UNIVERSITÀ DEGLI STUDI
DI TRENTO

DEPARTMENT OF INFORMATION ENGINEERING AND COMPUTER SCIENCE
ICT International Doctoral School

Strategic Reasoning for Enterprise Architectures: The SIENA Modeling Framework

Evellin Cristine Souza Cardoso

Advisor

Prof. John Mylopoulos

Università degli Studi di Trento

Co-Advisor

Assistant Prof. Jennifer Horkoff

Chalmers and the University of Gothenburg

January 2018

Abstract

This thesis contributes to the area of Enterprise Modeling by proposing the SIENA modeling framework for the representation of strategic enterprise architectures and automated reasoning with such models. In this work, we provide the SIENA language that provides abstractions for capturing enterprise's motivational elements (i.e. goals of different shades like mission, vision, strategic, tactical and operational goals) and their connections with behavioral elements (i.e., operations, business processes, commitments and activities) through which they are operationalized. The SIENA language also introduces the distinguishing feature of dimensional refinement operators, a new operator that can be used for the refinement of strategic goals in terms of time, location and products/services dimensions. SIENA language is also accompanied by modeling guidelines for the construction of its models. Besides the SIENA language, we also propose a business process language called Azzurra which is founded on the primitives of commitments and protocols for the representation of business processes. The representation of business processes in terms of commitments is a distinguishing feature of our approach. Further, our framework also supports the design of business processes specified using the Azzurra language from SIENA operational goals. As one of the greatest advantages of conducting enterprise modeling is to gain the ability to perform automated analysis using enterprise models, we also propose a formal reasoning technique for the automated generation of strategic plans subject to constraints to satisfy *enterprise's strategic goals. The overall approach is validated by means of a number of different activities, including self-evaluation, experimentation and in-depth case studies with novices.*

Keywords

Conceptual Modeling, Enterprise Architecture, Business Process Management, Requirements, Automated Reasoning

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The management of organizations involves a high level of complexity since it aggregates several knowledge domains. Each of these domains may be influenced by potentially conflicting quality factors which affect the organization's overall performance. In order to allow the balancing and prioritization of these factors, using an enterprise architecture becomes indispensable.

Enterprise architectures have been initially proposed in 1987 within the Zachman Framework [213] as an instrument for comprehensively describing the key elements and relationships of an enterprise. Since then, the holistic nature of enterprise architectures led them to become a widespread asset for supporting the management of the complexity of organizations. Nowadays, their great importance has been acknowledged in the industry with the adoption of the ArchiMate language [78] and TOFAG framework [77] as complementary enterprise modeling standards from the Open Group Standardization Consortium. In academia, this importance is acknowledged by the creation of a number of venues (e.g. Business Process Management (BPM)[1], Practice of Enterprise Modeling (PoEM)[2]) where research in enterprise modeling can be presented.

[1] `https://bpm-conference.org/BpmConference/`

[2] `https://kuleuvencongres.be/poem2017`

Among the different types of enterprise architectures, *strategic enterprise architectures* receive special attention both in academia and industry due to the fundamental importance of goals and requirements within the architecture development process [160]. In this context, strategic enterprise architectures are characterized by the explicit incorporation of goals as requirements imposed by business on the enterprise architecture. Such explicit representation enables the understanding, structuring and analysis of how business requirements are realized by the overall enterprise architecture (i.e., organizational structure, business processes, software systems, technical infrastructure and data aspects).

This thesis contributes to the area of Enterprise Modeling by proposing the SIENA modeling framework for the representation of strategic enterprise architectures and automated reasoning with such models. In this work, we provide the SIENA language that provides abstractions for capturing enterprise's motivational elements (i.e. goals of different shades like mission, vision, strategic, tactical and operational goals) and their connections with behavioral elements (i.e., operations, business processes, commitments and activities) through which they are operationalized. The SIENA language also introduces the distinguishing feature of *dimensional refinement operators*, a new operator that can be used for the refinement of strategic goals in terms of time, location and products/services dimensions. In comparison with traditional OR-refinements that capture alternatives for achieving goals, dimensional refinement operators enable the specification of different alternatives to achieve strategic goals in different points of a given dimension. Further, SIENA language is accompanied by modeling guidelines for the construction of its models. Besides the SIENA language, we also propose a business process language called Azzurra which is founded on the primitives of commitments and protocols for the representation of business processes. The representation of business processes in terms of *commitments* is a distinguishing feature of our approach. Azzurra also includes business primitives such as delegations, deadlines, constraints over roles and the notion of initiation and termination of a protocol (business process). The language introduces a graphical notation for modeling the main elements of a business process and supports the construction of business process models with a prototype Eclipse-based modeling tool. Further, our framework also supports the design of business processes specified using the Azzurra language from SIENA operational goals.

As one of the greatest advantages of conducting enterprise modeling is to gain the ability to perform automated analysis using enterprise models, we also propose a formal reasoning technique for the automated generation of strategic plans subject to constraints to satisfy enterprise's strategic goals. The overall approach is validated by means of a number of different activities, including self-evaluation, experimentation and in-depth case studies with novices.

Much of this work is based on earlier approaches in goal and business process modeling in Goal-Oriented Requirements Engineering (GORE), Enterprise Modeling (EM) and Business Process Modeling (BPM). Previous experience and work in the alignment of goal models and business process models [24] has also served as an inspiration for the development of the strategic enterprise conceptual model. In relation to previous work, our approach advances the current state the art by providing a modeling framework that supports the definition of different shades of motivational concepts (strategic, tactical and operational goals), behavioral concepts (operations, business process and commitments) and refinements (dimensional refinement operators) that cannot be expressed in any other similar proposal to the best of our knowledge. The methodology for the specification of strategic enterprise architectures and the reasoning technique introduced in this work provides support for the enterprise planning process which is not supported by any approach to the best of our knowledge.

This chapter is organized as follows: Section 1.1 presents the concept of enterprise architectures as the background of this work. Section 1.2 introduces an example of metal manufacturing company that motivates the development of a strategic enterprise architecture approach to support strategic enterprise analysis. Section 1.3 states this problem by means of research questions, while Section 1.4 describes the support for representation and analysis of strategic enterprise architectures in current literature. Section 1.5 discusses the approach adopted in our research. Section 1.6 provides a general overview of the SIENA modeling framework, together with its main components and contributions and Section 1.7 presents the structure of this thesis and how each research question is tackled along the thesis. Finally, Section 1.8 shows the list of publications resulted from this research and how the author of this thesis contributed to each publication.

1.1 Enterprise Architecture Modeling

The increasing competitiveness drives organizations to constantly evaluate their position in the market and promote changes in an attempt to improve the quality of the services and products they offer. In recent years, companies started recognizing the benefits of adopting enterprise architectures as an important asset for the management of organizations [167, 113, 143, 175, 147]. In this context, a plethora of enterprise architecture frameworks have arisen, such as the Zachman Framework [213], TOGAF [77], ArchiMate [113], FEAF [32], DoDAF [146], TOVE [60] to support companies in such endeavor. Moreover, the value of an architectural approach for organizational governance has been also recognized by a number of studies [143, 147, 113, 167, 175] that corroborate the benefits of enterprise architectures, such as lowering IT costs, improving alignment of architecture with business strategy, improving change and asset management, among others [167]. In some cases, the adoption of enterprise architectures is not only one of the top priorities of senior management [175, p. 19], but it is even mandated by law (e.g., the Cohen - Clinger Act of 1996 in the United States) [113, p. 10] [212].

In this context, an *enterprise* is defined as a goal-oriented designed system that can be systematically adapted and/or re-engineered [113, 147]. In managing such complex system, enterprise architectures arise as the instrument to deal with enterprise complexity and govern constant business changes [212, 147]. An enterprise architecture consists of "a coherent whole of principles, methods and models that are used in the design and realization of an enterprise's organizational structure, business processes, information systems, and infrastructure" [113, p. 3].

Among the methods in enterprise architecture, techniques for describing architectures [113, 175] usually rely on conceptual modeling languages [113, 147, 212, 175] structured in terms of *architectural domains* or *viewpoints* [74]. While conceptual modeling languages enable the graphical representation of important dimensions of enterprises such as organizational structure, business process and applications, viewpoints are an effective mechanism for focusing on specific aspects of enterprises such as structural, functional and behavioral domains. Once represented, architectural models can be used as a decision-making instrument to perform integrated analysis across several enterprise architectural domains, thus revealing how local changes may affect other portions of the enterprise architecture and enabling one to make informed decisions considering such interrelations to achieve an integrated and well-balanced system.

Although the benefits of an architectural approach can clearly justify its adoption, enterprise architecture methods are still in their infancy [113, 147]. Although the ArchiMate [113] framework defines a standard modeling language for building enterprise descriptions, the diversity of enterprise modeling frameworks with different viewpoints and applicability leads to little consensus regarding the most appropriate enterprise modeling framework for a problem at hand [182, 175]. Consequently, practical effort requires guidance, such as those in [182, 175] to drive the selection of enterprise frameworks that focus on the most appropriate architectural domains, depending on their context.

Among all architectural perspectives, the perspective of "motivation" has been recognized as one of the most important elements of enterprise architectures [212], since it allows architects to systematically express the goals that govern the design of the enterprise as well as the motivations for adopting one particular enterprise configuration [208]. This is essential for business improvement once changes in a company's strategy have significant consequences within all domains of the enterprise [98]. While the goal domain of enterprise architectures focuses on "why" [208], the behavioral domain has also significant importance in enterprise architectures by expressing "how" the enterprise organizes work and resources to fulfill its strategies, focusing on its course-grained activities to jointly create a product or service.

This thesis focuses on the representation and analysis of the motivational and behavioral domains in enterprise architectures given the great importance of both domains. Although our main concerns are directed to these architectural domains, we do not totally exclude other domains, such as the organizational structure, resources and so forth, using them when required.

1.2 Strategic Enterprise Architectures Modeling

The development of a strategic enterprise architecture framework requires the development of a set of components that characterize the enterprise architecture. These components include the definition of enterprise modeling language which is structured in terms of motivational and behavioral perspectives, the specification of modeling guidelines to guide the development of models in such language and the selection of automated analysis and decision-making techniques to be performed with such enterprise models.

In order to motivate the characteristics of a given strategic enterprise language, its methodology and automated reasoning, we introduce the motivating example of the metal manufacturing company [34, p.222] from Management literature.

1.2.1 Motivating Example of Metal Manufacturing Company

The metal manufacturing company consists of an autonomous organization divided into different functional areas (finance, human resources, operations, marketing and research and development). Functional areas are divided into departments that deliver manufactured metal products to customers and departments are composed of different roles that execute company's work.

The success of the metal company requires managers to decide where the company should be in the future and to find a path to reach such desired future. In practice, company's management process is performed by exercising five overall functions (management functions) which consist of planning, organizing, staffing, leading and controlling [153, 51, 34]. Among the five management functions, the **planning function** is the first and one of the most crucial activities for the enterprise as it enables the company to plan for the future. In practice, it involves setting up organization's goals (motivational concepts) and allocating actions and resources (behavioral concepts) to achieve such goals, providing a sense of direction and unity of purpose for the organization and its sub-systems [153, 51]. Figure 1.1 depicts the basic steps of the enterprise planning process.

In the metal manufacturing company, the planning process (step 1) starts by managers setting up metal company's goals for all the levels of the organizational structure (i.e., the entire metal company, its functional areas and departments and roles). Such goal setting process is then reflected into a unified goal hierarchy with the division of organization's goals into mission, strategic, tactical and operational goals, each of them assigned to different levels of the organizational structure. Figure 1.2 depicts the metal manufacturing's hierarchy of goals and their assignments to members of the organizational structure.

Followed by the definition of a unified hierarchy of company's goals, managers attempt to predict which internal and external factors will support or hinder attainment of desired company's goals (Step 2). In this context, a key strength of the metal company identified by managers is the loyal and talented workforce that is able to adapt to business pressures. Inversely, a potential barrier in the environment to the continued success of the metal company is a low availability of steel in the market, what may hinder the continued mass production.

Besides performing analysis of how possible future scenarios may affect goals, the actual achievement of the integrated hierarchy of goals requires managers to identify their available strategic alternatives (step 3). The identification of available alternatives consists of selecting the goals to be achieved and subsequently generating the possible ways to achieve it. For example, by selecting the "12% of return on investment" (ROI) strategic goal, managers have to identify possible ways to increase the return of

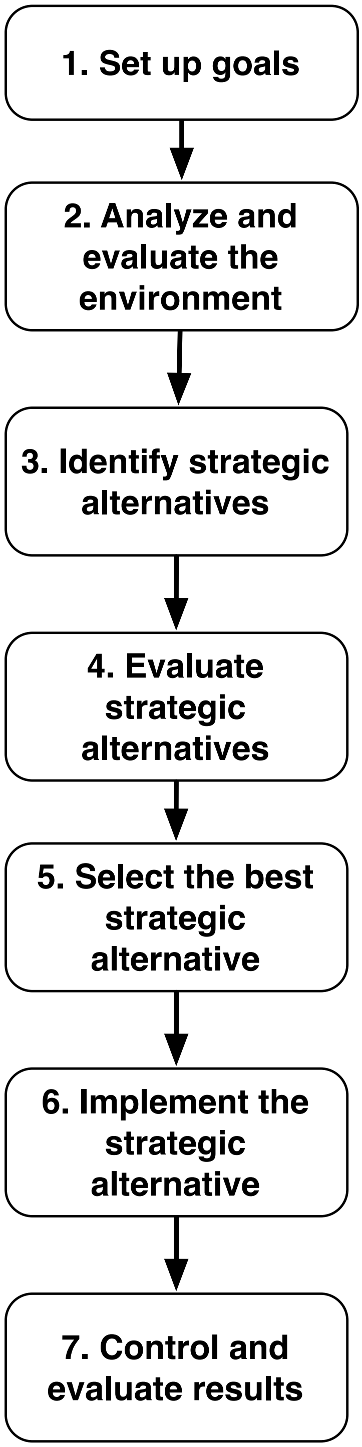


Figure 1.1: Steps in a Basic Planning Process [153]

investment using company's resources. One possible way to increase the overall company's ROI consists of increasing it by 12% in one of the lo-

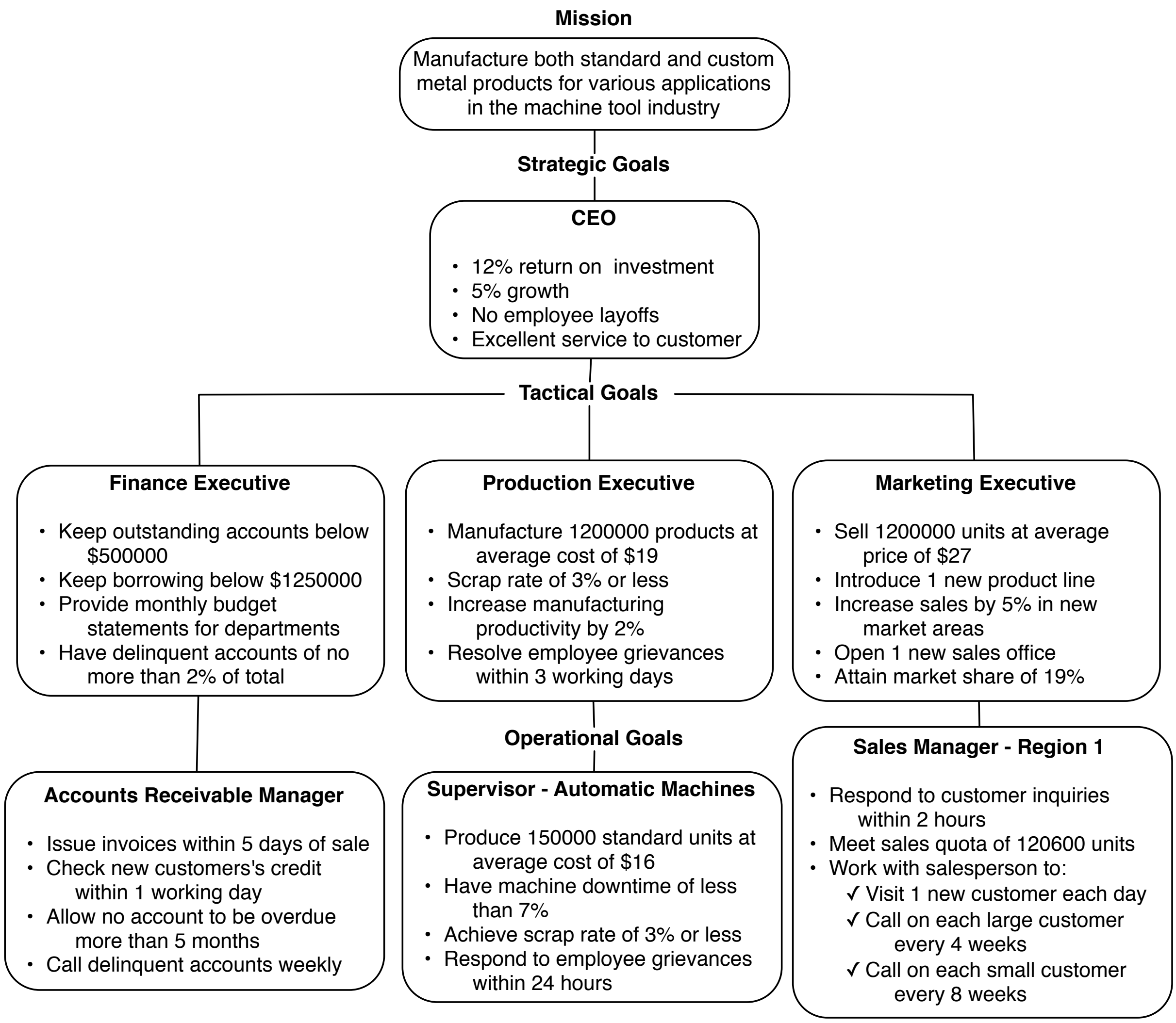


Figure 1.2: Goal Hierarchy Extracted from [34, p.222]

cations the company operates. Alternatively, the ROI might be uniformly increased in several locations where the company operates. Other variances regarding ROI increase might be also considered in terms of different products or services the company provides. As the alternatives are generated, they must be evaluated and selected according to different criteria (steps 4 and 5). For example, France might be chosen instead of Germany due to France's higher potential to attain the 12% of ROI increase (strategic goal) in an effective and efficient way. Alternatively, Germany may be also chosen due to the low effort required for the adoption of the alternative.

Once a strategic alternative is selected, it needs to be implemented accordingly. In order to implement strategic alternatives and indirectly achieve strategic goals (step 6), goals need to be assigned to members of different levels of the organizational structure that interact to perform their work, as the achievement of lower level (operational) goals entail the achievement of upper levels within the goal hierarchy. This interaction among organizational members is commonly coordinated on the basis of execution of a number of processes.

In order to gain a holistic overview of how enterprise's goals are achieved, managers usually intend to visualize a number of aspects regarding processes. First, the metal manufacturing company is interested in visualizing the social interactions carried out among its members with external members, like suppliers, logistics distributors and retailers. With such global overview, managers intend to understand the overall chain of company's processes, how they deliver value to the final customers and how they enable the achievement of lower-level goals. Second, managers are also interested in understanding and visualizing the interactions among its own internal organizational members (roles and departments) to achieve its lower level goals. For example, the sales manager - region 1 has to interact with the salesperson to achieve its goals (e.g., "work with salesperson to visit one customer each day" operational goals from the sales manager - region 1). Furthermore, besides interactions with other members to achieve goals, the visualization of operational steps would also help the company to understand issues related to goal achievement.

Overall, the example from the metal manufacturing company shows the importance of supporting the different steps of the organization's planning process. This support has advantages for different stakeholders. First, senior and middle managers can assess the means by which company's strategy is being implemented as company's processes. This allows them to assess how changes in enterprise's goals can impact their processes and vice-versa, enabling them to perform synchronized changes between both structures and adequately plan goal achievement. Middle-managers can gain a holistic overview of all processes performed by the company, being able to evaluate how changes in their goals and processes influence other departments, and thus manage risks accordingly. Equally, senior managers may spot problems in the overall chain of organization's process that may impact in the optimal delivery of products and services to the final customer. Operational managers and employees can understand the overall context of their work, why they need to achieve certain operational goals and how such goals relate to the overall enterprise's goal hierarchy.

1.3 Research Questions (RQs)

As outlined in the above motivating example of the metal manufacturing company, the support for a *strategic enterprise architecture* approach arises as an emerging research topic where a number of challenges need to be considered. In this section, we state these challenges by means of four research questions:

RQ1. How can we develop a strategic enterprise architecture approach to support business stakeholders in the exercise of the enterprise planning process?

In order to enable the development of a strategic enterprise architecture approach, we need a framework that contains a modeling language with well-defined semantics of concepts to support stakeholders in the representation of conceptualization inherent to enterprise planning process (i.e., motivational and behavioral concepts). Moreover, this framework should provide guidelines for the specification of modeling concepts in such strategic enterprise language. Finally, in order to get the benefits from the representation of strategic enterprise models, our approach requires the development of automated reasoning techniques to support business stakeholders in the exercise of the enterprise planning process.

Alternatively, we can decide to adapt existent enterprise modeling frameworks that already contain modeling languages, methodological guidelines and automated reasoning techniques that address the conceptualization inherent in the enterprise planning process. In the following, we further refine this general research question into two more specific research questions as follows:

RQ1.1. Which are the concepts required for expressing a strategic enterprise architecture? Or in other words, which are the abstractions to be captured for expressing the enterprise planning process? Which are the most suitable motivational-, behavioral-related concepts and how to interconnect them? Besides motivational and behavioral concepts, do we need other abstractions for capturing the overall enterprise planning process?

The modeling language should capture the right abstractions to express the conceptualization inherent to enterprise planning process. As can be seen by the description of our motivating example in the previous section (Section 1.2.1), motivational concepts (goals of different shades, such as strategic, tactical and operational goals) and behavioral concepts (business processes) consists of instrumental abstractions for expressing the enterprise planning process. Besides motivational and behavioral concepts, we have also to identify other abstractions that support the exercise of the enterprise planning process. In this context, our approach should inhere as much as possible concepts from existent languages in Goal Modeling, Enterprise Modeling and Business Process Modeling.

RQ1.2. Which types of analysis (reasoning techniques) should be performed on strategic enterprise models in order to support the exercise of the enterprise planning process?

The reasoning approach should initially start with strategic enterprise models expressed using formal semantics. Subsequently, a set of properties to be verified over strategic enterprise models should be defined and automated analysis for identification of such properties should be supported by means of tool support in an acceptable time. The reasoning technique should support as many as possible steps of the enterprise planning process described in the previous section (Section 1.2.1).

Finally, we must define whether our solution satisfactorily solves the problem at hand. This is expressed by the fourth research question:

RQ2. How can we evaluate whether our proposed framework satisfies the objectives of research stated by means of the research questions here expressed?

1.4 Existing Approaches and Their Limitations

In order to support the representation and analysis of strategic enterprise architectures, many approaches exist in a number of areas of Computer Science. More specifically, motivational modeling is mainly addressed by Goal-Oriented Requirements Engineering (GORE) and Enterprise Modeling (EM), whereas behavioral modeling is mainly addressed in Business Process Management (BPM). Hybrid approaches that acknowledge the benefits of integrating motivational and behavioral concepts also exist in BPM.

In terms of representational support, GORE frameworks [19, 208] use the concept of goal for capturing stakeholders' requirements for a target software system. Such representation enables the linkage between business requirements expressed as goals and the technical system requirements that address such business goals. Although such approaches provide suitable abstractions for the requirements engineering process, they are restricted to software engineering, not enterprise modeling.

In enterprise modeling, support for representation of motivational aspects is very rudimentary [8, 25]. This modeling support range from languages with unclear semantics (e.g., objective concept within the ARIS framework [39, 25] and several goal categories of BMM [75]) until very simplistic support (e.g. concepts of hard/soft-goals in i* [208], goals in EKD [102] and goals in BIM [89]). Even ArchiMate [8] (the standard language for enterprise modeling) presents a more refined set of modeling constructs like mission, vision, strategic goals, but still lacks essential modeling constructs like tactical and operational goals. Similarly, research is fragmented in the scope of hybrid proposals. Some proposals use the GORE concept of (operational) “goal” to either provide a motivational perspective for activities inside business processes [109, 125] or to generate alternative process variants on the basis of goals [117, 107], while other approaches [130, 125] recognize long-term, strategic goals. However, such approaches cannot represent the overall hierarchy of goals (like the one from the metal company, Section 1.2.1) in a single approach.

As a consequence of such rudimentary support for goal representation, the relations among goals and the set of behavioral elements that realize such goals are seriously impaired in enterprise modeling and hybrid approaches. In this context, although such approaches acknowledge the existence of business processes that realize such business goals, the absence of a single approach that addresses all goal categories leads to weak support in the integration between motivational and behavioral perspectives.

In BPM literature, support for behavioral modeling is provided in terms of the concept of business process (different ways in which a case can be handled [196]). Business process’s control-flow is represented in terms of different abstractions like activities [161, 176], data objects [16], messages [14], among others. However, such abstractions only capture the operational perspective of processes, devoting little attention to their strategic perspective. Consequently, the integration between the motivational and behavioral perspectives is not addressed in the scope of such approaches.

In terms of analysis of strategic enterprise architectures, GORE approaches use goal models as the starting point in automated reasoning techniques that generate alternative system designs. In this context, a number of GORE reasoning techniques quantify the level of satisfaction of top system goals depending on alternative system designs [65, 86]. Other approaches go beyond as they can not only quantify the level of satisfaction of top system goals, but can also recommend which designs to select [179, 142, 115]. Although GORE techniques allow one to perform advanced reasoning with goal models, their scope relies on the evaluation/generation of system designs, not strategic plans. In the scope of enterprise modeling and hybrid approaches, such approaches borrow the GORE automated reasoning techniques accordingly. In this context, as such approaches do not distinguish among goals of different types, the usage of GORE automated techniques cannot fully support the generation of strategic alternatives in the enterprise planning process.

Overall, the support for representation and reasoning for goal modeling presents a number of challenges in the acknowledgment of the existence of different shades of goals and in the integration between the motivational and behavioral perspectives. Further, although BPM research provides a good support for the representation of business processes and their control-flows, this is not reflected in good representational support for linking motivational and behavioral perspectives. Consequently, current approaches do not capture an integrated hierarchy of goals and behavioral concepts in a unique approach. In terms of reasoning support, techniques cannot automatically support the exercise of the enterprise planning process.

1.5 Approach

We have defined an approach for answering our research questions and by answering them, we reach the objectives of the thesis.

To answer the research question regarding the development of a systematic approach for strategic enterprise architecture (RQ1), we have conducted systematic studies and literature review in conjunction with other members of our research group in order to learn and clarify the conceptualization of motivational and behavioral perspectives. This study has been conducted on several areas of Computer Science such as Artificial Intelligence, Agent-Oriented Computing, Multi-Agent Systems, Goal-Oriented Requirements Engineering, among others in order to provided inspiration for the definition and usage of modeling concepts in the scope of our work and objectives. Such investigation is reported in publications 1 and 2 from referred journals and publications 3 and 5 from refereed conferences in Section 1.8.

Regarding the research question about the definition of the adequate set of modeling concepts for the representation of conceptualization inherent to enterprise planning process (i.e., motivational, behavioral-related concepts) (RQ1.1), we have conducted a literature investigation in the areas of Goal-Oriented Requirements Engineering (GORE), Enterprise Modeling (EM) and Business Process Management (BPM) to identify the support provided for the representation of motivational and behavioral concepts on current approaches. Current approaches have been also investigated to support the decision about creating or adapting existent strategic enterprise modeling frameworks. The focus on the three aforementioned areas was taken as a result of the insights acquired in the previous literature investigation with other members of the group.

Besides such studies, we have also used the insights gathered in our previous work [24] performed in a real-world hospital environment in the alignment of goal and business process models. From the analysis of the insights acquired with our previous experience together with the literature review in Computer Science (GORE, EM and BPM), we realized that a comprehensive conceptualization for the definition and interconnection of motivational and behavioral domains was still missing. For this reason, we performed extensive literature review in a number of areas of Management Sciences (in particular, Strategic Management, Management and Operations Management) to acquire such common conceptualization.

To answer the research question about the automated techniques to be performed on strategic enterprise models (RQ1.2), we have used the same literature study in Management Sciences of the previous phase to identify the enterprise planning process and the need of providing automated support for as many as possible steps of the enterprise planning process. Subsequently, we have investigated GORE reasoning techniques in order to understand their reasoning capabilities. On the basis of such investigation, we have adapted the Constrained Goal Model (CGM) formalism in order to develop our automated reasoning technique.

1.6 Thesis Overview and Contributions

Figure 1.3 presents a general overview of our strategic enterprise architecture approach, the SIENA (StrategIc ENterprise Architecture) Modeling Framework for Strategic Enterprise Modeling and Analysis. The framework is composed of two modeling languages, their corresponding modeling guidelines for the usage of modeling concepts and an automated reasoning technique.

In the remainder, we detail each of these contributions:

- A **strategic enterprise architecture modeling language**, called

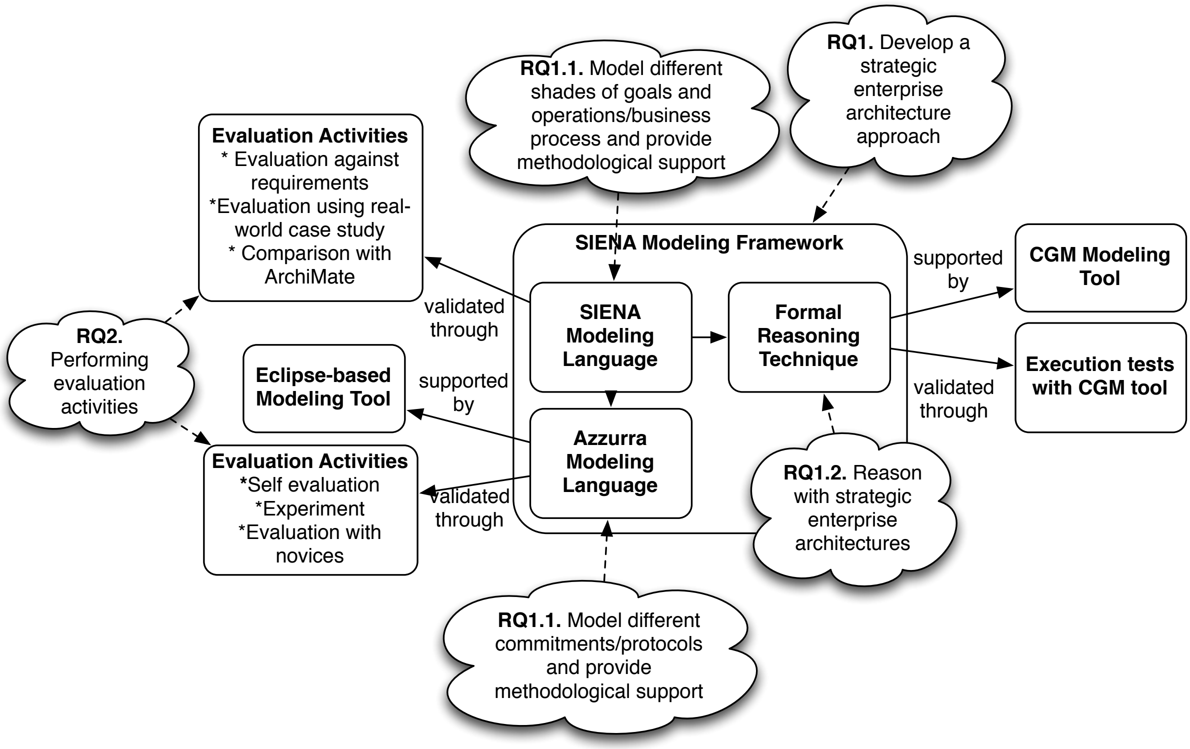


Figure 1.3: General overview of our strategic enterprise architecture approach

SIENA (StrategIc ENterprise Architecture) that delineates the differences in semantics and usage of different shades of goals (motivational elements) and operations/business processes (behavioral elements). Our proposal also introduces the distinguishing feature of *dimensional refinement operators*, a new operator that can be used for the refinement of strategic goals in terms of time, location and products/services dimensions. In comparison with traditional OR-refinements that capture alternatives for achieving goals, dimensional refinement operators go beyond by enabling the specification of different alternatives to achieve strategic goals in different points of a given dimension. With such contributions in hand, our language supports the definition of different shades of motivational concepts, behavioral concepts and refinements that cannot be expressed in any other similar proposal to the best of our knowledge. Further, this conceptual model also proposes a *hierarchical architecture for strategic enterprise models* that includes goals and the operations/processes through which they are operationalized. Finally, methodological guidelines are provided that explain how to elaborate such strategic enterprise models and when each concept should each be used in enterprise modeling.

- An **automated reasoning strategic planning technique** that takes as input a strategic enterprise architecture model and automatically generates optimum strategic plans (with respect to some objective function) subject to constraints to achieve strategic goals. To achieve this, a semantic mapping from strategic planning concepts (strategic, tactical goals, dimensional operators, etc.) into the CGM formalism [142] (formalism used for reasoning with goal models in software engineering) is defined. Then, strategic planning constraints and objective functions are specified in the CGM tool. With this approach in hands, we provide an automated strategic planning tool that explores enterprise variability with the use of dimensional operators. Here, our main contribution rests on the formalization of strategic planning concepts (e.g. strategic, tactical goals, strategic plans, etc.) and provisioning of an algorithm for performing strategic planning analysis (such as generation of strategic plans, SWOT analysis, strategic analysis). While there is a lot of work about the topic in the literature of Management Sciences (Chapter 3), such literature provides informal treatment for strategic planning, in contrast with our approach that performs strategic planning algorithmically.

- A **business process modeling language**, called Azzurra modeling language, founded on the primitives of *commitments* and *protocols* is defined for the representation of the internal logic (control-flow) of business processes. The representation of business processes in terms of *commitments* is a distinguishing feature of our approach. Azzurra also includes business primitives such as delegations, deadlines, constraints over roles and the notion of initiation and termination of a protocol (business process). The language introduces a graphical notation for modeling the main elements of a business process and supports the construction of business process models with a prototype Eclipse-based modeling tool. The language has been developed in a work in conjunction with other members of the research group, including Fabiano Dalpiaz and Paolo Giorgini. Our approach also provides a **business process design approach** for the generation of business processes' control-flow from SIENA operational goals and their subsequent specification using the Azzurra language.

In order to define whether the framework meets the research objectives, we have performed different types of evaluation according to the artifact under consideration enumerated as follows:

- **SIENA modeling language:** the language has undergone by three evaluation phases. In Chapter 2, we detail the research questions of Section 1.3 into the requirements to be met by strategic enterprise architectures. Consequently, the first phase evaluates the SIENA and Azzurra languages against the achievement of such requirements. This phase also uses the scenario of the integrated goal hierarchy from Management literature [34] (Section 1.2.1) to illustrate the evaluation. In the second phase, we use the real-world case study [24] from the Rheumatology department of the university hospital from our previous work to build models using SIENA and Azzurra. With the usage of such scenario, we intend to evaluate SIENA expressiveness and applicability for modeling a real use case. Finally, the third evaluation phase compares the SIENA and Azzurra modeling languages with the ArchiMate modeling language. The ArchiMate modeling language has been chosen due to its relevance as a standard language for enterprise modeling. After this modeling phase, we have compared models in both languages with respect to expressiveness.

- **Formal reasoning technique:** we evaluate the correctness of our approach by running the CGM tool and verifying the obtained results against the expected results. Further, this evaluation phase has been conducted in multiple models in order to investigate the behavior of the CGM software in a number of different models, thus stressing out our approach.

- **Azzurra modeling language:** Three different types of evaluations have been conducted to assess the language. First, an evaluation of the language has been performed by modeling two real-world scenarios from the medical domain and subsequent comparing the Azzurra's features with mainstream business process modeling approaches. Second, we have designed and conducted a preliminary experiment performed with a class of masters students at the University of Trento to examine the suitability of Azzurra for unstructured processes, a special class of business processes from literature. Third, we have conducted an in-depth study with the supervision of three master students (novices) that intended to first model clinical guidelines (CGs) (a special type of unstructured process within the medical domain) and subsequent compare the representation with their counterparts modeled in BPMN, the standard language for the representation of business processes.

1.7 Structure of the Thesis

This section depicts the structure of the thesis and shows in which chapters the research questions are answered.

- **Chapter 2: Requirements for Strategic Enterprise Architectures (RQ1).** This chapter starts with the research questions introduced in Section 1.3 and refines them in terms of the desirable requirements for strategic enterprise architectures. Such desirable characteristics are derived from an observation of the representational and analysis needs from the metal manufacturing company introduced in Section 1.2.1.

- **Chapter 3: Research Baseline (RQ1).** This chapter introduces the baseline of our work as the set of approaches that better meet the requirements for strategic enterprise architectures stipulated in the previous chapter. For the development of SIENA, the Business Intelligence Model [89] and the Business Motivation Model (BMM) [75] have been used as a foundation for the modeling concepts, while a number of proposals in Management Sciences provide the proper conceptualization to characterize different shades of goals and operations. For the development of the formal reasoning technique, the Constrained Goal Models (CGM) formalism [142] has been used as the formal reasoning technique for goal models. *Commitments* [184] and *protocols* [207] from multi-agent systems have been used as conceptual primitives for representing the social perspective of business processes, whereas the KAOS framework has been used for the business process design approach for the generation of Azzurra specifications from SIENA operational goals.

