both case studies confirmed that Qualigram is capable of supporting the three levels of abstraction. However, BPMN was not capable of fully supporting the strategic level of abstraction (see subsections 3.2.2.2 and 3.6.2.3).

4. What would be a proposed BPM approach for consistently representing the various needs and constraints at their appropriate level of abstraction?

This thesis presented the BPM$^+$ modeling approach (refer to Appendix XI) that was designed based on an abstraction hierarchy (see Figure 3.1) that includes the three levels of abstraction: strategic, tactical and operational. In BPM$^+$ a business process is composed of a set of procedures, a procedure is composed of a set of activities, and an activity is composed of a set of atomic tasks. All the operational-level models are integrated through tactical-level models, the tactical-level models are integrated through strategic-level models, and the entire business process models are integrated through a main strategic-level model. The modeling concepts to be represented at each level of abstraction (Tables 3.30, 3.31 and 3.32) have been delimited based on the findings from the literature review (see section 1.4), the results of a survey of practitioners with experience (see section 3.4) and the results of representational analyses (see section 3.3). Moreover, BPM$^+$ modeling concepts take into consideration the set of ontological concepts specifically selected to represent the modeling needs and constraints when conducting a software requirements elicitation activity (refer to Table 3.11). BPM$^+$ has proven to represent in a consistent way the modeling needs of the various stakeholders involved in two case studies (refer to sections 3.2 and 3.6). These two case studies also helped to assess the scope of BPM^+ levels of abstraction and the modeling concepts defined at each level of abstraction.

5. If a business process model represents software functional requirements, then can it be used for measuring the functional size of the software it represents? If so, is there some notation-specific business process modeling guidelines required to allow this measurement?

This thesis has shown through three measurement case studies the technical feasibility of using a business process model as a basis for measuring the functional size of the software it represents. All the measurements were obtained based on the COSMIC FSM method – ISO 19761. Two of the measurement case studies were conducted for the business application software domain: one with Qualigram (see subsection 4.2.1) and the second one with BPMN (see subsection 4.2.2). The third measurement case study was conducted for the real-time software domain (see section 4.3). For each case study a set of notation-specific modeling guidelines and a set of notation-specific mapping rules were defined. The results obtained in these three measurement case studies were evaluated through their comparison with the results obtained in previous works for the same software applications (see section 4.4). The analysis of the evaluation of the measurement results (see section 4.4) drove us to formulate additional modeling guidelines and to improve the mapping rules. The additional modeling guidelines and the final version of the mapping rules (see Table 4.9 and Table 4.12) were applied in order to obtain final measurement results in conformity with all the rules of the COSMIC ISO 19761 standard. The final measurement results (see Table 4.10 and Table 4.14) showed that, following the modeling guidelines and using the mapping rules, a business process model can be used successfully for FSM in both software domains. The results also confirmed that the modeling guidelines need to be notation-specific (e.g. BPMN guidelines, Qualigram guidelines), and software-domain oriented (i.e. business application domain oriented, real-time domain oriented).

6. What would be the set of notation-independent business process modeling guidelines for measuring the software functional size?

The measurement results obtained in this thesis using two different BPM notations are evidence that the measurement results are not affected by the BPM notation selected. This finding motivated us to derive a set of notation-independent BPM guidelines for FSM purposes. The set consists of eleven (11) modeling guidelines that are described in subsection 4.2.3 and section 4.4. In addition, a set of mapping rules (see Table 4.13) were also derived in order to be used together with the notation-independent modeling guidelines. These notation-independent modeling guidelines may be used to derive other notation-specific BPM guidelines for FSM purposes. For instance, Erasmus (2012) has used the notation-independent guidelines proposed in this thesis to derive a set of EPC-specific modeling guidelines (i.e. modeling guidelines for the Event-driven Process Chain notation) for FSM purposes as one of the components of his Master thesis.

7. What would be the procedure for measuring functional size using a business process model?

The procedure for measuring the functional size of a software application from the business process model that represents it has been described in CHAPTER 4 of this thesis. First, a business process model of each of the functional procedures of the software to be measured must be generated based on its requirements specifications and following the modeling guidelines available for the BPM notation used to generate the model. Once the business process models have been generated, the mapping rules available for the used BPM notation need to be applied to each of the models in order to identify and to tag the data movements (i.e. E, X, R, and W). Finally, the functional size of the software is obtained by adding the data movements identified.

Contributions and outcomes of this research

The main contributions of this thesis are:

1. A business process modeling approach named BPM^+ designed based on a multi-level hierarchical structure where the modeling concepts have been defined based on the results of ontological analyses. BPM^+ proposes to model at three levels of abstraction (i.e. strategic, tactical and operational) to generate business process models that represent in a consistent and structured way the needs and constraints of the various stakeholders involved in a typical BPM project. The resulting models are easy to be understood and shared by the various stakeholders, easing the communication between them.
2. A procedure for measuring the functional size of a software application from the business process model representing it. The measurement procedure is compatible with the COSMIC FSM method (ISO 19761) and it can be applied independently of the modeling notation used to represent the business process. The procedure includes the use of a set of modeling guidelines and a set of mapping rules. The modeling guidelines help the modeler to elaborate models that can be used for FSM purposes. The mapping rules allows to identify the data movements in the business process model.

In addition to these two main contributions, the following contributions have also been brought out in this thesis as a result of the process of developing it:

1. A set of ontological concepts specifically selected to represent the relevant concepts of the software requirements elicitation domain as described by the SWEBOK and the BABOK (Table 3.11).
2. An assessment of the capability of Qualigram notation and BPMN to represent this set of ontological concepts (subsection 3.3.1.4).
3. The meta-model of Qualigram notation (Appendix V).
4. The representational analysis of Qualigram notation (Appendix VI).
5. A set of Qualigram modeling guidelines for FSM purposes in the business application software domain (subsections 4.2.1.1 and 4.4.1)

6. The mapping rules between the COSMIC concepts and the Qualigram modeling constructs (Table 4.9).
7. A set of BPMN modeling guidelines for FSM purposes in the business application software domain (subsections 4.2.2.1 and 4.4.1).
8. The mapping rules between the COSMIC concepts and the BPMN version 1.2 modeling constructs (Table 4.12).
9. A set of Qualigram modeling guidelines for FSM purposes in the real-time software domain (subsections 4.3.1 and 4.4.2)
10. A set of notation-independent business process modeling guidelines for FSM purposes (subsections 4.2.3 and 4.4.1)

Finally, the development of this thesis has produced outcomes that have been published at the following conferences and journal:

1. Monsalve, C., A. April and A. Abran. 2010. «Representing unique stakeholder perspectives in BPM notations»[30]. 8th ACIS International Conference on Software Engineering Research, Management and Applications, SERA 2010. (Montreal, May 24-26), p. 42-49.
2. Monsalve, C., A. Abran and A. April. 2010. «Functional Size Measurement with Business Process Models: the Business Application Domain»[31]. International Workshop on Software Measurement IWSM/MetriKon/Mensura Conference 2010. (Stuttgart, Germany, November 10-12), p. 270-290. (Best paper award and the best presentation award).
3. Monsalve, C., A. April and A. Abran. 2011. «BPM and requirements elicitation at multiple levels of abstraction: a review»[32]. IADIS International Conference Information Systems 2011. (Avila, Spain, March 11-13), p. 237-242. International Association for Development of the Information Society (IADIS).

---

30 (Monsalve, April and Abran, 2010)
31 (Monsalve, Abran and April, 2010)
32 (Monsalve, April and Abran, 2011a)

4. Monsalve, C., A. April and A. Abran. 2011. «Requirements elicitation using BPM notations: focusing on the strategic level representation»[33]. 10th WSEAS International Conference on Applied Computer and Applied Computational Science (ACACOS '11). (Venice, Italy, March 8-10), p. 235-241. WSEAS Press.
5. Monsalve, C., A. Abran and A. April. 2011. «Measuring Software Functional Size from Business Process Models»[34]. International Journal of Software Engineering and Knowledge Engineering (IJSEKE), vol. 21, no 3, p. 311-338.
6. Monsalve, C., A. April and A. Abran. In press. «On the Expressiveness of Business Process Modeling Notations for Software Requirements Elicitation»[35]. 38th Annual Conference of the IEEE Industrial Electronics Society, IECON 2012. (Montreal, October 25-28).