- **Chapter 4: Related Work (RQ1).** This chapter presents the state-of-the-art by presenting a summary of the most relevant proposals in a number of areas of Computer Science that use the concepts of goals, operations and business process as an abstraction.

- **Chapter 5: SIENA Modeling Language (RQ1.1).** This chapter presents SIENA, our strategic enterprise architecture modeling language together with its modeling guidelines for the representation of different shades of goals and operations/business processes.

- **Chapter 6: Planning with Strategic Goals (RQ1.2).** This chapter presents our formal reasoning technique performed over SIENA models.

- **Chapter 7: Azzurra Modeling Language (RQ1.1).** This chapter presents Azzurra, our business process modeling language for the representation business processes' control-flow. The chapter also presents our business process design approach that generates Azzurra specifications from SIENA operational goals.

- **Chapter 8: Evaluation of SIENA Modeling Framework (RQ2).** This chapter presents the several types of evaluation we have performed with the SIENA modeling language.

- **Chapter 9: Evaluation of Azzurra Modeling Language (RQ2).** This chapter presents the several types of evaluation we have performed with the Azzurra modeling language.

- **Chapter 10: Conclusions and Future Work.** This chapter concludes the thesis by discussing its main contributions and the drawbacks of the proposal. Finally, we propose topics for further investigation as part of future work.

1.8 List of Publications

This section presents the list of publication related to this thesis divided by type of publication. We also clarify the contribution of the author of this thesis to each publication.

Refereed Journal

1. Horkoff, J.; Aydemir, F.; Cardoso, E.; Li, T.; Paja, E.; Salnitri, M.; Piras, L.; Mate, A.; Mylopoulos, J.; Giorgini, P. *Goal-Oriented Requirements Engineering: An Extended Systematic Mapping Study.* Requirements Engineering Journal (REJ), 2017.

   My contribution: The first author of this paper designed and guided the study and me and the other PhD students selected and revised the papers to be included in the study.

2. Horkoff, J.; Li, T.; Li, F.; Salnitri, M.; Cardoso, E.; Giorgini, P., Mylopoulos, J. and Pimentel, J. *Using Goal Models Downstream: A Systematic Roadmap and Literature Review.* International Journal of Information System Modeling and Design (IJISMD 2014), v.6(2), pp. 1-42, 2014.

   My contribution: The first author of this paper designed and guided the study and me and the other PhD students selected and revised the papers to be included in the study.

Refereed Conferences

1. Cardoso, E.; Mylopoulos, J.; Mate, A. and Trujillo, J. *Strategic Enterprise Architectures.* In Proceedings of the 9th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM 2016), Sweden, 2016.

My contribution: I reviewed the literature, developed the core conceptual model and its modeling guidelines under the supervision of my co-authors and wrote the paper with feedback from my co-authors.

2. Cardoso, E.; Labunets, K.; Dalpiaz, F.; Mylopoulos, J.; Giorgini, P.; *Modeling Structured and Unstructured Processes: An Empirical Evaluation.* Proceedings of the 35th International Conference on Conceptual Modeling (ER 2016), Japan, 2016.

   My contribution: I designed the experiment, applied it in the master student class and wrote the paper with feedback from my co-authors. The second author of this paper chose and performed the statistical methods to evaluate the results of the experiment and I also evaluated the results under the supervision of my co-authors.

3. Horkoff, J.; Aydemir, F.; Cardoso, E.; Li, T.; Mate, A.; Paja, E.; Salnitri, M.; Mylopoulos, J.; Giorgini, P. *Goal-Oriented Requirements Engineering: A Systematic Literature Map.* Proceedings of the 24th Requirements Engineering Conference (RE 2016), China, 2016.

   My contribution: The first author of this paper designed and guided the study and me and the other PhD students selected and revised the papers to be included in the study.

4. Dalpiaz, F.; Cardoso, E.; Canobbio, G.; Giorgini, P.; Mylopoulos, J. *Social Specifications of Business Processes with Azzurra.* Proceedings of 9th IEEE International Conference on Research Challenges in Information Science (RCIS 2015), Greece, 2015. **(Best Paper Award)**

   My contribution: I re-wrote the full paper based on previous ideas and included some new content under the supervision of my co-authors. More specifically, Sections 1, 2, 3 and 4 have been written based on previous versions of the paper. Section 5 (implementation of the modeling tool), 6 (evaluation of the modeling language using unstructured scenarios in the medical domain), 7 (related work) and Section 8 (conclusion) have been included by me in this version.

5. Horkoff, J.; Li, T.; Li, F.; Pimentel, J.; Salnitri, M.; Cardoso, E.; Giorgini, P. and Mylopoulos, J. *Taking Goal Models Downstream: A Systematic Roadmap.* Proceedings of 8th IEEE International Conference on Research Challenges in Information Science (RCIS 2014), Morocco, 2014. **(Best Paper Award)**

   My contribution: The first author of this paper designed and guided the study and me and the other PhD students selected and revised the papers to be included in the study.

Working Papers

1. Cardoso, E.; Sebastiani, R.; Horkoff, J. and Mylopoulos, J. *Planning with Strategic Goals.* unpublished.

   My contribution: I reviewed the literature, implemented the formalization and reasoning into the CGM tool under the supervision of my co-authors and wrote the paper with feedback from my co-authors.

Chapter 2

Requirements for Strategic Enterprise Architectures

This chapter details the research questions introduced in Section 1.3 in terms of the desirable requirements of strategic enterprise architectures, their relevant concepts, methodology and automated analysis, thus motivating the contributions provided in this thesis. We start in Section 2.1 with the description of the process for requirements acquisition. In Section 2.2, we present the metal manufacturing company introduced in Chapter 1 as the motivating example that will be used in this thesis. In Section 2.3, we present the desired requirements that a strategic enterprise architecture must meet.

2.1 The Process For Requirements Acquisition

This chapter presents the requirements for strategic enterprise architectures, which have been acquired in two distinct stages. In the first stage, we conducted an exploratory case study in the hospital environment [24] of our previous approach. This exploratory case study investigated the alignment of goal models with other enterprise models (organizational structure, business processes, and data objects).

As we have argued in [24], a case study research method consists of an empirical inquiry that investigates a contemporary phenomenon within its real-life context, especially when the boundaries between phenomenon and context are not clearly evident [206]. In particular, when the phenomenon is not clearly defined a priori, exploratory case studies are conducted to gather preliminary information and to support the definition problems and suggestion of hypotheses, providing insights and comprehension of an issue or situation [110]. In this context, the research begins with the observation of the social setting and subsequent explanation for the phenomenon under consideration. Rather than testing a pre-formulated hypothesis, the research aims at developing general principles to account for the previous observations. The idea is to let questions emerge from the situation itself [177]. Hence, this exploratory case study was instrumental for the development of an initial insight of our model in terms of existent goal categories and attributes (reported in [27]) and the relations between motivational and behavioral perspectives (reported in [26]).

With such results in hands, a discussion that naturally arises refers to the generality of results acquired in such case study. As argued in [206, 178], the results of case studies are generalizable to the theoretical propositions and in our case, they have proven fruitful for both generating and testing hypothesis in a process called as analytical generalization [206]. Hence, this exploratory case study was instrumental for the enrichment of the theory in enterprise modeling, both with the conceptualization of the goal perspective as well as the relations of the goal perspective with the other architectural domains.

However, although our exploratory case study contributed to enrich the literature in enterprise modeling, this effort was a single case study and therefore, we still had a simplistic enterprise model. On the basis of such realization, we decided to expand the investigation of the literature of other fields of Computer Science beyond enterprise modeling. Therefore, in the second stage (i.e., the present thesis), the author of this thesis conducted systematic studies and literature review in conjunction with other members of the research group in a number of areas of Computer Science, such as Artificial Intelligence, Agent-Oriented Computing, Multi-Agent Systems, Goal-Oriented Requirements Engineering, among others, in order to investigate the pertinent conceptualization of motivational and behavioral perspectives. By conducting such study, we concluded that literature in Goal-Oriented Requirements Engineering (GORE), Enterprise Modeling (EM), Business Process Management (BPM) and Multi-Agent systems could be further investigated due to their scope related to our research goals. Subsequently, we have conducted the literature investigation in such areas to identify the support provided for the representation of motivational and behavioral concepts on current approaches. We concluded that, although such areas could provide some conceptual support aligned with our research purposes, this support was still limited.

Therefore, from the analysis of the insights acquired with our previous experience together with the literature review in Computer Science (GORE, EM, BPM and Multi-Agent systems), we realized that a comprehensive conceptualization for the definition and interconnection of motivational and behavioral domains was still very limited. For this reason, we performed extensive literature review in a number of areas of Management Sciences (in particular, Strategic Management, Management and Operations Management) to acquire such common conceptualization.

Hence, in a nutshell, Management literature provided the main conceptualization for the development of the conceptual model, reasoning and methodology in the work reported in this thesis. However, although we strived to adopt as much as possible the conceptualization from such literature to avoid inconsistencies, we have also incorporated insights from our previous stages, such as our exploratory case study in the hospital environment and from the literature review in GORE, EM, BPM and Multi-Agent systems.

2.2 Motivating Example

This section introduces the motivating example used in this thesis. This motivating example consists of the metal manufacturing company [34, p.222] from Management literature introduced in Chapter 1. The metal manufacturing company consists of an autonomous organization divided into different functional areas (finance, human resources, operations, marketing and research and development). Functional areas are divided into departments that deliver manufactured metal products to customers and departments are composed of different roles that execute company's work.

The success of the metal company requires managers to decide where the company should be in the future and to find a path to reach such desired future. In practice, company's management process is performed by exercising five overall functions (management functions) which consist of planning, organizing, staffing, leading and controlling [153, 51, 34]. Among the five management functions, the **planning function** is the first and one of the most crucial activities for the enterprise as it enables the company to plan for the future. In practice, it involves setting up organization's goals and allocating actions and resources to achieve such goals, providing a sense of direction and unity of purpose for the organization and its subsystems [153, 51]. Figure 2.1 depicts the basic steps of the planning process.

In the metal manufacturing company, the planning process (step 1) starts by managers setting up metal company's goals for all the levels of the organizational structure (i.e., the entire metal company, its functional areas and departments and roles). Such goal-setting process is then reflected in a unified goal hierarchy with the division of organization's goals into a mission, strategic, tactical and operational goals, each of them assigned to different levels of the organizational structure. Figure 2.2 depicts the metal manufacturing's hierarchy of goals and their assignments to members of the organizational structure.

Followed by the definition of a unified hierarchy of company's goals, managers attempt to predict which internal and external factors will support or hinder attainment of desired company's goals (Step 2). In this context, a key strength of the metal company identified by managers is the loyal and talented workforce that is able to adapt to business pressures. Inversely, a potential barrier in the environment to the continued success of the metal company is a low availability of steel in the market, what may hinder the continued mass production.

Besides performing analysis of how possible future scenarios may affect goals, the actual achievement of the integrated hierarchy of goals requires managers to identify their available strategic alternatives (step 3). The identification of available alternatives consists of selecting the goals to be achieved and subsequently generating the possible ways to achieve it. For example, by selecting the "12% of return on investment" (ROI) strategic goal, managers have to identify possible ways to increase the return on investment using company's resources. One possible way to increase the overall company's ROI consists of increasing it by 12% in one of the locations the company operates. Alternatively, the ROI might be uniformly increased in several locations where the company operates. Other variances regarding ROI increase might be also considered in terms of different products or services the company provides. As the alternatives are generated, they must be evaluated and selected according to different criteria (steps 4 and 5). For example, France might be chosen instead of Germany due to France's higher potential to attain the 12% of ROI increase (strategic

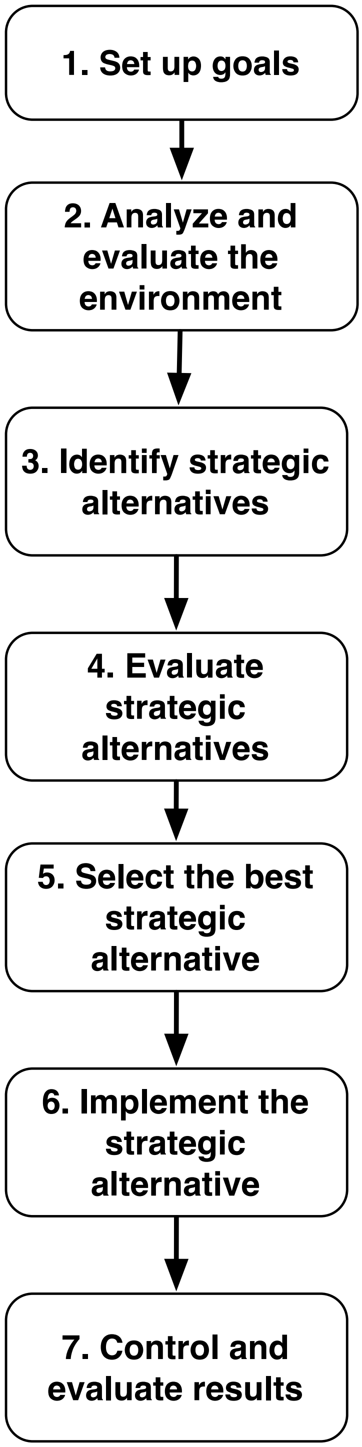


Figure 2.1: Steps in a Basic Planning Process [153]

goal) in an effective and efficient way. Alternatively, Germany may be also chosen due to the low effort required for the adoption of the alternative.

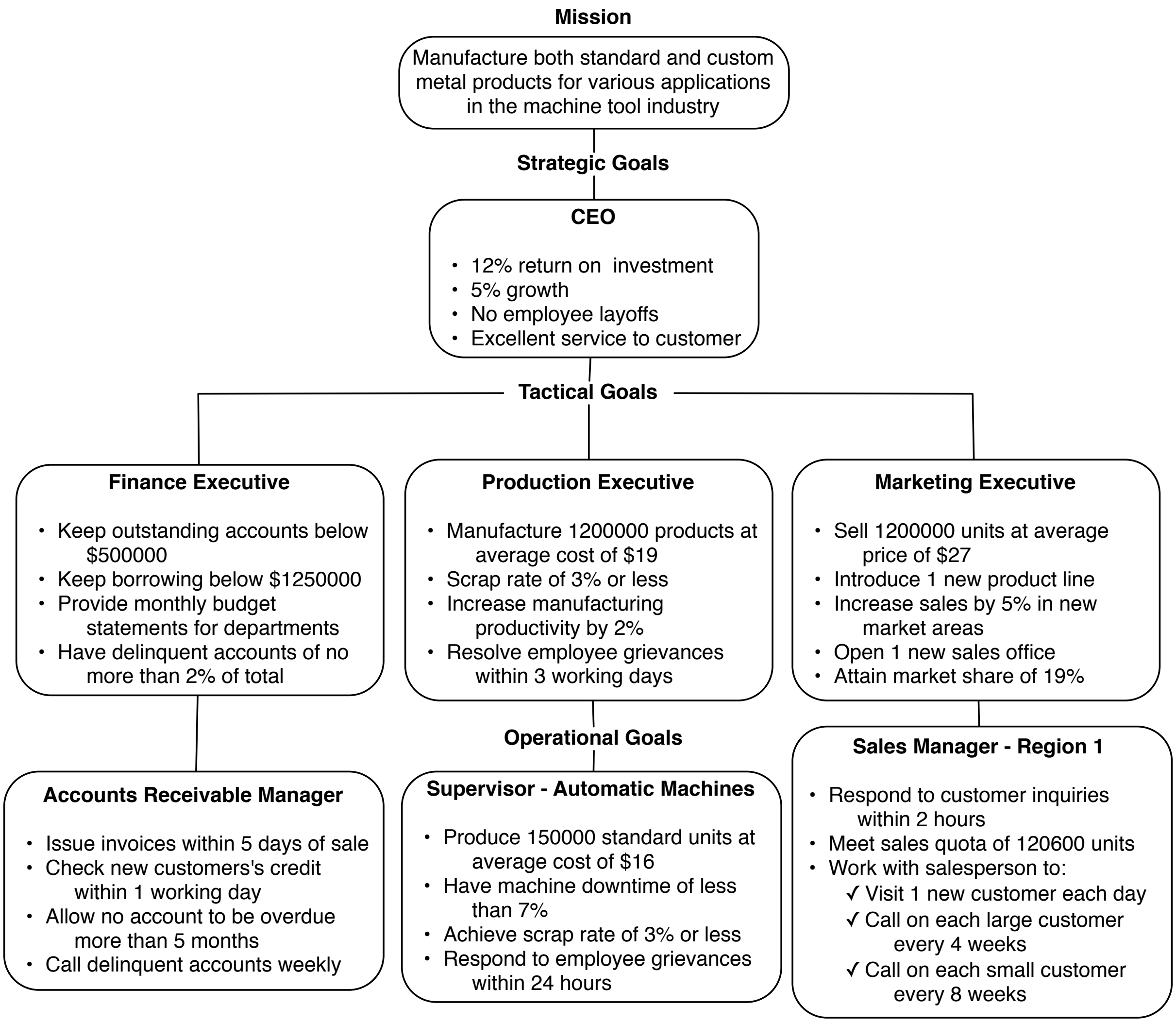


Figure 2.2: Goal Hierarchy Extracted from [34, p.222]

Once a strategic alternative is selected, it needs to be implemented accordingly. In order to implement strategic alternatives and indirectly achieve strategic goals (step 6), goals need to be assigned to members of different levels of the organizational structure that interact to perform their work, as the achievement of lower level (operational) goals entail the achievement of upper levels within the goal hierarchy. This interaction among organizational members is commonly coordinated on the basis of execution of a number of processes.

In order to gain a holistic overview of how enterprise's goals are achieved, managers usually intend to visualize a number of aspects regarding processes. First, the metal manufacturing company is interested in visualizing the social interactions carried out among its members with external members, like suppliers, logistics distributors and retailers. With such global overview, managers intend to understand the overall chain of company's processes, how they deliver value to the final customers and how they enable the achievement of lower-level goals. Second, managers are also interested in understanding and visualizing the interactions among its own internal organizational members (roles and departments) to achieve its lower level goals. For example, the sales manager - region 1 has to interact with the salesperson to achieve its goals (e.g., "work with the salesperson to visit one customer each day" operational goals from the sales manager - region 1). Furthermore, besides interactions with other members to achieve goals, the visualization of operational steps would also help the company to understand issues related to goal achievement.

Overall, the example from the metal manufacturing company shows the importance of supporting the different steps of the organization's planning process. This support has advantages for different stakeholders. First, senior and middle managers can assess the means by which company's strategy is being implemented as company's processes. This allows them to assess how changes in enterprise's goals can impact their processes and vice-versa, enabling them to perform synchronized changes between both structures and adequately plan goal achievement. Middle-managers can gain a holistic overview of all processes performed by the company, being able to evaluate how changes in their goals and processes influence other departments, and thus manage risks accordingly. Equally, senior managers may spot problems in the overall chain of organization's process that may impact in the optimal delivery of products and services to the final customer. Operational managers and employees can understand the overall context of their work, why they need to achieve certain operational goals and how such goals relate to the overall enterprise's goal hierarchy.

2.3 Requirements

Taking into account the needs of the previous scenario from the metal manufacturing company (Figure 2.2), a number of desirable requirements have been identified regarding the representation of strategic enterprise architectures, their methodology and automated reasoning.

Given the importance of the SMART criteria [50, 132, 183] for goal setting in Management literature, our requirements are aligned with these criteria. SMART consists of a mnemonic with little consensus about the meaning of the words on the acronym. Here, we adopt the definition from [50] in which SMART denotes: (i) *Specific*: goals must target a specific area for improvement, (ii) *Measurable*: goals must have an indicator and targets that quantify their progress towards achievement, (iii) *Assignable*: goals must specify who is going to achieve it, (iv) *Realistic*: goals must be realistically achievable given the available resources and (v) *Time-Related*: goals must have a deadline for their achievement. In the remainder of this section, we enumerate and detail our requirements, making the connections with the SMART criteria, when needed.

Regarding the **representation and methodology of strategic enterprise architectures**, we have identified a number of desirable requirements that we detail as follows:

R1. Expressiveness. The language to represent strategic enterprise architectures must have high expressiveness to capture goals of different types (strategic, tactical, operational), such as those exemplified in the metal manufacturing company. Further, the set of processes by which such goals are achieved should be also captured by our language. Therefore, our language should be expressive with respect to three different dimensions: (i) the representation of motivational domain (ii) the representation of behavioral domain and (iii) the representation of the interconnection between both perspectives. In the following, we detail the sub-requirements that each perspective should encompass:

R1.1 Motivational Perspective. Regarding the conceptual definition of the motivational perspective, the concept of "goal" represents the central concept of such perspective. Moreover, as can be noticed from the metal manufacturing example, goals have a number of conceptual attributes to characterize them together with a number of relations. In order to achieve the *specific* property from the SMART criteria by making goals as specific as possible, we enumerate a number of goal's conceptual attributes, the type of relations that should be captured among them and the methodology for their specification as follows:

1. **Representation of goal formulations and conceptual attributes**

    1.1. **Description.** As the metal manufacturing example demonstrates, each goal must have a description that encompasses what must be achieved (e.g. the strategic goal "12% return on investment");

    1.2. **Ownership.** Once elaborated by the enterprise, each goal must be assigned to the members of the organizational structure to ensure their achievement in order to achieve the *assignable* property from the SMART criteria. Therefore, the language has to be expressive enough to capture goals together with the organizational members responsible for their fulfillment, such as the whole enterprise, organizational units or roles, thus allowing accountability for organizational goals. For example, in the motivating example, the goal hierarchy follows the company's organizational structure;

1.3. **Time Frame for Goal Achievement.** Goals need to have a certain time frame (deadlines) in which they need to be achieved. For example, in the motivating scenario, the supervisor of automatic machines need to respond to employees grievances within 24h ("Respond to employee grievances within 24h" operational goal). By incorporating time frames for goals, we achieve the *time-related* property from the SMART criteria;

1.4. **Goal Pattern.** Different patterns may be required for considering goals to be satisfied. For example, when managers elaborate the "12% return on investment" strategic goal to be achieved in a time frame of three years, it is implicitly assumed that the goal is currently not achieved and needs to be achieved in the future. After three years, the achievement of the goal can be checked. Alternatively, when managers elaborate the "Keep borrowing below $1250000" tactical goal to be maintained, it is also implicitly assumed that the goal is currently being achieved and needs to be maintained within a specific timeframe;

1.5. **Target.** Each goal should have quantitative targets in order to determine in which extent this goal is being fulfilled, thus achieving the *measurable* property from the SMART criteria. For example, the "Achieve scrap rate of 3% or less" operational goal from the supervisor of automatic machines states a goal to achieved (Decrease scrap rate) and a measure with a target range (3% or less) to determine how well the goal should be achieved. By incorporating targets on the definition of goals, we also achieve the *realistic* property from the SMART criteria, once it forces managers to estimate realistic targets for goal achievement;

1.6. **Multiple Levels of Abstraction.** The example of metal manufacturing company shows that the definitions of goals may be stated in a broad scope within the organization, ranging from high-level concerns such as the mission statement in Figure 2.2 to the operational results (operational goals) that must be achieved by roles in Figure 2.2. Therefore, the motivational perspective must be able to capture the four levels of abstraction (mission, strategic, tactical and operational goals), also including precise criteria to build goal models with different layers of abstraction and methodological guidelines that enable the derivation of lower level goals from higher level goals.

2. **Representation of relations among goals.** As the metal manufacturing scenario and its planning demonstrate, three types of relations among goals should be captured within strategic enterprise architectures. First, goals need to be decomposed into a finer grained structure to enable their assignment to organizational members for their achievement. Second, as the enterprise's planning also demonstrates, different alternatives for goal achievement should be captured in order to address the generation of multiple alternatives during the planning process. Third, refinement and alternatives relationships usually capture a goal structure in terms of well-defined relations, with the achievement of lower level goals fully implying the satisfaction of upper goals in the hierarchy. However, many relations among goals in business analysis are usually not formalizable in terms of well-defined refinements and alternatives, presenting often partial relations among them. For example, increasing sales ("increase sales by 5% in new market areas") certainly demands a corresponding increase in the manufacturing productivity ("increase manufacturing productivity by 2%") that should be captured within the strategic enterprise architecture model. This relation is partial as an increase in sales should be accompanied by an increase in the manufacturing productivity, but

an increase in manufacturing does not fully imply the satisfaction of the "increase sales" goal.

3. **Representation of environmental factors that impact goal achievement.** As the step 2 of planning process demonstrates, strategic enterprise architectures should be able to represent factors that impact the achievement of enterprise's goals, by either supporting or hindering their achievement. The representation of such aspects is fundamental to capture uncertainties which naturally arise in the course of the enterprise planning process.

R1.2 Behavioral Perspective. Regarding the conceptual definition of behavioral perspective, "process" consists of the central abstraction to represent behavior in strategic enterprise architectures. As the metal manufacturing example evidence, behavioral representation also involves other conceptual characteristics complementary to the concept of "process" that we enumerate as follows:

1. **Social Perspective of Process.** The concept of "process" must have abstractions to capture the social perspective of the internal logic of processes (process' control-flow). These social abstractions are fundamental for the representation of the processes in social terms, thus enabling managers to visualize the interactions and compromises among different roles in the achievement of operational goals;

2. **Operational Perspective of Process.** Besides the social perspective, the operational perspective of processes should be also captured in order to reveal the operational steps that are required for the achievement of each operational goal;

3. **Business Process Architecture (BPA).** Although the concept of process is instrumental in capturing the isolated behavior required

to achieve a given enterprise's goal, in practice, companies have a set of processes that interact among themselves to *jointly* realize the company's hierarchy of goals. Therefore, the behavioral perspective should be able to capture the entire set of enterprise's processes, together with their corresponding relations, i.e., the business process architecture (BPA). Furthermore, as business processes exist at different levels in organizations (e.g. intra-organizational, operational, etc.) and some principle should be also used to organize the entire set of company's processes;

R2 Traceability Between Motivational and Behavioral Perspectives. Regarding the interconnection between motivational and behavioral perspectives, the metal manufacturing description emphasizes the need of representing the behavioral perspective in terms of three elements: (i) the overall chain of company's processes, (ii) the interactions between internal and external members and (iii) the interactions among internal members to achieve lower-level goals. In this way, as the behavioral perspective only exist to achieve organization's goals, these elements must be derived from the company's hierarchy of goals. Alternatively, in order to ensure the achievement of such organization's goals, consistency among motivational and behavioral perspectives must be kept in order to identify how changes in a specific business process may affect the achievement of certain strategic goals. Therefore, to ensure traceability between motivational and behavioral perspectives, the following sub-requirements are elaborated:

R2.1 Traceability in the Representation of Motivational and Behavioral Perspectives. A strategic enterprise architecture language should contain clear and precise relations between motivational and behavioral perspectives in order to ensure traceability between them. More precisely, the minimum requirement is to connect at least one motivational layer (usually the lowest level, i.e., the operational goal layer) with company's

BPA. Further, the abstractions from motivational layer should be also connected with the abstractions used for representing process' control-flows, thus enabling the full derivation of behavioral elements from the enterprise's motivational perspective;

R2.2 Traceability in Methodological Consistency Between Motivational and Behavioral Perspectives. The representation of the relations among the motivational and behavioral perspectives should be accompanied by clear methodological guidelines on how the abstractions in each perspective are related with the other perspective. For example, the methodological guidelines should allow one to derive the BPA together with its hierarchical levels from the motivational perspective or the alignment between both perspectives could be checked with the application of the methodological guidelines. By ensuring this traceability between both perspectives, we support the execution of the enterprise planning process, given the fact that organization's goals are implemented by means of processes (step 6);

Regarding the development of **automated analysis techniques with strategic enterprise architectures**, we have the following requirements:

R3. Support for Automated Reasoning with Strategic Enterprise Architectures. Besides capturing all the aforementioned concepts for the comprehensive representation of strategic enterprise architectures, the metal manufacturing company is also interested in using automated techniques for supporting the several steps of its planning process. Therefore, the use of automated reasoning first requires the **representation of strategic enterprise architecture models in terms of specifications with formal rigor (R3.1)**. Such formal specifications can then be used as input for automated analysis techniques. Second, **automated techniques must support the execution of the several steps of the planning process (R3.2)**. In particular, in order to support the planning process, such techniques should: (i) reason with different shades of goals **(R3.2.1)**, reason with environmental factors **(R3.2.2)**, support evaluation and selection of best strategic alternatives according to different criteria **(R3.2.3)** and support control and evaluation of the implemented strategic alternatives **(R3.2.4)**.

2.4 Summary

In this chapter, we have presented the requirements regarding the conceptual characteristics of strategic enterprise architectures and automated reasoning (summarized in Table 2.1) that serve as motivation for the contributions presented along this thesis.

Summary of Requirements for Strategic Enterprise Architectures	
Expressiveness in mot. perspective	Description (1.1)
	Ownership (1.2)
	Time Frame (1.3)
	Goal Pattern (1.4)
	Targets (1.5)
	Multiple Levels of Abstraction (1.6)
	Representation of Goal Relations (2)
	Representation of Factors that Impact Goal Achievement (3)
Expressiveness in beh. perspective	Social Perspective of Process (1.1)
	Operational Perspective of Process (1.2)
	Business Process Architecture (BPA) (1.3)
Traceability between mot/beh persp.	Traceability in Representation (2.1)
	Traceability in Methodological Consistency (2.2)
Support for Automated Reasoning	Formal rigor in specifications (R3.1)
	Support for Execution of Planning Process (R3.2)
	Reason with Different Shades of Goals (R3.2.1)
	Reason with Environmental Factors (R3.2.2)
	Support Selection of Best Strategic Alternatives (R3.2.3)
	Support Implementation of Strategic Alternatives (R3.2.4)

Table 2.1: Summary of Requirements for Strategic Enterprise Architectures

Chapter 3

Baseline

This chapter introduces existent concepts and formalisms that provide foundations for the achievement of the requirements from Chapter 2 with the development of our strategic enterprise architecture approach. We start by introducing the Business Intelligence Model (BIM) (Section 3.1), the Business Motivation Model (BMM) (Section 3.2) and relevant concepts from literature in Management Sciences (Section 3.3) as foundations for the development of the motivational and behavioral perspectives of our strategic enterprise architecture approach. Subsequently, we introduce the Constrained Goal Models formalism (Section 3.4), a formal reasoning technique for goal models in Requirements Engineering, which is used as the formal foundations for our strategic planning approach. Finally, commitments and protocols (Section 3.5) and the KAOS approach (Section 3.6) are also used as foundations for the development of behavioral perspective of our strategic enterprise architecture approach.

3.1 Business Intelligence Model (BIM) Framework

The *Business Intelligence Model (BIM)* [89] consists of an enterprise modeling approach for linking the business-level representation of an enterprise with the data stemmed from databases and data warehouses. The main purpose of the framework is to bridge the gap between data-oriented representation models and their corresponding business representation. As such, the approach provides abstractions close to business decision-makers, such as *goals*, *situations*, *processes* and *domain assumptions*. Such abstractions capture many notions prescribed by requirements from Chapter 2 and therefore, they are used for the development of strategic enterprise architectures.

In BIM, a *goal* represents an objective of a business which captures strategic enterprise's concerns, such as "Increase sales". Goals may be related by either *refinement* of *influence relationships*. In a refinement relation, goals are decomposed into a finer-grained structure by means of AND/OR relationships, with an AND decomposition supporting a goal to be decomposed in a series of sub-goals and an OR decomposition allowing analysts to model alternative ways of achieving a goal. Influence relationships ($G_i \rightarrow G_j$) among goals specify how the satisfaction/denial of the source goal G_i implies in the (partial) satisfaction/denial of the target goal G_j. Influence strengths are modeled using qualitative values: + (weak positive), ++ (strong positive), - (weak negative) and – (strong negative). A partial influence ($G_i \xrightarrow{+/-} G_j$) denotes that the satisfaction of the source goal G_i implies in the partial satisfaction/denial of the target goal G_j (respectively), whereas the full influence ($G_i \xrightarrow{++/--} G_j$) denotes that the satisfaction of the source goal G_i implies in the full satisfaction/denial of the target goal G_j (respectively). Decompositions are intentional as they allow designers to refine goals in a structural manner, whereas influence relationships depict the impacts of achievement of goals on each other, representing the side-effects among goals [179].

Goals and their relations are captured in goal models which may be enriched with *domain assumptions*, *processes* and *situations*. *Domain assumptions* indicate properties that are assumed to be true for some goal to be achieved. For example, "High demand" must be true for the "Increase Sales" goal to be satisfied. if such assumptions are false, then its associated goal is not satisfied. *Processes* can be associated with a particular goal via an *achieves* relation to denote that a process is intended to achieve a goal.

Besides domain assumptions and processes, managers are usually interested in foreseeing other aspects that influence the fulfillment strategic goals during enterprise planning. In that respect, SWOT analysis [40] consists of a useful tool to identify internal and external factors that may impact positively or negatively the achievement of strategic goals. SWOT stands for Strengths (internal and favorable factors), Weaknesses (internal and unfavorable factors), Opportunities (external and favorable factors) and Threats (external and unfavorable factors). BIM propose to model SWOT factors in terms of the concept of *situation*. A situation characterizes a state of affairs (state of the world) in terms of the entities that exist in that state, their properties and interrelations. Favorable situations are represented via positive influence links on goals, whereas unfavorable situations are represented via negative influence links.

The BIM concepts of *goal*, *situation*, *domain assumption* and *process* precisely capture many notions prescribed by the requirements for strategic enterprise architectures from Chapter 2 and therefore, the framework has been selected as the starting point of our strategic enterprise architecture. More specifically, the BIM concept of goal precisely captures the goal concept from the motivational perspective, whereas BIM AND/OR refinements respectively capture refinements and alternative relationships among goals. In its turn, partial relations prescribed by requirements are captured by BIM (partial) influences. BIM situations, their SWOT relations and domain assumptions are also instrumental notions for representing factors that impact goals, either hindering or supporting their achievement. Finally, BIM processes also consist of a useful abstraction to represent behavior in strategic enterprise architectures.

Despite the great match between BIM concepts and their respective counterparts in the requirements for strategic enterprise architectures, the simplistic nature of goals does not allow the differentiation among multiple levels within the goal hierarchy as described in the motivating scenario (Section 2.2, Chapter 2). Therefore, BIM *situation*, *domain assumption* and *process* are directly inhered from the framework, whereas BIM *goal* is further refined with other strategic concepts.

3.2 Business Motivation Model (BMM)

Contrasting with BIM's simplistic ontology for goals, BMM offers a rich vocabulary of goal-related concepts that is used as the starting point for the differentiation among the several goal types.

The *Business Motivation Model (BMM)* [75] is a conceptual specification adopted by OMG for schematizing or structuring the development, communication and management of business plans in enterprises. The specification is structured in terms of four essential concepts: *means*, *ends*, *influencers* and *assessments* that are further refined into other concepts.

Broadly speaking, an *end* is something the business intends to accomplish, whereas a *means* represents something that must be activated or enforced to achieve an end. An *influencer* consists of some factor that has the capability to cause changes in the employment of means or in the achievement of ends, whereas *assessments* correspond to the judgment of how influencers drive the enterprise to articulate its decisions. Means are further refined into *mission*, *strategy* and *tactic*, while ends are refined into *vision*, *goals* and *objectives*.

Table 3.1 introduces definitions for those concepts together with an example of each of concept relative to a Consulting Company.

Concept	Description
Vision (End)	A Vision comprises an overall image of organization's future state. Vision: "Be the premier consulting company in the industry"
Goal (End)	A Goal is an attainable statement about a state of the enterprise to be brought about or sustained through appropriate Means. It indicates what must be satisfied on a continuing basis to effectively attain the Vision. Goal: "To improve customer satisfaction (over the next five years)"
Objective (End)	An Objective is a statement of an attainable, time-targeted and measurable target (explicit criteria to determine satisfaction) that the enterprise seeks to accomplish.The main difference between objectives and goals is that objectives are always time-targeted and measurable, while goals are not so specific. Objective: "By June 30, 2008, create an operational customer call center"
Mission (Mean)	A Mission indicates the ongoing operational activity of the enterprise, describing what the business is or will be on a daily basis. A Mission makes a Vision operative as it indicates the ongoing activity that makes the Vision a reality. Mission: "Provide consulting, outsourcing, and staff augmentation services to companies in North America"
Strategy (Mean)	A Strategy represents how resources, skills or competencies are combined to achieve enterprise Goals, given the environmental constraints and risks. Although strategies tend to be longer-term and broader in scope than tactics, the model does not make a hard distinction between them [75, p.13] and the enterprise must define their own criteria. Strategy: "Implement a Customer Relationship Management System"
Tactic (Mean)	A Tactic represents part of the detailing of Strategies. Tactic: "Call first-time customers personally" implements the strategy "Increase repeat business"
Influencer	Factor that may cause changes in the employment of means or in the achievement of ends. BMM provides three Influencers categories: Internal Influencer (from within the enterprise that can impact the employment of Means), External (from outside the enterprise boundary) and a set of general categories to allow enterprises to define their own set. Among the External Influencers, there are competitors, customers, environment, partners, regulations, suppliers and technology and as Internal Influencers, there are assumptions, corporative values, habits, infrastructure, issues, management prerogatives and resources
Assessment	Correspond to the judgment of how influencers drive the enterprise to articulate its decisions

Business Process	Although the concept of "business process" is present in the BMM specification, it actually belongs to the Business Process Definition Metamodel (BPDM) in which it is defined and associated accordingly with goal-related concepts

Table 3.1: The Description of Means-Ends Concepts from BMM [75]

Although BMM offers a rich ontology of goal-related concepts which can be used for the refinement of BIM *goals*, the definition of BMM concepts lacks formal rigor, as can be observed in Table 3.1. Consequently, practical efforts of refining BIM goals in terms of finer-grained goal distinctions from BMM are hindered by the absence of well-defined, concrete criteria in BMM.

Furthermore, the refinement of the core BIM notions requires not only a conceptualization that provides finer-grained distinctions for goal concepts but also enables a coherent and consistent integration between the motivational and behavioral perspectives. In this context, although the BMM specification foresees associations of its concepts with behavioral elements, in practice, business process are references to elements that are defined and maintained outside the scope of an enterprise's Business Motivation Model (in the case of business processes, such elements are defined in the Business Process Definition meta-model (BPDM)). Consequently, to overcome the lack of concrete criteria in BMM and to acquire a common semantic foundation for integrating motivational and behavioral perspectives, we referred to conceptualization provided by Management Sciences Theories in this thesis.

3.3 Management Theories

Our literature review in Management literature started by reviewing the historical roots of business strategy and competition [63, 138] and resulted into valuable insights about the existent frameworks and influential authors in the area, such as Henry Mintzberg [135, 133, 134], Michael Porter [153, 34, 97, 156, 157], Peter Drucker [153], Kenneth Andrews [135], Gary Hamel [172], Prahalad [172], Birger Wernerfeld [172], Edith Penrose [172], among others. In principle, we thought this approach would be fruitful to gain an overview of the area and acquire the required conceptualization to distinguish among different types of motivational and behavioral concepts.

This initial study enabled us to acquire frameworks and methodologies that discuss different ways of analyzing a given company and to generate its corresponding strategy but lacked the actual concepts to promote conceptual integration between motivational and behavioral perspectives. Despite the usefulness of such literature, here we only include the management conceptualization required for the development of our strategic enterprise architecture approach and therefore, this literature is not included.

This lack of actual concepts led us to turn our attention to undergraduate textbooks in the areas of Management, Strategic Management and Operations Management with the intent of understanding by which means organization's goals are connected with their corresponding operations. In these textbooks, Management literature conceptualizes an organization as a system composed of a set of interrelated parts that jointly work to achieve stated goals [153, 51, 34]. This system typically receives external influences, such as competitors, industry segment needs, regulative pressures or alliances with other cooperating companies. In order to operate successfully, the management process involves setting up goals and their corresponding resource requirements and operations by exercising five management functions (planning, organizing, staffing, leading and controlling) [153, 51].

Among those management functions, the planning function provides conceptualization that allows us to provide support for our requirements from Chapter 2. Basically, the planning function refers to the process of planning the goals to be achieved by the company, followed by the corresponding planning of operations to achieve such goals. Therefore, this section summarizes a number of theories in Strategic Management, Management and Operations Management that particularly focus on the description of the planning function in companies. Such theories are used as foundations for the development of our strategic enterprise architecture approach, by providing us with the finer-grained distinctions for different types of goals and by providing conceptualization for connecting motivational and behavioral perspectives. In the remainder of this section, the goal planning activity is described in Section 3.3.1, whereas operations planning is described in Section 3.3.2.

3.3.1 Goal Planning in Management Sciences

In order to describe the concepts required for performing goals planning in companies, we have analyzed multiple examples of strategic goals and goal hierarchies from Strategic Management and Management literature, such as [172, 99, 97, 153]. Given the divergence of definitions and examples, we first analyze each concept together with the semantics of multiple examples of goals and operations in order to provide a consolidated definition for each concept. We select the example [34, p.222] of the integrated goal hierarchy of metal manufacturing company from Management literature (depicted in Figure 3.1). This example was first introduced as our motivating scenario for strategic enterprise architectures (Chapters 1 and 2) and is used throughout this section and next chapters (Chapters 5, 6 and 7) to illustrate our definitions, strategic planning approach and to demonstrate how goals are represented in our framework.

In order to perform goal planning, organizations distinguish among three levels of decision-making, *Strategic*, *Tactical* and *Operational* [153, 34]. Inside each level of abstraction, managers have to specify strategic, tactical and operational goals that focus on different enterprise concerns and must be achieved within distinct time frames.

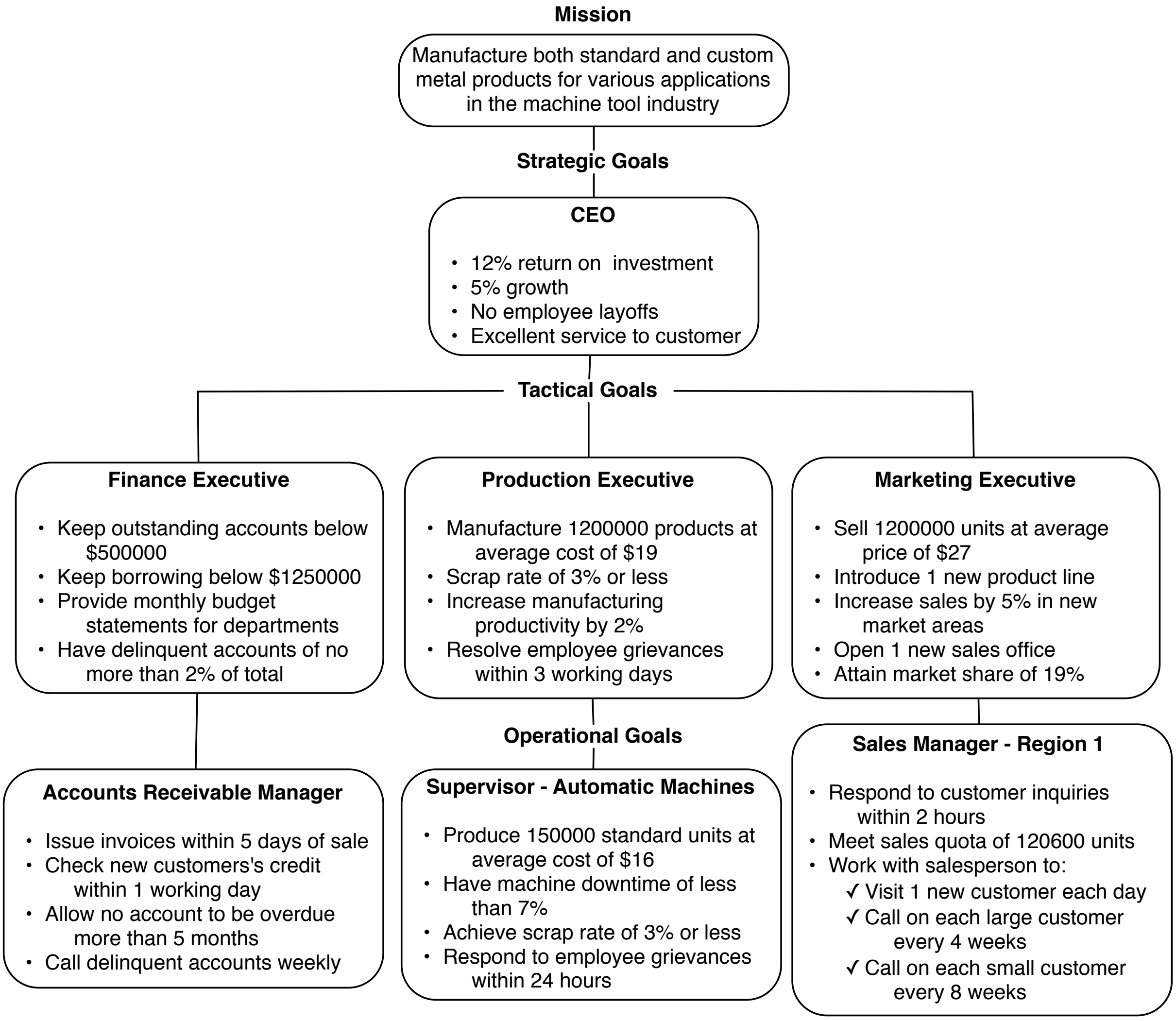


Figure 3.1: Goal Hierarchy Extracted from [34, p.222]

Strategic Level. This level is concerned with the long-term direction of the organization, determining the range of businesses (scope) in which the organization operates and how it copes with external influences [187, 97, 172]. Further, strategic planning tends to be long-term and may define organizational action steps between two and five years [34]. Within the Strategic Layer, the four important concepts are *Mission*, *Vision*, *Strategic Goals* and *Strategy*:

Mission [97, 172, 99, 34]. A mission defines a formal expression of an organization's purpose, i.e., the reason why the organization exists. Management literature also mentions that organizations exist to aggregate some value [172]. On the basis of both views, we interpret that organizations exist to aggregate some sort of value to the external world. An analysis of the mission statement from Figure 3.1 reveals that the organization exists to provide some value by producing standard and custom metal products ("Manufacture both standard and metal products").

Vision [172, 99]. Comprises a description of a desired future state of the company, meant to close the gap between the current reality and a potential future. An example of vision could be: "To be the market leader of standard and custom metal products in the machine tool industry" [34] (not shown in Figure 3.1).

Strategic Goals [97, 172]. Represent concrete outcomes or status to be achieved to measure whether mission statements are being achieved [99, 153]. They are directional as they guide the strategy towards achieving the organization's mission [172, 99]. By analyzing the definition and a number of examples (including Figure 3.1), we interpret strategic goals as statements about external and internal company's conditions that reflect company's strategy to succeed in business. Other examples of strategic goals that result of an internal evaluation of business environment are "Improve market share from 15 to 20% over the next three years" and "Increase gross margin on current sales" [34].

Strategy [187, 97, 172]. A Strategy can be defined as a course of action created to achieve an organization's strategic goals [153]. The purpose of strategy is to provide a framework crafted to guide decision-making about which actions should be executed to adjust the internal company's context (i.e., how to effectively deploy organizational resources) according to the corresponding external context (i.e. competitive and non-market pressures) in order to satisfy organization's strategic goals [187, 97, 172].

Corporate and Business Unit Strategy [157, 172, 153, 97]. Within the strategic level, literature in Strategic Management further distinguishes between two levels of strategy for a diversified company [157, 172, 153], i.e., a Corporate Strategy and Business Unit Strategy. A Strategic Business Unit (SBU) can be defined as a part of an organization for which there is a distinct external market for goods or services [172, 97, 153]. Examples of companies that operate in a single SBU includes Coca-cola as a purveyor of coke and Dell as a computer company [172]. Unlike Coca-Cola and Dell, Hewlett Packard has distinct businesses, including Unix server, laser printers and inkjet printers businesses [172], while Yahoo!'s business units include Yahoo! Photos and Yahoo! Music [97].

Therefore, a Corporate Level Strategy is concerned with the overall purpose and scope of an organization and how value will be added to the different parts (business units) of the organization. This could include issues of geographical coverage, diversity of products/services or business units, and how resources are to be allocated between the different parts of the organization [97]. In general, the corporate-level strategy may also include expectations of owners (shareholders and stock market) which may be an explicit or implicit "mission statement" that reflects such expectations [97]. In its turn, a Business Unit Strategy concerns how the various businesses included in the corporate strategy should compete in their particular markets [97]. This distinction arises due to the fact that each business unit has different assets and competitors and managers have to business unit strategies have to create competitive advantage. Usually, as the majority of companies operate in single businesses, there is a match between the corporate and the business unit strategies.

Tactical Level. While the *Strategic Level* refers to the direction of the business, providing a framework for guiding decisions that lead to a desired future state of the organization (i.e., the strategy), the tactical level involves the planning of the actual steps required to implement such strategy by the company's functional areas or organizational units [153, 163]. In Management, a tactic is a concept that appears as a product acquired from the planning at the tactical level. The concept of tactics steams from the military domain in which a military tactic is the concept of organizing a military force (troops), combining and using weapons and military units to engage and defeat an enemy in battle [34]. In this context, a military tactic is concerned about the role developed by weapon systems and troops in delivering value to the country's overall strategic plan [34, p.218]. Similarly, in business, every strategic plan requires a series of related tactical plans to achieve the strategic goals [153, p.159]. In companies, tactical plans are elaborated by middle managers, typically having shorter time horizons (usually from one to three years) and narrower scopes than strategic plans [153, p.159].

Tactical Goals (or Objectives) [34, 163, 153]. Tactical goals define the outcomes to be achieved by major divisions and departments in the context of strategic goals. An analysis of examples in Management literature reveals that tactical goals focus either on the specification of responsibilities of functional areas (or departments) in the context of overall achievement of strategic goals or on the specification of tactics to achieve strategic goals. Figure 3.1 takes the first view by segmenting strategic goals into tactical goals specified by responsibilities of functional areas (Finance, Production and Marketing), e.g. "Manufacture 1200000 products at an average cost of $19" from Operations). In [153], marketing responsibilities usually include tactical decisions like advertisements (e.g. "Execute promotions for golf apparel with Tiger Woods" from Nike or "Diversify sales channels" from Celestial Seasonings [153, p.183]), while the other functional areas also have their responsibilities accordingly. In contrast, in [172], the definition of tactics takes the second interpretation by specifying the Border Inc.'s tactical goal ("Open 20 new stores by the end of the planning period") as a tactics for its corresponding strategic goal ("Borders will be the leading retail distribution outlet for books in the US").

Operational Level. Concerns the planning and management of daily operations responsible for delivering products and services on behalf of the company [153]. Operations implement the tactical initiatives that are elaborated for supporting organization's strategy. Such tactical initiatives are then scheduled and eventually emerge as the set of organization's operation specifications [133]. The operational plan is the first-line manager's tool for executing daily, weekly and monthly activities [153], specifying how to accomplish the organization's operational goals.

Operational Goals [153, 34]. Operational goals consist of quantitative, measurable and daily results expected from departments, workgroups and individuals within the organization. In Figure 3.1, operational goals are specified according to different roles. An interpretation of definition and examples reveals a mixture of definitions for operational and tactical goals in terms of the responsibility for their achievement. Although tactical goals are assigned to departments and operational goals are assigned to roles in Figure 3.1, most of the approaches [153, 34] mention that both types of goals should be achieved by departments. Further analysis also reveals that both types of goals can be scheduled (e.g. "Resolve employee grievances within 3 working days" and "Respond to employee grievances within 24h"). As tactical and operational goals in Management literature present similar conceptual characteristics, it is also not clear how operational goals are connected with their respective tactical goals, i.e., how the achievement of operational goals entail the achievement of tactical goals. Finally, there is also a lack of clear connection between operational goals with their respective operations and the activities that compose such operations.

3.3.2 Operations Planning in Management Sciences

In order to plan the operations that achieve the enterprise's goals hierarchy, the company is usually seen as a transformation entity that receives inputs and generates outputs by means of operations. In this context, the concept of *Operation* can be defined as:

Operation [163, 123]. An activity conducted in order to transform a set of inputs into useful outputs using some sort of transformation process. An example of Operation for the company under consideration could be an "Assemble standard metal products".

For a given operations system to be considered successful, it must support the organization to achieve competitive advantage by aggregating value to final customers during the transformation process [163, 123]. In this context, in order to systematically devise which activities aggregate value and which do not, Value Added Chains (VACs), introduced in the work of Michael Porter [156] are helpful concepts for managers to gain such understanding:

Value Added Chain (VACs) [97, p.110] [156]. Regarding a specific enterprise E, its Value Added Chain (VAC) comprehends all the activities the enterprise conducts to create a product or deliver a service [97, p.110]. A VAC has a rich internal structure that is represented by different categories of activities (Figure 3.3). These high-level categories (e.g., human resources management) can be broken down into smaller functional units, spanning a hierarchical structure of business activities of different granularities.

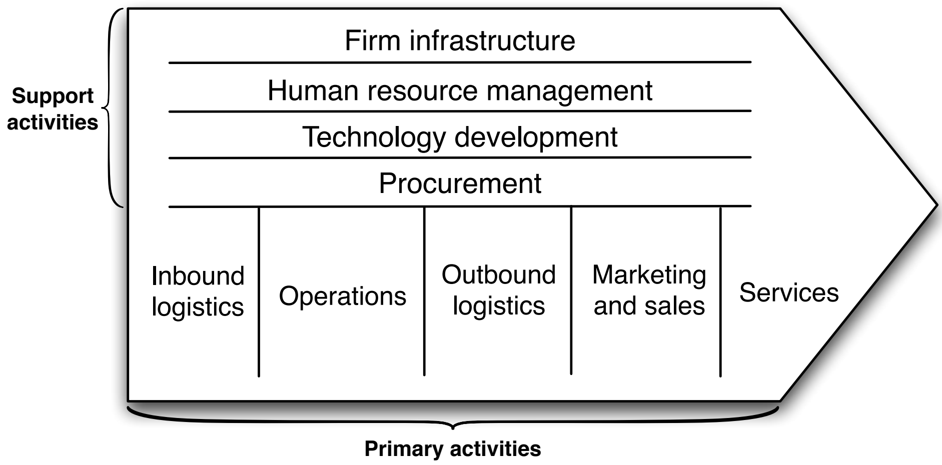


Figure 3.2: VAC from Enterprise E Extracted from [97, p.110]

According to Porter, activities conducted by a given company can be categorized into primary and support activities, as shown in Figure 3.3 [135, p.104] [97, p.110]. Primary activities are directly involved in the flow of product or service to the customer, directly contributing to the competitive advantage of the company. It includes inbound logistics (receiving, storing, etc.), operations (or transformation like machining, packaging and assembly), outbound logistics (order processing, physical distribution, etc.), marketing and sales, and service (installation, repair, etc.). Support activities exist to support primary activities, providing the environment in which they can be effectively executed. They include procurement, technology development, human resources and firm's infrastructure (including finance, accounting, general management, etc.).

3.4 Constrained Goal Models (CGM) Formalism

Once we have acquired the required conceptualization from Management Sciences, requirements from Chapter 2 also require us to provide automated support for strategic enterprise architectures. Consequently, the Constrained Goal Models (CGM) approach is the formalism used in this

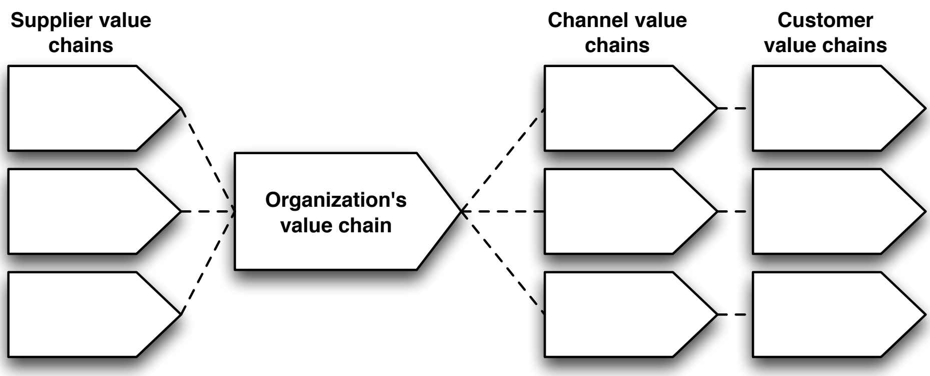


Figure 3.3: Value Network Extracted from [97, p.114]

thesis to provide a formal semantics for strategic enterprise architecture models and reason with such models. In particular, the CGM formalism is used for the generation, evaluation and selection of strategic alternatives during the planning process described in Chapter 2.

The Constrained Goal Models (CGM) formalism [142] consists of a modeling and automated-reasoning suite for decomposing stakeholders' goals as an AND/OR graph of alternative refinements and relations, and for automatically finding the optimum set of sub-goals according to a combination of objective functions.

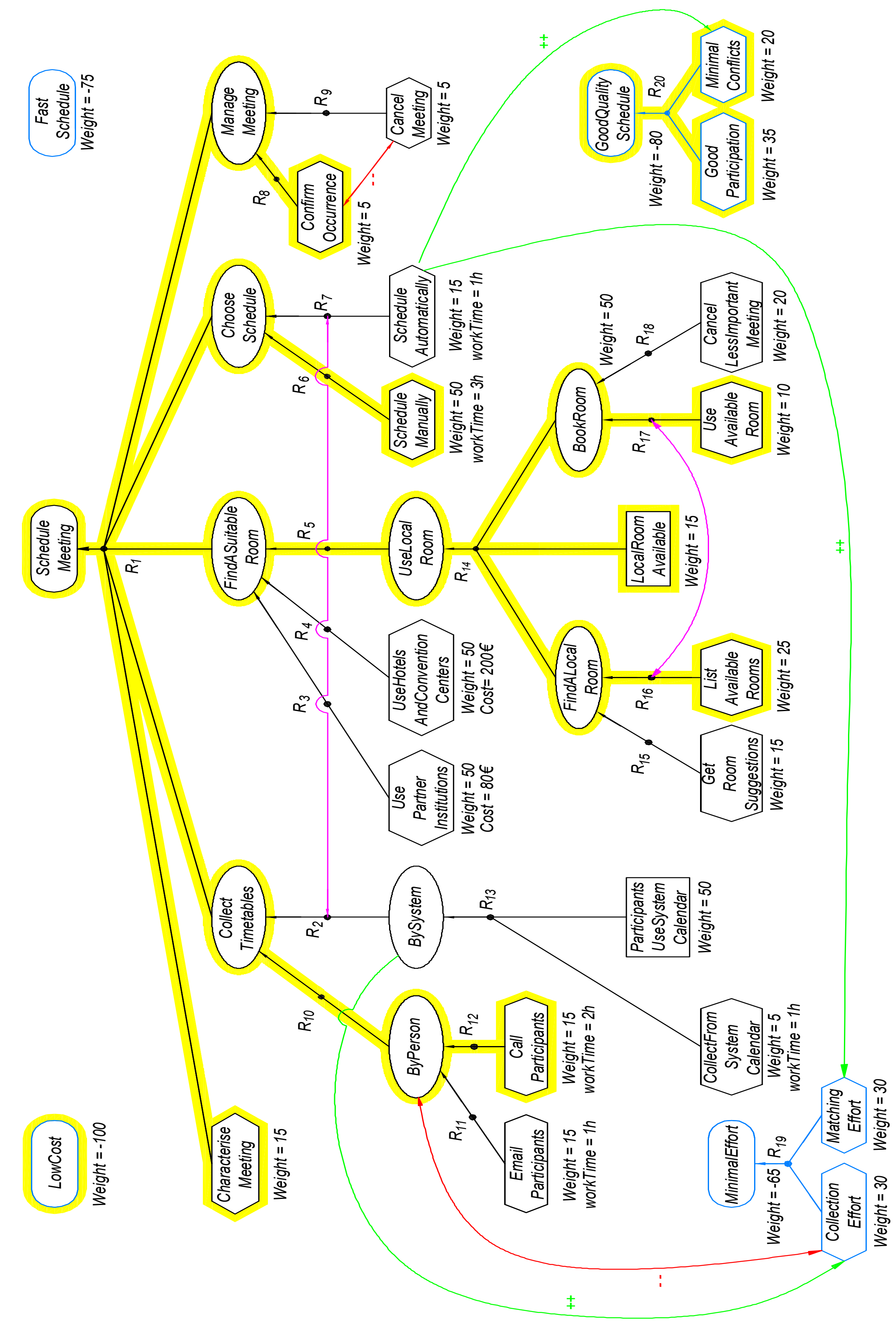


Figure 3.4: An Example of Constrained Goal Model From [142]

Regarding CGM modeling functionalities, goal and their relations are represented in terms of *CGM nodes*, *CGM refinements* and *CGM relation edges*. Starting with CGM nodes, CGM considers stakeholders' *requirements* as a desired state of affairs that the system has to achieve. Such *requirements (root goals)* need to be progressively refined into *intermediate goals* until producing *tasks* (actionable/leaf goals) that can be executed. *Domain assumptions* can be attached to express propositions about the domain that need to be true for a goal to be achieved, whereas *users' assertions* enable the specification of optional requirements that would be interesting to be fulfilled in the case they are not conflicting with other requirements.

Figure 3.4 exemplifies the modeling constructs from CGM. Among *CGG nodes*, $ScheduleMeeting$ consists of a *root goal* that can be refined into *intermediate goals* (e.g. $ChooseSchedule$ and $ManageMeeting$) until reaching tasks (e.g. $ScheduleManually$). *Domain assumptions* are represented as rectangles (e.g. $ParticipantsUseSystemCalendar$), while *user's assertions* are directly set on the tool by means of colors.

Regarding CGM relation edges, *goal refinements* allow one to decompose each element (non-leaf goal or domain assumption) into a conjunction of sub-elements necessary to achieve it. For example, in Figure 3.4, the *GoodQualitySchedule* goal is refined into two sub-goals (*GoodParticipation* and $MinimalConflict$) by means of a goal refinement (GoodParticipation, MinimalConflict) $\xrightarrow{R20}$ GoodQualitySchedule. Furthermore, by adding a number of *goal refinements*, one can also represent the alternatives of how to achieve an element.

Besides *goal refinements*, other relations can be expressed among elements by means of *relation edges.* In CGM, relations edges can be categorized into the following types:

- **Contribution edges ($\mathbf{E}_i \xrightarrow{++} \mathbf{E}_j$).** Express that if the source element

E_i is satisfied, then the target element E_j must be satisfied (but not vice versa);

- **Conflict edges ($\mathbf{E}_i \dashleftarrow\dashrightarrow \mathbf{E}_j$).** Express that the two elements E_i and E_j cannot be both satisfied simultaneously;
- **Refinement bindings ($\mathbf{R}_i \longleftrightarrow \mathbf{R}_j$).** Intuitively, denote that two refinements are bound, i.e., if two elements E_i and E_j are satisfied, then E_i must be refined by R_i and E_j refined by R_j, respectively;

In Figure 3.4, contributions, conflicts edges and refinement bindings are respectively represented as green, red and purple edges among goals.

Besides graphical constraints expressed as relation edges, goal models can be enriched with arbitrary *prerequisite logic formulas* (constraints) ϕ_G^+ (resp. ϕ_G^-), indicating that ϕ_G^+ (resp. ϕ_G^-) must be satisfied when G is satisfied (resp. denied). Further, *numerical constraints* on elements and refinements can be also used to express user-defined constraints among nodes. For example, if one estimates that goal $G_{UsePartnerInstitutions}$ costs 80 € and goal $G_{UseHotelsAndConventionCenters}$ costs 200 €, one can use such information to write constraints in terms of costs among goals $G_{UsePartnerInstitutions}$ and $G_{UseHotelsAndConventionCenters}$. In order to do that, s/he first creates a global numerical variable called "cost" and the system automatically generates a numerical attribute $cost_E$ for each element E (whose default value is set to 0) and a global default constraint cost $= \sum_{E \in \varepsilon} cost_E$. Then, for each element E of interest, one can set the value for $cost_E$ in case E is satisfied (or denied) and also manipulate the global default constraint accordingly. In this case, we set $cost_{G_{UsePartnerInstitutions}} := 80$ and $cost_{G_{UseHotelsAndConventionCenters}} := 200$ and the system automatically adds the following constraints:

$$\begin{aligned} \phi^+_{G_{UsePartnerInstitutions}} &= ... \wedge (cost_{G_{UsePartnerInstitutions}} = 80) \\ \phi^+_{G_{UseHotelsAndConventionCenters}} &= ... \wedge (cost_{G_{UseHotelsAndConventionCenters}} = 200) \end{aligned} \tag{3.1}$$

Finally, a singular or combination of *objective functions* to optimize (i.e. maximize or minimize) can also be specified as functions of boolean and numerical variables. For example, a user might be interested in $maximize(cost)$ in the example of Figure 3.4.

Regarding CGM reasoning functionalities, once goals are modeled and objective functions are specified accordingly, stakeholders may request the CGM solver to automatically generate *realization(s)*. Realizations (depicted in yellow in Figure 3.4) correspond to one of the alternative ways of refining the mandatory requirements (and potentially some of the optional ones) in compliance with the user's assertions and constraints. The CGM solver then generates realizations that optimize one single objective or a lexicographically ordered combination of objectives. For example, by defining boolean or numerical variables (e.g. $cost$, $workingTime$, $TotalCost$) and using one objective function ($minimize(cost)$), the solver finds the realizations with minimal cost among all realizations. In a lexicographically ordered combination of objectives, the user defines a list of ordered objective functions (e.g. $minimize(cost)$ AND $minimize$ ($working\ Time$)) and the solver first finds a realization with minimum $cost$. Among all realizations with minimum cost, the solver then finds a realization with minimum $workingTime$.

CGMs is supported by the CGM-tool [129], which is implemented as a standalone java application based on the Eclipse RCP engine. CGM-tool uses a state-of-the-art *Optimization Modulo Theories* (OMT) solver [180, 181] as backend automated-reasoning engine. The tool provides functionalities to create CGM models as graphical diagrams and to perform different forms of reasoning, including interactive search for realizations.

3.5 Commitments and Protocols

In our motivating scenario (Chapter 2), once strategic alternatives are selected, the planning process requires them to be implemented accordingly by means of a number of processes. As the requirements for strategic enterprise architecture demand processes to be captured in terms of their social dimension (expressiveness in behavioral perspective requirement (R1.2)), we need to find primitives for capturing processes in terms of their social dimension.

In the context of this thesis, we rely on the concept of *commitments*. Commitments have been studied as a fundamental social primitive in a number of areas, including social sciences [13], computer-supported collaborative work [58] and multi-agent systems [184]. They are social abstractions as they carry a contractual nature with social meaning.

Formally, a (social) commitment [184] c(x,y,p,q) is a promise with contractual validity made by an agent x (debtor) (the agent who is committed) to another agent y (creditor) (the agent who receives the commitment) that, if proposition p is brought about (antecedent), then proposition q will be brought about (consequent). If p is true ($\top$), the commitment is unconditional; otherwise, it is conditional.

Commitments change when their interacting agents exchange messages. Messages constitute *commitment operations*: (i) *creation*: the debtor commits to the creditor that the consequent will be brought about; (ii) *cancellation*: the debtor cancels an existing commitment; (iii) *release*: the creditor releases the debtor from a previous commitment; (iv) *delegation*: the debtor delegates the commitment to a third party; and (v) *assignment*: the creditor assigns its credit to another actor.

Moreover, *declare* operations allow an agent to inform another that a certain proposition has changed its truth value (e.g., the book has been sent). *Declare* operations enable the change of commitment state. A commitment is *detached* when the debtor is informed (through a *declare*) that the antecedent has been brought about, and the commitment becomes unconditional. A commitment is discharged/fulfilled, when the creditor is informed that the consequent has been brought about.

We adopt a version of commitments [124] where antecedent and consequent are expressed in propositional logic extended with a temporal precedence operator "·". Thus, $(\mathsf{p} \wedge \mathsf{q}) \cdot \mathsf{r}$ means that p and q occur (in any order) before r occurs.

Commitments can be used to define an interaction in a *protocol* between roles [31, 42]. For instance, given roles R_1 and R_2, a protocol may include a commitment such as $\mathsf{C}(\mathsf{R}_1, \mathsf{R}_2, \mathsf{P}, \mathsf{Q})$. Thus, an agent playing role R_1 is expected to create instances of this commitment to some agent playing R_2. The propositions in such commitment will be instantiated too: if P is "Book sent", a possible instance p is "copy 123 of book Dracula sent".

3.6 Knowledge Acquisition in autOmated Specification (KAOS) Approach

Besides capturing the social perspective of processes during the implementation step of the planning process, methodological support for performing the transition from motivational to behavioral perspective is also required (traceability in methodological consistency between motivational and behavioral perspectives requirement (R2.2) requirement). In this thesis, methodological support to promote such transition between perspectives is acquired from refinement goal patterns [38] and operationalization process [118, 10] within the KAOS framework.

The Knowledge Acquisition in autOmated Specification (KAOS) approach [37, 118] consists of a Goal-Oriented Requirements Engineering

(GORE) methodology that supports the requirements engineering process by means of formal analysis techniques. In KAOS, the content of goals is represented as admissible system states (behaviors) formalized using Linear Temporal Logics (LTL). Further, the language also introduces the notion of goal patterns (achieve/cease, maintain/avoid and optimize). Goal patterns allow the requirements engineer to specify a pattern to be checked in order to evaluate goal satisfaction in terms of possible behaviors of the system.

In order to exemplify the specification of goals in KAOS, we take the management of London Ambulance System as an example. In this system, the top system goal [119] can be written as:

Goal$Achieve[AmbulanceIntervention]$

Informal Definition. For every urgent call reporting an incident, there should be an ambulance at the incident scene within 14 minutes after receiving the first call

FormalDef $\forall inc : Incident$

$$Reported(inc) \implies \Diamond_{\leq 14min}(\exists amb : Ambulance) Intervention(amb, inc) \quad (3.2)$$

This specification fragment introduces the $AmbulanceIntervention$ goal together with its natural language definition. The *Achieve* keyword declares a goal pattern, in this case, it states that some target property must eventually hold in the future ($P \implies \Diamond Q$). The formal counterpart may be specified optionally in LTL [37, 119]. Other goal patterns might have been specified, such as *Cease* (to disallow achievement "some time in the future", $P \implies \Diamond \neg Q$), *Maintain* (to denote that a property must hold "at all times in the future", $P \implies \Box Q$), *Avoid* (to prescribe that a property must not hold "at all times in the future", $P \implies \Box \neg Q$) and *Optimize* (to denote that some property must be maximized/minimized) [37, 118].

In KAOS, goals can be refined in terms of AND/OR relations and enriched with domain properties [118]. The framework also provides goal refinement patterns that document most common goal refinements, thus supporting the refinement process and the identification of missing goals in refinements [38]. For example, the refinement pattern (RP1) [38] in Figure 3.5 states the top goal $P \implies Q$ must be AND-refined into other three sub-goals.

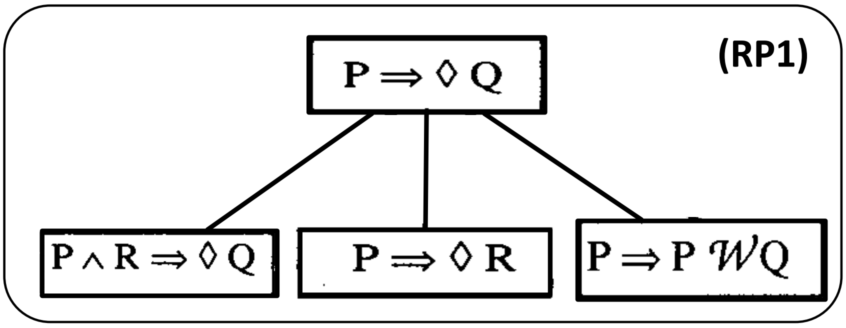


Figure 3.5: Example of KAOS Refinement Patterns Extracted From [38]

Once refined accordingly, goals reach a level of abstraction in which they can be operationalized by means of operations. In operation models, operations represent the various services to be provided by software agents to business stakeholders. The operationalization process thus consists of assigning operations for the satisfaction of goals by prescribing pre-, post-conditions and triggers on operations in order to fulfill goal specifications [118].

For example, in order to ensure the achievement of the *Ambulance Intervention* goal, an operation needs to be select and pre-, post-conditions and triggers must be assigned on this operation. Selecting the CallAmbulance operation (Equation 3.3) and adding a trigger on it, we have the *AmbulanceIntervention* goal operationalized into Equation 3.4:

Operation CallAmbulance

Input a: Ambulance; **Output** a: Ambulance/AmbulanceRequested

DomPre a.AmbulanceRequested ≠ '*Requested*'

DomPost a.AmbulanceRequested = '*Requested*'

(3.3)

Operation CallAmbulance

Input a: Ambulance; **Output** a: Ambulance/AmbulanceRequested

DomPre a.AmbulanceRequested ≠ '*Requested*'

DomPost a.AmbulanceRequested = '*Requested*'

ReqTrig for AmbulanceIntervention: i.Reported = 'yes'

(3.4)

where i in Equation 3.4 refers to the reporting an incident.

Regarding KAOS reasoning techniques [119, 86], Letier et al. [119] first enrich the KAOS goal model with a probabilistic layer in which degrees of partial goal satisfaction are captured by means of quality variables, objective functions and impacts of alternative system designs on goals. Subsequently, authors [86] present automated techniques for evaluating a number of alternative systems designs in order to find the optimal ones. In this context, the framework allows the automatic quantification of the level of satisfaction of top system goals depending on alternative system designs, thus allowing stakeholders to manually find the optimal ones.

3.7 Discussion

In order to justify the selection of frameworks used as the baseline of this thesis, this section analyzes in which extent such frameworks achieve the requirements for strategic enterprise architectures from Chapter 2. Table 3.2 summarizes this assessment.

Regarding the conventions for Table 3.2, a ✓sign indicates that the proposal fully addresses the requirement, a - sign indicates that the proposal does not address the requirement and a ∼ sign indicates that the proposal partially addresses the requirement.

	R1.Expressiveness											**R2.Trac.**	
	R1.1 Motivational Domain								**R1.2 Beh. Dom.**				
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**2**	**3**	**1.1**	**1.2**	**1.3**	**2.1**	**2.2**
BIM approach [89]	✓	∼	-	-	✓	-	✓	✓	-	-	-	∼	∼
BMM approach [75]	✓	-	-	-	∼	∼	-	∼	-	-	-	-	-
Management Theories	✓	✓	✓	∼	✓	∼	∼	∼	-	✓	✓	∼	∼
Commitments and Protocols [124, 184, 31, 42]	-	-	-	-	-	-	-	-	✓	-	-	-	-
CGM formalism [142]	✓	-	-	-	-	-	✓	✓	-	-	-	∼	∼
KAOS approach [37, 118]	✓	✓	✓	✓	-	-	∼	∼				∼	∼

Table 3.2: Assessment of baseline approaches against the requirements from Chapter 2

BIM Framework. The BIM framework presents a great match with the concepts required by strategic enterprise architectures, as can be noticed in Table 3.2. Consequently, it was considered as our starting point among the Conceptual Modeling approaches. In particular, we have used the concepts of *goal*, *relations among goals*, *situation*, *domain assumption* and *process* for the development of motivational perspective of strategic enterprise architectures. However, among BIM shortcomings, we include no support for the specification of multiple levels of abstraction among goals (denoted by a - sign in sub-requirement 1.6 from motivational perspective). Furthermore, although a process “achieves” a goal, the framework refrains from representing process-control flow (denoted by a $\sim$ sign in both sub-requirements from traceability).