Expected industry impacts

The results and findings of this thesis can provide the following benefits to the requirements elicitation, business process modeling and software measurement communities, including software engineers, information systems and information technology professionals, business analysts, BPM vendors and practitioners, and software measurers:

- A business process modeling approach (BPM^+) that contributes to the efforts of closing the gap between IT and non-IT stakeholders when modeling a business process and that allows representing in a consistent and structured way their modeling needs and constraints.
- New insights on the capability of BPMN to support the levels of abstraction and to represent the modeling concepts found in this thesis as determinants for generating business process models that contribute to high-quality SRS.
- A meta-model of Qualigram notation that helps to formalize its textual specifications. The benefit of this meta-model is twofold: 1) it may help the designers of Qualigram's

[33] (Monsalve, April and Abran, 2011b)
[34] (Monsalve, Abran and April, 2011)
[35] (Monsalve, April and Abran, In press)

tools to improve the specifications of Qualigram notation; and 2) it can help Qualigram's practitioners (users of Qualigram's tools) to better understand the notation.

- An assessment of Qualigram notation that illustrates some improvement opportunities of this notation.
- A comparison of BPMN and Qualigram notations that shows that there are no major differences between them if the modeling goal is the documentation of business processes for software requirements elicitation or functional size measurement purposes.
- A new procedure for measuring the functional size of software with the COSMIC FSM method.
- Modeling guidelines for BPMN and Qualigram notation in order to generate or update business process models to be used for FSM.
- Mapping rules for BPMN and Qualigram notation that can be implemented in their corresponding modeling tools in order to facilitate the identification of COSMIC data movements in the business process models generated by using these BPM notations.
- Modeling guidelines and mapping rules that can be used as a basis for automating the functional size measurement of software applications according to the COSMIC ISO 19761 standard.

Limitations

Some limitations and validity threats of this research have been identified and described throughout this thesis. An overview of these limitations and validity threats was presented in Table 2.8. We discuss below some additional limitations:

- The literature showed that there are many business process modeling notations available. However, this thesis has used only two of them (BPMN and Qualigram) as discussed in section 1.3h 1.3. This might threaten the generalization of the two main results of this thesis. To overcome this threat, future research work has to study the applicability of these results with other BPM notations.
- The various research activities of this thesis have used version 1.2 of BPMN, but the current 2012 version is 2.0. This might require updating some of the findings and results

of this thesis. However, whenever it has been considered necessary, this thesis has incorporated some comments related to BPMN version 2.0. It has been shown that the impact of the new version of BPMN in the results of this thesis is small.

- The development of the BPM^+ approach could have included other types of research activities; for instance: interviewing experts. However, various other types of research methods were conducted (i.e. survey, representational analysis, case study) looking for a triangulation of evidence. Moreover, a theoretical framework was used as a basis for the development of BPM^+.
- The case studies conducted to develop BPM^+ were selected based on the availability and willingness of the participant organizations. As a consequence, none of the two case studies reported an experience in the use of BPM during a software requirements elicitation activity; affecting the significance of the results obtained from these case studies. However, the case studies were useful to evaluate the various concepts of BPM^+ with different types of stakeholders; and the obtained results were complemented with the results of other research activities conducted as part of this thesis.
- The empirical evidences obtained in this thesis to support the scope and modeling concepts of the operational level of abstraction of BPM^+ are limited. We expected more specific evidences from our second case study, but time limitations and the lack of experience of the participants in business process modeling impacted the research data collected during this case study. To overcome this limitation, future research work has to test more in-depth the operational level of abstraction.
- The FSM procedure developed in this thesis, its modeling guidelines, and the mapping rules, can only be applied with the COSMIC FSM method. Therefore, this opens new avenues of research to study the technical feasibility of developing similar measurement procedures using other FSM methods.

Recommendations for further research work

The following discussion aims to motivate the undertaking of new research to build on or to develop the contributions to the knowledge generated in this thesis:

- The BWW representation model can be revised and improved if necessary based on the findings from the representational analyses conducted in this thesis.
- The survey can be replicated with BPM and software requirements experts on a world-wide basis.
- The adaptability of BPM^+ to other BPM notations can be studied by mapping the modeling concepts proposed by BPM^+ with the modeling constructs of other BPM notations and conducting new case studies where the business process models are generated using these other BPM notations.
- The author of this thesis recognises that additional empirical research should be conducted in order to further test the operational level of abstraction of BPM^+. Moreover, additional empirical research would allow collecting more evidences to support BPM^+ and the generalization of its benefits.
- The additional empirical research to test BPM^+ would also allow verifying the need expressed by the participants of our second case study to represent internal roles at the detailed type of model of the strategic level of abstraction.
- A set of translation rules or translation mappings between BPMN and Qualigram notation might be derived based on a comparison of the representational analyses performed in this thesis. This research work is underway, including an algorithm for the correct use of these translation rules.
- The notation-independent BPM guidelines for FSM purposes can be adapted to develop modeling guidelines and mapping rules for other popular BPM notations. The new modeling guidelines and mapping rules should be tested with additional case studies. The results of these case studies would help to increase the validity of the notation-independent BPM guidelines.
- The measurement procedure proposed in this thesis should be tested with new measurement case studies rather than only testing it with the C-Registration System and the Rice Cooker system. These new case studies should cover business processes that are typically modeled in the industry. The results obtained may be compared to the results obtained by expert COSMIC measurers, if that is the case.

- The new measurement case studies can also help to study the additional effort required by organizations for generating or updating business process models in order to be used for FSM purposes.
- The measurement procedure developed in this thesis might be used at the early stages of a software project. Therefore, it needs to be tested against other case studies that are based on high-level specifications typically used at an early stage of a software project.
- The advantages and disadvantages of using FSM results as a vehicle to estimate effort based on business processes can be studied.
- A prototype tool can be developed as a proof of concept of the automation of the FSM procedure proposed in this thesis.
- The technical feasibility to develop new FSM procedures based on BPM can be studied for other FSM methods recognized as ISO standards.

ANNEX I

LIST OF APPENDICES ON CD-ROM

The following is the list of appendices referenced within this thesis and that can be found on the attached CD-ROM:

Table-A I-1 List of appendices referenced within the thesis

Appendix #	File name	Description
I	SYS869.zip	SYS869 research activities protocol
II	CÉR_sys869.pdf	CÉR approval within the ethics domain of the SYS869 research activities protocol
III	case_study.zip	Case study protocol
IV	CÉR_case_study.pdf	CÉR approval within the ethics domain of the case study protocol
V	Qmetamodel.zip	Qualigram Meta-model
VI	qBWW.xlsx	Results of Qualigram representational analysis
VII	Recker_work.pdf	Results of BPMN representational analysis performed by Recker *et al.*
VIII	questionnaire.pdf	Survey questionnaire
IX	marketing.pdf	Survey announcements and invitations
X	BWWconcepts.zip	The BWW representation model concepts
XI	BPM+.pdf	Final version of BPM^+

LIST OF BIBLIOGRAPHICAL REFERENCES

Abran, A., JW Moore, P. Bourque and R. Dupuis. 2004. *SWEBOK: Guide to the Software Engineering Body of Knowledge 2004 Version*. Los Alamitos, California: IEEE Computer Society, 202 p.

Abran, Alain. 2010. *Software Metrics and Software Metrology*. New Yersey: John Wiley & Sons Interscience and IEEE-CS Press, 328 p.