BMM. The simplistic notion of BIM goals led us to adopt the BMM specification due to its rich conceptualization of distinct goal categories. However, although this refined goal ontology can provide us initial ideas, its lack of formal rigor hinders the practical application in enterprise modeling efforts. Furthermore, BMM also presents limited and informal support for integration of goals and behavioral concepts by means of an association with elements external to the specification (this characteristic is demonstrated in Table 3.2 by a - sign for all behavioral and traceability sub-requirements). Therefore, we turned our attention to Management Theories, striving to find clear criteria for differentiating among goals of different shades and for clear conceptual integration of goals with behavioral concepts.

Management Theories. In Management Theories, our initial objective was to find clear differentiation among goals and conceptual integration between motivational and behavioral perspectives. However, we faced similar issues of unclear distinctions as in BMM, although Management Theories provided richer conceptualization. In this context, Management literature contains a rich set of conceptual tools like assumptions, scenario analysis, goal and operations planning. Nevertheless, the field does not use models with precise semantics and for this reason, many requirements are only met partially in Table 3.2. For example, the field recognizes scenario analysis, but no modeling concepts are employed for performing such kind of analysis (denoted by a $\sim$ sign at sub-requirements 3 from motivational perspective). Tactical and operational goals are not clearly distinguishable from each other (denoted by a $\sim$ sign at sub-requirements 1.6 and 2 from motivational perspective). Further, precise conceptual relations between operational goals and the set of company's operations (VACs) are also missing (denoted by a $\sim$ sign at sub-requirements from traceability requirement). This lack of clear semantics in Management Theories then motivated the contributions of this thesis with the development of motivational and behavioral perspectives of strategic enterprise architectures.

Despite this main shortcoming of Management literature, some conceptualization was still considered very useful to achieve our final intent. In particular, we have used this literature for providing finer-grained distinctions for goal concepts (using the owner (1.2), time frame (1.3), targets (1.5) and multiple levels (1.6) sub-requirements from motivational perspective) and for the development of behavioral perspective by integrating goals with the overall chain of company's operations (VACs)(sub-requirement 1.3 of behavioral perspective).

While the usage of Management Literature intended to provide a foundation to distinguish among different types of motivational and behavioral concepts, an attentive reader may argue that foundational ontologies might be considered as an alternative approach towards achieving such endeavor, as they are largely recognized as a foundation to adequately integrate the concepts of different languages, defining these concepts in an unambiguous way [144]. Such ontologies could then be used to precisely characterize the concepts of motivational and behavioral perspectives, thus enabling a subsequent integration between both perspectives. Although this is a valid argument, enterprise ontologies are not used in this thesis due to the absence of a unique ontology that addresses all the semantic distinctions required in this work [24]. In particular, regarding the usage of foundational ontologies for the integration of motivational and behavioral perspectives of the enterprise, we refer the interested reader to our previous efforts [24, 2, 30, 26] for a deeper discussion.

In order to explain in more specific terms how BIM, BMM and Management Theories have been used in this thesis, we have interpreted the semantics of definitions and examples of each concept found on them, which allowed us to find overlaps and gaps in the conceptualization provided by the three aforementioned proposals. Table 3.3 summarizes the respective correspondences among concepts from BIM, BMM and Management Sciences. Such overlaps and gaps have been used as input in our framework to promote a consistent integration of all concepts in the fourth column of Table 3.3.

<table>
<tr><th colspan="4">Development of motivational and behavioral perspectives</th></tr>
<tr><th>Manag. Sciences (Section 3.3)</th><th>BIM (Section 3.1)</th><th>BMM (Section 3.2)</th><th>SIENA framework (Chapter 5)</th></tr>
<tr><td>Mission, Vision</td><td>-</td><td>Mission, Vision</td><td>Mission, Vision</td></tr>
<tr><td>Strategic Goal</td><td rowspan="3">Goal</td><td>Goal</td><td>Strategic Goal</td></tr>
<tr><td>Tactical Goal</td><td>Objective, Strategy, Tactics</td><td>Tactical Goal</td></tr>
<tr><td>Operational Goal</td><td>-</td><td>Operational Goal</td></tr>
<tr><td>-</td><td>Goal Refinements and Influences</td><td>-</td><td>Goal Refinements and Influences</td></tr>
<tr><td>Operation</td><td>Process</td><td>-</td><td>Operation</td></tr>
<tr><td>-</td><td>-</td><td>-</td><td>Business Process</td></tr>
<tr><td>-</td><td>Domain Assumption</td><td>-</td><td>Domain Assumption</td></tr>
<tr><td>-</td><td>Situation</td><td>Influencers</td><td>Situation</td></tr>
</table>

Table 3.3: Summary of Concepts from Literature together with Concepts from Our Framework

Commitments and Protocols. Commitments and protocols have been used in this thesis in order to specify processes in social terms as prescribed by the expressiveness in behavioral perspective requirement (R1.2) from Chapter 2.

CGM. Given our final interest in automatically reasoning with strategic enterprise architectures (requirement R3), we needed to either assign our own formal semantics for strategic enterprise architecture models or find existing languages with already well-established semantics for reasoning (sub-requirement 3.1).

As we opted for finding an existing language with well-established semantics, the next step consists in deciding which specific modeling language should be selected. In this case, given our requirement of supporting the several steps of the planning process (sub-requirement 3.2), we have to find a language that meets most of its sub-requirements. Therefore, we are interested in a language that can reason with goals (R3.2.1 sub-requirement) and environmental factors (R3.2.2 sub-requirement) in supporting evaluation and selection of best strategic alternatives (R3.2.3 sub-requirement) and supporting the implementation of strategic alternatives (R3.2.4 sub-requirement).

In this context, the CGM approach has configured as a suitable candidate towards such endeavor due to two reasons. First, the CGM approach allows the representation of complex relations among goals like contributions, conflicts and refinement bindings among goals (denoted by a ✓sign at sub-requirement 2 from motivational perspective). Further, the approach also allows the representation of domain assumptions that may capture the environmental factors that impact the achievement of goals (denoted by a ✓sign at sub-requirement 3 from motivational perspective). Both types of primitives open up the possibility of performing complex reasoning with goal models. Second, the CGM approach allows the generation of realizations (alternative ways of achieving goals), thus consisting in an appropriate language for the generation of multiple strategic alternatives (step 3) in the planning process.

KAOS Approach. The KAOS approach consists in an expressive approach for goal modeling, evidenced by the achievement of sub-requirements 1.1, 1.2, 1.3 and 1.4 from the motivational perspective. However, its focus on software engineering leads to no support for multiple levels of abstraction (sub-requirement 1.6), only partial support for complex relationships among goals (sub-requirement 2) and environmental factors (sub-requirement 3).

Due to its advanced goal primitives and methodological support, KAOS refinement patterns and operationalization process have been used in this thesis to provide methodological support. This methodological support concerns the transition from motivational to behavioral perspective during the implementation of strategic alternatives in the enterprise planning process.

Chapter 4

Related Work

Chapter 3 reviewed a number of approaches that are used as foundations for the work developed in this thesis. As investigation in motivation and behavioral modeling has a long tradition in a number of research areas, this chapter starts by reviewing the support provided for the representation of motivational concepts (i.e. goals of different shades like mission, vision, strategic goals, etc.) and behavioral concepts (i.e., operations, business processes, business process relations, actions, tasks, activities) in a number of areas of Computer Science. More specifically, Section 4.2 describes approaches that incorporate the representation of motivational concepts, whereas Section 4.3 covers proposals that address behavioral representation. Section 4.4 presents approaches that incorporate both motivational and behavioral modeling primitives. Subsequently, Section 4.5 assesses in which extent such approaches meet the requirements for strategic enterprise architectures defined in Chapter 2. As our work uses Management Sciences as the starting point, a number of theories in Management Science are also related to our approach. These theories are presented in Section 4.6. Finally, Section 4.7 draws general conclusions about the gaps within the state of the art, thus motivating and contextualizing our contributions. This chapter is based on the published research papers [91, 90, 88, 87].

4.1 Introduction

Information modeling "is concerned with the construction of computer-based symbol structures which model some part of the real world" [137]. The area dates back to the 50s, where the first data processing systems started using records and file structures to model and organize information [137]. Since then, a number of proposals for information models and their corresponding techniques have proliferated, covering different areas of Computer Science and Information Systems Engineering [137]. As each information modeling technique supports a different range of applications, each information model has its own ontology. This ontology captures some part of the real-world, together with its assumptions about the reality which are relevant for a given range of applications.

Given our interest in the development of motivational and behavioral perspectives of strategic enterprise architectures, this chapter is particularly interested in reviewing information models whose ontologies capture motivational and behavioral concepts. Therefore, Section 4.2 reviews the state of the art in motivational modeling, while Section 4.3 covers behavioral modeling proposals. Inspired by the representation of motivational and behavioral concepts, a number of approaches in Computer Science have proposed their joint representation as means of performing synchronized changes between both enterprise's perspectives. This group of approaches is described in Section 4.4.

4.2 Motivational Modeling

The notion of "goal" represents the core abstraction in motivational modeling. Historically, goal modeling has its roots in the area of requirements engineering in which the concept of "goal" consists of an abstraction to justify the existence of data and operations requirements of a given software system [200]. In this context, goals have been used not only as a motivation for data and operations requirements, but also as a criterion for requirements completeness, as such requirements only exist for the achievement of higher-level business objectives that naturally arise in the requirements engineering process [200]. Around the core notion of goals, motivational modeling usually incorporates other complementary abstractions, such as *agents* in the software-to-be and in the environment which are responsible for achieving such goals, *objects* that capture domain entities that are relevant to goal formulations and *operations* (or *plans*, *tasks*, *functions*, *activities*, *actions*, *business processes*) whose execution entail the achievement of goals [200].

This section describes the most influential works that support the representation of motivational concepts in Goal-Oriented Requirements Engineering (GORE) (Section 4.2.1) and Enterprise Modeling (EM) (Section 4.2.2). In each work, we describe the semantics of motivational concepts and their interconnections with other complementary abstractions. In order to employ motivational concepts meaningfully, we also discuss available reasoning techniques in the context of each work.

4.2.1 GORE Frameworks

Goal-Oriented Requirements Engineering (GORE) approaches use the concept of goal as the core abstraction for capturing and structuring the content of requirements for a target software system. Therefore, goal models consist of diagrammatical representations of stakeholders' goals and how such business goals are linked to the technical requirements of a system [93]. Although a number of frameworks, techniques, or methodologies for goal modeling exist (e.g., KAOS, GBRAM, NFR, Techne, Tropos, GRL, i*, AGORA, among others) [93], here we sole focus on the description of the

Tropos and i* frameworks. This decision can be accounted by the direct relation of Tropos and i* reasoning techniques with the CGM formalism (Section 3.4) which is used as the baseline of our work. Similarly, the KAOS framework is also another prominent goal technique which is also one of the baselines of our work and therefore, it has been presented in Section 3.6.

Tropos Methodology. The Tropos Methodology [19] allows the representation of requirements in terms of informally-defined goals using natural language, such as "Have a highly reliable system" [65]. Such goals can be further distinguished among hard-goals of soft-goals, with a hard-goal being a goal with clear criteria for determining its satisfaction, while a soft-goal has no clear-cut definition and/or criteria for its satisfaction. (Soft)goals can be related by AND/OR refinement relations and (partial) positive/negative contributions, whose semantics is similar to the *refinement* and *influence relation* (respectively) from BIM framework (Chapter 3, Section 3.1). Together with (soft)goals, Tropos also allows the representation of plans that represent a specific way of doing something, thus capturing a means to satisfy an end ((soft)goal) [19].

The Tropos methodology allows the derivation of detailed UML activity diagrams from business goals. For that, starting with the specification of (soft)goals and plans during the early requirements analysis, the Tropos methodology allows the derivation of global system architecture in the architectural design phase. Subsequently, the methodology also proposes the detailing of agents' plans of the system architecture in terms of UML activity diagrams in the detailed design phase.

For reasoning with goal-related concepts, Tropos proposes two types of label propagation algorithms, forward reasoning [65] and backward reasoning [179]. The forward reasoning algorithm introduces a formal framework in which a goal is axiomatized as a proposition and the relations among goals are axiomatized in terms of propagation rules. The algorithm takes as input lower level goals annotated with labels that denote satisfiability/deniability and forwardly propagates such labels to infer the satisfiability/deniability of higher-level goals depending on alternative system designs.

In contrast, the backward reasoning algorithm finds the minimum cost label assignment to leaf goals that satisfies (or denies) all root goals, thus recommending which system designs to select. In order to do that, the algorithm uses the same axiomatization of the forward reasoning algorithm and then, it encodes the goal model as a boolean formula ϕ. This boolean formula ϕ together with the assignment of satisfiability/deniability values of boolean variables are given to a SAT solver. This solver then determines whether the boolean formula ϕ admits at least one satisfying truth assignment μ to its variables A_i. Both types of algorithms have its qualitative and quantitative counterparts. Furthermore, as the forward reasoning starts with initial values assigned to leaf goals and forwardly propagates such values to infer the satisfaction level of root goals, they are said to work in a *bottom-up* fashion, while the backward reasoning starts with values in the root goals and find the satisfaction values of leaf goals, thus working in a *top-down* fashion.

i* Framework. In the i* modeling framework (first proposed in Yu's PhD thesis [208]), Strategic Dependency (SD) models capture dependency relationships among various actors, while Strategic Rationale (SR) models represent the internal actors' rationales. In both types of models, the framework offers the same modeling constructs of Tropos framework as the latter has been inspired by i*. While Tropos has been developed with focus on an agent-oriented paradigm for software development, i* has a broader applicability in a number of different areas, such as requirements engineering [208], business process re-engineering [209, 210, 211, 208], organizational impacts analysis [208] and software process modeling [208]. For that, i* proposes a conceptual framework that goes beyond the traditional representation of a process in terms of a number of operational steps (i.e., "how" the process is executed) to incorporate the underlying motivation behind those operational steps (i.e., "why" certain activities exist and why a particular order between them is required).

Similarly to Tropos, i* evaluation techniques consists of forward [92] and backward [92, 94] reasoning techniques, in which the goal satisfaction technique allows the initial assignment of evaluation labels for goals (satisficed, denied, and others) and based on the semantics of links among goals, such values are propagated either forwardly (in the direction of the link) [93] or backwardly, allowing the answering of questions like "what is the effect of this alternative?" (forward) or "can these goals be satisfied?" (backward) [93]. Compared to Tropos' reasoning techniques, i* evaluation techniques [94] gives a step beyond Tropos, as it allows the involvement of human judgment in the evaluation of goal satisfaction.

4.2.2 Enterprise Modeling (EM) Frameworks

Inspired by the benefits of goals and requirements modeling in the scope of GORE approaches, goals and requirements modeling have been incorporated in a number of academic and industrial enterprise architecture frameworks. In this context, goal models are used as requirements for an enterprise architecture to be constructed or re-designed, capturing the stakeholder's motivations behind the whole enterprise architecture.

In previous work [25], we have investigated a number of enterprise modeling frameworks, examining their expressiveness in terms of the representation of modeling motivational and behavioral concepts. In this work, we have only included enterprise modeling frameworks that are supported by enterprise modeling languages. Here, we have selected the most prominent industrial frameworks (ARIS and ArchiMate) from our previous effort and also included i* and EKD due to their relevance in academia. Other enterprise modeling frameworks from our previous effort (e.g. Zachman, ISO RM-ODP, DoDAF and MODAF) can be found at [25].

ARIS Framework. The ARIS framework [39, 174] consists of an enterprise modeling framework, popular both in academia and industry, that provides a tool for describing enterprises by means of different views. In the Objective View, goals correspond to the definition of future business objectives which are supposed to be reached by supporting the critical factors and realizing new business processes [174]. They can be related to each other by means of "belongs to" relationship whose semantics is informal according to the ARIS documentation [25]. Summing up, it is a N:N relationship among goals, i.e., an (overriding) goal can be overridden by N (subordinate) goals and a (subordinated) goal can override N (overriding) goals.

In order to capture how enterprise's goals can be achieved, the framework provides the Functional and Process Views. The Functional View captures functions that represent processes, activities or tasks which must be executed for the production of goods or services [174]. While the Functional View captures the enterprise's functional structure in various hierarchical levels, independently of belonging to specific business process, the Process View (or Control View) reflects the dynamic behavior of processes and how they are related to resources, goals and functions [39]. Within the Process View, the ARIS framework uses Value-Added Chains (VACs) from Management Theories (Section 3.3) to represent all enterprise's macro-processes used for delivering goods or services and Event-driven Process Chains (EPCs) [104] to capture the internal logical steps of such processes. The language has opted for modeling the relationship between goals and functions also in the Process View since the execution of functions can be seen as operations applied to objects for the purpose of supporting one or more goals [174]. Although it is possible to represent the relationships between goals and functions by means of a "supports of" relationships, its semantics is also informal in the ARIS literature [25]. Consequently, an informal semantics among goals and among goals and functions hinders the development of reasoning techniques to check satisfaction of goals. For this reason, goal reasoning techniques have not been developed using the ARIS framework, to the best of our knowledge.

ArchiMate Motivational Extension. ArchiMate is a modeling language for describing enterprise architectures which presents a well-defined set of concepts and relationships between architectural domains [113, 78]. The core language distinguishes among three main layers or abstraction levels: the *Business Layer* which offers products and services to external customers realized by business processes executed by actors or roles; the *Application Layer* which supports the business layer with software applications services; and the *Technology Layer* which offers infrastructural services for software applications (composed by software systems, computer and communication devices).

The marginal support devoted to "motivation" in the core ArchiMate modeling framework led researchers to extend it by proposing the ArchiMate Motivational Extension (AME) [160] with common GORE concepts like (soft)goals, AND/OR refinements and contribution relations among goals. Goals are connected to other concepts of ArchiMate by means of a "realization relation" with services and business processes, but processes are not further detailed within the approach, relying on specific process representation languages like BPMN or EPCs to represent their internal logics [78]. In [8], authors analyze strategic planning literature to extend AME with finer grained concepts such as mission, vision, strategic goals and their relations (AND/OR decompositions and refinements), targets and time interval for goal achievement. Although the core AME has been extended with strategic planning concepts in such approach, no reasoning techniques to check goal satisfiability have been proposed in ArchiMate to the best of our knowledge.

EKD. The EKD (Enterprise Knowledge Development) framework consists of a systematic approach to develop and document enterprise knowledge, supporting enterprises in deliberated implementing changes [102, 166]. In order to support a synchronized changing process, the proposal starts with the elicitation of AS-IS enterprise processes by analysts, producing business processes documented using activities and routing gateways, such as AND/OR gateways [22]. Second, AS-IS goals realized by existing processes are abstracted from process descriptions, thus producing an enterprise goal hierarchy. This goal hierarchy captures goals expressed in natural language related by means of AND/OR relations. Such goals are also connected by means of realization relations to the corresponding processes that achieve them in the goal hierarchy [166]. Subsequently, TO-BE goals are elicited and alternative future scenarios that addressed such future goals are modeled. Finally, the proposal comparatively evaluates the current and future scenarios to check the feasibility and aggregated value of the proposed changes.

Regarding reasoning techniques, the EKD approach offers a semi-formal way to specify goals, as it allows the representation of the meaning of goal entities in terms of natural language, and the relationships between such goals as AND/OR and realization relations [103]. Consequently, this semi-formal nature for specifications hinders the development of formal reasoning techniques with goal models, for example, to check the consistency and satisfiability of goal models [103].

4.3 Behavioral Modeling

Historically, behavioral modeling has its roots in three different areas (Quality Control in Operations Management, Management and Computer Science) triggered by needs of work representation within the three areas [196, 201]. In Quality Control, behavioral modeling originated from the necessity of systematic identification of the optimal way of executing the physical work (e.g., assemble a machine or produce shoes), together with the need of development of quality control systems for measuring work efficiency. In contrast, Management has focused on the overall performance of the company, emphasizing the alignment of the strategy with the means of realizing it, such as the organization of work to achieve corporate goals. More recently, the advent of information technology also triggered efforts in Computer Science regarding the development of automated techniques to support operations. In Computer Science, research in behavioral modeling is made in different areas, like Business Process Management (BPM), cooperative work, multi-agent systems, among others.

Section 3.3.2 has already reviewed the state of the art of operations planning in Quality Control and Management from Management Sciences. This section addresses the support in Computer Sciences, more specifically, in BPM.

In BPM, behavioral modeling is based on the central notion of "process" [196]. Different definitions exist for the concept of process, each of them mainly emphasizing a representational method, rather than providing a conceptual definition for the concept. For example, certain definitions like "a business process consists of a set of activities that are performed in coordination in an organizational and technical environment" [202, 162] emphasizes the representation of business processes in terms of activities, while others like "a sequence of unstable states leading to a stable state" [185], highlights a state-based representation of a process. Here, we adopt the definition from [196] in which "a process model intends to capture the different ways in which a case (i.e., process instance) can be handled", as this definition does not assume any particular representational method.

A process model is usually conceived as being composed by a number of *perspectives* [196, 52]. The *control-flow perspective* (or behavioral, in some proposals) is often regarded the backbone of a process model [196] by commonly capturing the behavior in terms of *activities* (unit of work that takes time), *tasks* (simple activities that correspond to a single unit of work), *decisions*, *events* (stimulus from the environment that happen atomically with no time duration) [196, 52]. The *resource perspective* captures who perform the work (e.g. human actors, organizations, roles), the *data perspective* captures data handled along the process (material objects like equipment, materials, products, paper documents and informational objects like electronic documents and electronic records). The *time perspective* models temporal aspects of the process such as durations, deadlines, etc., whereas the *function perspective* describes the computer support provided to activities (applications) [196, 52]. More recently, as processes may span multiple organizations and simultaneously be supported by multiple information systems, there is ever increasing need of representing all the processes performed by the company, together with the required interactions between them. The set of all process performed by the company and their interrelations is commonly called as *Business Process Architecture (BPA)* (or *process portfolio* or *process landscape*) in BPM literature [49, 52].

This section describes the most influential works that support the representation of behavioral concepts. In a nutshell, Section 4.3.1 describes approaches that capture the internal logic (control-flow) of business processes by means of a modeling language, while Section 4.3.2 describes approaches that go beyond the representation of one single process to represent multiple business processes, together with their corresponding interconnections. Similarly to motivational representation, we describe the semantics of behavioral concepts and their complementary abstractions in each work.

4.3.1 Business Process Modeling Approaches

In business process modeling approaches, the process control-flow is represented by means of a modeling language, whose main abstraction diverges depending on the area. Therefore, this section presents a number of approaches for business process modeling and their corresponding abstractions in different areas of Computer Science.

Business Process Modeling Languages in BPM. Proposals targeting the representation of processes are mainly stemmed from the Business Process Management (BPM) area. In BPM, the work performed by roles and individuals in companies is captured by means of business process modeling languages.

Historically, the well-known Turing machine can be considered the first notion of process model [196], by describing how to manipulate the symbols on the tape according to a table of rules. Within the umbrella of business process modeling languages, Petri-nets were the first formalism to treat concurrency as first-class citizen [196]. Despite the availability of well-established formalisms like Petri-nets and process calculi in BPM, industry needs pushed the adoption of a plethora of conceptual languages like BPMN, BPEL, EPCs, workflow nets, etc [196].

In this context, the majority of conceptual process modeling languages represents business processes using the **imperative paradigm** (e.g. BPMN, BPEL, EPCs, workflow nets). In this paradigm, a process model captures which activities need to be executed to handle each process instance. The execution order between such activities also needs to be explicitly described via links (control flows or connectors) between activities and/or data conditions associated with them [176].

Differently from the imperative paradigm that requires the explicit definition of "how" the process is executed, the **declarative paradigm** focuses on the specification of "what" has to be done to handle the case like in the Declare language [176, 198]. In declarative process languages, the process model represents which activities need to be executed, with all the execution paths (ordering between activities) being allowed by default. As all execution paths are allowed by default, there is no explicit need to represent the causal links that activate activities. Prohibited execution paths are specified by adding constraints that restrict the execution order between activities [162, 176].

The realization that the primary driver for the progress of certain types of business processes is not the event related to the completion of activities, but instead the availability of certain values of *data* [162] led to the creation of the **artifact-centered paradigm**. This paradigm proposes a hybrid approach for the representation of business processes, by capturing them in terms of activities and artifacts (data objects). Several variants of the artifact-centered paradigm exist, including the case handling paradigm [199], object-aware processes [162], including also the Guard-Stage Milestone (GSM) meta-model from IBM [95, 16, 33].

Although activity- and data-centered process languages in BPM can be considered conceptual process languages, and therefore, closer to human cognition, as they have been historically originated from mathematical formalisms like Petri-nets, their abstractions (activities and data) are closer to machine abstractions. For example, when a customer hands out an order form to a supplier and the supplier delivers the goods, this logic is represented in the model by means of activities, routing links, events and data objects. However, the real essence of such social interaction is lost in this type of representation, i.e., the act of handling the form represents a request for the supplier to perform some actions and entail other actions to be performed by the customer in return (e.g. payment). Therefore, other trends of research also started recognizing the need of producing specifications whose abstractions are closer to the real essence of business processes as a social interaction among process participants.

Choreographies. Service-Oriented Architectures (SOA) provide descriptions of services (pieces of functionality) and methodological guidelines on how to combine them [196]. In this context, the business process model can be interpreted as a specification on how to combine multiple services. Choreographies approaches capture business processes as a social interaction that takes place in terms of a *flow of messages* among autonomous actors. For example, van der Aalst [197] models cross-organizational business processes using choreographies. Khalaf [105] shows how to map the RosettaNet PIPs business protocols to abstract BPEL processes. Decker et al. [41] extend BPEL with choreography-related constructs. WS-CDL [205] and BPMN 2.0 both support the specification of choreographies. Benatallah et al. [14] propose a transition-based conversation model to conceptualize web service conversations.

Commitment-Based Approaches. *Commitments* also consist of an important abstraction in the representation of business processes as social interactions in SOA. For example, Desai et al. [43] and Yolum [207] use commitments and protocols as design abstractions for business processes. Both works inspire the REGULA framework [124], which introduces temporal operators to represent more expressive commitments and reasoning about them. A number of works also use commitments for specifying cross-organizational business processes. In [42], Desai at al. describe the Amoeba methodology for specifying business processes based on business protocols. Robinson and Purao [164] propose a framework for specifying and monitoring cross-organizational business processes that relies upon commitments enriched with a temporal logic. Nandi and Sanz [139] employ sets of commitments as cross-organizational contracts that lead to value creation. Ferrario et. al. [57] propose an ontological model of services in terms of the notion of commitments in service systems, while Nardi et al. [140] propose a reference ontology for service-oriented enterprise architecture (UFO-S) whose main abstraction is the concept of commitment. Finally, Chopra et. al [31] presents a business level conceptual model that represents process participants in a service-oriented application in terms of goals and commitments.

Resource Management Approaches. By extending activity-centered languages with constructs for modeling allocation of activities to different roles in a business process, social and resource management perspectives in BPM also strive to provide a more social perspective for business processes. For example, Cabanillas et al. [23] introduce the Resource Assignment Language (RAL) whose formal semantics is defined in Description Logics. The RAL language extends the BPMN language with RAL expressions in order to provide mechanisms for history-based human resources management within business processes. Similarly, Brambilla et al. [18] also extend the BPMN notation for capturing social requirements, by including new events and task types together with some annotations for pools and lanes.

Compliance Management Approaches. Compliance management approaches in BPM incorporate the representation of legal, contractual aspects into the behavioral dimension of business processes and consequently, a social perspective is also provided. In that respect, Ghose and Koliadis [64] semantically annotate business processes in BPMN with constraints on their execution in order to define normative compliance. Sadiq et al. [170] enrich imperative business process models with obligations that members of an enterprise must fulfill in order to remain compliant. In order to model such obligations, the paper proposes the Formal Contract Language

(FCL) which is a combination of a non-monotonic formalism (defeasible logic) and a deontic logic of violations.

Language Action (L/A) Perspective Approaches. While in activity- and data-centered approaches the business process control-flow is represented in terms of activities, L/A approaches shift the representation of business process from activities structure to coordination structure. In L/A perspective, the business process is seen as a social intercourse composed by customers and performers who interact by means of generic interaction phases (e.g., negotiation, fulfillment, completion phases). Such generic interaction phases called as *business actions* or *action workflow loops* [127, 47], *business acts* [122, 121], *communicative actions* [96] are considered the basic primitives for the representation of business processes [127]. In such approaches, business actions are represented by loops of multiple activities, each of them representing a generic phase of the business action, and the overall structure of the business process is represented in terms of business actions. Therefore, as business processes are represented in terms of business actions, it is said that they are represented in terms of coordination structures. Such coordination structures may be highly recurrent (done in a structured way time after time), while others may be ad-hoc (unique to a situation) [127]. Consequently, although activities are still the basic primitives for the representation of processes, the different starting point leads to different potentials for representation and support of activities [127].

4.3.2 Business Process Architecture (BPA) Approaches

While substantial efforts have been developed for representing and analyzing individual business processes, more recently, a number of approaches started going beyond the sole representation of business processes control-flow (together with other perspectives) and striving to incorporate the representation of a process in the context of the relations that it establishes with other processes in the company. The existence of multiple business processes at different levels of abstraction also motivated such approaches to introduce principles to organize the overall company's set of processes, which is usually a hierarchical structure. Therefore, this section reviews the state of the art in the representation of multiple processes, their interrelations and organizing principles, which is usually called as the *Business Process Architecture (BPA).*

In [202], Weske proposes a hierarchy of processes, starting with (i) a *strategic level* in which business goals are defined by the enterprise (but not represented in any specific language), (ii) an *organizational level* composed by organizational processes that depicts organization's suppliers and consumers modeled in an informal way by means of textual language, (iii) an *operational level* composed by operational business processes modeled in a business process modeling language and finally (iv) an *implemented level* in which business processes are implemented in an information system.

Similarly, Dumas et al. [52] define the notion of process architecture as a conceptual model that shows the processes of a company and makes their relationships explicit, depicting the BPA in three levels. The *business process landscaped model* represents the entire BPA with all the processes executed by the company together with their relations, the *abstract level* depicts the organizational process modeled as process maps that contain only the essential steps of the process, usually represented in a linear way, abstracting from alternative/exception paths, iterations and the roles that execute each step. Finally, the *detailed level* shows the process models that can be represented using the BPMN language.

Eid-Sabbagh [54] presents a formal conceptual framework for representing and analyzing BPAs. The approach starts by formalizing the notion of BPA in terms of its elements (business processes, events, etc.), relations among processes (composition, specialization, etc.) and structuring levels (BPA compendium, BPA subset, detailed process models). Then, it defines desired BPA properties (e.g. BPA correctness criteria) in terms of structural and behavioral patterns of relations between process models, together with anti-patterns that represent erroneous relations between process models. A transformation from BPAs to Open Nets (ON)(a subclass of Petri nets) is defined and the (anti)-patterns are then used as input in a Petri-net verification tool (LoLA (Low Level petri net Analyzer)) that check the soundness of the whole BPA, allowing one to analyze process interdependencies in a facilitated way.

4.4 Motivational and Behavioral Modeling

Besides well-consolidated GORE, EM and BPM approaches, this section reviews approaches that combine existing representation methods in goal modeling (e.g., i*, EKD, etc.) and process modeling (e.g. BPMN, EPCs, etc.), also including approaches that cover reasoning involving motivational and behavioral concepts.

Ontologies for Goals. Based on the realization that existing goal modeling frameworks require more expressive ontologies for enterprise goals, some proposals create novel ontologies in enterprise modeling:

- **Mendes et al.** [**130**] propose a goal meta-model that distinguishes among strategic vs. operational goals and quantitative vs. qualitative goals in enterprise modeling;
- **Markovic et al.** [**125**] propose a business goal modeling ontology that also distinguishes among strategic vs. operational goals and quantitative vs. qualitative goals. In this proposal, a goal G has also a number of attributes like description, measure, deadline, prior-

ity, achieved. Further, the paper also proposes modeling patterns to integrate goals into process models. Regarding automated reasoning with goals and processes, although authors argue that such ontology can be used for query answering (e.g., find a process that does not support any goal or filter goals by deadline and priorities), the technical queries are not proposed in the paper.

Integration of Existing Methods in Goal and Process Modeling. Other proposals go beyond the creation of novel goal ontologies to propose the integration of existing goal modeling constructs from goal modeling techniques (e.g., i*, EKD, BMM, etc.) into process modeling (e.g. EPCs, BPMN, etc.):

- **Korherr et al. [109]** extend BPMN and EPCs meta-models with the concepts of process goals and measures.

- **Greenwood et al. [72, 73]** also extend BPMN with a conceptual modeling language called Goal-Oriented Business Process Modeling Notation (GO-BPMN). In GO-BPMN, a business process is represented in terms of goals to be achieved/maintained (that can be refined in terms of AND/OR relations), plans (activities) and the hierarchical relationships among goals. The proposal also incorporates an execution engine for the enactment of business processes represented using GO-BPMN. In [73], an autonomous agent controller is assigned to each business process and the execution engine is capable of coordinating the execution of multiple processes that are executed/controlled autonomously by independent agent controllers;

- **Yu et al. [212]** perform a mapping between i* intentional concepts into BMM concepts to capture key BMM's distinctions for enterprises by means of an intentional modeling language (i*). Subsequently, it uses i* reasoning techniques to determine the level of achievement of top-goals based on the assignment of labels to lower-level goals.

- **Koubarakis at al. [111]** proposes a formal enterprise and business process modeling framework built on the basis of situation calculus (AI formalism) and F^3 and EKD enterprise modeling frameworks. Enterprise models consist of five interconnected sub-models (organizational, objectives and goals, process, concepts and constraints sub-models). Within the objectives and goal sub-model, the concept of enterprise goal captures a desired state of affairs, whereas behavioral concepts are captured in terms of business processes and actions using situation calculus and concurrent logic programming language ConGolog concepts;

- **Aburub et al. [1]** model the non-functional requirements of a process (e.g. service-time, responsiveness) in terms of the Non-Functional Requirements (NFR) modeling constructs from the NFR framework[1]. In the proposal, the NFR model is mapped into a process model represented in Role Activity Diagrams (RADs) in order to support the identification and modification of process entities (e.g. activities, interactions and roles) to better meet the desirable qualities or properties (NFR) of the business process;

- **Neiger at al. [141]** formalize EPCs for the representation of processes and Value Focused Thinking (VFT) framework (a conceptual tool from Decision Sciences) for the representation of goals. In VFT, fundamental goals (structured as a hierarchy) describe business values from the company and means goals (structured as a network) describe

[1]Although we do not describe the NFR framework in detail among the GORE approaches (Section 4.2.1), for the sake of this thesis, it is sufficient to say that NFR framework represents the non-functional requirements of a given software system in terms of soft-goals.

the means of achieving the fundamental goals. Further, the formalization of EPCs and VFTs opens up the possibility for the paper to propose rules for the consistency between goals and multiple business processes;

- The **User Requirements Notation (URN) standard [158, 159]** integrates a goal-oriented notation (Goal-oriented Requirement Language (GRL)), based on the i* and NFR modeling concepts and a scenario-oriented notation (Use Case Maps (UCM)). In URN, goals, soft-goals, their contribution links and decomposition relations (AND, XOR, IOR) are captured in GRL. In its turn, UCM notation enables the representation of scenario behavior by specifying paths. The concept of path is very similar to a business process as it expresses the causal flow of the system behavior, by representing actions, sequence, alternatives, and concurrency as well as the beginning and end of scenarios;

- **Guizzardi at al. [82]** propose a method that integrates models expressed using Tropos and BPMN;

- **Popova et al. [154, 155]** present a framework for modeling goals and performance indicators in the context of a general organization modeling framework. In [154], a goal describes a desired state or development of the company or an individual. Label propagation rules for evaluating goals satisfaction are similar to NFR framework rules for evaluating the satisfaction of higher-level goals on the basis of information about the degrees of satisfaction/satisficing of lower level goals. Processes are modeled in terms of tasks, processes and resources. In [155], a formal technique for analysis of executions of organizational scenarios is also proposed. Such techniques establish the correspondence between a formalized execution (i.e., a trace) and the corresponding specification that describe (or prescribe) ordering constraints on organizational processes, resources, allocations of actors to processes, etc. using TTL (temporal trace language) model checkers. Although goal evaluation and trace evaluation formal techniques are proposed in the approach, they are not integrated to evaluate the satisfaction of high-level goals based on trace executions to the best of our knowledge.

Top-Down Business Process Design from Business Goals. While the motivational characteristic of goals express "why" a certain activity of process should be executed, the behavioral dimension captures "how" this goal can be achieved by means of alternative behaviors. This intrinsic feature of goals and process models have inspired many top-down approaches that design the internal logic of one or multiple business processes from business goals:

- **Kueng et al. [112]** suggest an informal approach for generating process models on the basis of goals. The proposal distinguishes among functional goals that determine the structure of process design, whereas non-functional goals are used to evaluate quality properties of the process model, such as correctness of a given design. The informal approach starts by representing business process-related goals, goal measurement criteria and restrictions. Further, the activities that achieve such business goals are derived and represented accordingly in input/output table that captures logical and temporal dependencies between activities as well as sequences, alternations or concurrency. Then, authors propose to apply Petri-nets to depict the execution order among activities. Once this process design is ready, it is evaluated against some independent business-process requirements proposed in the paper;

- **Bleistein et al. [17]** propose the B-SCP (business strategy, context, and process) requirements engineering framework for business-IT alignment in the specification of e-business systems. The framework is composed of three parts by combining Jackson problem frames, goal modeling expressed using i* language and process modeling in RADs. Jackson problem diagrams describe the world in terms of two parts, i.e., a domain context that describes the business domain in terms of domain entities (problem context) and an optative part that describes the effects in the real world that should be guaranteed by the machines (the requirements). The approach uses Jackson's problem diagrams to describe the problem context and goal modeling to capture the optative properties of the system (requirements). In terms of methodology, the approach starts with the elicitation strategy that employs the VMOST analysis technique (Vision, Mission, Strategies, Objectives and Tactics) for eliciting business strategy in terms of core concepts. Such core concepts are extracted from the BRG-Model (Vision, Mission, Goals, Strategies, Objectives and Tactics) and modeled using i* modeling language. Second, it uses a progression of problems for refining requirements from high levels of abstraction down to lower levels. In each level of abstraction, goals consist of requirements that are linked to context diagrams describing the entities of the business domain. As in each level of abstraction, goals are part of a larger goal model; and the goal model enables the explicit connections of requirements and problem context at adjacent levels in terms of super goals and sub-goals. The refinement process ends when it is possible to connect goals and business processes. Overall, business processes are mapped to the organizational goal model and context diagrams by means of correspondence rules to enforce the alignment between business strategy and business processes that support such strategy.

Business Goals and BPA Approaches. While top-down proposals for business process design have recognized the benefits of proposing an integrated approach between organization's goals and one or multiple business processes, this line of work explicitly incorporate the representation of business process architectures and goals in their proposals:

- **Lapouchnian et al. [116]** propose a modeling framework for designing business process architectures (BPAs) that focuses on the representation of different BPA alternatives and supports the selection of alternative architectural choices. Variability in the space of architectural alternatives can be obtained by moving across several dimensions. In particular, the temporal dimension regards the placement of process elements (PE) (an activity or a decision) earlier or later in relation to other PEs within a process, whereas the recurrence dimension considers the placement of a PE into a process that is executed with a different frequency than other processes. Therefore, a concrete BPA can be derived by selecting different points of the design dimensions. Further, PE placement options (i.e., alternative process architecture configurations) are represented in terms of functional goals and alternative BPA configurations can be chosen on the basis of the prioritization of soft-goals and conflicts handling.

- **Morrison et al. [136]** formalize the relationships between the Strategy Modeling Language (SML) and the process portfolio or BPA. In SML, company's strategic goals are modeled in terms of functional goals, plans (that describe milestones in an organizational strategy to achieve such goals) and optimization goals (maintain, maximize, minimize) (used to discriminate preferences for strategic outcomes). The set of company's processes are modeled in terms of BPMN model with activities that are semantically annotated with pre- and post-conditions. In terms of reasoning, the approach formalizes the notion of "strategic alignment" as the minimal set of business processes that must be aligned with a given functional goal, plan and optimization goal. Subsequently, this notion of strategic alignment is used to test the alignment of goals and processes against a rule base.

Business Process Re-Engineering from Business Goals. Either because operational procedures are not fully delivering value to the company or they have low performance, they need to be re-engineered to better fulfill company's standards. As business goals define "why" something needs to performed, instead of "how", the usage of goal models may reveal shortcomings in the process and thus, they can be used to re-engineer them. In particular:

- **Grau et al.** [**69**] propose an i*-based business process re-engineering method that builds on top of i* modeling language and enriches the framework with additional methodological steps for business process re-engineering.

- **Anton et al.** [**5**] also propose a business process re-engineering method in which goals are expressed using textual language and business processes are expressed by means of operational concept definitions (OCD) that describe a business process in terms of scenarios, critical incidents and examples of problems the organization must solve. The goal model distinguishes among a number of goal categories (achievement, maintenance, avoidance from KAOS, objective vs. adverbial goals and prescriptive vs. descriptive goals). Business processes are re-engineered by promoting an alignment between the goal structure and process scenarios by means of top-down and bottom-up strategies.

Goal Variability. Work in goal variability offers the possibility of performing synchronized movements between a goal and process models, by allowing one to design several alternative variants of a process model. The general idea involves building a goal models together with the activities responsible for the achievement of goals and then select different process variants on the basis of the prioritization of some particular goals. In particular:

- **Koliadis at al.** [**107**] extend BPMN with KAOS constructs and propose rules to maintain consistency among both models when changes in one of the models happen. In [**108**], the same approach is taken for guiding analysts to reflect changes in an i* model when changes occur in a BPMN model and vice-versa.

- **Santos et al.** [**173**] adopt the same approach for the re-configuration of BPMN models on the basis of goals expressed with constructs of Tropos language.

- **Lapouchnian et al.** [**117**] propose a systematic, tool-supported requirements-driven approach for business process design and configuration (expressed in BPEL) on the basis of goals models expressed in a Tropos style goal language.

- **Lapouchnian et. al** [**115**] enhance goal models (expressed in a Tropos style goal language) with contextual tags and use conventional Tropos goal analysis techniques (forward reasoning [65], Section 4.2.1) to reason whether system goals can be attained in currently active context(s).

State-Based Representations of Business Processes. Some approaches in BPM shifts the representation of business processes of common BPM abstractions, such as *activities*, *artifacts*, *messages*, to propose a state-based representation of business processes. This new type of representation opens up the possibility of representing the process goal in terms of a final desired state among the many states that compose the business process:

- **Soffer et al. [185]** provide a conceptual/formal framework in which a process is conceived as a set of unstable states that lead to a stable state. In the proposal, a goal is seen as a set of states that satisfy a condition. Conditions are defined over functions that assign values to state variables and goals can be potentially achieved by many alternative paths. The formalism also uses the concept of soft-goals from i*/Tropos (Section 4.2.1) to define a (process) soft-goal as an order relation among a number of states. In this context, soft-goals can be used both by designing and measuring a process. In process design, a given order relation on goal states can serve for evaluating and selecting the best process path among many alternative paths. In process measurement, actual values of state variables can be measured at execution time and serve for evaluating the specific instance of a process with respect to other instances or to target values.

- **Khomyakov et al. [106]** consider a business process as a dynamical system that moves in the multidimensional space that contains all possible states until reaching one of the final states (i.e., the goal). In this context, a process goal is defined as a set of final states defined over a criterion. Movements in the multidimensional space are done either via the execution of activities (e.g. build a wall) that moves the process towards the goal or by the occurrence of external events that may move the system in the opposite direction. The execution control is realized by means of the notion valid state in which each valid state contain all required, allowed and prohibited activities for moving the process to the next stipulated state.

- In **Nurcan et al. [145]**, a process model is captured as a map which consists of a non-deterministic ordering of intentions and strategies. An intention is an optative statement that expresses an expected state or result to be reached in the future. For example, "make room booking" represents an intention to make a reservation for a room in a hotel, whose booking is the expected result. Each map has two special intentions, start and stop, to denote the beginning and the end of the process, respectively. Intentions can be achieved by means of a strategy that represents an approach or manner to achieve this intention. For example, in order to make the room reservation, bookings can be made via internet or by visiting a travel agency.

Planning Approaches in Artificial Intelligence (AI). In AI planning, the planning problem consists of finding a sequence of actions to be performed by an agent to achieve some goal. It starts in a given state of the world and the agent has to choose a sequence of actions that will very likely bring about the final desired state [169]. Therefore, goal-oriented approaches reproduce the same idea by finding sequences of actions or design alternatives in goal models to achieve the goals represented in the model. Here, we focus on works that represent goal models graphically in terms of goals and their relationships (AND/OR and contribution links).

- **Liaskos et al. [120]** represent goal models in terms of goals, tasks, AND/OR refinements, enriched with temporal constraints, optional goals (goals that can be optionally achieved in an AND-refinement) and preference goals. Then, the paper uses a planner to find a sequence of tasks which would satisfy all mandatory goals, respecting the precedence constraints.

- **Bryl et al. [20, 21]** propose a framework in which goal models are represented using simplified i* syntax (goals, agents responsible for satisfaction of goals and delegations) and uses a planner to find satisfactory delegations (design alternatives) among agents.

- In **Asnar et al. [7]**, Asnar et al. use Tropos reasoning algorithms (Section 4.2.1) to find the best design alternative taking into account risks. This work [7] is combined with the work of Bryl et al. [20, 21] in AI planning in **Asnar et al. [6]**. Basically, Asnar and colleagues [6] start from the first-order logic formalism used by Bryl et al. [20] and add formalisms to define the notions of goal criticality and goal relaxation. While goal criticality level defines the minimum level of trust required for the goal to be delegated, goal relaxation indicates that the goal criticality needs to be relaxed (i.e., lowered) if a dependency which satisfies the required level of trust cannot be found. Then the approach uses the same AI planning approach proposed in Bryl et al. [21] to find a plan (design alternative), evaluate it and re-plan if the plan does not meet the required criteria.

4.5 Assessment of Surveyed Approaches Against Requirements

Taking as starting point the proposals introduced in Sections 4.2, 4.3 and 4.4, we here analyze their capabilities against the achievement of the requirements for strategic enterprise architectures (Chapter 2). In order to remind the reader and facilitate our evaluation, we repeat Table 2.1 (shown in Table 4.1) from Chapter 2 with requirements R1 and R2 for strategic enterprise architectures:

Summary of Requirements for Strategic Enterprise Architectures	
Expressiveness in mot. perspective	Description (1.1)
	Ownership (1.2)
	Time Frame (1.3)

	Goal Pattern (1.4)
	Targets (1.5)
	Multiple Levels of Abstraction (1.6)
	Representation of Goal Relations (2)
	Representation of Factors that Impact Goal Achievement (3)
Expressiveness in beh. perspective	Social Perspective of Process (1.1)
	Operational Perspective of Process (1.2)
	Business Process Architecture (BPA) (1.3)
Traceability between mot/beh persp.	Traceability in Representation (2.1)
	Traceability in Methodological Consistency (2.2)

Table 4.1: Summary of Requirements R1 and R2 for Strategic Enterprise Architectures

Each group of approaches previously introduced in this chapter puts special emphasis on a specific enterprise's perspective and certain requirements are not addressed by all groups of approaches. Table 4.2 summarizes the groups of proposals and the requirements addressed by each of them.

	R1. Expressiveness		**R2. Traceability**	**R3. Reasoning**
	R1.1 Mot. Domain	**R1.2 Beh. Domain**		
GORE Frameworks	✓	-	✓	✓
EM Frameworks	✓	-	✓	✓
Business Process Modeling Approaches	-	✓	-	-
BPA Approaches	-	✓	-	-
Motivational and Behavioral Modeling	✓	✓	✓	✓

Table 4.2: Assessment of approaches against the requirements from Chapter 2

Regarding the conventions for Table 4.2, a ✓sign indicates that the proposal fully addresses the requirement, a - sign indicates that the proposal does not address the requirement and a ∼ sign indicates that the proposal partially addresses the requirement.

In the remainder of this section, each group of approaches is assessed against the requirements using the same sign conventions of Table 4.2. More specifically, GORE approaches are assessed in Section 4.5.1, EM approaches in Section 4.5.2, Business Process Modeling and BPA approaches in Section 4.5.3 and motivational and behavioral approaches in Section 4.5.4. Finally, automated analysis with motivational and behavioral within the aforementioned areas are evaluated in Section 4.5.5.

4.5.1 GORE Frameworks

Table 4.3 depicts the assessment of GORE frameworks (Section 4.2.1) against the expressiveness in motivational perspective (R1.1) requirement and traceability between motivational and behavioral perspectives (R2) requirement. For both requirements, Tropos and i* present the same level of support for the requirements of strategic enterprise architectures, as can be evidenced by Table 4.3.

Regarding the **expressiveness in motivational perspective (R1.1) requirement**, Tropos and i* addresses sub-requirements 1.1 and 1.2, but neglect sub-requirements 1.3-1.6. This is explained by their focus on the early stages of the software engineering process in which requirements on operations and data are represented together with their corresponding goals, motivating "why" such operations and data are represented. Consequently, such approaches are able to capture goals (sub-requirement 1.1) and the agents responsible for their achievement (sub-requirement 1.2). However, with this focus on early stages of software engineering, they are not able to represent sub-requirements for strategic enterprise architectures (sub-requirements 1.3-1.6), such as multiple levels of goal abstraction and environmental factors that impact goal achievement.

Regarding **traceability between motivational and behavioral perspectives (R2) requirement**, although the concept of “process” can be modeled as a plan in Tropos and i*, capturing which behavioral elements are responsible for the achievement of a goal, the detailing of processes in terms of its control-flow and the corresponding linkage between goal, process and the elements in the process control-flow cannot be captured in GORE approaches. Therefore, the three approaches only partially support the integration between the motivational and behavioral perspectives as denoted by a ∼ sign at property 2.1 in Table 4.3.

	R1.Expressiveness								**R2.Trac.**	
	R1.1 Motivational Domain									
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**2**	**3**	**2.1**	**2.2**
Tropos Methodology [19]	✓	✓	-	-	-	-	✓	-	∼	✓
i* framework [208]	✓	✓	-	-	-	-	✓	-	∼	✓

Table 4.3: Assessment of GORE approaches against the requirements from Chapter 2

4.5.2 EM Approaches

Table 4.4 depicts the assessment of EM frameworks (Section 4.2.2) against the expressiveness in motivational perspective (R1.1) requirement and traceability between motivational and behavioral perspectives (R2) requirement.

Regarding the **expressiveness in motivational perspective (R1.1) requirement**, inspired by the benefits of goal-orientation in GORE approaches, EM frameworks borrow the GORE concept of goal to represent strategic concerns from a given company such as “Increase sales”. Those goals can then be linked to their corresponding process, thus motivating “why” these processes exist in the context of a given enterprise architecture. Consequently, like GORE approaches, ARIS and EKD modeling approaches capture goals and the organizational actors responsible for their achievement (sub-requirements 1.1 and 1.2), but cannot address the representation of sub-requirements for strategic enterprise architectures, such as time frame, goal patterns, targets and multiple levels of abstraction. Both approaches also do not fully recognize the existence of complex relationships among goals and environmental factors that impact their achievement.

In fact, the ArchiMate language started recognizing the needs of providing a more expressive ontology for the representation of goals in enterprise architectures in [8]. Consequently, the language covers most of the motivational requirements in Table 4.4, although it still does not recognize the different types of goal patterns and different types of behavior required for the achievement of such goals (sub-requirement 1.4). The existence of lower goal levels of abstraction (e.g., tactical and operational) (sub-requirement 1.6) are also not acknowledged. Due to its advance towards a more expressive goal ontology, ArchiMate can be considered the most expressive language for the goal representation in enterprise architectures, as can be seen in the motivational column of Table 4.4.

Regarding **traceability between motivational and behavioral perspectives (R2)** requirement, ARIS and EKD modeling approaches capture the concept of goal attached to a process and further refine this process in terms of its control-flow (evidenced by a ✓sign at traceability requirement for both frameworks in Table 4.4). However, although ArchiMate can be considered the most expressive EM framework for the representation of different shades of goals, it captures the behavioral domain in terms of processes, but refrains from detailing such processes in terms of their control-flow. In order to capture processes' control-flow in ArchiMate, specific process representation languages like BPMN or EPCs have to be adopted [78].

	R1.Expressiveness								**R2.Trac.**	
	R1.1 Motivational Domain									
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**2**	**3**	**2.1**	**2.2**
ARIS framework [39, 174]	✓	✓	-	-	∼	-	-	-	✓	✓
EKD framework [102, 166]	✓	✓	-	-	-	-	∼	-	✓	✓
ArchiMate [160, 8]	✓	✓	✓	-	✓	-	✓	-	∼	-

Table 4.4: Assessment of EM approaches against the requirements from Section 2.3

4.5.3 Business Process Modeling and BPA Approaches

Table 4.5 depicts the assessment of business process modeling and BPA approaches against the expressiveness in behavioral perspective (R1.2) requirement and traceability between motivational and behavioral perspectives (R2) requirement.

Regarding **expressiveness in behavioral domain (R1.2) requirement**, all approaches provide abstractions for capturing the operational perspective of processes (sub-requirement 1.1). Such abstractions depend on the language under consideration, i.e., imperative and declarative paradigms capture processes in terms of activities, whereas artifact-centered approaches use data objects, choreographies use messages, etc. This result seems to be reasonable as such languages are intended to define operational details required by computer systems that support process execution.

Nevertheless, in face of our ultimate purpose of integrating motivational and behavioral perspectives, business processes need to be also specified in terms of social abstractions (sub-requirement 1.2) as means of capturing the social interactions among organizational members to achieve their goals. In this context, although the need of social specifications has been recognized by some proposals (e.g., choreographies, L/A approaches, etc.), processes' control-flow is typically represented in terms of activities. This lack of abstractions for the representation of business processes in social terms represents a serious impairment for the integration of the motivational and behavioral perspectives because it does not allow a seamless transition between both domains. This transition is not seamless because the idea of goal involves the representation of "what" has to be achieved (regardless "how"), whereas the activity-centered paradigm precisely specifies "how" to achieve a final goal by requiring the specification of certain steps to be performed. As the mapping of goals to activities usually involves complex relations (e.g., one goal may be associated to one activity, multiple activities, loops, etc.), the direct association of goals to activities does not promote a seamless transition between both domains.

Regarding the representation of the set of processes performed by the company (sub-requirement 1.3), the concept of BPA has appeared in more recent approaches as means of capturing and analyzing the whole integrated set of company's business processes. In this context, the BPA represents an important abstraction to promote the integration of both motivational and behavioral elements in strategic enterprise architecture models as it captures the whole set of behavioral elements required for the achievement of the company hierarchy of goals. The approaches usually organize the BPA into three hierarchical levels: (i) the process landscape level that depicts a holistic view of all business processes executed by the enterprise, together with their consumer-producer relations, (ii) abstract process models that represent processes and their relations and (iii) detailed process models that capture the control-flow of each process. As argued before, although this comprises in an important initiative towards capturing the integrated set of processes that achieve company's goals, detailed process models are usually represented in terms of imperative models, thus lacking a social perspective of business processes and hindering a seamless integration between motivational and behavioral domains.

	R1.Expressiveness		
	R1.1 Beh. Domain		
	1.1	**1.2**	**1.3**
Petri-nets [196]	-	✓	-
Imperative languages (BPMN, BPEL, UML, EPCs, workflow nets) [196, 104]	-	✓	-
Declarative languages [176, 198]	-	✓	-
Artifact-centered languages [162, 16]	-	✓	-
Choreographies [197, 105, 41, 205, 14]	∼	✓	-
Resource Management Approaches [23, 18]	∼	✓	-
Compliance Management Approaches [64, 170]	∼	✓	-
L/A Approaches [127, 47, 122, 121, 96]	∼	✓	-
Weske [202]	-	✓	∼
Dumas et al. [52]	-	✓	✓
Eid-Sabbagh [54]	-	✓	✓

Table 4.5: Assessment of Business Process Modeling and BPA Approaches Against the Requirements from Section 2.3

4.5.4 Motivational and Behavioral Modeling

Table 4.6 depicts the assessment of hybrid approaches that combine the representation of motivational and behavioral concepts against the expressiveness in motivational perspective (R1.1), expressiveness in behavioral perspective (R1.2) and traceability between motivational and behavioral perspectives (R2) requirements.

Regarding **expressiveness in motivational domain (R1.1) requirement**, the hybrid proposals follow the same approach of EM frameworks by inhering the GORE conceptualization and thus, sub-requirements 1.1-1.6 are roughly supported by such approaches. In this context, Table 4.6 reveals that approaches commonly inhere some GORE conceptualization (e.g. goals, goals' owner, goal patterns, etc.) in order to perform some application (e.g. to develop some goal ontologies, to integrate existing methods in goal and process modeling, to design business processes from business goals, etc.), but none of them provide complete support for the representation of motivational requirements. For example, Grau et al. [69], Anton et al. [5] and Koliadis at al.(a) [107] inherit the concept of goal patterns (achieve/cease, maintain/avoid, optimize) (sub-requirements 1.4) from KAOS and apply such conceptualization in business process (re)design from business goals. However, the three approaches do not provide support for the representation of targets (sub-requirements 1.5) and multiples levels of among goals (sub-requirements 1.6).

It is interesting to notice that only five approaches partially acknowledge the existence of multiple levels by either inheriting BMM conceptualization (e.g. Yu et al. [212] and Bleistein et al. [17]) or by creating their own levels with incomplete number of layers and unclear criteria to differentiate among them (e.g., Mendes et al. [130], Neiger at al. [141], Guizzardi et al. [82]). Approaches that do not recognize the existence of multiple levels commonly use the GORE concept of goal to either represent process goals (desired state to be achieved by a certain business process) [125, 109, 72, 107, 108, 117, 185, 106, 145, 120, 20, 21, 6] or to represent strategic and process goals interchangeably [111, 1, 158, 159, 112, 116, 165, 136, 69, 5]. Furthermore, any approach acknowledges the representation of environmental factors that affect goals achievement (denoted by a - sign at sub-requirement 3 from motivational perspective).

Regarding **expressiveness in behavioral domain (R1.2) requirement**, we conclude that none of the approaches consider social abstractions to capture the control-flow of business process (denoted by a - sign at sub-requirement 1.1 from behavioral perspective), centering their representations in terms of operational abstractions (activities) (denoted by a ✓sign at sub-requirement 1.2 in most of the approaches from behavioral perspective). Furthermore, very few approaches consider the importance of representing the enterprise's whole set of processes by means of the concept of BPA. In this context, although some approaches recognize the existence of multiple processes to achieve enterprise's goals [125, 158, 159, 112, 141], very few consider the explicit representation of the company's BPA (with the exceptions of [116, 165, 136]). As a result of that, they lack to provide a holistic overview of all company's processes and its relations.

Regarding **traceability between motivational and behavioral perspectives (R2)** requirement, most of the hybrid approaches link motivational and behavioral concepts, although many of them do not provide a representation of process control-flow. In this sense, approaches that support the representation of the concept of "process" together with its control-flow have scored ✓, whereas approaches that support the representation of processes, but with no control-flow representation scored ∼. Such process representation can be accompanied or not by a corresponding methodology (e.g. top-down business process design, goal variability, etc.).

	R1.Expressiveness											**R2.Trac.**	
	R1.1 Motivational Domain								**R1.2 Beh. Dom.**				
	1.1	**1.2**	**1.3**	**1.4**	**1.5**	**1.6**	**2**	**3**	**1.1**	**1.2**	**1.3**	**2.1**	**2.2**
Mendes et al. [130]	✓	-	-	-	✓	∼	-	-	-	-	-	∼	-
Markovic et al. [125]	✓	-	✓	-	✓	-	∼	-	-	✓	-	✓	-
Korherr et al. [109]	✓	∼	✓	-	✓	-	-	-	-	✓	-	✓	-
Greenwood et al. [72, 73]	✓	✓	✓	∼	-	-	∼	-	-	✓	-	✓	-
Yu et al. [212]	✓	✓	-	-	-	∼	✓	-	-	✓	-	∼	-
Koubarakis at al. [111]	✓	✓	-	-	-	-	∼	-	-	✓	-	✓	✓

Aburub et al. [1]	✓	✓	-	-	-	-	✓	-	-	✓	-	✓	✓
Neiger at al. [141]	✓	~	-	-	-	~	~	-	-	✓	~	✓	✓
URN [158]	✓	✓	-	-	✓	-	✓	-	-	✓	~	✓	✓
Guizzardi et al. [82]	✓	✓	-	-	-	~	✓	-	-	✓	-	✓	✓
Kueng et al. [112]	✓	✓	-	-	-	-	~	-	-	✓	-	✓	✓
Bleistein et al. [17]	✓	-	-	-	-	~	✓	-	-	✓	-	✓	✓
Lapouchnian et al. (a) [116]	✓	-	-	-	-	-	✓	-	-	✓	~	✓	✓
Morrison et al. [136]	✓	-	✓	~	-	-	~	-	-	✓	~	✓	-
Grau et al. [69]	✓	✓	-	✓	-	-	✓	-	-	✓	~	✓	✓
Anton et al. [5]	✓	✓	-	✓	-	-	~	-	-	✓	-	✓	-
Koliadis at al.(a) [107]	✓	-	✓	✓	-	-	~	-	-	✓	-	~	✓
Koliadis et al. (b) [108]	✓	✓	-	-	-	-	✓	-	-	✓	-	~	✓
Santos et al. [173]	✓	-	-	-	-	-	✓	-	-	✓	-	~	-
Lapouchnian et al. (b) [117]	✓	-	-	-	-	-	✓	-	-	✓	-	~	✓
Lapouchnian et. al (c) [115]	✓	-	-	-	-	-	✓	-	-	-	-	-	-
Soffer et al. [185]	-	-	-	-	-	-	-	-	-	✓	-	✓	✓
Khomyakov et al. [106]	-	-	-	-	-	-	-	-	-	✓	-	~	✓

Nurcan et al. [145]	-	-	-	-	-	-	-	-	-	✓	-	✓	✓
Liaskos et al. [120]	✓	-	-	-	-	-	✓	-	-	✓	-	∼	✓
Bryl et al. [20]	✓	✓	-	-	-	-	✓	-	-	✓	-	∼	✓
Asnar et al. [6]	✓	✓	-	-	-	-	✓	-	-	✓	-	∼	✓

Table 4.6: Assessment of Motivational and Behavioral approaches against the requirements from Section 2.3

4.5.5 Automated Analysis of Surveyed Approaches

In this section, we review the state of the art of automated reasoning techniques regarding the achievement of **support for automated reasoning with strategic enterprise architectures (R3)** requirement. In order to remind the reader and facilitate our assessment, we repeat Table 2.1 (depicted in Table 4.7) from Chapter 2 with requirement R3 for strategic enterprise architectures:

Summary of Requirements for Strategic Enterprise Architectures	
Support for Automated Reasoning	Formal rigor in specifications (R3.1)
	Support for Execution of Planning Process (R3.2)
	Reason with Different Shades of Goals (R3.2.1)
	Reason with Environmental Factors (R3.2.2)
	Support Selection of Best Strategic Alternatives (R3.2.3)
	Support Implementation of Strategic Alternatives (R3.2.4)

Table 4.7: Summary of Requirement R3 for Strategic Enterprise Architectures

As we are interested in automated techniques that fulfill all the requirements from Table 4.7, here we include only those automated techniques that minimally covers the concept of *goal* (due to sub-requirement R3.2.2), thus excluding techniques that only perform reasoning with process models and BPAs (e.g. business process modeling and BPA approaches, Section 4.5.3).

Table 4.8 depicts the assessment of approaches regarding the achievement of the **support for automated reasoning with strategic enterprise architectures (R3)** requirement and its sub-requirements. This table does not include requirement R3.1 (specifications with formal rigor) since all approaches achieve this requirement. After careful analysis of automated techniques, we can draw the following conclusions:

GORE reasoning techniques (lines 1-5). GORE techniques can deliver complex reasoning functionalities with goals by means of forward and backward reasoning algorithms. Forward reasoning algorithms work in a bottom-up fashion by estimating the level of satisfaction of top goals on the basis of the partial achievement values of lower level goals (Tropos, i* and KAOS forward reasoning). Conversely, backward reasoning algorithms work in a top-down fashion by recommending specific goal subtrees that achieve top system goal in a specified level (Tropos and i* backward algorithm) or by recommending specific goal subtrees that satisfy certain user-defined constraints (CGM). Although both types of techniques provide interesting insights about the satisfaction of goals in software engineering, they refraining from addressing goals in enterprise modeling. Consequently, such techniques are neither able to reason with multiple levels of abstraction in goal hierarchies nor able to take into account environmental factors that impact the achievement of enterprise goals (as can be observed in Table 4.8). They are also not able to support the enterprise planning process, like supporting the selection of best strategic alternatives (R3.2.3), according to prescribed by requirements from Table 4.7.

EM reasoning techniques (lines 6-8). EM frameworks borrow the

GORE concept of goal and its forward and backward reasoning techniques accordingly (e.g. BIM, URN). Consequently, a deficiency in the representation of multiple levels of goals leads to an inability to reason with goals in multiple levels of abstraction and to support reasoning with factors that may impact the achievement of enterprise's goals. As a consequence of that, such techniques cannot fully support the planning process. For example, BIM forward reasoning (line 6) is able to propagate lower-level values to infer the achievement of top company's goals, but is not able to generate specific strategic alternatives (denoted by a − sign at sub-requirement R3.2.3). In [126], the paper uses BIM forward technique for systematically analyzing business strategies in different scenarios (set of situations) in a framework for stress testing and similarly, the approach cannot generate strategic alternatives. In contrast, BIM backward reasoning (line 7) generates specific strategic alternatives (denoted by a ✓sign at sub-requirement R3.2.3), but it is not able to reason with multiple levels of abstraction in goal hierarchies.

Motivational and behavioral reasoning techniques (lines 9-16). Most of the techniques that combine motivational and behavioral concepts consider the automated design of process-control flows based on (operational) goals (lines 9, 12-16), by either applying planning techniques like Liaskos et al. [120], Bryl et al. [20, 21] and Asnar et al. [6] or using other variability techniques like Lapouchnian et al. [117]. Greenwood et al. [72, 73] (line 10) proposes a technology suite for controlling the execution of multiple processes controlled by autonomous agents. However, such techniques ignore the impacts of high-level, strategic goals on process design, also ignoring environmental factors that may impact strategic goals achievement. Consequently, such techniques cannot fully support the enterprise planning process, with support for the selection of best strategic alternatives (denoted by a ∼ sign at sub-requirement R3.2.3).

Lapouchnian et al. [116] and Morrison et al. [136] are two exceptions as both approaches are able to generate strategic alternatives from enterprise goals. Such strategic alternatives are subsequently implemented as business processes in a BPA. Nevertheless, both proposals ignore the existence of multiple goal levels and the existence of factors that impact the achievement of enterprise goals, and therefore, we consider them to not fully support the enterprise planning process (denoted by a $\sim$ sign at sub-requirement R3.2.3).

4.6 Other Relevant Theories

As the work presented in this thesis is grounded on conceptualization from Management literature, a number of theories in Management Sciences also present some connections with the research here reported. Although such theories do not necessarily incorporate motivational and behavioral conceptualization, they are included here due to their interconnections with the Management literature used as the baseline of our work (Section 3.3). Such theories inspired a number of works in Conceptual Modeling which are also presented in this section.

In particular, Section 4.6.1 reviews the concept of *value propositions*, whereas Section 4.6.2 discusses the concept of *capability-based* approaches. Section 4.6.3 describes the work of Michael Porter in competitive analysis. Although the work of organizational ontologies described in Section 4.6.4 slightly diverges from the previous sections, as ontologies are originated within Computer Science, the influences received from Management Sciences (e.g., Strategic Management, Organizational Theory, Marketing, etc.) justify their inclusion in this section. Finally, Section 4.6.5 concludes the description of other relevant theories by discussing their relations with the Management literature used in Section 3.3.

		R3. Support for Automated Reasoning			
		R3.2.1	R3.2.2	R3.2.3	R3.2.4
1	KAOS forward reasoning [119, 86]	–	–	–	–
2	Tropos forward reasoning [65]	–	–	–	–
3	Tropos backward reasoning [179]	–	–	✓	–
4	i* forward reasoning [92]	–	–	–	–
5	i* backward reasoning [92, 94]	–	–	✓	–
6	BIM forward reasoning [89]	–	✓	–	–
7	Mate et al. [126]	–	✓	–	–
8	BIM backward reasoning [89]	–	✓	✓	–
9	URN standard [158, 159]	–	–	–	–
10	Greenwood et al. [72, 73]	–	–	∼	–
11	Lapouchnian et al. [116]	–	–	∼	✓
12	Morrison et al. [136]	∼	–	∼	✓
13	Lapouchnian et al. [117]	–	–	∼	–
14	Liaskos et al. [120]	–	–	∼	–
15	Bryl et al. [20, 21]	–	–	∼	–
16	Asnar et al. [6]	–	–	∼	–

Table 4.8: Assessment of approaches against requirement R3 from Section 2.3

4.6.1 Value Theories and Business Modeling Ontology

Value Propositions in Management Sciences. The concept of *value proposition* was firstly proposed by Lanning and Michaels in their seminal work [114] as a benefit perceived by some target customer in acquiring a good minus the price paid for this good (i.e., value = benefit minus price). In order to exemplify the concept, they have used the example from IBM in the market segment of computers. They argue that although IBM does not provide the leading edge computer technology (in terms of power, speed and user-friendliness), IBMs technology reliability in terms of product robustness and customer service allows the company to lead its market segment. In achieving such trade-off between product technology vs. robustness and customer service, IBM grasped what the customer perceives as a benefit that s/he can modestly pay more for that.

Using the concept of value proposition as the basic ground, authors propose that a business (company) consists of a system for delivering superior value. Moreover, they defend that value propositions are one of the most basic factors for companies achieving and sustaining competitive advantage. As customers select products/services based on their perceived value compared to the competing alternatives, understanding what customers perceive as benefits is crucial to understand their choices and thus, to achieve competitive advantage.

Furthermore, tailoring company's value proposition according to customer segment's perceived benefits is not the only source for delivering superior value, but also propagating the value delivery in every element and activity performed by the company. Since its creation in 1988, the concept of value proposition has been enhanced by other researchers, such as Kambil et al. [101].

The remarkable role of value propositions in Strategic Management drove incorporation of the concept in some enterprise modeling approaches. These approaches are:

e^3 Value Approach. The e^3 value approach [66] is an economic value-based ontology derived from economic and business science literature that represents and analyze value models. In this ontology, the concept of value proposition is most closely related to the concept of *value offering* that captures a set of exchanges among actors of value objects. In this context, value objects correspond to objects which are valuable to one or more actors. They can be of different types, such services, products, monetary resources or even consumer experiences [67].

Value in ArchiMate. Another enterprise modeling framework that incorporates value-related concepts is ArchiMate. In ArchiMate [78], although the concept of value propositions is not explicitly incorporated, the *value* concept can capture the value of a product or service that makes some stakeholder to appreciate it (either for providing or for acquiring the product/service). For instance, *Be Insured* value represents the value that the *Provide Insurance* service aggregates to some client.

Since ArchiMate is a service-oriented enterprise framework in which the concept of service is the linking among its several layers (business, application and infrastructure layers), the association of the concepts of value and service allows the representation of the value aggregated by each layer to the overall architecture (thus, capturing the reasons for the existence of the layers). Moreover, by using the value concept, it is also possible to represent the value delivered by the overall architecture to the final customer.

Value Delivery Modeling Language (VDML). The Value Delivery Modeling Language (VDML) [76] from OMG provides a standard modeling language for the description and analysis of the operations of an enterprise with special emphasis on value creation and exchange. In VDML, a value consists of a measurable benefit delivered to a customer together with a business item (deliverable). The benefit may represent any intrinsic feature of the deliverable (e.g. composition, performance or weight) or other benefits such as price, a commitment to future purchases, trustworthiness, warranty or environmental impact. Typically, a business item contains multiple values and a given exchange may involve multiple business items. Therefore, the concept of *value proposition* in VDML subsumes the perceived benefit by the customer as a bundle of items and their corresponding values.

Value Ontology in UFO. On the basis of a careful literature review, Sales et al. [171] propose an ontological analysis and conceptual clarification of the notion of *value proposition* using the Unified Foundational Ontology (UFO). The analysis and clarifications build upon the Value Ascription Ontology (VAO) [4].

In the UFO-based analysis, value can be assigned to either value objects or value experiences (experience with some value object, such as using a good or service). The valuation ascription judgment depends on: (i) the value beholder (person who ascribes the value) and value beneficiary (person who enjoys the value) (e.g., in the case in which a father prepares a dinner for daughter, the father is the beholder, while the daugther is the beneficiary), (ii) the intrinsic properties of the object being valuated (e.g. a car with airbag, the softness of a mattress, the good quality of a class due to the content being conveyed), (iii) the context (e.g. water in the desert is more valuable than water in a dinner).

On the basis of the aforementioned theory of value ascription, the paper distinguishes between *value propositions* and *value offerings*. Value propositions are characterized as a value assertion resulted from the trade-off between benefits and sacrifices of acquiring some object, while a value offering consists of a promise made by an agent (offeror) to a group of agents (eligible market) to execute actions (e.g. allow to use a TV stream of service) under certain specific conditions (e.g. payment in return). Value propositions consist of “what” and “why” a customer values something, whereas value offerings consist of “how” value is delivered by the company.

Business Model Ontology (BMO) and Business Model Canvas

(BMC). An enhancement of the work of Lanning and Michaels in value propositions influenced the development of the Business Model Ontology (BMO) [148], which consists of the conceptual basis for the Business Model Canvas (BMC) Approach [149], both developed by Alexander Osterwalder in his PhD thesis.

The Business Model Canvas consists of a template for documenting business models of different companies by means of a visual notation. As described in Osterwalder's PhD Thesis [148], a business model consists of "a conceptual tool that contains a set of elements and their relationships and allows expressing a company's logic of earning money. It is a description of the value a company offers to one or several segments of customers and the architecture of the firm and its network of partners for creating, marketing and delivering this value and relationship capital, in order to generate profitable and sustainable revenue streams."