Abran, Alain, Lucie Laframboise and Pierre Bourque. 2003. « A Risk Assessment Method and Grid for Software Measurement Programs ». *Institute of Chartered Financial Analysts of India (ICFAI) Journal of Systems Management*, n° 3, p. 34.

Aguilar-Saven, RS. 2004. « Business process modelling: Review and framework ». *International Journal of Production Economics,* vol. 90, n° 2, p. 129-149.

Albani, A., and J. Dietz. 2006. « The Benefit of Enterprise Ontology in Identifying Business Components ». In *The Past and Future of Information Systems: 1976–2006 and Beyond*, ed. Avison, David, Steve Elliot, John Krogstie and Jan Pries-Heje. Vol. 214, p. 243-254. Coll. « IFIP International Federation for Information Processing ». Santiago, Chile: Springer Boston.

Ami, T., and R. Sommer. 2007. « Comparison and evaluation of business process modelling and management tools ». *International Journal of Services and Standards,* vol. 3, n° 2, p. 249-261.

Anthony, Robert N. 1965. *Planning and Control Systems: A Framework for Analysis*. Coll. « Studies in Management Control ». Boston: Division of Research, Graduate School of Business Administration, Harvard University, 180 p.

Antonellis, V. De, and B. Zonta. 1990. « A Disciplined Approach to Office Analysis ». *IEEE Transactions on Software Engineering,* vol. 16, n° 8, p. 822-828.

Bandara, W., and M. Rosemann. 2005. « What Are the Secrets of Successful Process Modeling? Insights From an Australian Case Study ». *Systèmes d'Information et Management,* vol. 10, n° 3, p. 47-68.

Basili, V R, R W Selby and D H Hutchens. 1986. « Experimentation in software engineering ». *IEEE Transactions on Software Engineering,* vol. 12, n° 7, p. 733-743.

Basili, Victor R., and Richard W. Selby. 1991. « Paradigms for experimentation and empirical studies in software engineering ». *Reliability Engineering & System Safety,* vol. 32, n° 1-2, p. 171-191.

Baskerville, R, and MD Myers. 2004. « Special issue on action research in information systems: Making IS research relevant to practice - foreword ». *MIS Quarterly,* vol. 28, n° 3, p. 329-335.

Becker, J., M. Rosemann and C. von Uthmann. 2000. « Guidelines of Business Process Modeling ». In *Business Process Management*, ed. Aalst, Wil van der, Jörg Desel and Andreas Oberweis. Vol. 1806, p. 241-262. Coll. « Lectures Notes in Computer Science », 1806/2000: Springer Berlin / Heidelberg.

Berger, C., and S. Guillard. 2000. *La rédaction graphique des procédures : démarche et techniques de description des processus*. Paris: Association Française de Normalisation, AFNOR, 250 p.

Bhat, Jyoti M., and Nivedita Deshmukh. 2005. « Methods for Modeling Flexibility in Business Processes ». In *Sixth Workshop on Business Process Modeling, Development, and Support, BPMDS'05*. (Porto, Portugal, June 13-14).

Burton-Jones, A., and PN Meso. 2006. « Conceptualizing Systems for Understanding: An Empirical Test of Decomposition Principles in Object-Oriented Analysis ». *Information Systems Research,* vol. 17, n° 1, p. 38-60.

Casewise. 2011. *The Business Process Management Marketplace by Numbers*. Infographic. Casewise Systems, 1 p. < http://blog.casewise.com/2011/10/the-business-process-management-marketplace-by-numbers/ >. Consulted on August 15, 2012.

Castro, Jaelson, Manuel Kolp and John Mylopoulos. 2002. « Towards requirements-driven information systems engineering: the Tropos project ». *Information Systems,* vol. 27, n° 6, p. 365-389.

CEPHAS. 2009. « CEPHAS Consulting Corp.- Enterprise Computing ». < http://cephas.cc/sitecontent/ >. Consulted on December 2009.

COSMIC. 2003. *The COSMIC-FFP, Measurement Manual (The COSMIC Implementation Guide for ISO/IEC 19761: 2003) Version 2.2.*: The Common Software Measurement International Consortium (COSMIC), 81 p. Consulted on August 15, 2012.

COSMIC. 2008. *Case Study: Rice Cooker*. Case study. Montréal: COSMIC group. < http://www.cosmicon.com/portal/dl_showall.asp?cat_id=69&parent_id=17&parent_name=04+-+Case+Studies&sub_name=Real-time >. Consulted on October 2010.

COSMIC. 2009. *The COSMIC Functional Size Measurement Method Version 3.0.1, Measurement Manual (The COSMIC Implementation Guide for ISO/IEC 19761: 2003).* The Common Software Measurement International Consortium (COSMIC). < http://www.cosmicon.com/portal/public/COSMIC%20Method%20v3.0.1%20Measurement%20Manual.pdf >. Consulted on August 15, 2012.

Courtney, James F. 2001. « Decision making and knowledge management in inquiring organizations: toward a new decision-making paradigm for DSS ». *Decision Support Systems,* vol. 31, n^{o} 1, p. 17-38.

Curtis, B., M. I. Kellner and J. Over. 1992. « Process modeling ». *Communications of the ACM,* vol. 35, n^{o} 9, p. 75-90.

Dahlander, Linus. 2005. « Appropriation and Appropriability in Open Source Software ». *International Journal of Innovation Management,* vol. 09, n^{o} 03, p. 259-285.

Daneva, Maya. 1999. « Measuring reuse of SAP requirements: a model-based approach ». In *1999 Symposium on Software Reusability*. (Los Angeles, California, United States), p. 141-150. ACM.

Daoudi, Feriel, and Selmin Nurcan. 2007. « A benchmarking framework for methods to design flexible business processes ». *Software Process: Improvement and Practice,* vol. 12, n^{o} 1, p. 51-63.

Davenport, T. H. 1993. *Process innovation: reengineering work through information technology*. Boston, Ma: Harvard Business School Press, 337 p.

Davies, I., P. Green, M. Rosemann, M. Indulska and S. Gallo. 2006. « How do practitioners use conceptual modeling in practice? ». *Data & Knowledge Engineering,* vol. 58, n^{o} 3, p. 358-380.

Davies, Islay G., Michael Rosemann and Peter Green. 2004. « Exploring proposed ontological issues of ARIS with different categories of modellers ». In *Australasian Conference on Information Systems, (ACIS 2004)*. (Hobart, Australia, 1-3 December). Australasian Chapter of the Association for Information Systems <http://aisel.aisnet.org/acis2004/50 >.

Davis, Rob. 2008. *ARIS Design Platform: Advanced Process Modeling and Administration*, 1st. Edition. Coll. « Business and Economics ». London: Springer London.

Davison, R, MG Martinsons and N Kock. 2004. « Principles of canonical action research ». *Information Systems Journal,* vol. 14, n^{o} 1, p. 65-86.

Decreus, Ken, and Geert Poels. 2009. « Mapping semantically enriched Formal Tropos to business process models ». In *2009 ACM Symposium on Applied Computing (SAC' 09)*. (Honolulu, Hawaii, March 9-12), p. 371-376. ACM.

Demirors, O., and C. Gencel. 2004. « A comparison of size estimation techniques applied early in the life cycle ». In *11th European Software Process Improvement Conference,* *EuroSPI 2004*. (Trondheim, Norway, November 10-12). Vol. 3281, p. 184-194. Berlin / Heidelberg: Springer.

Demirors, O., C. Gencel and A. Tarhan. 2003. « Utilizing business process models for requirements elicitation ». In *29th Euromicro Conference, New Waves in System Architecture*. (Belek-Antalya, Turkey, September 3-5), p. 409-412. IEEE Computer Society

Desel, J. 2005. « Process modelling using Petri nets ». In *Process-Aware Information Systems-Bridging People and Software through Process Technology*. p. 147-176. Hoboken, N.J.: Wiley, Chichester.