In order to describe business models, BMO identified nine building blocks that conceive a business model. These building blocks are segmented around four basic areas: (i) product (concerns the businesses in which the company is in), (ii) customer interface (concerns the target customers of the company), (iii) infrastructure management (address how the architecture of the company works together with its network of partners), and (iv) financial aspects (concerns the revenue model, cost structure and profit sustainability). Table 4.9 depicts the definition of the nine building blocks together with their respective areas:

The importance of BMO for the representation of companies business models has been acknowledged in Conceptual Modeling and the concept has been incorporated in a number of efforts, such the mapping from BMO to the ArchiMate enterprise framework [128] and the effort towards establishing a reference ontology for business models [3].

Area	Building Block	Description
Product	Value Proposition	A value proposition consists of a collection of products or services offered by a certain company that represent a benefit for a given customer segment
Customer Interface	Target Customer	A customer segment captures a given segment of customers that the company provides value
	Distribution Channel	A distribution channel consists of a means to get in touch with the customer
	Relationship	A relationship describes the kind of link the company establishes with customers
Infrastructure Management	Value Configuration	A value configuration describes the disposition of activities and resources necessary to create value for a given customer
	Capability	A capability consists of an ability to execute actions necessary to create value for the customer
	Partnership	A partnership is a cooperative agreement between two or more companies to create value for the customer
Financial Aspects	Cost Structure	A cost structure is the representation of all the financial means employed in the business model
	Revenue Model	A revenue model describes the logic employed by the company make to make money through a variety of revenue flows

Table 4.9: The nine BMO building blocks (extracted from [149])

4.6.2 Capability-Based Organizational Design

Resources and Capabilities in Management Sciences. Resource-centric theories in Strategic Management consider organizations as an aggregation of resources that allow the organization to gain or sustain competitive advantage [68, 11, 55]. On the basis of such definition, capabilities theories consider that the mere existence of strategic resources does not necessarily give organizations a competitive advantage, but rather it depends how such resources are organized. While resource-centric theories focus on the accumulation of resources, capability-based theories focus on configuring resources and capabilities towards adapting them to the environment [191].

The increased interest of capability-based theories in Strategic Management has been acknowledged also in Conceptual Modeling with the development of capability-driven approaches or the incorporation of the "capability" abstraction in enterprise modeling approaches. In particular, we can cite two approaches:

Capability-Driven Development. In [215, 15], authors propose a business/IT alignment approach for designing information systems from enterprise models. The approach is denominated Capability Driven Development (CDD), as capabilities consist one of the main abstractions of enterprise models.

The approach is divided into three phases. In the *enterprise and capability modeling* phase, enterprise modeling is performed for capturing a set of generic solutions (system designs) applicable to many different business situations. For that, the approach uses the concepts of *capability* (enterprises ability to achieve a business goal in a given certain context), *goals*, *key performance indicators (KPI)* (for monitoring goal achievement), *business processes* needed to accomplish the goals and *resources* required to execute a process.

In the *capability delivery context modeling*, the approach captures the different contexts in which the solutions should be applied. In its turn, context consists of any information that characterizes a situation in which a capability is provided. It is captured by means of *context elements* (e.g. geographical location). In this phase, *context indicators* are also captured in order to monitor the occurrence of a specific context situation in which a capability must be delivered.

Finally, in the *capability delivery patterns* phase, reusable solutions for achieving business goals under different contexts are represented. Each context defined for a given capability should match with the context in which the pattern is applicable.

Resources and Capabilities in ArchiMate. The ArchiMate language [9] incorporates *capabilities* and *resources* as means of aligning strategic decisions with the actual implementation of such strategic in terms of a target enterprise architecture. For that, authors use capabilities and resources as abstractions of more detailed enterprise elements (e.g., business processes, IT artifacts, etc.) used as the realization of the enterprise architecture. This incorporation thus enables one to build more stable enterprise descriptions which require less effort to maintain. In latter stance, this capability-based planning approach facilitates the discussion of business managers in terms which outcomes to achieve (e.g. better quality, lower costs, higher returns on investment), instead of the detailed means (processes, projects and IT applications) to achieve such outcomes.

In the approach, the capability-based planning approach is illustrated by means of two use cases. The first use case consists of an approach for capability enhancement of Toyota in order to better meet business goals, while the second approach performs IT portfolio consolidation in an energy supplier company. The latter uses a capability-based planning approach to eliminate redundant IT resources after three different company have been merged.

4.6.3 Porter Five Forces

Porter's Five Forces model [135, 172] consists of an approach that strives to explain the nature of competition within an industry environment. For that, Porter identifies five forces in a given organization's external environment that may influence competition. These are represented in Figure 4.1 and described as follows:

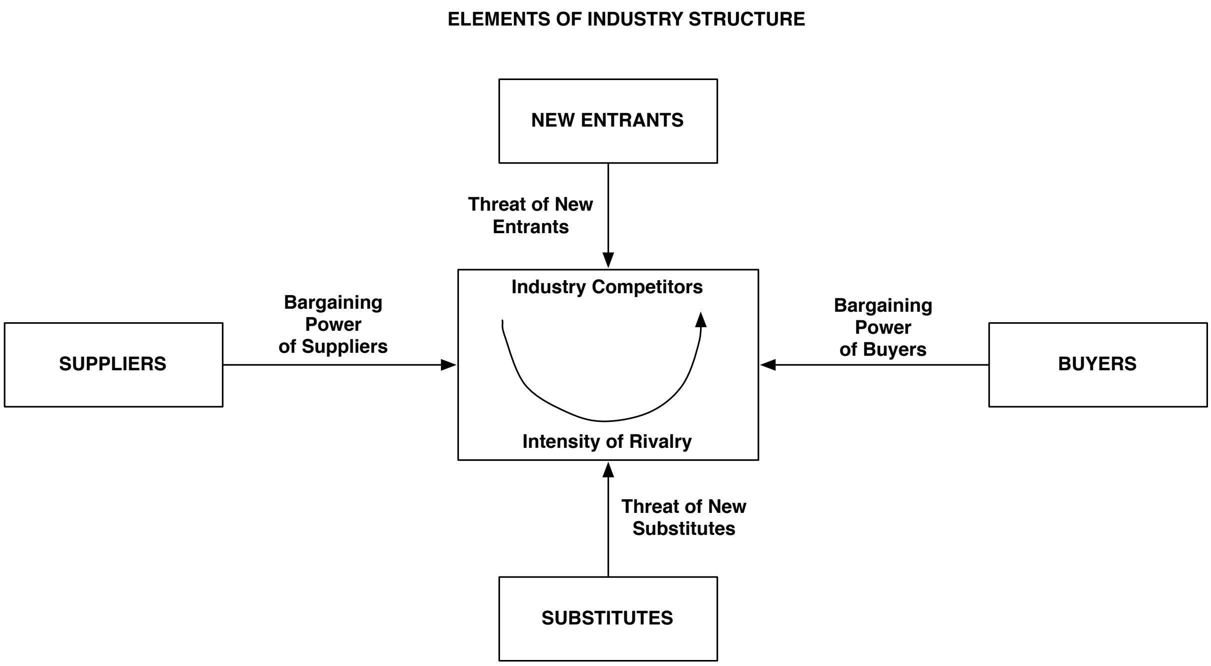


Figure 4.1: Porter's 5 Forces (extracted from [135]

- **Threat of New Entrants.** Porter argues that an industry sector is similar to a "club" in which companies are admitted by overcoming certain "barriers to entry" (e.g. economies of scale, customer loyalty and basic capital requirements). If high barriers exist, this encourages friendly competition among a reduced number of firms, while low barriers lead to a big group with fierce competition;

- **Bargaining Power of Firm's Suppliers.** As companies intend to charge as much as they can for their products, a tension naturally arises between companies and their suppliers. The winner is the one who has more choices (e.g. companies with multiple suppliers) or less to lose in the case the relationship ends;

- **Bargaining Power of Firm's Customers.** Customers intend to get the lowest prices with the highest quality. Their ability in achieving this depends on a number of factors, such as how much they buy, how informed they are, their willingness to try new alternatives, etc.;

- **Threat/Substitute Products.** Competition depends on which extent products from one company are replaceable by the ones from another (e.g. taxi services vs. Uber);
- **Intensity of Rivalry Among Competing Firms.** All of the previous factors contribute to the existence of rivalry among companies. This rivalry may converge to different situations, such as the competing companies to attack each other or peacefully co-exist or even form alliances, depending on the aforementioned factors.

The particularities of each industry may explain why companies adopt one particular strategy and thus such forces shape the industry structure and environment.

4.6.4 Organizational Ontologies

In the field of enterprise engineering, organizational ontologies lay down a common conceptualization and terminology for organizational environments in an attempt to capture the key elements of enterprises. In this context, although goal-related concepts intuitively appear as important building blocks of enterprises, few works in organizational ontologies define the concept of goal from an ontological point of view [2].

Among the works that include goal-related concepts, the widespread dissemination of goal-orientation paradigm in a number of areas of Computer Science inspired the development or incorporation of goals and objectives in organizational ontologies. The support for motivational and behavioral representation in some organizational ontologies approaches has been reviewed in (Section 4.4).

Contrasting with the works that define goals from an ontological point of view, a number of works in enterprise ontologies indeed mention goals and objectives, but in an informal and brief manner. In fact, in the course of defining organizations from an ontological standpoint, such works recognize the existence of goals, but use a simplistic notion of goals as building blocks of organizations.

For example, the Unified Foundational Ontology (UFO) defines an organization as an institutional agent formed by a number of other agents (physical, artificial or institutional agents) and a goal as the set of desired state of affairs (desired by the organization (agent)) [81]. In this context, although the concept of goal is foreseen by the ontology, UFO does not provide a refinement structure of the goal concept in terms of a more refined goal ontology. In the same vein, the TOVE Ontology [61, 79] defines an organization as a set of constraints on actions, resources, organizational units (including roles, positions and agents of the enterprise), goals, products, services, policies and the set of constraints that defines the external environment. In this context, a goal is a future state to be achieved by the enterprise, but no refinement structure is also defined.

In the OperA framework for the specification of multi-agent systems [48], an organization is defined by its externally observable objectives and by the means to achieve such objectives. Although the social structure (SS) model supposedly specifies objectives of the society, society's goals are indeed specified indirectly by means of roles. In its turn, roles are captured in terms of objectives, norms, rights and type of enactment. Such role's objectives are states of affairs expected to be achieved in the environment, but they do not present any refinement structure in terms of different types of goals.

The Enterprise Ontology [194] is slightly more refined ontology with the incorporation of different types of *purposes* (strategic purpose, mission, vision, objectives and goals). Although such purposes are ordered in terms of measurability and time-horizons following the order (objective, goal, mission, vision), a sharp distinction between those concepts is not provided. Finally, another enterprise ontology [46] proposed by Jan Dietz totally refrains from discussing goal-related concepts.

4.6.5 Comparative Analysis of Relevant Theories With Management Literature

This section analyzes the theories of this section in a perspective with the Management literature used as the baseline of our work (Chapter 3, Section 3.3).

The Strategic Management discipline is concerned with the long-term direction of the organization, determining the range of businesses in which the organization operates, the nature of economic and non-economic value the company delivers and how companies intend to gain and sustain competitive advantage over competing firms that provide similar services [187, 97, 172, 153]. In this context, the Management theories here reported share the common feature of striving to explain the nature of competition as well as to predict how to acquire and sustain competitive advantage.

In this context, value proposition theories (Section 4.6.1) capture how a given company delivers value to its customer's segments and thus, how this company can achieve and sustain its competitive advantage in its market segment. In its turn, the BMO incorporated the idea of value propositions in order to capture the overall business logic of some company to earn money. In order to represent such business logic, BMO makes use of a number of primitives, such as value propositions, customers, distribution channels, cost structure, revenue models, etc.

While value propositions theories explain that companies should craft its internal environment to deliver value to the customer and thus, achieve competitive advantage, capabilities-based theories presented in Section 4.6.2 take a similar approach by defending that company's success can be achieved by analyzing its internal environment as means of tailoring its capabilities and resources. In this context, capability-based theories offer a suitable abstraction to hide the complexity of enterprise descriptions. As business managers are solely interested in goals and how to achieve such goals at the strategic level, a detailed representation of the means to achieve such goals (processes, projects, applications) has little relevance at this level. Thus, the representation of capabilities and resources enables business managers to abstract from detailed implementations of business goals and focus on the means. This enables them to understand the current enterprise's status and which improvements should be made in order to leverage organization's competitive advantage by means of capabilities and resources. In its turn, Porter's Five Forces model (Section 4.6.3) explains the forces that shape competition in a given industry segment. Slightly diverging from such attempt to explain the internal and external forces of a given company and the nature of competition, organizational ontologies (Section 4.6.4) provide a definition of the internal organization's environment from an ontological standpoint, characterizing the organization in terms of key distinctions, but with little focus on the definition of goals and processes from an ontological point of view.

Comparing the four aforementioned approaches with the Management Theories used as the baseline of this thesis (Section 3.3), we can draw a parallel with conceptualization from the Strategic Layer, more specifically, with the concepts of *Mission* and *Strategic Goals*. Starting with value proposition theories/BMO, when performing goal planning at the Strategic Level (Section 3.3.1), Management literature mentions that organizations exist to aggregate some value to the external world by means of products or services. Indeed, this "value" mentioned by mission statements captures the "value proposition" notion from value theories. Furthermore, propagating down this value by means of the planning methodology corresponds to echoing the value down in every element and activity performed by the company, as defended by value proposition approaches. In comparison with capability-based approaches, Management literature cites the existence of the hierarchy of goals and the means (processes) to achieve such goals. In this context, capabilities and resources consist of suitable abstractions that could be also used for the representation of the means to achieve such goals in our approach. In relation to Porter's work, as the Five Forces model explains the forces that shape competition in a given industry segment, this could have supported us to elaborate strategic goals from the perspective of a given company. Although value propositions theories, BMO, capabilities and resources and Porter's model present some overlapping aspects with the conceptualization used in this thesis, they are not here incorporated. We leave the investigation of the conceptual integration between the work developed in this thesis and such conceptualization for future work. Finally, although organizational ontologies characterize the organization from an ontological standpoint, such definition has little focus on goals and processes and thus, they have not been used in this thesis as well.

4.7 Conclusions

By analyzing the four aforementioned areas (GORE frameworks, EM frameworks, Business Process Modeling and BPA approaches and motivational and behavioral approaches), we draw a number of conclusions regarding the state of the art in the representation and reasoning with strategic enterprise architectures. We enumerate such conclusions as follows:

1. Regarding the representation of motivational perspective, GORE approaches (Section 4.5.1) use the concept of goal to represent requirements of a target software system. Therefore, as multiple levels of

abstraction among goals (strategic, tactical, operational) and environmental factors that impact goal achievement are particular concerns to strategic enterprise architectures, they do not cover both types of concerns;

2. Interestingly, although both types of concerns are crucial in the representation of strategic enterprise architectures (as depicted by our motivating example (Chapter 2) and the enterprise planning process), EM approaches (Section 4.5.2) also refrain from recognizing them. Even ArchiMate that consists of the most expressive EM approach does not cover tactical and operational goals. Regarding motivational and behavioral approaches (Section 4.5.4), the same weaknesses have been noticed. Although some approaches inherit BMM conceptualization to represent multiple levels of goals, they do not provide clear distinctions for such BMM concepts. Other approaches create their own goal ontologies to represent either strategic goals (e.g. "Increase sales") or operational goals (e.g. "create order") interchangeably. Further, none of the approaches acknowledge the existence of environmental factors that may impact the achievement of goals during enterprise planning;

3. Regarding the interconnection between motivational and behavioral perspectives, GORE approaches capture the concept of process in terms of plans, but do not provide constructs for detailing the process-control flow. In contrast, EM and hybrid (motivational and behavioral) approaches capture both process and the refinements of their control-flows in terms of activities. However, none of the approaches (including business process modeling and BPA approaches) incorporate modeling constructs for capturing the social perspective of business processes;

4. There is a gap in the representation of all required properties in moti-

vational and behavioral perspectives in the same strategic enterprise architecture approach. A number of approaches either focus on motivational representation, like GORE (Section 4.5.1) and EM frameworks (Section 4.5.2) or behavioral perspective like Business Process Modeling and BPA Approaches (Section 4.5.3). An exception are the hybrid approaches (Section 4.5.4) that focus on both perspectives simultaneously, although those approaches also lack a comprehensive proposal that encompasses all the expressiveness requirements elaborated in Chapter 2;

5. A direct consequence of deficient support in the representation of strategic enterprise architectures is reflected in its automated reasoning. Some steps of the enterprise planning process are not supported by any approach. For example, reasoning techniques cannot explore the existence of goals of different shades (requirement R3.2.1) and analysis regarding the achievement of goals based on environmental factors (requirement R3.2.2). Two exceptions here are BIM forward and backward reasoning techniques (line 6-7) that are able to reason with such environmental factors, but cannot address reasoning with multiple levels of abstraction in goal hierarchies. Most of the approaches cannot support the generation of strategic alternatives (requirement R3.2.3). Only three approaches (Tropos, i* and BIM backward reasoning) can generate strategic alternatives, but cannot explore goals of different shades. Furthermore, only two approaches (Lapouchnian et al. [116] and Morrison et al. [136]) can support the implementation of strategic alternatives by means of processes (requirement R3.2.4). Overall, any approach can fully support the execution of all steps of the enterprise planning process (requirement R3.2).

Chapter 5

StrategIc ENterprise Architecture (SIENA) Modeling Language

In order to achieve the requirements discussed in Chapter 2 (Section 2.3), this chapter introduces the StrategIc ENterprise Architecture (SIENA) modeling language for the hierarchical representation of motivational and behavioral concepts in enterprise architectures. Methodological guidelines that specify how to elaborate, refine and operationalize motivational concepts by means of behavioral concepts are also provided. The content of this chapter is an updated and revised version of the paper [29]. The chapter is organized as follows: Section 5.1 introduces the general structure of the SIENA modeling language, describing concepts and relations of both Goal and Operations View, whereas Section 5.2 presents the methodological guidelines for the elaboration of models using the concepts provided by the language.

5.1 The SIENA Modeling Language

As presented in the introduction (Section 1.6), Figure 5.1 depicts a schematic representation of the main contributions of this thesis which consists of the StrategIc ENterprise Architecture (SIENA) Modeling Language, an automated strategic planning reasoning technique with SIENA and the Azzurra Modeling Language.

This chapter presents the contributions of the SIENA Modeling Framework by introducing the SIENA Modeling Language (highlighted in Figure 5.1 by a red circle). In particular, we present the SIENA modeling language, its abstract syntax (meta-model), concrete syntax (notation), the semantics of modeling concepts [83] and methodology for the specification of concepts. The modeling constructs of SIENA are exemplified by means of the motivating example of the metal manufacturing company introduced in Chapters 1 and 2. Next chapters use the SIENA language introduced in this chapter to present the other contributions of this thesis presented in Figure 5.1.

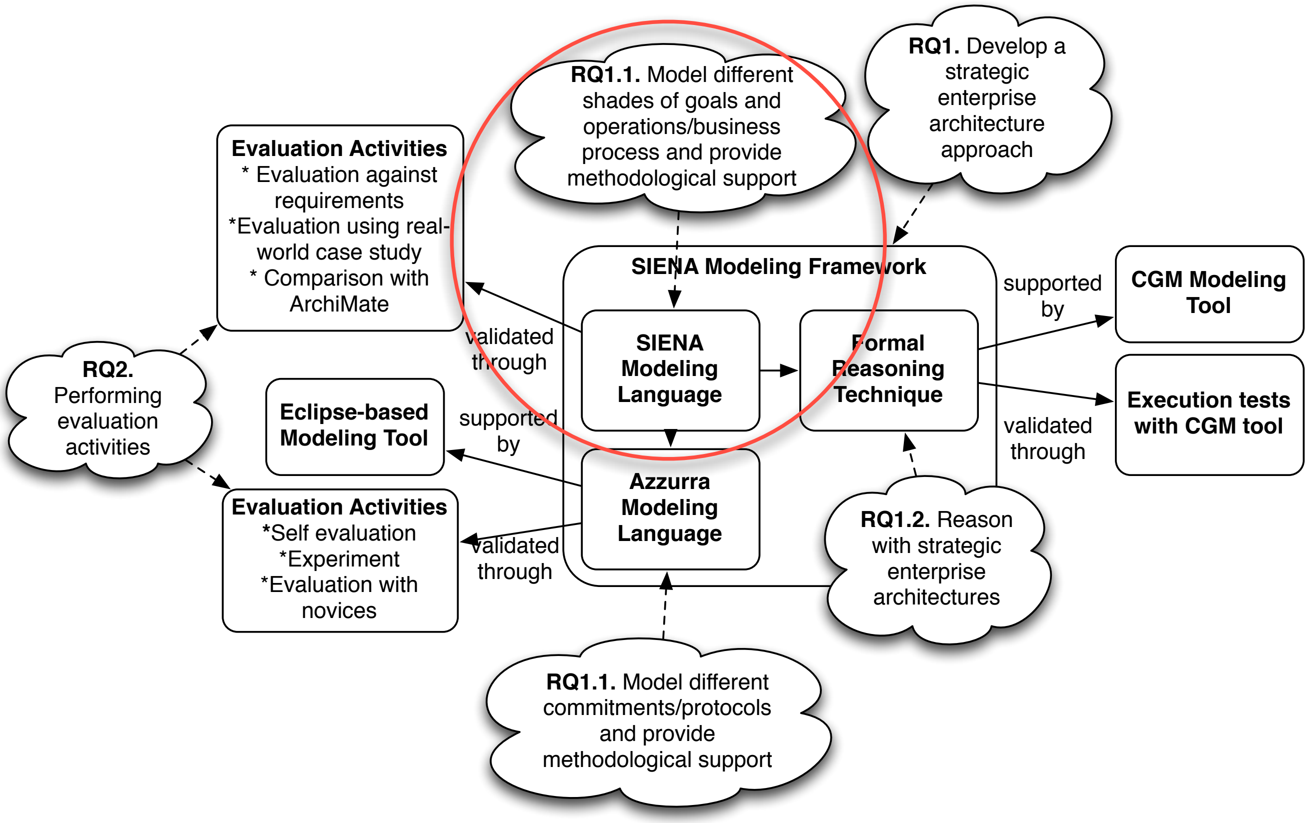


Figure 5.1: The Contribution of this Chapter in the Context of the Overall Thesis

In order to achieve expressiveness in motivational and behavioral perspectives (R1) and part of the traceability between motivational and behavioral perspective requirements (R2) described in Section 2.3, the SIENA modeling language is structured in terms of a hierarchical, layered structure of motivational and behavioral concepts. Motivational concepts are modeled within the Goal View, whereas behavioral concepts are captured within the Operations View. Within the Goal View, the framework follows the same three-layered distinction proposed by Management Sciences (i.e., *Strategic*, *Tactical* and *Operational* Layers)(Chapter 3, Section 3.3), whereas the Operations View is structured in four layers of abstraction (*Operations*, *BPA - Level 0*, *BPA - Level 1* and *BPA - Level 2* Layers).

Figures 5.2 and 5.3 depict SIENA's abstract syntax (meta-model) with its concepts, relations and cardinality constraints, while Figure 5.4 shows the concrete syntax (notation) used for each modeling concept presented in the meta-model.

The remainder of the chapter describes the semantics of SIENA's concepts and relations in Sections 5.1.1 and 5.1.2. Furthermore, in order to achieve the traceability between motivational and behavioral perspectives requirement (R2), the methodology for the specification and refinement of SIENA's concepts is also specified in Section 5.2. In particular, the methodological steps starts from the top layer (*Strategic* Layer), drilling further down until the description of the *BPA - Level 0* Layer in Chapter 7.

5.1.1 Goal View

This section introduces the goal-related concepts of our framework. Within our *Strategic Layer*, SIENA contains the concepts of *Mission*, *Vision* and *Strategic Goals*, defined on the basis of a consolidation of different views in Management Sciences (Section 3.3).

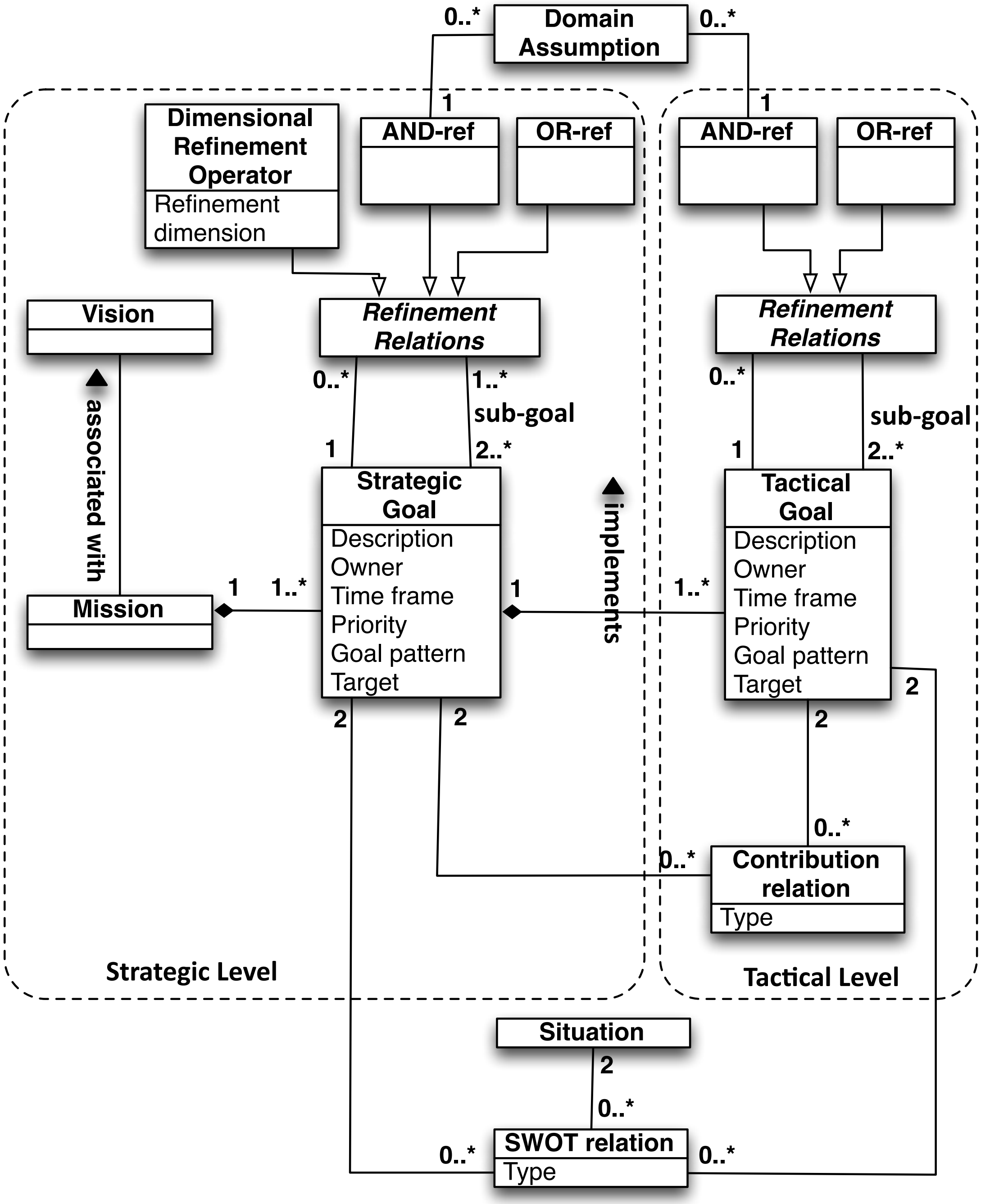


Figure 5.2: The SIENA Modeling Framework Meta-Model (Strategic and Tactical Levels from Goal View)

Mission. A mission defines a formal expression of an organization's purpose, i.e., the reason why the organization exists by aggregating some

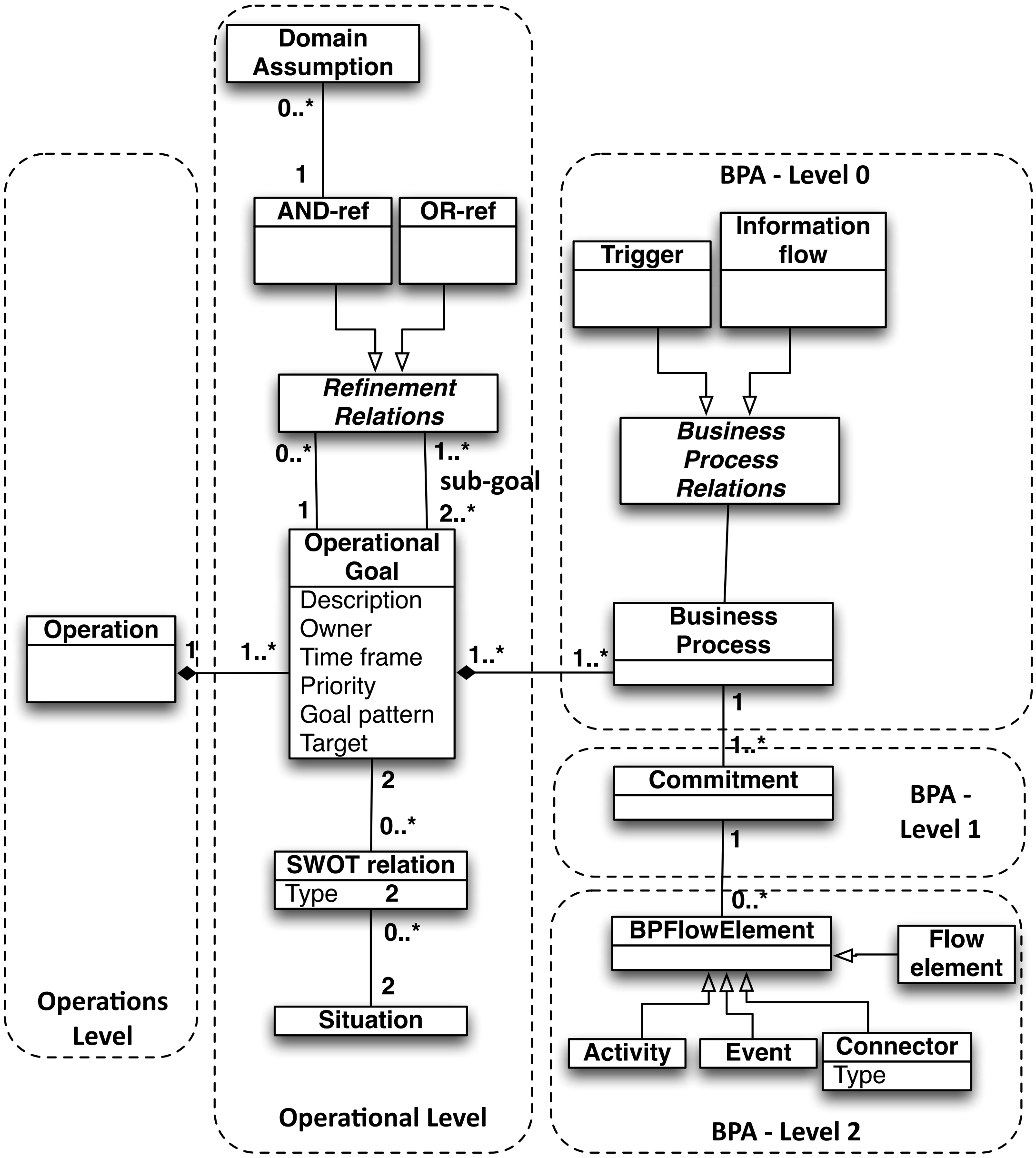


Figure 5.3: The SIENA Modeling Framework Meta-Model (Operational Level from Goal View and Operations View)

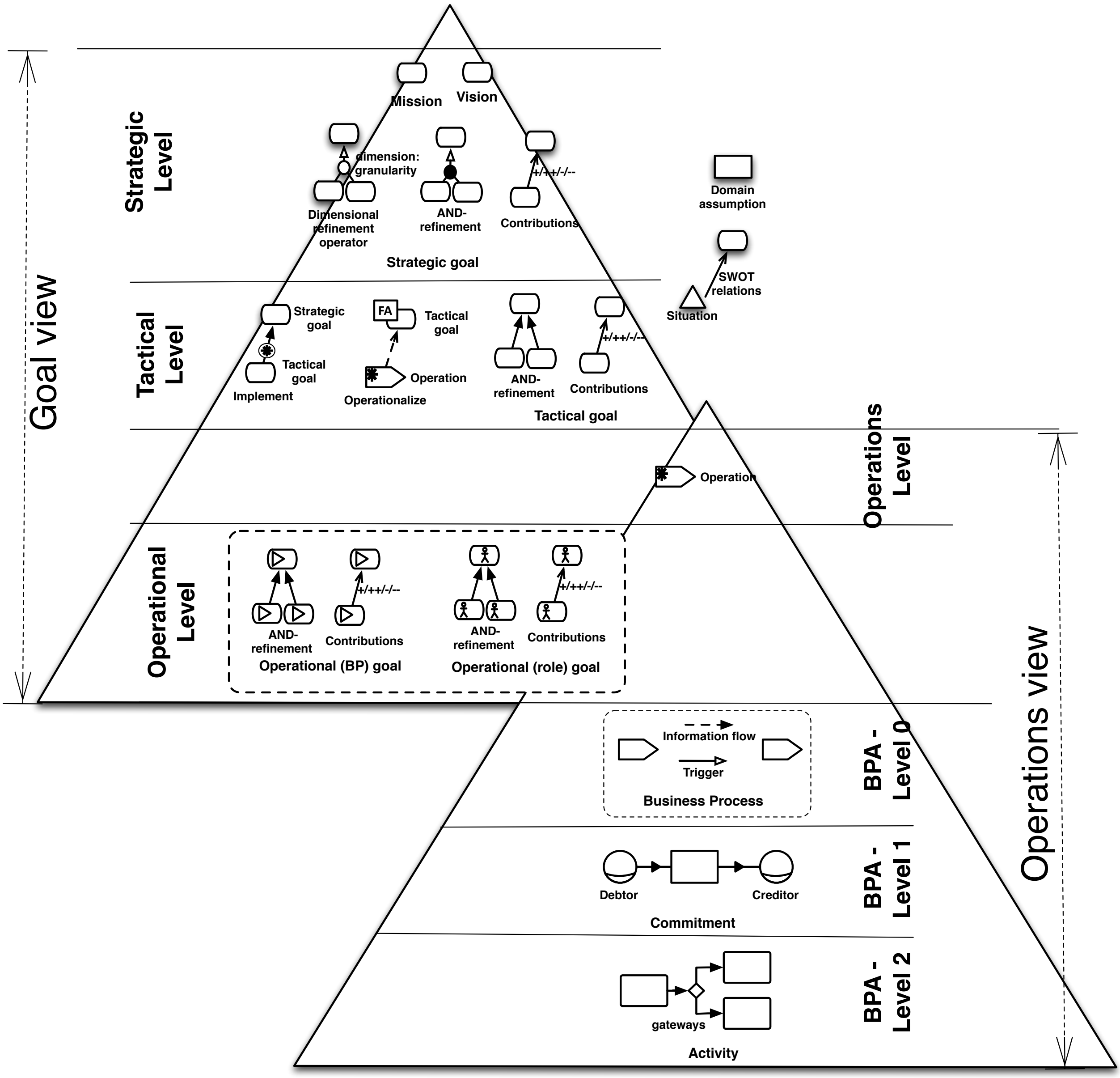


Figure 5.4: Hierarchy of Layers and Graphical Concepts in the SIENA Modeling Framework

sort of value to the external world.

Vision. Comprises a description of a desired state of the company, meant to close the gap between the current reality and a potential future.

Strategic Goals present key characteristics in Management that we consolidate as follows:

Strategic Goals. Represent goals that specify concrete outcomes that must be achieved to measure the achievement of the mission, reflecting the organization's strategy to achieve success in business. Strategic goals are global to the *overall organization* as the entire organization is responsible for their achievement. They are also long-term, lasting between two and five years.

Dimensional Refinement Operator. As Strategic Goals are global to the entire organization, they represent the problem space of a given enterprise, defining the space of *all* alternatives goals that can be implemented by the enterprise. To precisely characterize such variability and unambiguously characterize Strategic Goals, our framework introduces the distinguishing feature of *refinement dimensions* and *dimensional refinement operators*. *Refinement dimensions* correspond to different properties along which goals can be characterized (e.g, location, time or product types properties) extracted from data warehouse literature [192]. They are used to guide the refinement of Strategic Goals using *dimensional refinement operators*. To exemplify the use of *refinement dimensions* and *dimensional refinement operators*, consider the "Increase sales by 2% over 3 years" goal in Figure 5.5. This parent goal defines the space of all possible *locations* (countries, in this example) in which the company operates. Therefore, this parent goal can be refined into the following sub-goals: "Increase sales in Italy by 2% over 3 years", "Increase sales in Germany by 2% over 3 years" and "Increase sales in France by 2% over 3 years". Another refinement of the same parent goal across *time* (within the *year* granularity) is also depicted in Figure 5.5, yielding the "Increase sales by 2% over 1st year", "Increase sales by 2% over 2nd year" and "Increase sales by 2% over 3rd year" sub-goals.