Dixon, John, and Teresa Jones. 2011. *Hype Cycle for Business Process Management, 2011* Coll. « Hype Cycles », G00214214 USA: Gartner, Inc., 89 p. < http://www.gartner.com/id=1751119 >. Consulted on August 15, 2012.

Dreiling, Alexander, Michael Rosemann, Wil M. P. van der Aalst and Wasim Sadiq. 2008. « From conceptual process models to running systems: A holistic approach for the configuration of enterprise system processes ». *Decision Support Systems,* vol. 45, nº 2, p. 189-207.

Dumas, M., W. van der Aalst and A. Ter Hofstede. 2005. *Process-aware information systems: Bridging people and software through process technology*. Hoboken, N.J.: Wiley-Interscience, 409 p.

Dumas, P, and G Charbonnel. 1990. *La méthode OSSAD: pour maîtriser les technologies de l'information*, Tome 1: Principes. Paris, France: Les Editions D'Organisation, 160 p.

Dwyer, T. 2006. *BPMInstitute's State of Business Process Management: "Assessing the Current State of BPM Awareness and Usage"*. BPMInstitute.org, 14 p. < www.bpminstitute.org >. Consulted on May 2010.

Easterbrook, S., J. Singer, M.-A. Storey and D. Damian. 2008. « Selecting Empirical Methods for Software Engineering Research ». In *Guide to Advanced Empirical Software Engineering*. p. 285-311. Coll. « Computer Science ». London: Springer.

Eikebrokk, Tom R., Jon Iden, Dag H. Olsen and Andreas L. Opdahl. 2011. « Understanding the determinants of business process modelling in organisations ». *Business Process Management Journal,* vol. 17, nº 4, p. 639-662.

Elzinga, D. J., T. Horak, Chung-Yee Lee and Ch. Bruner. 1995. « Business process management: Survey and methodology ». *IEEE Transactions on Engineering Management,* vol. 42, nº 2, p. 119-128.

Engels, G., A. Förster, R. Heckel and S. Thöne. 2005. « Process modeling using UML ». In *Process-Aware Information Systems*. p. 85-117. John Wiley & Sons, Inc.

Erasmus, Pierre Izak. 2012. « The COSMIC EPC method-An ERP functional size measurement method delivering time and cost estimates ». Master Thesis. Göteborg, Sweden, University of Gothenburg, 75 p. < http://hdl.handle.net/2077/29163 >. Consulted on July 2012.

Eriksson, H., and M. Penker. 2000. *Business modeling with UML: business patterns at work*. New York: John Wiley & Sons, 480 p.

Gehlert, Andreas, and Werner Esswein. 2007. « Toward a formal research framework for ontological analyses ». *Advanced Engineering Informatics,* vol. 21, n^{o} 2, p. 119-131.

GELOG-ETS. 2008. *Software Functional Size with ISO 19761:2003 COSMIC-FFP Measurement Method, Proposed Measurement Etalon: C-Registration System*. Case Study. Canada, 47 p. < http://www.cosmicon.com/portal/public/CRS_RUP_Case_%20Study_version_Jan_04_2007_web_%20version_update_feb_2008.pdf >. Consulted on September 2010.

Georgakopoulos, D., M. Hornick and A. Sheth. 1995. « An overview of workflow management: From process modeling to workflow automation infrastructure ». *Distributed and parallel Databases,* vol. 3, n^{o} 2, p. 119-153.

Giaglis, G. M. 2001. « A Taxonomy of Business Process Modeling and Information Systems Modeling Techniques ». *International Journal of Flexible Manufacturing Systems,* vol. 13, n^{o} 2, p. 209-228.

Goedertier, Stijn, and Jan Vanthienen. 2007. « Declarative process modeling with business vocabulary and business rules ». In *On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS*, ed. Meersman, Robert, Zahir Tari and Pilar Herrero. Vol. 4803, p. 603-612. Coll. « Lecture Notes in Computer Science ». Berlin, Heidelberg: Springer-Verlag. < http://dx.doi.org/10.1007/978-3-540-76888-3_83 >.

Gorry, G.A., and M.S. Scott Morton. 1989. « A framework for management information systems revisited ». *Sloan Management Review,* vol. 30 n^{o} 3, p. 49-61.

Green, P., and M. Rosemann. 2000. « Integrated process modeling: an ontological evaluation ». *Information Systems,* vol. 25, n^{o} 2, p. 73-87.

Green, P., M. Rosemann, M. Indulska and JC Recker. 2006. « Improving representational analysis: an example from the enterprise systems interoperability domain ». In *17th Australasian Conference on Information Systems (ACIS 2006)*. (Adelaide, Australia, December 6-8), ed. Spencer, Su, and Adam Jenkins. Vol. 17, p. 2-11. Australasian Association for Information Systems.

Green, P., M. Rosemann and A. Ipswich. 2001. « Ontological analysis of integrated process models: testing hypotheses ». *Australasian Journal of Information Systems (AJIS),* vol. 9, n° 1, p. 30-38.

Green, P.F., and M. Rosemann. 2002. « Perceived ontological weaknesses of process modeling techniques: Further evidence ». In *10th European Conference on Information Systems: Information Systems and the Future of the Digital Economy, ECIS 2002*. (Gdansk, Poland, 6-8 June), ed. Wrycza, Stanislaw, p. 312-320.

Grüninger, Michael, Katy Atefi and Mark S. Fox. 2000. « Ontologies to Support Process Integration in Enterprise Engineering ». *Computational & Mathematical Organization Theory,* vol. 6, n° 4, p. 381-394.

Guizzardi, Giancarlo, Heinrich Herre and Gerd Wagner. 2002. « On the General Ontological Foundations of Conceptual Modeling ». In *21th International Conference on Conceptual Modeling, ER 2002*. (Tampere, Finland, October 7–11), ed. Spaccapietra, S., March, S. T., Kambayashi, Y., p. 65-78. London, UK, UK: Springer-Verlag.

Gulla, Jon Atle, and Terje Brasethvik. 2000. « On the Challenges of Business Modeling in Large-Scale Reengineering Projects ». In *4th International Conference on Requirements Engineering, ICRE'00*. (Schaumburg, IL, June 19-23), p. 17-26. IEEE Computer Society.

Gulledge, TR, and RA Sommer. 2002. « Business process management: public sector implications ». *Business Process Management Journal,* vol. 8, n° 4, p. 364-376.

Hammer, M., and J. Champy. 1993. *Reengineering the corporation: a manifesto for business revolution*. New York: Harper Business, 223 p.

Haque, B., K. S. Pawar and R. J. Barson. 2003. « The application of business process modelling to organisational analysis of concurrent engineering environments ». *Technovation,* vol. 23, n° 2, p. 147 - 162.

Harmon, P., and C. Wolf. 2010. *The state of business process management 2010*. Coll. « A BPTrends Report ». USA: Business Process Trends, 52 p. < http://www.bptrends.com/surveys_landing.cfm >. Consulted on May 2010.

Harmon, Paul, and Celia Wolf. 2011. *Business Process Modeling Survey*. Coll. « BPTrends Report ». USA: Business Process Trends, 36 p. < http://www.bptrends.com/surveys_landing.cfm >. Consulted on February 2012.

Hepp, M., and D. Roman. 2007. « An Ontology Framework for Semantic Business Process Management ». In *Wirtschaftsinformatik 2007*. (Karlsruhe, Germany, February 28 - March 2), ed. Oberweis, Andreas, Christof Weinhardt, Henner Gimpel, Agnes Koschmider, Victor Pankratius and Björn Schnizler, p. 423-440.

Hommes, BJ, and V. Van Reijswoud. 2000. « Assessing the quality of business process modelling techniques ». In *33rd Hawaii International Conference on System Sciences (HICSS 2000)*. (Maui, Hawaii, January 4-7) Vol. 1, p. 1007. IEEE Computer Society.

Hoyle, D. 2006. *ISO 9000 quality systems handbook*, 5th Edition. Oxford, U.K.; Burlington, Mass.: Butterworth-Heinemann/Elsevier, 686 p.

IIBA. 2009. *A Guide to the Business Analysis Body of Knowledge (BABOK Guide)*, Version 2.0. Toronto: International Institute of Business Analysis (IIBA), 272 p.

Indulska, M., P. Green, J. Recker and M. Rosemann. 2009. « Business Process Modeling: Perceived Benefits ». In *28th International Conference on Conceptual Modeling - ER 2009*. (Gramado, Brazil, November 9-12), p. 458-471. Berlin, Heidelberg: Springer-Verlag.

Indulska, M., M. zur Muehlen and J. Recker. 2009. *Measuring Method Complexity: The Case of the Business Process Modeling Notation*. Coll. « BPM Center Report », BPM-09-03: BPMcenter.org. < http://bpmcenter.org/wp-content/uploads/reports/2009/BPM-09-03.pdf >. Consulted on August 15, 2012.

ISO. 2008. *ISO 9000 Introduction and Support Package: Guidance on the Concept and Use of the Process Approach for Management Systems*. Coll. « Introduction and Support Package », ISO/TC 176/SC 2/N 544R3. London: International Standardization Organization, 12 p. Consulted on August 15, 2012.

ISO. 2010. *ISO 9000 Introduction and Support Package: Guidance on the Documentation Requirements of ISO 9001:2008*. Coll. « Introduction and Support Package », ISO/TC 176/SC 2/N 525R2. London: International Standardization Organization, 7 p. Consulted on August 15, 2012.

ISO/IEC. 2006. *Information technology — Software measurement — Functional size measurement — Part 6: Guide for use of ISO/IEC 14143 series and related International Standards*. ISO/IEC 14143-6:2006. Switzerland: International Organization for Standardization, 21 p.

ISO/IEC. 2007. *Information Technology - Software Measurement - Functional Size Measurement. Part 1: Definition of Concepts*. ISO/IEC 14143-1:2007. Switzerland: International Organization for Standardization, 6 p.

ISO/IEC. 2010. *Information technology — Systems and software engineering — FiSMA 1.1 functional size measurement method*. ISO/IEC 29881:2010. Switzerland: International Organization for Standardization, 16 p.

ISO/IEC. 2011. *Software engineering -- COSMIC: a functional size measurement method*. ISO/IEC 19761:2011. Switzerland: International Organization for Standardization, 14 p.

Kaplan, Robert S., and D. P. Norton. 2007. « Using the Balanced Scorecard as a Strategic Management System ». *Harvard Business Review,* vol. 85, n° 7/8, p. 150-161.

Kaschek, Roland, Roman Pavlov, VladimirA Shekhovtsov and Sergiy Zlatkin. 2007. « Characterization and Tool Supported Selection of Business Process Modeling Methodologies ». In *Technologies for Business Information Systems*, ed. Abramowicz, Witold, and Heinrich Mayr. p. 25-37. Netherlands: Springer. < http://dx.doi.org/10.1007/1-4020-5634-6_3 >.

Kaya, Mahir. 2010. « E-cosmic: a business process model based functional size estimation approach ». MSc Thesis. Ankara, Middle East Technical Universtiy. Consulted on August 15, 2012.

Kazhamiakin, R., M. Pistore and M. Roveri. 2004. « A framework for integrating business processes and business requirements ». In *Eighth IEEE International Enterprise Distributed Object Computing Conference EDOC 2004*. (Monterey, California, USA, September 20-24), p. 9-20. IEEE Computer Society.

Kettinger, WJ, JTC Teng and S. Guha. 1997. « Business process change: a study of methodologies, techniques, and tools ». *Management Information Systems Quarterly,* vol. 21, p. 55-80.

Kitchenham, Barbara A., and Shari L. Pfleeger. 2008. « Personal opinion surveys ». In *Guide to Advanced Empirical Software Engineering*, ed. Shull, Forrest, JaniceSinger and Dag I. K. Sjoberg. p. 63-92. Coll. « Computer Science ». London: Springer London. < http://dx.doi.org/10.1007/978-1-84800-044-5_3 >.

Ko, Ryan K. L. 2009. « A computer scientist's introductory guide to business process management (BPM) ». *Crossroads,* vol. 15, n° 4, p. 11-18.

Ko, Ryan K. L., Stephen S. G. Lee and Eng Wah Lee. 2009. « Business Process Management (BPM) standards: A survey ». *Business Process Management Journal,* vol. 15, n° 5, p. 744 - 791.

Krogstie, J. 2003. « Evaluating UML using a generic quality framework ». In *UML and the unified process*, ed. Favre, Liliana. p. 1-22. Hershey, PA, USA: IRM Press (an imprint of Idea Group Inc.).

Kueng, P., and P. Kawalek. 1997. « Goal-based business process models: Creation and evaluation ». *Business Process Management Journal,* vol. 3, n° 1, p. 17-38.

Lankhorst, M. 2005. *Enterprise Architecture at Work: Modelling, Communication, and Analysis*. Berlin, Heidelberg: Springer-Verlag, 334 p.

Lapouchnian, A., Y. Yu and J. Mylopoulos. 2007. « Requirements-Driven Design and Configuration Management of Business Processes ». In *Business Process Management*. p. 246-261.

Lavazza, Luigi, and Vieri Bianco. 2009. « A Case Study in COSMIC Functional Size Measurement: The Rice Cooker Revisited ». In *IWSM '09 /Mensura '09: International Conferences on Software Process and Product Measurement*. (Amsterdam, The Netherlands), p. 101-121. Berlin, Heidelberg: Springer-Verlag.

Lin, Fu-Ren, Meng-Chyn Yang and Yu-Hua Pai. 2002. « A generic structure for business process modeling ». *Business Process Management Journal,* vol. 8, n° 1, p. 19-41.

Lind, M., and U. Seigerroth. 2010. « Multi-Layered Process Modeling for Business and IT Alignment ». In *43rd Hawaii International Conference on System Sciences (HICSS)*. (Poipu, Kauai, Hawaii, 5-8 January), p. 1-10. IEEE Computer Society.

Lindsay, Ann, Denise Downs and Ken Lunn. 2003. « Business processes-attempts to find a definition ». *Information and Software Technology,* vol. 45, n° 15, p. 1015-1019.

List, B., and B. Korherr. 2005. « A UML 2 Profile for Business Process Modelling ». In *Perspectives in Conceptual Modeling: ER 2005 Workshops AOIS, BP-UML, CoMoGIS, eCOMO, and QoIS*, ed. Akoka, Jacky *et al.* Vol. 3770/2005, p. 85-96. Coll. « Lecture Notes in Computer Science ». Germany: Springer Berlin / Heidelberg.

List, B., and B. Korherr. 2006. « An evaluation of conceptual business process modelling languages ». In *2006 ACM symposium on Applied computing*. (Dijon, France, April 23-27), p. 1532-1539. ACM New York.

Luo, W., and YA Tung. 1999. « A framework for selecting business process modeling methods ». *Industrial Management and Data Systems,* vol. 99, n° 7, p. 312-19.

Marín, Beatriz, Giovanni Giachetti and Oscar Pastor. 2008. « Measurement of Functional Size in Conceptual Models: A Survey of Measurement Procedures Based on COSMIC ». In *IWSM/Metrikon/Mensura '08: International Conferences on Software Process and Product Measurement*. (Munich, Germany, November 18-19), ed. Dumke, Reiner R. *et al.*, p. 170-183. Berlin, Heidelberg: Springer-Verlag.

Mayr, Heinrich C., Christian Kop and Daniela Esberger. 2007. « Business Process Modeling and Requirements Modeling ». In *First International Conference on the Digital Society, ICDS '07*. (Guadeloupe, 2-6 Jan. 2007), p. 8-8. Washington, DC, USA: IEEE Computer Society.