Strategic Goals Relations. Besides dimensional refinement operators, Strategic Goals can be also related by AND/OR-relationships and positive and negative (+/++/-/–) contributions. AND-refinements structurally decompose goals into sub-goals following domain particularities, while OR-refinements depict alternatives for goals to be achieved. For example, "Increase sales by 2% over 3 years" is AND-refined in terms of "Maintain gross margin over 3 years" and "Increase volume sales by 2% over 3 years" in Figure 5.5. Furthermore, the three refinements (refinement by time, AND-refinement and refinement by location) of "Increase sales by 2% over 3 years" consists of three different alternatives to achieve this top goal. Finally, positive and negative (+/++/-/–) contributions among Strategic Goals may be used to depict how they influence each other inside the *Strategic Layer*. For instance, increasing volume sales may positively affect a sales increase every year. For this reason "Increase volume sales by 2% over 3 years" positively (+) contributes to "Increase sales by 2% over 1st year", "Increase sales by 2% over 2nd year" and "Increase sales by 2% over 3rd year" sub-goals in Figure 5.5.

Within the *Tactical Layer*, Management literature (Section 3.3) commonly specifies tactical goals either as responsibilities of functional areas or tactics to achieve strategic goals. We consolidate both views in our definition of *Tactical Goals*:

Tactical Goals. Represent goals that specify particular ways for fulfilling Strategic Goals with the available resources and capabilities of the company. Tactical Goals have no dimensions, but rather depict particular solutions ("tactics") for each point of the refinement dimension in order to fulfill a Strategic Goal. Tactical goals typically have shorter time horizons (usually from one to three years) than strategic goals.

In order to exemplify this discussion, we use the refinement of "Increase sales by 2% over 3 years" Strategic Goal across the *location* dimension (depicted in Figure 5.6). For one of the points of the *location* dimension (Italy) represented by the Strategic sub-goals ("Increase sales in Italy by

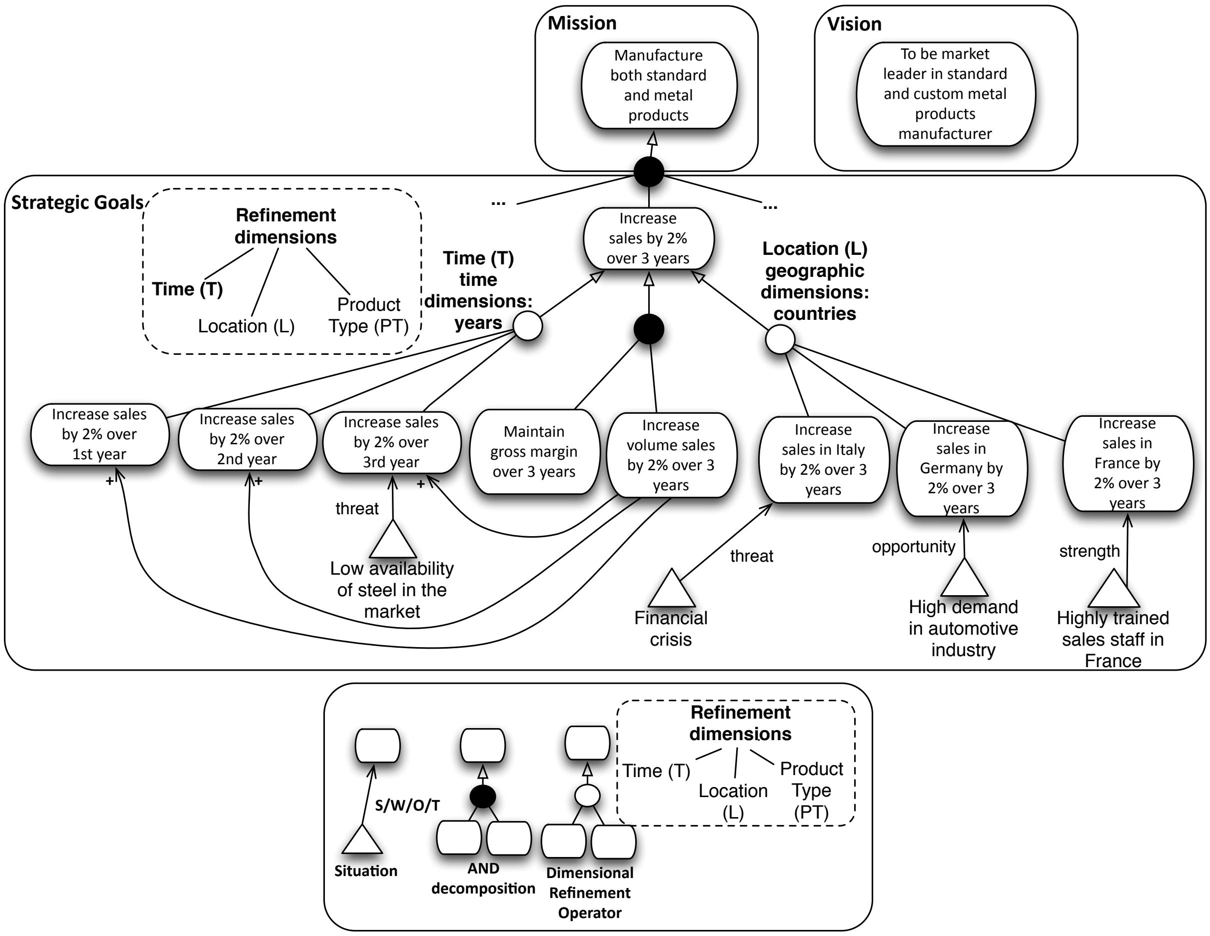


Figure 5.5: Strategic Goal Hierarchy, illustrating Strategic Goals and Dimensional Refinement Operators

2% over 3 years"), there are two tactics for increasing sales, i.e., promotions ("Increase sales in Italy by 2% over 3 years through promotions") and create new sales channel ("Increase sales in Italy by 2% over 3 years by opening new sales channels" Tactical Goal). For the other point of the *location* dimension (France), training sales people corresponds to a tactic for increasing sales ("Increase sales in France by 2% over 3 years by training sales staff" Tactical Goal).

Tactical Goals Relations. Concerning the relation of Strategic and Tactical Goals, it said that Tactical goals *implement* Strategic Goals. In the example, it is said that promotions ("Increase sales in Italy by 2% over

3 years through promotions") is the tactic that *implements* the increase of sales ("Increase sales in Italy by 2% over 3 years"). Further, Tactical Goals may be structurally refined into sub-goals by means of AND-relationships and several alternative Tactical Goals may be also represented by means of OR-relationships. Finally, they can be also related by means of positive and negative (+/++/-/–) contributions that depict how Tactical Goals influence each other inside the *Tactical Layer*. For example, "Diversify customers" is one of the alternatives for opening new sales channels ("Increase sales in Italy by 2% over 3 years by opening new sales channels") and positively contributes to a sales increase through promotions ("Increase sales in Italy by 2% over 3 years through promotions") in Figure 5.6.

Once the organization has established its competitive requirements to achieve success in business (Strategic Goals) and subsequently has devised particular ways (Tactical Goals) for implementing such requirements, it has to plan the implementation of such goals with the available company's capabilities by means of the concept of operation. This discussion is reflected in Figure 5.6 with the Tactical Goals connected to operations in the *Operations Layer*.

Within the *Operational Layer*, as Management literature (Section 3.3) provides a simplistic treatment for the specification of operational goals, our framework starts with the same definition of this discipline and subsequently refines it:

Operational Goals. Operational goals correspond to the results that must be achieved in the course of performing the organization's operations. Our framework further details their definition by arguing that they represent a description of milestones the operation must reach in order to ensure that they are indeed planning the execution of tactics. Operational goals can be further refined with respect to the entities that are responsible for

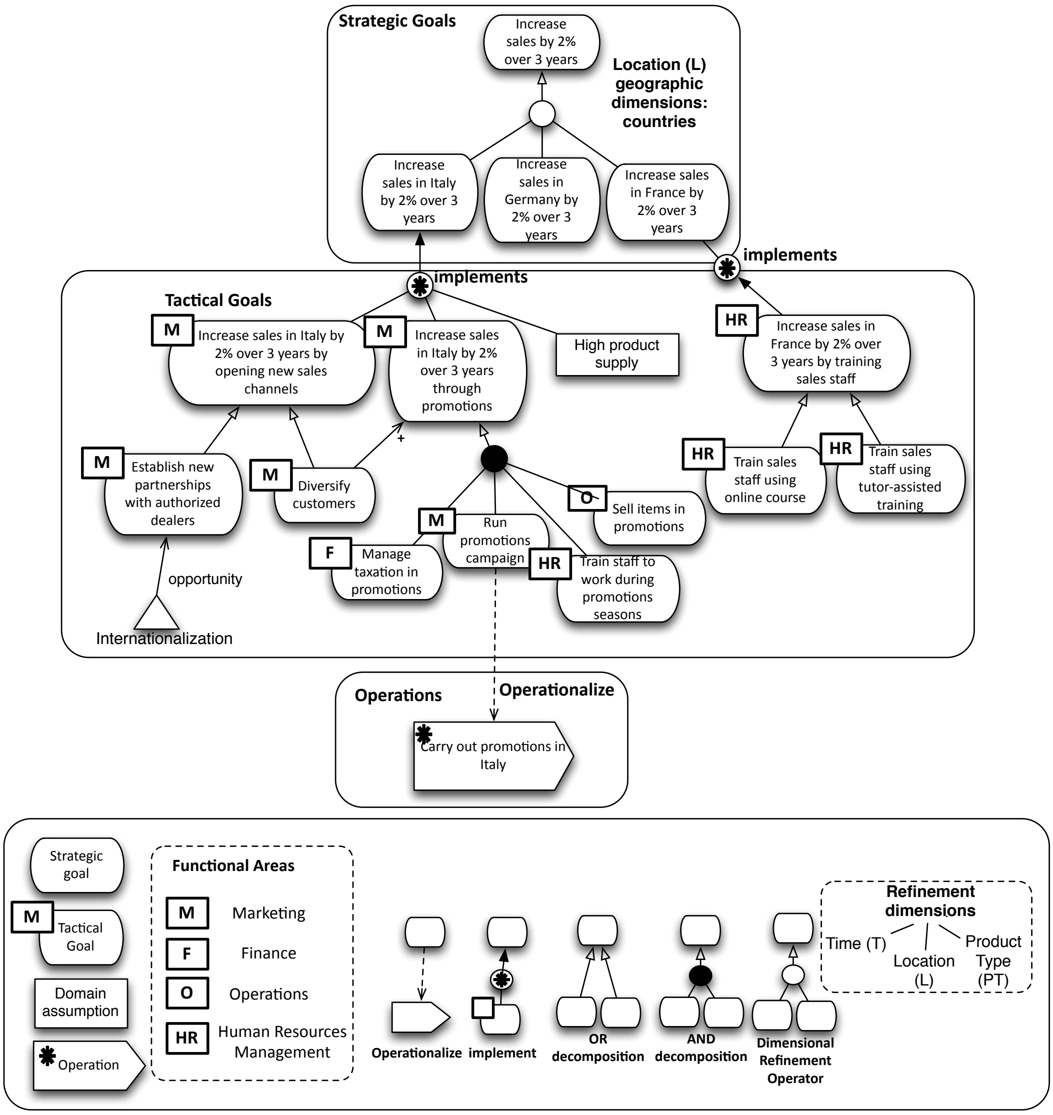


Figure 5.6: Tactical Goal Hierarchy, illustrating Tactical Goals and Operations

their achievement as follows:

(Operational) Role Goals. Correspond to goals that specify the results to be achieved by *roles and individuals* in the course of the performing their daily work. In Figure 5.7(b), "Choose items for promotion" and "Choose promotions price" consist of operational goals assigned to roles of the company.

(Operational) Business Process Goals. Correspond to goals that represent the final state to be achieved by a *business process*. The concept of *Business Process* is explained in Section 5.1.2. In Figure 5.7(b), “Advertise items in promotion” is a business process goal as it reflects the final state to be achieved by the “Advertise items in promotion” business process.

Operational Goals Relations. Operational Goals may be related by AND/OR-relationships to represent refinements among them as well as positive and negative (+/++/-/–) contributions.

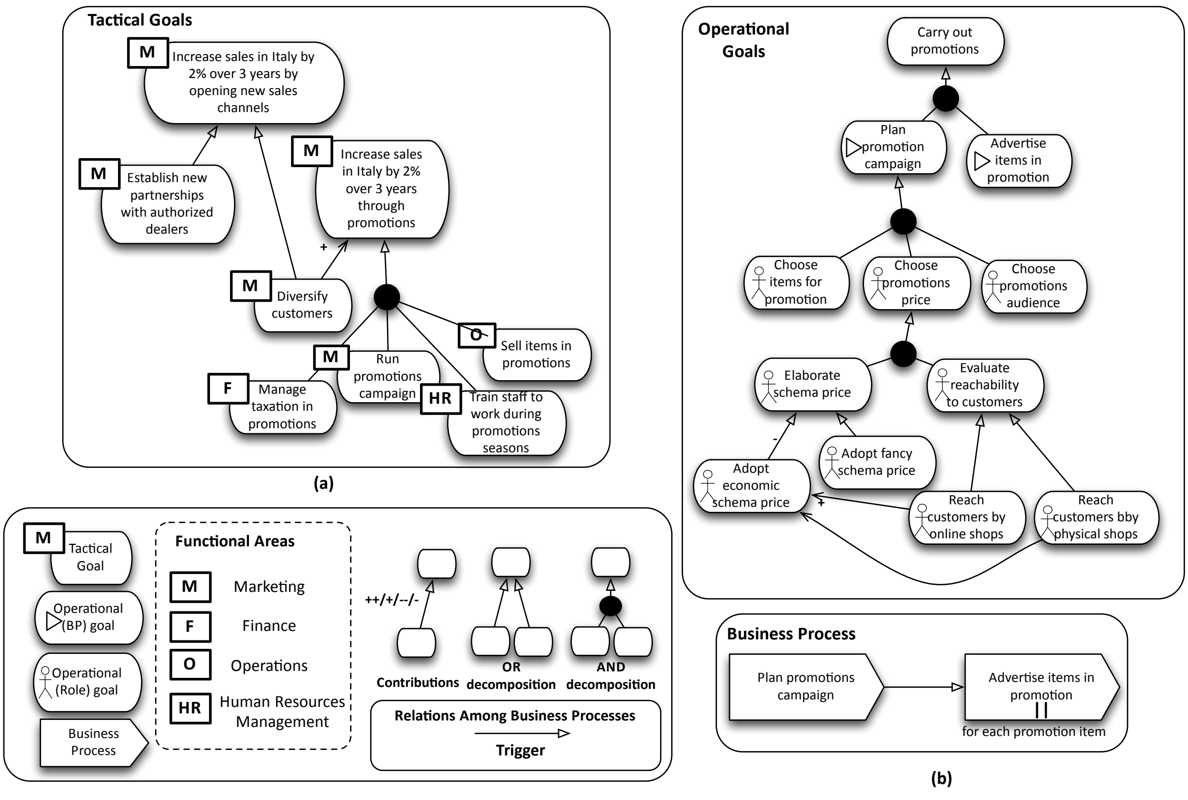


Figure 5.7: Operational Goals and Business Processes Hierarchy

In order to graphically represent the relations of Operations and Operational Goals and Business Processes, every SIENA Operation has a container for its respective Operational Goals and Business Processes. For example, “Carry out promotions in Italy” Operation (Figure 5.6) is refined into a container with its Operational Goals and Business Processes in Figure 5.7(b). Consequently, the root Operational Goal in the top of the container ("Carry out promotions") corresponds to the final state to be achieved by the "Carry out promotions in Italy" Operation. This top goal is then refined into Operational Role Goals and Operational Business Process Goals. Operational Business Process Goals correspond to the final state to be achieved by business processes and for this reason, every Operational Business Process Goal (e.g., "Plan promotions campaign", "Advertise items in promotion") has a business process associated. These business processes ("Plan promotions campaign", "Advertise items in promotion") are represented in the container below together with the respective relations among them.

Situation, SWOT Relations and Domain Assumptions. As one of the purposes of our modeling framework is to enable managers to adequately plan enterprise's goals and the corresponding operational elements that satisfy them, during the enterprise planning activity is important to foresee the potential future scenarios that facilitate or hinder the achievement of enterprise's goals (i.e., SWOT factors) together with assumptions about the environment. Therefore, our framework inheres the concepts of Situation and Domain Assumption from BIM framework. Situations are represented by triangles attached to goals by means of arrows annotated with the type of influence of situations on goals, whereas Domain Assumptions are represented by means of rectangles attached to goals. Figure 5.5 admits that a financial crisis may threaten the achievement of the "Increase sales in Italy by 2% over 3 years" Strategic Goal. Further, for both tactics to work for this goal (new sales channel and promotions), analysts assume a high supply of products for Italy (Figure 5.6).

5.1.2 Operations View

While the concept of *Operation* is central to the Management literature as a process that transforms inputs into useful outputs, our framework distinguishes between the concepts of *Operation* and *Business Process*. In this context, Operation and Business Process are the central concepts within the *Operations Layer*, whereas Business Process and its relations are the central concept within the *BPA - Level 0 Layer*:

Operation. Consists of a high-level process in charge of planning the execution of a specific tactic. A given operation encompass both *what* has to be achieved (Operational Goals) to concretize the tactics as well as *how* to conduct operational steps to achieve such tactics (business process). As operations plan the implementation of a given strategy, it is said that an operation *operationalizes* Strategic or Tactical Goals in our framework, i.e., operations are solutions for Strategic/Tactical goals.

The concept of business process inheres the same definition of *Operation* from Management Sciences as follows:

Business Process. Consists of an activity conducted with the purpose of transforming a set of inputs into useful outputs (products or services) using some sort of transformation process. In contrast with Operations, business processes intend to produce products or provide services to the final customer.

To exemplify the concepts of *Operation* and *Business Process*, we use Figures 5.6 and 5.7. In Figure 5.6, one can see that the organization decided to either use promotions or open new sales channel as tactics for increasing sales in Italy and therefore, "Carry out promotions" is the *Operation* used to plan the execution of the promotion tactics. In its turn, the "Carry out promotions" Operation consists of collections of operational goals and business processes (depicted in Figure 5.7(b)). The operational goals specify certain milestones to be achieved during the planning of promotions, such as to choose how many promotions are required and decide what to offer in each promotion ("Choose items for promotion"), choose promotions price and audience ("Choose promotions price" and "Choose promotions audience") and advertise items in a promotion ("Advertise items in promotion'). Finally, "Run promotions campaign" and "Advertise items in promotion" business processes are the entities that are responsible for indeed executing the promotions and advertising the items in promotion.

In order to depict all the value-adding activities conducted by the enterprise, such activities need to be connected accordingly. For that, we use Value-Added Chains from Management literature (Section 3.3) to define the following relations:

Relations Among Operations/Business Processes. Business processes may be related by means of horizontal relations that depict consumer-producer relations among them. Horizontal relations may be divided into trigger and information relations:

Trigger Relation. A trigger relation indicates that one instance of some business process triggers/starts another business process. Figure 5.7(b) and 5.8(b) shows an example of a trigger relation in which the process "Plan promotions campaign" triggers the execution of "Advertise items in promotion". When one instance of a business process activates multiple instances of the target element, this is distinguished by a double mark in the process that is activated multiple times together with its corresponding frequency of activation (in this case, the "Advertise item in promotion" is activated for each item in promotion).

Information Relation. An information flow relation depicts information or products exchange between the involved business processes. Figure 5.8(a) depicts this type of relation by indicating that instances of the three business processes exchange information during execution.

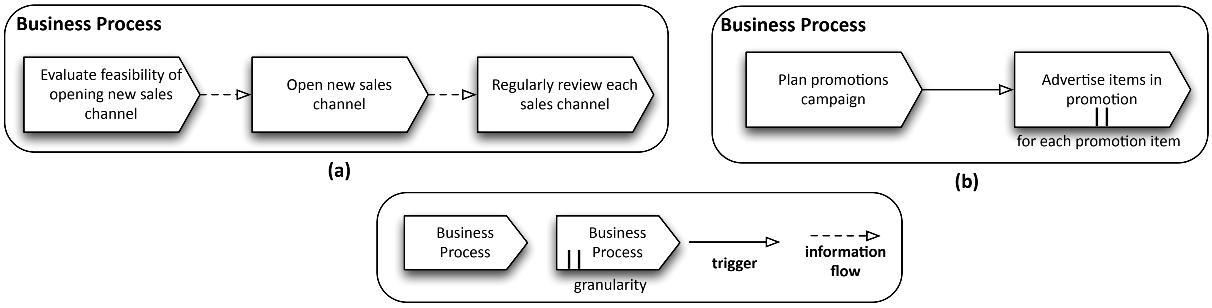


Figure 5.8: Trigger and Information Relations Among Operations/Business Processes

5.2 Methodological Guidelines for Goal-Driven Design of Operations Architecture

This section provides methodological guidelines that prescribe how to elaborate, refine and operationalize goals by means of operations and business processes in the SIENA modeling language. In order to prescribe such guidelines, as goals and operation planning occurs at formalized, step-by-step procedures in companies, we start by describing managers' concerns during goal and operations planning extracted from Strategic Planning literature. Subsequently, we explain how these concerns should be specified in our modeling framework. Although Strategic Planning literature mentions the existence of both a (top-down) deliberated and (bottom-up) emergent strategy formation process [135], we here focus on a traditional, top-down strategic planning for goal definition and implementation, leaving as future work the bottom-up strategy formation.

5.2.1 Guideline G1: Elaborate Mission and Vision Statements

At the *Strategic Level*, the first managers' step comprehends the articulation of organization's mission and vision as means of providing a general sense of direction for the company.

Mission and Vision Elaboration. The guideline is to elaborate a mission statement that reflects the value the organization intends to deliver to the external world. For *profit* companies, given that organizations can be either manufacturing or service organizations [163, 123], value aggregation is performed by enumerating the products or services the company produces. Furthermore, to "Make profit" [172] is always an additional mission that must be captured that also justifies the existence of *profit* companies.

For *non-profit* companies, the mission statement should capture other forms of value that provide social justification and legitimacy of the existence of the organization. For instance, Greenpeace's mission reflects this aggregation of value as "... Greenpeace's goal is to ensure the ability of the earth to nurture life in all its diversity..." [71].

The guideline for the elaboration of vision statements is to enumerate the products and services which are currently not implemented by the organization, but there is an intention to address them on the company's portfolio in the future.

Mission and Vision statements do not have a refinement structure in our framework as each of the concepts just refer to a specific product, service or aggregated value. When the company is engaged in the production or delivery of diversified products and services, this should be captured as distinct mission and vision statements.

In order to depict the full integrated hierarchy of mission, vision, strategic goals, tactical goals, operations and domain assumptions, Figures 5.5 and 5.6 are combined to produce Figure 5.9. In Figure 5.9, it is possible to see the metal company's mission and vision elaborated as a value delivered by a profit company.

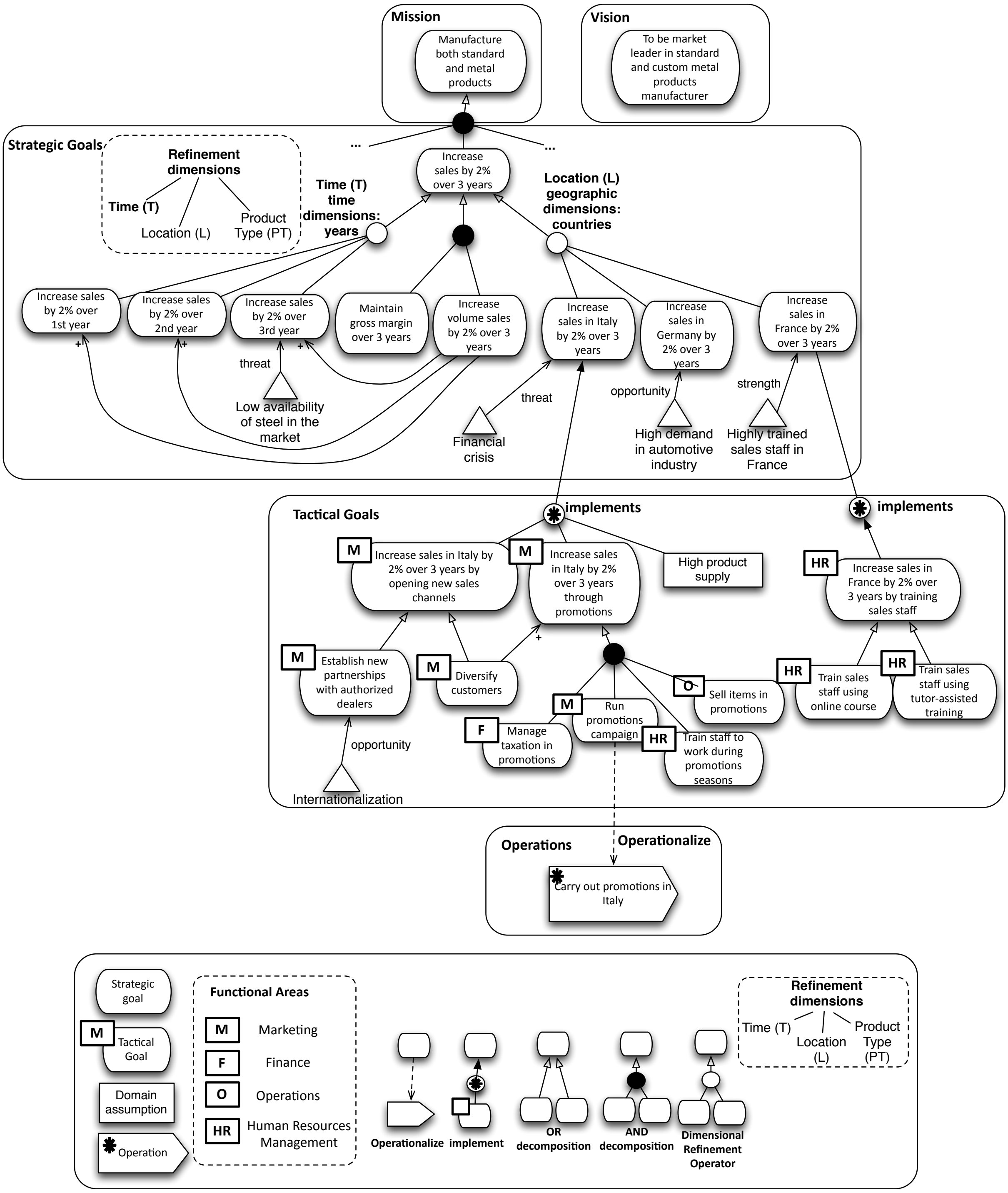


Figure 5.9: Hierarchy of Strategic Goals, Tactical Goals and Operations

5.2.2 Guideline G2: Elaborate Strategic Requirements

Strategic planning within the *Strategic Level* intends to guide the organization to achieve a successful position in a competitive environment, while achieving its goals [153].

Strategic Goals Elaboration. In order to compete, managers first identify *external aspects* that impact the ability of the organization to surpass its competitors. Such aspects includes three external sources of influence [97, p.55]: (i) the macro-environment (composed by regulative, political, economic, social, technological and environmental pressures), (ii) the industry sector (composed by the organizations producing the same products or services) and (iii) competitors and market (composed by organizations inside the same industry sector, with different characteristics and competing on different bases).

Following, *internal aspects* that enable the organization to gain a competitive advantage such as capabilities, resources and competencies are also evaluated. With such aspects in hands, the organization defines how it intends to compete and thus, a *strategic intent* is elaborated. This strategic intent is then used to elaborate company's Strategic Goals. For instance, the metal manufacturing company decided to compete on the basis of low manufacturing costs as this strategic intent will allow the company to achieve an advantage over its competitors and then, become a market leader. After identifying the strategic intent of low manufacturing costs, the company elaborates the "Increase sales" strategic goal in order to allow the company to become a market leader.

Other (real) example of strategic intent is the Acer PC manufacturer [34, p.492] that identified that Dell (competitor) competes on the basis of low manufacturing costs. This could represent an external threat for Acer that may lead Dell to become the market leader in computers. Based on an internal evaluation of its assets, Acer decided to gain competitive advantage based on management philosophy of highly motivated employers in order to increase production. Therefore, Acer elaborated the "Increase sales" Strategic Goal. With the elaboration of this Strategic Goal, Acer intended to become the market leader supported by an internal capability.

Our definition of Strategic Goals highlights them as *concrete outcomes* to be achieved by the *overall organization*. As outcomes, they need to be expressed in quantitative terms (targets) in order to reflect desired values for the organization, as for example, desired financial levels (e.g. desired sales, profit levels, rates of growth, dividend levels, share valuations), market-based outcomes (market share, customer service), among others [97, 34]. Their time frame must be also set up in order to allow their subsequent measure and determine their achievement. It is also important to mention that strategic goals always refine one specific mission, depicting the strategic requirements to be achieved in the context of that specific mission.

As the "Increase sales" strategic goal elaborated by the metal manufacturing company needs to be expressed in concrete terms to be achieved in a defined time frame, Figure 5.9 shows the "Increase sales by 2% over 3 years" strategic goal elaborated as a concrete outcome (target) of a desired level of sales (increase sales by 2%). As this strategic goal refines the "manufacture both standard and metal products" mission, the desired increase of 2% in sales refers to standard and metal products.

Strategic Goals Refinement Rules. Dimensional Refinement Operator. Dimensional refinement allows one to AND-decompose a goal with respect to a number of *refinement dimensions* introduced in Section 5.1.1. A dimension is introduced when a Strategic Goal has different operationalizations for different parts of the problem space. For example, there exist different solutions for increasing sales in Italy, Germany and France

("Increase sales by 2% over 3 years" goal in Figure 5.9) and therefore, the *location* is an eligible *refinement dimension*. In contrast, a dimension may be not applicable for a particular strategic goal, as for example, if we have a strategic goal "Build better products", but the company has only one manufacturing plant, location is not a refinement dimension for the goal. The following rules can be applied when using dimensional refinement: (i) **time dimension**: used when seasonal variations of business aspects (e.g., toys sales increase during Christmas season) may impose different operationalizations for the Strategic Goal; (ii) **location dimension**: used when the company presents a distributed organizational structure across distinct locations (e.g., sales departments for different countries) and the way in which the company pursue the Strategic Goal varies according to place under consideration; (iii) **product, service, customer type dimensions**: products, services and customers usually have a number of properties that characterize them (e.g., patients under 20 years old, different metal products, etc.) and operationalizations of the Strategic Goal varies according to the values that such properties may assume.

AND-refinement. Strategic Goals can be also AND-refined by following structural domain rules or based on dimensional refinement. Refinement based on structural domain rules is applied when there exists a mathematical formula that relates domain variables and enables one to structurally decompose a goal into sub-goals using this formula. For example, once we know the profit stemmed from sales can be described by the formula *salesProfit = numberSoldItems * profitMarginPerItem* and managers intend to increase this profit ("Increase sales profit by 2% over 3 years" goal), one can increase volume sales (*numberSoldItems*) and maintain profit margin, yielding the following goals: "Increase volume sales by 2% over 3 years" and "Maintain gross margin over 3 years" (Figure 5.9). An alternative decomposition of the same root goal could also consider an increase in the profit margin, yielding “Maintain volume sales over 3 years” and “Increase gross margin by 2% over 3 years” as sub-goals.

OR-refinement. Alternatives strategies (strategic goals) may be also considered for the achievement of strategic goals. For example, Figure 5.9 depicts three different strategies (alternative refinements) to achieve the “Increase sales by 2% over 3 years” strategic goal, i.e., the refinement by time, an AND-refinement and the refinement by location.

5.2.3 Guideline G3: Elaborate Tactical Requirements and Operations

Within the *Tactical Level*, the strategy is put into action by creating “tactics” that are particular ways of implementing the achievement of Strategic Goals with the deployment of organizational assets [187, 153].

Tactical Goals Elaboration. For the elaboration of Tactical Goals, Tactical goals specify particular solutions (“tactics”) for fulfilling Strategic Goals. In this sense, different tactics must be found to implement each point of the refinement dimensions introduced during the Strategic Goals Refinement. This discussion has been exemplified in Section 5.1.1 with the “Increase sales by 2% over 3 years” Strategic Goal refined in terms of the *location* refinement dimension and implemented by offering promotions and opening new sales channel (in Italy) or alternatively, by training sales people in France (depicted in Figure 5.9).

Every organization usually has two different types of tactics to implement Strategic Goals, i.e., *initiatives* or *established responsibilities*. Initiatives correspond to single projects that establish mechanisms for implementing Strategic Goals (e.g., create promotions, open a new sales channel, establish a new business process). Usually, such initiatives are executed one time and once completed, they have operated changes within the company environment that better enable the company to achieve its strategic goals. In contrast, established responsibilities correspond to the responsibilities of every functional area that need to be repeatedly executed so that the functional area accomplishes its part of the organization's strategy (e.g. manufacture products (Operations), provide monthly budget statements for departments (Finance), sell 1200000 units at average price of $27 (Marketing), etc.). Notice that responsibilities of functional areas may be the result of the adoption of initiatives. Furthermore, time frames for the achievement of tactical goals must be set up, depending on whether they consist of initiatives or established routines. For initiatives, this time frame is just a deadline, while for functional responsibilities, time frames are recurrent schedules.

To exemplify this distinction, as Figure 5.9 specifies initiatives (create promotions, new sales channels and training) for the "Increase sales in Italy by 2% over 3 years" and "Increase sales in France by 2% over 3 years" Strategic Goals, Figure 5.10 and 5.11 adds established responsibilities that implement both strategic goals in Italy and France. In Figure 5.10, responsibilities of functional areas are specified for Italy ("Manufacture 900000 products at average cost of $19 in Italy", "Increase manufacturing productivity by 2% in Italy" and "Have scrap rate of 3% or less in Italy" from Operations and "Keep outstanding accounts below $300000 in Italy", from Marketing). These functional responsibilities have been extracted from our motivating scenario from Figure 2.2 (Chapter 2). In Figure 5.11, the same four established responsibilities are represented for France. Observe that as tactics consist of particular solutions for achieving strategic goals for different points of the refinement dimension, different rates of increase in manufacturing productivity have been specified for Italy (Figure 5.10) and France (Figure 5.11).