McDonald, Mark, and Dave Aron. 2011. *Reimagining IT: The 2011 CIO Agenda*. Coll. « Gartner Executive Programs »: Gartner, 69 p. < http://www.gartner.com/technology/cio/index.jsp >. Consulted on June 2011.

Medina-Mora, R., T. Winograd, R. Flores and F. Flores. 1992. « The ActionWorkflow approach to workflow management technology ». In *1992 ACM Conference on Computer-Supported Cooperative Work CSCW '92: Sharing Perspectives*. (Toronto, Ontario, Canada, November 1-4), p. 281- 288. ACM, New York.

Mendling, J., H. Reijers and J. Cardoso. 2007. « What Makes Process Models Understandable? ». In *Business Process Management: 5th International Conference, BPM 2007*, ed. Alonso, Gustavo, Peter Dadam and Michael Rosemann. Vol. 4714/2007, p. 48-63. Coll. « Lecture Notes in Computer Science ». Berlin, Heidelberg: Springer-Verlag.

Mendling, J., and M. Strembeck. 2008. « Influence Factors of Understanding Business Process Models ». In *11th International Conference on Business Information Systems (BIS 2008)*. (Innsbruck, Austria, May 5-7), ed. Abramowicz, Witold, and Dieter Fensel Vol. 7, p. 142-153. Coll. « Lecture Notes in Business Information Processing »: Springer-Verlag.

Milanovic, M., D. Gasevic and L. Rocha. 2011. « Modeling Flexible Business Processes with Business Rule Patterns ». In *The 15th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2011)*. (Helsinki, Finland, Aug. 29 - Sept. 2), p. 65-74. IEEE Computer Society.

Mili, H., A. Leshob, E. Lefebvre, G. Lévesque and G. El-Boussaidi. 2009. « Towards a Methodology for Representing and Classifying Business Processes ». In *E-Technologies: Innovation in an Open World, 4th International Conference, MCETECH 2009*, ed. Wil van der Aalst, *et al.* Vol. 26, p. 196-211. Coll. « Lecture Notes in Business Information Processing ». Berlin, Germany: Springer Berlin Heidelberg.

Mili, Hafedh, Guy Tremblay, Guitta Bou Jaoude, Éric Lefebvre, Lamia Elabed and Ghizlane El Boussaidi. 2010. « Business process modeling languages: Sorting through the alphabet soup ». *ACM Computing Surveys,* vol. 43, nº 1, p. 1-56.

Miller, James. 2008. « Triangulation as a basis for knowledge discovery in software engineering ». *Empirical Software Engineering,* vol. 13, nº 2, p. 223-228.

Milton, S.K. 2007. « Ontological Foundations of Representational Information Systems ». *Scandinavian Journal of Information Systems,* vol. 19, n° 1, p. 109-134.

Monsalve, C, A April and A Abran. 2010. « Representing unique stakeholder perspectives in BPM notations ». In *8th ACIS International Conference on Software Engineering Research, Management and Applications, SERA 2010*. (Montreal, May 24-26), p. 42-49. Washington, DC, USA: IEEE Computer Society.

Monsalve, C, A April and A Abran. 2011a. « BPM and requirements elicitation at multiple levels of abstraction: a review ». In *IADIS International Conference Information Systems 2011*. (Avila, Spain, March 11-13), p. 237-242. International Association for Development of the Information Society.

Monsalve, C, A April and A Abran. 2011b. « Requirements elicitation using BPM notations: focusing on the strategic level representation ». In *Recent Researches in Applied Computer and Applied Computational Science: 10th WSEAS International Conference on Applied Computer and Applied Computational Science (ACACOS '11)*. (Venice, Italy, March 8-10), ed. S. Chen, *et al.*, p. 235-241. WSEAS Press. < http://www.wseas.us/conferences/2011/venice/acacos/ >.

Monsalve, C, A April and A Abran. In press. « On the Expresiveness of Business Process Modeling Notations for Software Requirements Elicitation ». In *38th Annual Conference of the IEEE Industrial Electronics Society (IECON 2012)*. (Montreal, October 25-28). Institute of Electrical and Electronics Engineers.

Monsalve, Carlos, Alain Abran and Alain April. 2010. « Functional Size Measurement with Business Process Models: the Business Application Domain ». In *Joined International Conferences on Software Measurement IWSM/MetriKon/Mensura 2010*. (Stuttgart, Germany, November 10-12), ed. Alain Abran, Günter Büren, Reiner R. Dumke, Juan J. Cuadrado-Gallego, Jürgen Münch, p. 270-290. Aachen: Shaker Verlag.

Monsalve, Carlos, Alain Abran and Alain April. 2011. « Measuring Software Functional Size from Business Process Models ». *International Journal of Software Engineering and Knowledge Engineering (IJSEKE),* vol. 21, n° 3, p. 311-338.

Muehlen, Michael Zur, and Jan Recker. 2008. « How Much Language Is Enough? Theoretical and Practical Use of the Business Process Modeling Notation ». In *CAiSE '08: 20th international conference on Advanced Information Systems Engineering*. (Montpellier, France, June 18-20), p. 465-479. Springer-Verlag.

Nysetvold, A. G., and J. Krogstie. 2005. « Assessing Business Process Modeling Languages Using a Generic Quality Framework ». In *Workshop on Evaluating Modeling Methods for Systems Analysis and Design (EMMSAD'05)*. (Porto, Portugal, June 13-14), ed. Halpin, T.A., K. Siau and J. Krogstie, p. 545-556. FEUP, Porto, Portugal.

OMG. 2009a. *OMG Business Process Model and Notation (BPMN), version 1.2*. OMG formal/2009-01-03. Needham, MA, U.S.A.: Object Management Group, 294 p. < http://www.omg.org/spec/BPMN/1.2 >.

OMG. 2009b. « UML Resource Page ». In *Unified Modeling Language*. < http://www.uml.org/ >. Consulted on July 29 2012.

OMG. 2010. *OMG Unified Modeling Language (OMG UML), Superstructure, Version 2.3*. OMG formal/2010-05-05. Needham, MA, U.S.A. : Object Management Group, 742 p. < http://www.omg.org/spec/UML/2.3/Superstructure/ >.

OMG. 2011. *Business Process Model and Notation (BPMN) version 2.0*. OMG formal/2011-01-03. Needham, MA, U.S.A.: Object Management Group (OMG). < http://www.omg.org/spec/BPMN/2.0 >.

Opdahl, AndreasL, and Brian Henderson-Sellers. 2002. « Ontological Evaluation of the UML Using the Bunge–Wand–Weber Model ». *Software and Systems Modeling,* vol. 1, n° 1, p. 43-67.

Ouanouki, Rafik, and Alain April. 2008. « IT Process Conformance Measurement: A Sarbanes- Oxley Requirement ». In *Software Process and Product Measurement: International Conference, IWSM - Mensura 2007*, ed. Cuadrado-Gallego, Juan Jose, René Braungarten, Reiner R. Dumke and Alain Abran. Vol. 4895, p. 26 - 37. Coll. « Lecture Notes in Computer Science ». Palma de Mallorca, Spain: Springer.

Paré, Guy. 2002. « Enhancing the rigor of qualitative research: application of a case methodology to build theories of IT implementation ». *The Qualitative Report*. Vol. 7, n° 4, p. 1-13. < http://www.nova.edu/ssss/QR/QR7-4/pare.html >. Consulted on August 4, 2012.

Pavlovski, Christopher J., and Joe Zou. 2008. « Non-functional requirements in business process modeling ». In *Fifth Asia-Pacific Conference on Conceptual Modelling (APCCM 2008)*. (Wollongong, New South Wales, Australia, January 22-25). Vol. 79, p. 103-112. Australian Computer Society, Inc.

Pichler, M., and H. Rumetshofer. 2006. « Business Process-based Requirements Modeling and Management ». In *First International Workshop on Requirements Engineering Visualization, 2006. REV '06* (Minneapolis/St.Paul, Minnesota, USA, September 11), p. 6-6. IEEE Computer Society.

Porter, Michel E., and Victor E. Millar. 1985. « How Information Gives You Competitive Advantage ». *Harvard Business Review,* vol. 63, n° 4, p. 149-160.

Prezel, Vid, Dragan Gašević and Milan Milanović. 2010. « Representational Analysis of Business Process and Business Rule Languages ». In *1st International Workshop on Business Models, Business Rules and Ontologies (BuRO 2010)*. (Bressanone/Brixen, Italy, 21 September), ed. T. Eiter, A. El Ghali, S. Fernández, S. Heymans, T. Krennwallner, F. Lévy, p. 15-28.

Recker, J, and M Indulska. 2007. « An Ontology-Based Evaluation of Process Modeling with Petri Nets ». *International Journal of Interoperability in Business Information Systems (IBIS),* vol. 2, nº 1, p. 45-64.

Recker, J. 2010. « Continued use of process modeling grammars: the impact of individual difference factors ». *European Journal of Information Systems,* vol. 19, nº 1, p. 76-92.

Recker, J., M. Indulska, M. Rosemann and P. Green. 2005. « Do process modelling techniques get better? A comparative ontological analysis of BPMN ». In *16th Australasian Conference on Information Systems*. (Sydney, Australia, November 30 - December 2). Australasian Chapter of the Association for Information Systems <http://aisel.aisnet.org/acis2005/36 >.

Recker, J., M. Indulska, M. Rosemann and P. Green. 2006. « How good is BPMN really? Insights from theory and practice ». In *14th European Conference on Information Systems: ECIS' 06*. (Göteborg, Sweden, June 12-14), ed. Ljungberg J, Andersson M, p. 1-12. < http://aisel.aisnet.org/ecis2006/135 >.

Recker, J., M. Rosemann, PF Green and MK Indulska. 2007. « Extending the Scope of Representation Theory: A Review and a Proposed Research Model ». In *3rd Biennial ANU Workshop on Information Systems Foundations*. (Camberra, Australia, September 27-28), ed. Hart, Dennis, and Shirley Gregor, p. 93-114. ANU E-Press.

Recker, J., M. Rosemann, M. Indulska and P. Green. 2009. « Business process modeling: a comparative analysis ». *Journal of the Association for Information Systems,* vol. 10, nº 4, p. 333-363.

Recker, Jan. 2006. « Process Modeling in the 21st Century ». USA: Business Process Trends, 1-8 p. < http://www.bptrends.com/publicationfiles/05-06-ART-ProcessModeling21stCent-Recker1.pdf >. Consulted on February 2009.

Recker, JC. 2008. « BPMN Modeling–Who, Where, How and Why ». Business Process Trends, 1-8 p. < http://www.bptrends.com/publicationfiles/05-08-ART-BPMN%20Survey-Recker-JR%20final.pdf >. Consulted on August 15, 2012.

Rosemann, M., and P. Green. 2002. « Developing a meta model for the Bunge–Wand–Weber ontological constructs ». *Information Systems,* vol. 27, nº 2, p. 75-91.

Rosemann, Michael, and Peter Green. 2000. « Integrating multi-perspective views into ontological analysis ». In *21st International Conference on Information systems, ICIS'00*. (Brisbane, Queensland, Australia, December 10-13), p. 618-627. Association for Information Systems.

Rosemann, Michael, Peter Green, Marta Indulska and Jan C. Recker. 2009. « Using ontology for the representational analysis of process modelling techniques ». *International Journal of Business Process Integration and Management,* vol. 4, no 4, p. 251-265.

Runeson, Per, and Martin Höster. 2009. « Guidelines for conducting and reporting case study research in software engineering ». *Empirical Software Engineering,* vol. 14, no 2, p. 131-164.

Russell, N., AHM Ter Hofstede, D. Edmond and WMP van der Aalst. 2004. *Workflow data patterns*. Coll. « QUT Technical Report », FIT-TR-2004-01. Brisbane: Queensland University of Technology, 75 p. < http://www.workflowpatterns.com/patterns/data/ >. Consulted on August 15, 2012.

Russell, N., AHM Ter Hofstede, WMP van der Aalst and N. Mulyar. 2006a. *Workflow control-flow patterns: A revised view*. Coll. « BPM Center Report », BPM-06-22: BPMcenter.org, 134 p. < http://bpmcenter.org/wp-content/uploads/reports/2006/BPM-06-22.pdf >. Consulted on August 15, 2012.

Russell, N., WMP van der Aalst, AHM Ter Hofstede and P. Wohed. 2006b. « On the suitability of UML 2.0 activity diagrams for business process modelling ». In *3rd Asia-Pacific Conference on Conceptual Modelling APCCM '06*. (Hobart, Australia, January 16-19). Vol. 53, p. 95-104. Darlinghurst, Australia: Australian Computer Society, Inc.

Salant, P., and D.A. Dillman. 1994. *How to conduct your own survey*. New York: John Wiley & Sons Inc, 232 p.

SAP, AG. 2008. « SAP and IDS Scheer partner to bring customers easier modeling and visualization of enterprise services ». In *SAP*. < http://www.sap.com/canada/about/press/press.epx?pressid=9129 >. Consulted on August 12 2012.

Sarshar, K., and P. Loos. 2005. « Comparing the Control-Flow of EPC and Petri Net from the End-User Perspective ». In *Business Process Management: 3rd International Conference, BPM 2005*, ed. Aalst, Wil M.P. van der, Boualem Benatallah, Fabio Casati and Francisco Curbera. Vol. 3649/2005, p. 434-439. Coll. « Lecture Notes in Computer Science ». Germany: Springer Berlin / Heidelberg.

SCC. 2008. *Supply-Chain Operations Reference-model. SCOR Overview*. Supply Chain Council. < http://www.supply-chain.org >. Consulted on August 15, 2012.

Scheer, August-Wilhelm, and Kristof Schneider. 2006. « ARIS — Architecture of Integrated Information Systems ». In *Handbook on Architectures of Information Systems*, ed. Bernus, Peter, Kai Mertins and Günter Schmidt. p. 605-623. Coll. « International Handbooks on Information Systems ». Germany: Springer Berlin Heidelberg. < http://dx.doi.org/10.1007/3-540-26661-5_25 >.

Scheer, AW, O. Thomas and O. Adam. 2005. « Process Modeling Using Event-driven Process Chains ». In *Process-aware Information Systems: Bridging People and Software through Process Technology*, ed. Dumas, Marlon, Wil M. P. van der Aalst and Arthur H. M. ter Hofstede. p. 119-145. Hoboken, New Jersey: John Wiley & Sons, Inc.

Scheer, IDS. 2009. « ARIS Business Performance Edition ». < http://www.ids-scheer.com/en/ARIS_ARIS_Business_Performance_Edition/89359.html >. Consulted on October 2010.

Scheer, IDS. 2012. « ARIS Platform ». In *Software AG*. < http://www.softwareag.com/corporate/products/aris_platform/default.asp >. Consulted on August 12 2012.

Schlauderer, Sebastian, and Sven Overhage. 2011. « How perfect are markets for software services? an economic perspective on market deficiencies and desirable market features ». In *19th European Conference on Information Systems, ECIS 2011*. (Helsinki, Finland, June 9-11), ed. Tuunainen, Virpi Kristiina, Matti Rossi and Joe Nandhakumar.

Sedera, W., G. Gable, M. Rosemann and R. Smyth. 2004. « A success model for business process modeling: findings from a multiple case study ». In *Eighth Pacific Asia Conference on Information Systems, PACIS 2004*. (Shanghai, China, July 8-11). < http://aisel.aisnet.org/pacis2004/38/ >.

Sellami, Asma, and Hanene Ben-Abdallah. 2009. « Functional Size of Use Case Diagrams: A Fine-Grain Measurement ». In *ICSEA '09: 2009 Fourth International Conference on Software Engineering Advances*. p. 282-288. Washington, DC, USA: IEEE Computer Society.

Shang, Shari, and Peter B. Seddon. 2002. « Assessing and managing the benefits of enterprise systems: the business manager's perspective ». *Information Systems Journal,* vol. 12, n° 4, p. 271-299.

Sharp, A., and P. McDermott. 2009. *Workflow Modeling: Tools for Process Improvement and Applications Development*, second edition. Norwood, MA, USA: Artech House, 449 p.

Silver, Bruce. 2009. *BPMN method and style*. Aptos, California, USA: Cody-Cassidy Press, 213 p.

Simon, C. 2005. « Incremental development of business process models ». In *Workshop Enterprise Modelling and Information Systems Architectures*. (Klagenfurt, Austria, October 2005), ed. Desel, Jörg, and Ullrich Frank Vol. 75, p. 222-235. Coll. « Lecture Notes in Informatics »: German Informatics Society.

Sjoberg, DIK, T Dyba and M Jorgensen. 2007. « The future of empirical methods in software engineering research ». In *Future of Software Engineering, 2007. FOSE '07*. (Minneapolis, USA, May 23-25), p. 358-378. IEEE Computer Society.

Smart, P. A. , H. Maddern and R. S. Maull. 2009. « Understanding Business Process Management: Implications for Theory and Practice ». *British Journal of Management,* vol. 20, n^{o} 4, p. 491-507.

Smith, H., and P. Fingar. 2007. *Business process management: the third wave*, 4th Anniversary Edition. Tampa, Fla., USA: Meghan-Kiffer Press, 292 p.

Spanyi, A. 2003. *Business process management is a team sport: play it to win!* , 1st Edition. Tampa, FL: Anclote Press, 173 p.

SPARKX. 2009. « UML tools for software development and modelling.- Enterprise Architect ». < http://www.sparxsystems.com.au/ >. Consulted on December 2009.

StarUML. « StarUML - The Open Source UML/MDA Platform ». < http://staruml.sourceforge.net/en/ >. Consulted on December 2009.

Susman, G.I., and R.D. Evered. 1978. « An Assessment of the Scientific Merits of Action Research ». *Administrative Science Quarterly,* vol. 23, n^{o} 4, p. 582-603.

van den Berg, K. G., T. Dekkers and R. Oudshoorn. 2005. « Functional size measurement applied to UML-based user requirements ». In *2nd Software Measurement European Forum (SMEF 2005)*. (Rome, Italy, March 16-18), p. 69-80.

van der Aalst, WMP, AHM ter Hofstede and M. Dumas. 2005. « Patterns of Process Modeling ». In *Process-Aware Information Systems: Bridging People and Software through Process Technology*, ed. Dumas, Marlon, Wil M. van der Aalst and Arthur H. ter Hofstede. p. 179-203. Hoboken, NJ, USA: John Wiley & Sons, Inc.

Van Nuffel, Dieter, and Manu De Backer. 2012. « Multi-abstraction layered business process modeling ». *Computers in Industry,* vol. 63, n^{o} 2, p. 131-147.

Vara, Jose Luis, Juan Sánchez and Óscar Pastor. 2008. « Business Process Modelling and Purpose Analysis for Requirements Analysis of Information Systems ». In *Advanced*

Information Systems Engineering: 20th international conference, CAiSE 2008. Vol. 5074/2008, p. 213-227. Coll. « Lecture Notes in Computer Science ». Berlin, Heidelberg: Springer-Verlag.

Vasconcelos, André, Artur Caetano, João Neves, Pedro Sinogas, Ricardo Mendes and José Tribolet. 2001. « A Framework for Modeling Strategy, Business Processes and Information Systems ». In *5th IEEE International Conference on Enterprise Distributed Object Computing. EDOC '01*. (Seattle, WA, USA, September 4-7), p. 69-80. IEEE Computer Society.

Wand, Y., and R. Weber. 1990. « An ontological model of an information system ». *IEEE Transactions on Software Engineering,* vol. 16, n° 11, p. 1282-1292.

Wand, Y., and R. Weber. 1993. « On the ontological expressiveness of information systems analysis and design grammars ». *Information Systems Journal,* vol. 3, n° 4, p. 217-237.

Wand, Y., and R. Weber. 1995. « On the deep structure of information systems ». *Information Systems Journal,* vol. 5, n° 3, p. 203-223.

Wand, Y., and R. Weber. 2002. « Research commentary: information systems and conceptual modeling -- a research agenda ». *Information Systems Research,* vol. 13, n° 4, p. 363-376.

Weber, Ron. 1997. *Ontological foundations of information systems*. Coll. « Coopers & Lybrand accounting research methodology monograph », no. 4. Melbourne, Australia: Coopers & Lybrand and the Accounting Association of Australia and New Zealand, 212 p.

White, SA. 2004. « Introduction to BPMN ». Business Process Trends, 11 p. < http://www.bptrends.com/publicationfiles/07-04%20WP%20Intro%20to%20BPMN%20-%20White.pdf >. Consulted on August 15, 2012.

Wohed, P., WMP van der Aalst, M. Dumas, AHM Hofstede and N. Russell. 2006. « On the suitability of BPMN for business process modelling ». In *Business Process Management: 4th International Conference, BPM 2006*, ed. Dustdar, Schahram, José Luiz Fiadeiro and Amit P. Sheth. Vol. 4102/2006, p. 161-176. Coll. « Lecture Notes in Computer Science ». Germany: Springer Berlin / Heidelberg.

Yin, Robert. 2002. *Case Study Research: Design and Methods*, Vol. 5, Third Edition. Coll. « Applied Social Research Methods Series ». USA: Sage Publications, Inc.

Yu, E. S. K., and J. Mylopoulos. 1994a. « Using goals, rules, and methods to support reasoning in business process reengineering ». In *Twenty-Seventh Hawaii*

International Conference on System Sciences, Information Systems: Collaboration Technology Organizational Systems and Technology. (Wailea, HI, USA, 4-7 January) Vol. 4, p. 234-243.

Yu, Eric S. K., and John Mylopoulos. 1994b. « From E-R to "A-R" - Modelling Strategic Actor Relationships for Business Process Reengineering ». In *13th International Conference on the Entity-Relationship Approach - ER '94: Business Modeling and Re-Engineering*. (Manchester, United Kingdom, December 13–16), ed. Loucopoulos, Pericles. Vol. 881/1994, p. 548-565. London, UK: Springer-Verlag.

Zairi, M, and D Sinclair. 1995a. « Business process re-engineering and process management. A survey of current practice and future trends in integrated management. ». *Business process management journal (Online),* vol. 1, nº 1, p. 8-30.

Zairi, M. 1997. « Business process management: a boundaryless approach to modern competitiveness ». *Business Process Management Journal,* vol. 3, nº 1, p. 64-80.

Zairi, M., and D. Sinclair. 1995b. « Business process re-engineering and process management: A survey of current practice and future trends in integrated management. ». *Business Process Re-engineering & Management Journal,* vol. 1, nº 1, p. 8-30.

zur Muehlen, M, and M Rosemann. 1998. « Evaluation of Workflow Management Systems-a Meta Model Approach ». *Australian Journal of Information Systems,* vol. 6, nº 1, p. 103-116.

zur Muehlen, M. 2004. *Workflow-based Process Controlling: Foundation, Design, and Application of Workflow-driven Process Information Systems*. Berlin: Logos Verlag, 300 p.

zur Muehlen, Michael, and Danny T. Ho. 2008. « Service Process Innovation: A Case Study of BPMN in Practice ». In *41st Annual Hawaii International Conference on System Sciences*. (Waikoloa, Big Island, Hawaii, January 7-10), p. 372. IEEE Computer Society.

zur Muehlen, Michael, and Marta Indulska. 2010. « Modeling languages for business processes and business rules: A representational analysis ». *Information Systems,* vol. 35, nº 4, p. 379-390.