Implements-relationship. Implements-relationship needs to be specified in order to denote that strategic goals are implemented by tactical

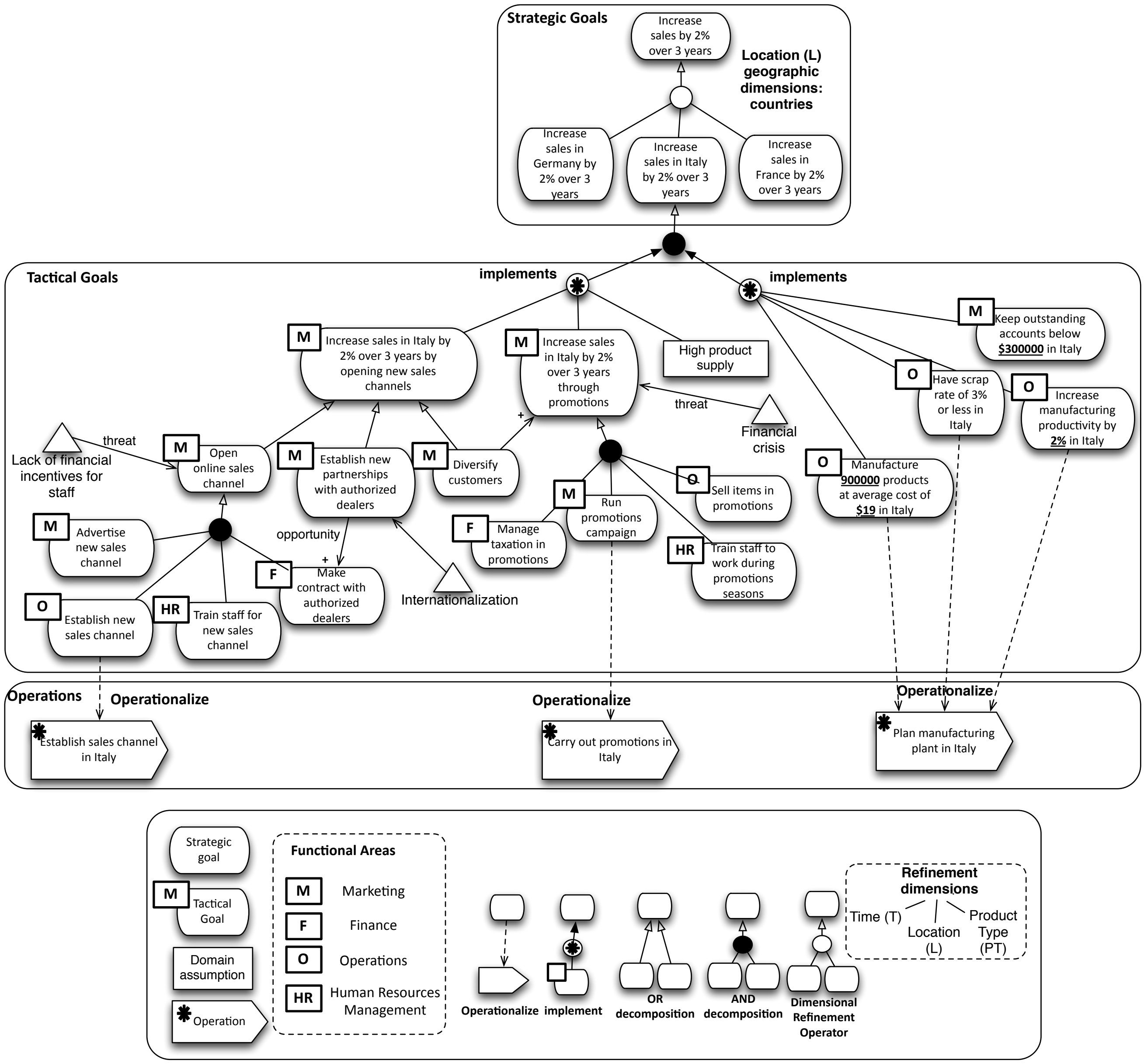


Figure 5.10: Tactical Goals Divided into Initiatives and Established Responsibilities in Italy

goals accordingly. As an implements-relationship has an AND semantics, as all of its sub-elements (sub-goals and domain assumptions) need to be satisfied in order to satisfy the parent goal. For example, in order to increase sales in Italy ("Increase sales in Italy by 2% over 3 years" strategic goal), new sales channels and promotions need to be adopted ("Increase

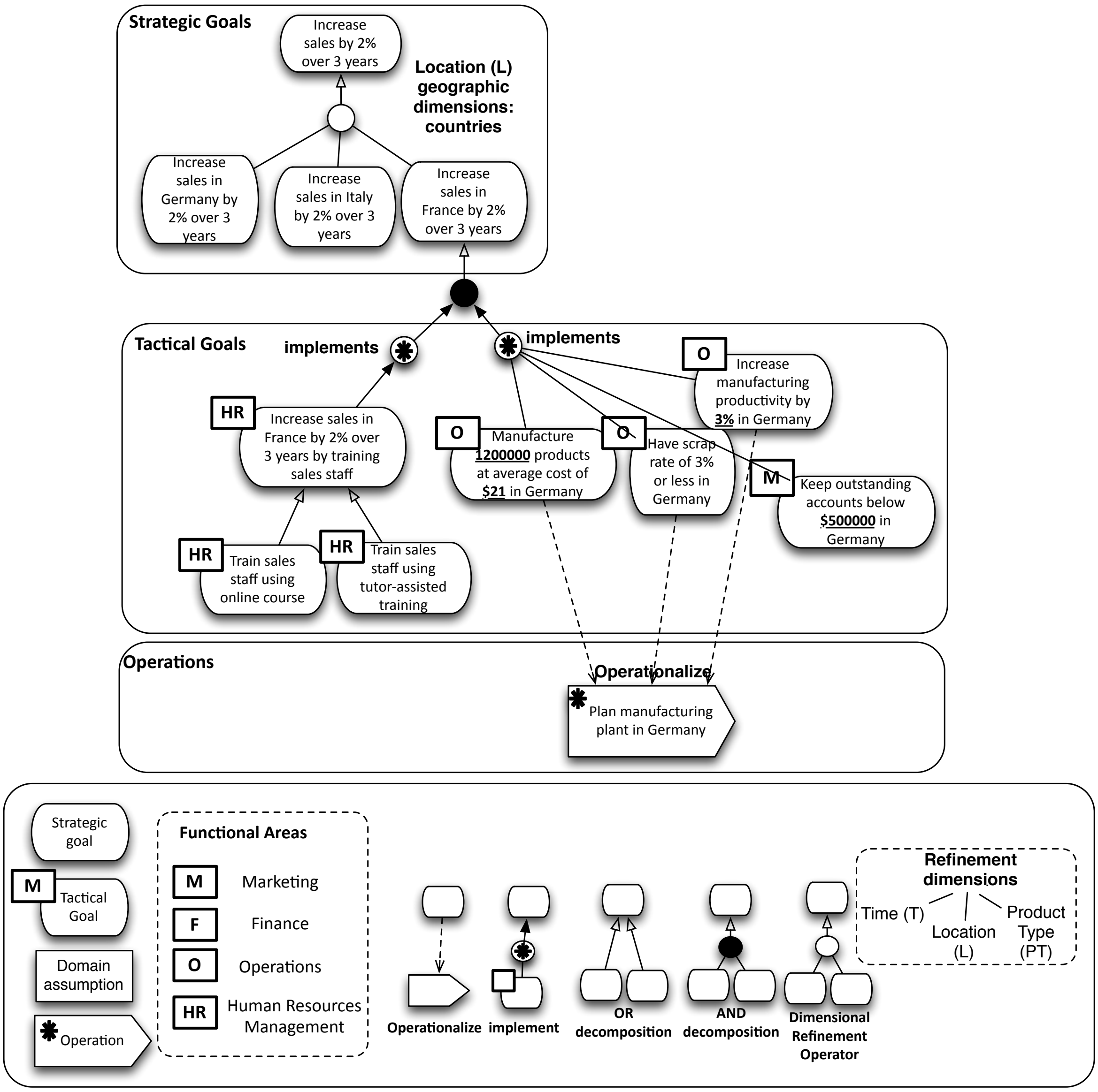


Figure 5.11: Tactical Goals Divided into Initiatives and Established Responsibilities in France

sales in Italy by 2% over 3 years by opening new sales channel" and "Increase sales in Italy by 2% over 3 years through promotions", respectively) under the assumption that there will be a "High product supply" (Figure 5.9).

Observe that Tactical Goals inhere the properties of parent goals that have been refined through dimensional refinement, i.e., the Tactical Goal

"Increase sales in Italy by 2% over 3 years through promotions" inheres the same properties of the refinement across location from the 'Increase sales in Italy by 2% over 3 years" Strategic Goal.

Each leaf level Strategic Goal has to be *implemented* by one or more Tactical Goals (tactics), otherwise, strategies will be not effective. Among such Tactical Goals, initiatives or established responsibilities must be specified. Inversely, each Tactical Goal *implements* one and just one Strategic Goal to avoid confusions between tactics that implement different Strategic Goals.

Tactical Goals Refinement Rules. AND-refinement. After finding solutions for points of refinement dimensions (tactical refinement), managers must AND-refine such solutions across the responsibilities of each *functional area* of the company. For instance, in order to increase sales in Italy, offering promotions or opening sales channel correspond to two tactics that pertain to the responsibilities of the Marketing area. In its turn, other functional areas of the company have also responsibilities in the context of promotions. This is reflected in Figure 5.9 with the "Increase sales in Italy by 2% over 3 years through promotions" AND-refined into four distinct goals, each of them representing the responsibility of each functional area. Functional areas are represented in our model by attaching squares with their first letter to goals (see Figure 5.9).

OR-refinement: a Tactical goal is OR-refined if there are different alternatives for achieving the same Tactical Goal. In our example, two alternative types of sales channels can be opened, i.e., by finding new partners to distribute the products or by finding new customers. Therefore, the "Increase sales in Italy by 2% over 3 years by opening new sales channels" is OR-refined into "Establish new partnerships with authorized dealers" or "Diversify customers" (Figure 5.9).

Tactical Goal Operationalization and Operations Modeling.

The refinement of Tactical Goals finishes when it is possible to plan and schedule the achievement of a Tactical Goal by assigning it an operation. In this case, it is said that an operation *operationalizes* a Tactical Goal which corresponds to the final state to be achieved by its corresponding operation. Tactical operations can be scheduled and executed with a certain frequency in order to achieve the Tactical Goal.

In order to ensure the tactics are indeed implemented, for each tactical goal, there must be at least one operation responsible for planning the execution of the corresponding tactics. For example, Figure 5.9 depicts a refinement of the tactical goal "Increase sales in Italy by 2% over 3 years through promotions" into other four tactical goals and an operationalization of such goal by one operation ("Carry out promotions in Italy").

5.2.4 Guideline G4: Elaborate Operational Requirements and Business Processes

At the *Operational Level*, the execution of tactics is planned by planning the expected results from organization's daily operations [153]. In our framework, expected results are delivered by means of setting the Operational Goals together with the business processes that deliver such results and their corresponding time frames for achievement. In order to facilitate the reading, we here repeat Figure 5.7, thus resulting in Figure 5.12.

Operational Goals Elaboration. As the Tactical Goal corresponds to the final state to be achieved by the operation that *operationalizes* such Tactical Goal, the elaboration of Operational Goals indeed starts by refining this Tactical Goal into intermediate milestones that compose its corresponding operation. These milestones are elaborated by specifying *which* results the operation must accomplish, regardless *how* this is accomplished. For the company's operations to be valuable, milestones must be elaborated considering that they need to add value to the final product.

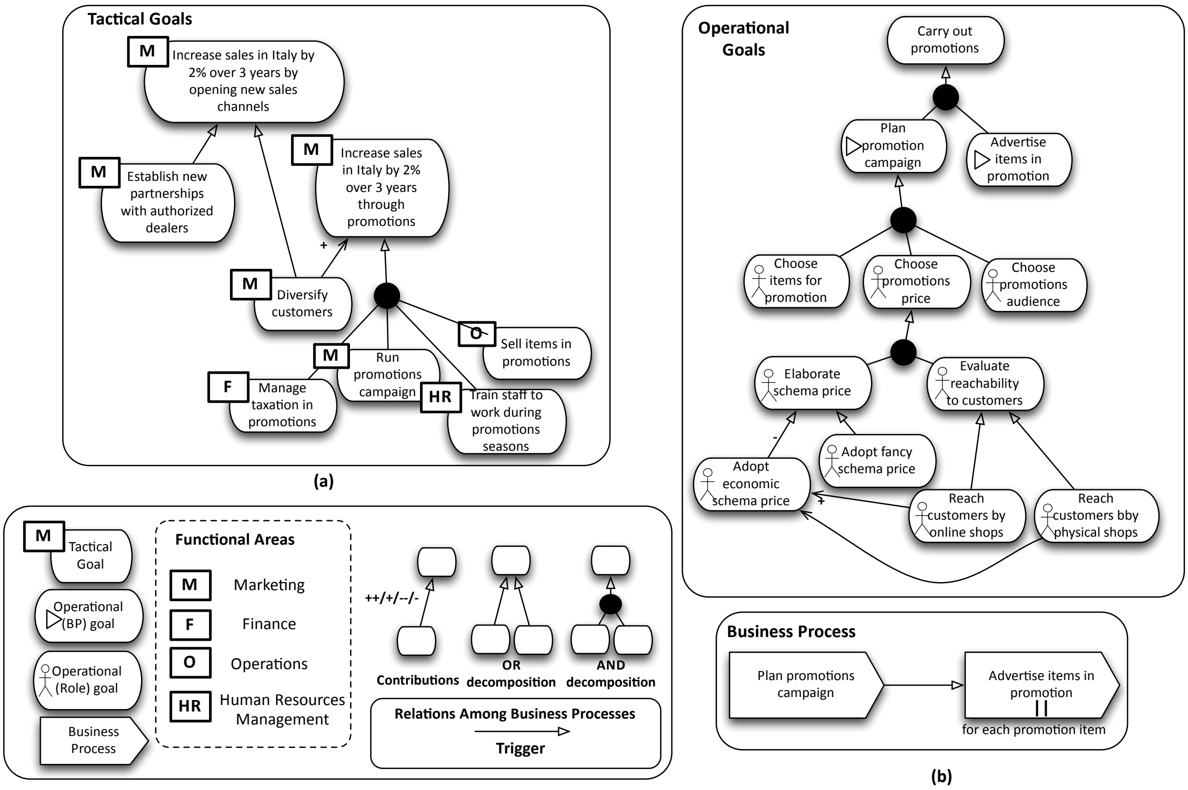


Figure 5.12: Operational Goals and Business Processes Hierarchy

In order to make such discussion more concrete, consider the “Carry out promotions in Italy” Operation in Figure 5.9 that represents the Operation responsible for planning the execution of the promotions tactics. Besides representing an Operation in our framework, “Carry out promotions in Italy” can be also interpreted as the final state to be achieved by such Operation, thus originating the “Carry out promotions” Operational Goal in Figure 5.12(b). This Operational Goal must be refined into operational milestones that correspond to value-adding responsibilities and therefore, this refinement yields the “Choose items for promotion”, “Choose promotions price”, etc. Operational Goals in Figure 5.12(b).

Observe also that the level of granularity reached in the representation of operational goals (i.e., whether the modeler decides to represent role goals or business process goals) depends on the modeler’s purposes with such enterprise model. In other words, if the modeler desires to have the finest-grained level of representation, the representation may reach the level of role goals to depict the value adding responsibilities of business processes. Alternatively, if the intention is to merely represent how the goals of business processes are linked to the overall company's strategy, then the level of business process goals may be reached. Section 7.4 provides a deeper discussion about how to refine operational goals and link them to business processes.

Operational Goals Refinement Rules. AND-refinement. An AND-refinement is used for structurally decompose a Tactical Goal (operationalized by a given operation) into intermediate Operational Goals (milestones) necessary for the execution of some tactics. An example of milestones refinement has been provided in Section 5.1.2. **OR-refinement.** An Operational goal is OR-refined if there are different alternatives for achieving the same Operational Goal.

Operational Goals Operationalization and Business Process Architecture Modeling. As Operational Goals may be achieved by either roles or business processes, the refinement of Operational Goals finishes when it is possible to find a business process whose final state corresponds to the Operational Goal under consideration. When a greater level of granularity should be considered, the refinement may finish when it is possible to assign roles for the satisfaction of Operational Goals (Figure 5.12(b)).

Besides the representation of business process goals together with their corresponding business processes, the relations among business processes need to be represented accordingly. For that, managers have to connect business processes by means of trigger/information relations to form a Value Added Chain (VAC) (Section 3.3), i.e., a sequence of activities (business processes) conducted by the enterprise to create a product or deliver a service.

5.2.5 Guideline G5: Elaborate Situations and Domain Assumptions

Situation Modeling. As SWOT analysis intends to spot the conditions in company's environment that affect the achievement of its goals and the nature of this impact, analysts should spot the internal enterprise's conditions (strengths/weaknesses) and external (opportunities/threats) and represent them as situations and domain assumptions attached to goals. In particular, situations may be suitable for devising SWOT factors that affect the ability of the company to surpass competitors in the *Strategic Layer*. In the *Tactical Layer*, situations may be useful for reasoning about the applicability of certain tactics in certain specific contexts. In Figure 5.9, one can see the "high demand in automotive industry" as an opportunity for increasing sales in Germany and the "low availability of steel in the market" as a threat for increasing the sales in the 3rd year.

5.3 Summary

This chapter introduced the StrategIc ENterprise Architecture (SIENA) Modeling Language that consists of a Goal and an Operations View for the respective representation of motivational and behavioral concepts in strategic enterprise architectures. Within the Goal View, the framework distinguishes among three layers of abstraction (Strategic, Tactical and Operational Levels), whereas the Operations View is structured in terms of four layers of abstraction (Operations Level, BPA - Level 0, BPA - Level 1 and BPA - Level 2 Levels). Within the Goal View, the framework distinguishes among goals of various shades (mission, vision, strategic, tactical and operational goals), offering the concepts of *refinement dimensions* and *dimensional refinement operators* for the refinement of strategic goals, inspired by data dimensions in data warehouse literature. Within the Operations View, the framework distinguishes among operations and business processes and their relations. Furthermore, SIENA also includes methodological guidelines on how to build such strategic enterprise process models.

Regarding the achievement of the requirements for strategic enterprise architectures (Chapter 2), the SIENA language achieves the expressiveness in motivational and behavioral perspectives (R1) and part of the traceability between motivational and behavioral perspectives requirements (R2). Starting with expressiveness in the motivational perspective, SIENA allows the representation of goals in terms of labeled descriptions (1.1) segmented in multiple levels of abstraction (strategic, tactical and operational levels) (1.6). Each goal layer has a target (1.5) to be achieved and distinct time frames (1.3) in which they need to be accomplished, i.e., strategic goals are usually long-term (between two and five years), whereas tactical goals have typically shorter time horizons (usually from one to three years) and operational goals consists of daily results. Each goal category from the hierarchical level is assigned to some member of the organizational structure (1.2) responsible for its achievement. Strategic goals are assigned to the overall organization, while tactical goals are assigned to functional areas and organization units. In its turn, operational goals are assigned either to a role or to multiple roles (business process operational goal). Furthermore, SIENA also captures a number of relations among goals, such as refinements (AND-relations and dimensional refinement operators), alternatives (OR-relations) and partial, qualitative relations (positive and negative contributions) (2) among the goals of all layers. Factors that impact the achievement of goals are also captured as situations and domain assumptions (3). The achievement of goal patterns (1.4) sub-requirement is further explained in Chapter 7.

In relation to expressiveness in behavioral perspective, SIENA offers the concepts of operations and business processes (1.2), also capturing the relations among business processes within the business process architecture (1.3). The achievement of social (1.2) and operational perspective (1.2) of processes is further explained in Chapter 7. The distinction between operations and business processes has been motivated by the acknowledgment that enterprise's strategies have to be planned in advance (thus motivating the introduction of the concept of operation) before business processes that indeed deliver company's products and services.

The achievement of expressiveness in motivational and behavioral perspectives opens up the possibility of also (partially) achieving traceability between motivational and behavioral perspective with the derivation of operations from tactical goals and business processes from operational goals.

Compared to GORE, EM and motivational and behavioral approaches (Chapter 4), SIENA advances the state of the art by providing *multiple levels of abstraction in goal hierarchies*, *dimensional refinement operators* for the refinement of strategic goals and the *explicit connection between motivational and behavioral concepts*. These three SIENA's features advance the state of the art in the representation of motivational and behavioral modeling in different ways.

First, the existence of goals of *multiple levels of abstraction* allows the representation of goal hierarchies like the ones of our motivating example (Section 2.2) which cannot be captured by current approaches (i.e., GORE, EM and motivational and behavioral approaches (Chapter 4)). As explained in Section 4.7, GORE approaches (e.g. Tropos, i*) use the concept of "goals" to capture stakeholders' interests and requirements and link them to the technical requirements of the system. For example, a stakeholder's concern in the context of a scheduler meeting system could be represented in terms of a goal "Schedule meeting". Enterprise modeling approaches (e.g. ARIS, EKD and ArchiMate) borrow GORE goals to represent enterprise's strategic concepts such as "Increase company's sales", whereas motivational and behavioral approaches (e.g. [111, 1, 72, 185, 109, 117, 116, 165]) borrow GORE goals to represent either "strategic goals" or "process goals" (desired state to be achieved by a certain business process) interchangeably. Consequently, as the three groups of approaches simply inherit the GORE concept of goal, they cannot distinguish among multiple levels of abstraction and thus, they cannot cover the representation of goal hierarchies like our motivating example.

Second, within the three groups of approaches, goals may be decomposed into a finer-grained structure by means of AND/OR relationships, with an AND decomposition supporting a goal to be decomposed in a series of sub-goals and an OR decomposition allowing analysts to model alternative ways of achieving a goal. Partial positive/negative relations may be also specified in most of the approaches. SIENA incorporates all those relations, but also proposes the refinement of strategic goals in terms of *refinement dimensions* and *dimensional refinement operators*. The usage of dimensional refinement operators allows the refinement of strategic goals in terms of different enterprise's dimensions, such as time, geographical distribution of the company and properties of company's products/services. Consequently, alternatives to achieve strategic goals differently on each enterprise dimension can be expressed (e.g., sales can be increased differently in different locations where the company operates). In contrast with current approaches that can specify different alternatives for achieving goals with the usage of OR-refinements, SIENA goes further by allowing the representation of different alternatives for achieving strategic goals for different points of a given refinement dimension.

Third, the *explicit connection between motivational and behavioral constructs* allows SIENA to capture the planning of the achievement of strategic by means of *operations*. This connection also allows the derivation of operational goals from operations and the subsequent derivation of business processes and business process architecture from operational goals, thus capturing how enterprise's goals are realized by the set of company's processes. The interconnections between operations and tactical goals and business processes and operational goals are also instrumental in the specification of the implementation of strategic alternatives during the enterprise's planning process. In relation to the current state of the art, although the concept of BPA and its layered structure already exist in BPM literature [202, 52, 54], it refrains from connecting such BPA with its corresponding enterprise's goals. Very few approaches [116, 136, 165] indeed recognizes the integration of enterprise's goals and the BPA, but their hierarchy of goals is very simplistic by simply inheriting the GORE concept of goal.

In relation to the original publication [29], the following features have been further refined in this chapter:

1. SIENA's meta-model and concrete syntax have been specified;

2. As business process and operations had the same concrete syntax, a differentiation has been proposed in order to allow the graphical discrimination between both concepts. The concrete syntax of implements relation (between strategic and tactical goals), dimensional refinement operators and AND-refinement (among strategic goals) have also been differentiated;

3. Relations among business processes have been introduced together with their respective modeling guidelines, thus allowing the representation of the BPA;

4. The differentiation of tactical goals between *initiatives* and *established routines* allowed us to specify responsibilities of functional areas that

either implement strategic goals one single time (initiatives) or responsibilities that recur (established routines).

Chapter 6

Planning with Strategic Goals

In order to achieve the requirements discussed in Chapter 4 (Section 2.3) regarding the development of automated analysis techniques with strategic enterprise architectures, this chapter introduces an automated reasoning strategic planning technique that allows the generation of optimum plans with respect to an objective function for fulfilling strategic goals, taking into account constraints and scenarios. The content of this chapter is an updated version of a paper to be submitted [29]. The chapter is organized as follows: Section 6.1 introduces our strategic planning approach with the SIENA modeling language using the metal manufacturing example as a motivating scenario, Section 6.2 depicts the formalization of strategic goals and their AND/dimensional refinements, whereas Section 6.3 shows the formalization of optimization goals. Section 6.4 presents the mapping of strategic planning concepts into the CGM formalism introduced in Chapter 3 (Section 3.4), while Section 6.5 shows two illustrative examples of the generation of optimum strategic plans using the CGM tool.

6.1 The Strategic Planning Approach

In order to achieve the support for automated reasoning with strategic enterprise architectures requirement (R3) described in Section 2.3, this chapter presents the automated reasoning strategic planning technique (highlighted in Figure 6.1 by a red circle) and illustrates the technique in the metal manufacturing scenario modeled in Chapters 1, 2, 3 and 5.

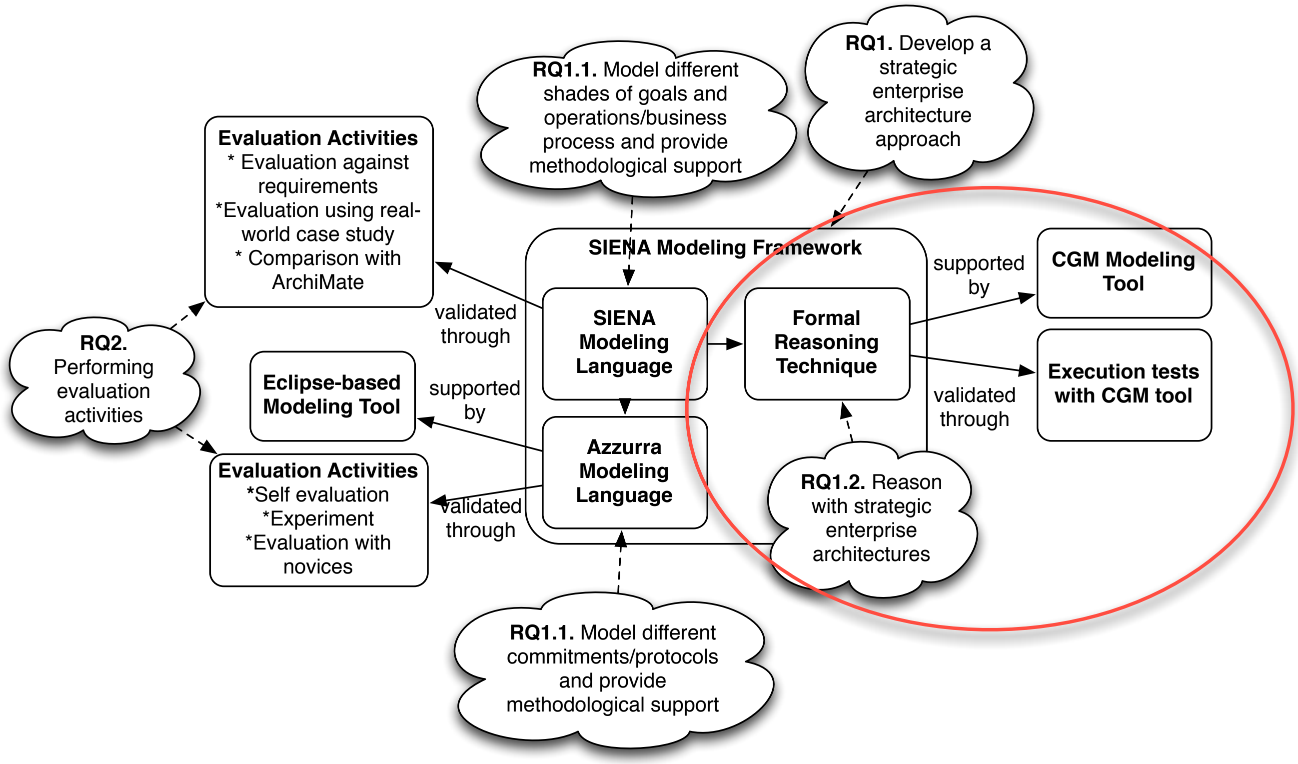


Figure 6.1: The Contribution of this Chapter in the Context of the Overall Thesis

The support for automated reasoning with strategic enterprise architectures requirement (R3) has been elaborated in order to support the enterprise's planning process of Section 2.2, Chapter 2. In SIENA, the enterprise's planning process starts with the elaboration of *strategic goals* that reflect organization's strategy to achieve success in business (step 1).

For example, starting with the "Increase sales over 3 years" strategic goal from the metal manufacturing company (Figure 6.2), managers decided to perform a refinement by *location* (countries) in which the company operates, thus yielding the "Increase sales in Italy over 3 years", "Increase sales in Germany over 3 years" and "Increase sales in France over 3 years" sub-goals. This dimension has been chosen due to the existence of different tactics to increase sales in each country the company operates.

A direct consequence of refining strategic goals in terms of refinement dimensions is the ability to specify different "solutions" ("tactics" or *tactical goals*) for the same strategic goal along different points of interest of a dimension. Therefore, *tactical goals* are specified by means of *implements relations*, representing particular ways of fulfilling strategic goals with the available resources and capabilities of the company. Following in Figure 6.2, it is possible not only to specify the alternatives for increasing sales for a given company, but rather, one can explore the alternatives for increasing sales depending on the country the company operates. Therefore, managers can increase sales in Italy by opening new sales channels or through promotions (represented in Figure 6.2 by an *implements relation* from the tactical goal to the strategic goal), while in France one can train new sales staff. Once it is possible to schedule the achievement of tactical goals, managers create an operation for each tactic. In this context, operations are in charge of planning the execution of tactics (also depicted in Figure 6.2).

Following in the planning process, managers elaborated *situations* that impact the achievement of goals either positively or negatively (step 2). In Figure 6.2, a risk of a financial crisis may threaten an increase in sales in Italy, while the high demand of automotive industry in Germany may be a favorable situation to increase the metal sales in this country. Further, a highly trained sales staff in France may also be favorable for this country in its sales increase.

Once the refinement process of strategic goals has ended and strategic planning concepts (strategic, tactical goals, operations and situations) have been specified accordingly, the metal company has a strategic goal model with several strategic plans (different sets of strategic, tactical goals, operations and situations) to achieve its strategic goals. In the context of the enterprise planning process, the generation of these several strategic plans corresponds to the identification of strategic alternatives in step 3. For example, Figure 6.2 depicts a red and blue strategic plans for the achievement of the top strategic goal “Increase sales over 3 years”. In this context, our strategic planning approach is interested in: How to automatically find different ways of satisfying strategic goals, i.e., different strategic plans?

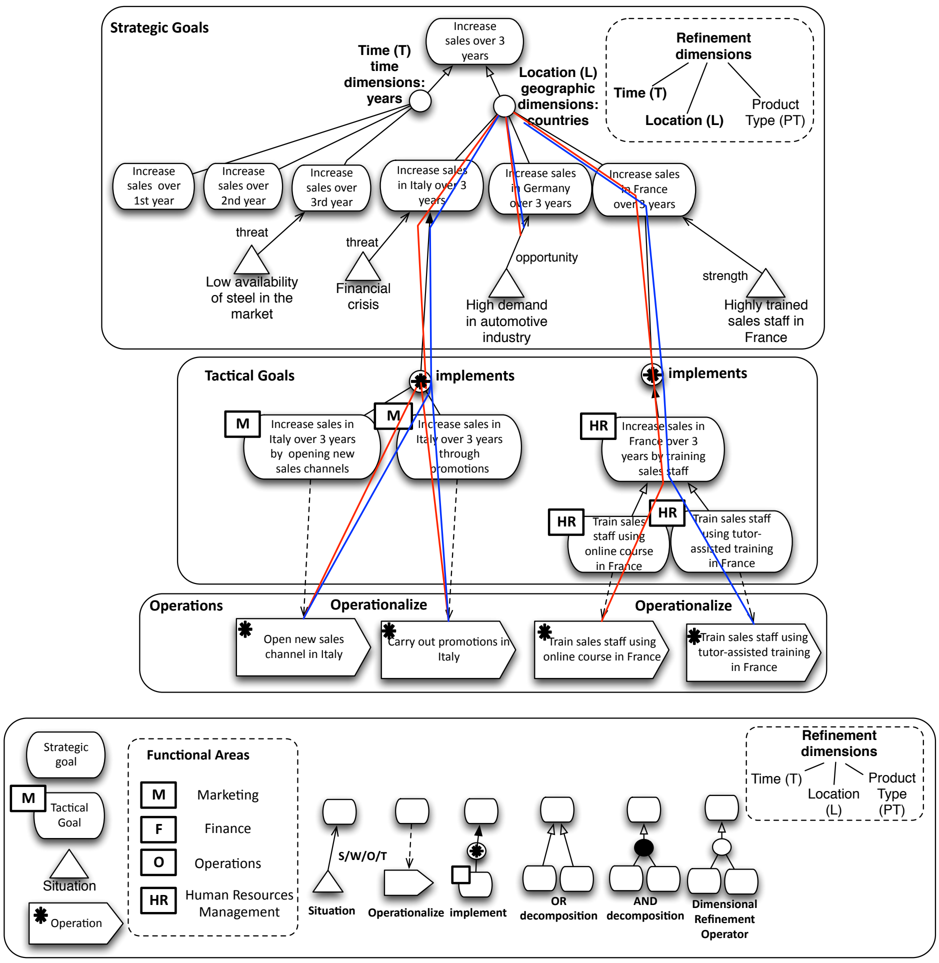


Figure 6.2: Strategic goals and Strategic Plans

In the remainder of this chapter, we first formalize strategic goals and their dimensional/AND/OR refinements in Section 6.2. This formalization enables us to perform automated strategic planning reasoning with SIENA models. In Section 6.3, we present the notion of optimization goals that allow us to select best strategic plans to achieve strategic goals. Subsequently, as SIENA models from Chapter 5 have no formal semantics for automated reasoning, we need to either assign our own formal semantics for reasoning or find existing goal-modeling languages with already well-established semantics for reasoning. In particular, we opted for the second choice by choosing the Constrained Goal Models (CGM) formalism described in Chapter 3, Section 3.4. Then, we propose a mapping from strategic planning concepts (strategic, tactical goals, dimensional operators, etc.) into the CGM formalism in Section 6.4. Finally, we illustrate our strategic planning automated reasoning with the generation of optimum plans with respect to some objective function for fulfilling strategic goals in Section 6.5.

6.2 Formal Strategic Goal Models

This section presents the formalization of SIENA's strategic goals, their AND/OR/d-refinements and implement relations with the purpose of enabling the automated strategic planning reasoning with SIENA models.

In SIENA, strategic goal models span one or more dimensions of the domain and can be realized differently along different points of each dimension (Section 5.1.1). The notion of dimension defines different levels of granularity at which the planning of strategic goals needs to be conducted. The dimensional schema is a star-shaped schema where different dimensions are represented by the branches of the star and its center represents the strategic goal to be viewed at different levels of granularity. Figure 6.3 shows a star schema for the goal "Increase sales by 2% over 3 years". The instantiation of this schema along the LOCATION dimension might have different countries of Europe as instances of countries, below each region of each country, below the cities of each region and at the bottom layer the particular stores in each city. It is interesting to note that a typical star schema instantiation may be large. For example, an instantiation of the schema of Figure 6.3 might include $\mathcal{O}(1K)$ elements along the PRODUCT and LOCATION dimensions, and $\mathcal{O}(1K)$ elements along the TIME dimension. This means that strategic planning needs to be conducted for up to $\mathcal{O}(10M)$ "Increase sales" goals.

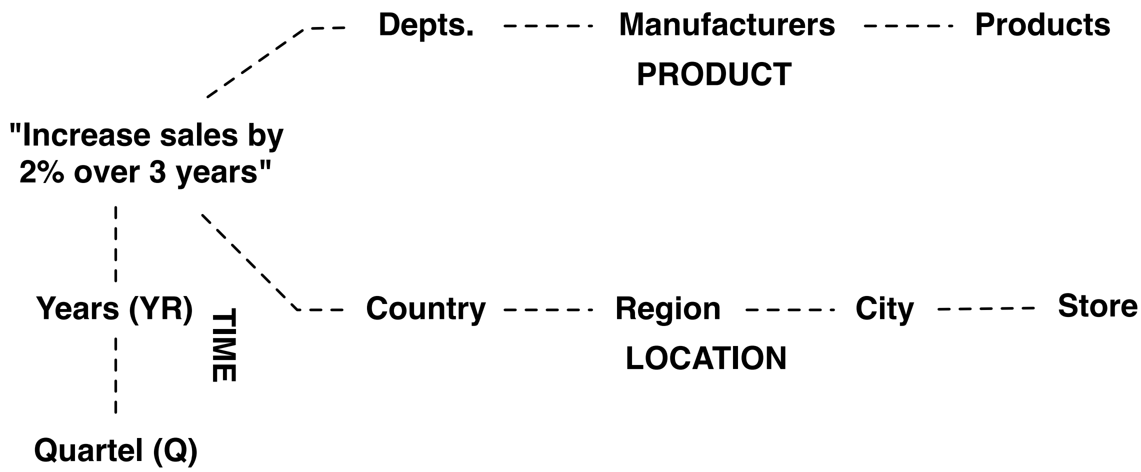


Figure 6.3: A star schema for the "Increase sales" strategic goal

A strategic goal is represented formally as a parameterized goal with one parameter per dimension. For example, the "Increase sales" goal is represented as IncrSales(3YR, Europe, AllProducts), where 3YR, Europe, AllProducts are respectively the parameters of the three dimensions spanned by the goal. To define the goal, we need to express the increase of 2% in sales for all products throughout Europe over three years. For this, we assume an indicator "sales" associated with each IncrSales goal and define

$$IncrSales(3YR, Europe, AllProducts) = \\ = 1.02 * sales(now, Europe, AllProducts) \quad (6.1)$$

The dimensional refinement (d-refinement) conjunctively refines a strategic goal along with a dimension. For example, IncrSales might be refined along the LOCATION dimension with specific targets for sales for each country in Europe, or a subset thereof since there might not be ambitions for certain countries. When determining targets for the subgoals of a d-refinement, we want the targets to be based on real past data. As such, we use a *criterion* to set metrics for each target calculation for each dimensional subregions over which we are refining. This criterion is selected by managers following his/her preferences. For example, for IncrSales, suppose we are interested in increasing sales in only three countries (Germany, France and Italy) and use the growth of the economy last year of each country ("growth") as a criterion to estimate the targets for each country. An alternative criterion, in this case, would be the growth of sales last year of each country ("growthSales"). Going forward with the first criterion, then:

$$\begin{aligned} IncrSales(3YR, Europe, AllProducts) = \\ IncrSales(3YR, Germany, AllProducts), \\ IncrSales(3YR, France, AllProducts), \\ IncrSales(3YR, Italy, AllProducts) \end{aligned} \tag{6.2}$$

where

$$\begin{aligned} IncrSales(3YR, Germany, AllProducts) = \\ = \frac{1}{3}\frac{growth(Germany)}{growth(Germany) + growth(France) + growth(Italy)} * \\ 1.02 * sales(now, Germany, AllProducts) \end{aligned} \tag{6.3}$$

$$
\begin{aligned}
&IncrSales(3YR, France, AllProducts) = \\
&\quad = sales(now + 3YR, France, AllProducts) \\
&\quad = \frac{1}{3}\frac{growth(France)}{growth(Germany) + growth(France) + growth(Italy)} * \\
&\qquad\qquad 1.02 * sales(now, France, AllProducts) \quad (6.4)
\end{aligned}
$$

$$
\begin{aligned}
&IncrSales(3YR, Italy, AllProducts) = \\
&\quad = sales(now + 3YR, Italy, AllProducts) \\
&\quad = \frac{1}{3}\frac{growth(Italy)}{growth(Germany) + growth(France) + growth(Italy)} * \\
&\qquad\qquad 1.02 * sales(now, Italy, AllProducts) \quad (6.5)
\end{aligned}
$$

Here, the third line of equations (6.3), (6.4) and (6.5) corresponds to the normalization factor that allocates to each country a sales growth normalized by the size of economic growth of the corresponding country last year. Figure 6.4(b) shows the refinement by LOCATION, using equation (6.3), with the actual growth of economy last year ("growth") criterion: *growth(Germany)* = 2%, *growth(France)* = 1% and *growth(Italy)* = 0.5% and current sales as: *sales(now, Germany, AllProducts)* = 10000, *sales(now, France, AllProducts)* = 7000 and *sales(now, Italy, AllProducts)* = 7000. For example, using equation (6.3), IncrSales(3YR, Germany, AllProducts) has a target value of 1942.85, which is a targeted increase of 19.42% in sales over the current value (sales(now, Germany, AllProducts) = 10000)(depicted in Figure 6.4(b)). Currently, although these numbers are manually acquired together with their corresponding calculations, we intend to automatically generate goal refinements and check their consistency.

Orthogonal to this d-refinement, one may want to d-refine IncrSales along the TIME dimension using expected growth of the economy of each country ("expGrowth") as criterion

$$IncrSales(3YR, Europe, AllProducts) = IncrSales(YR1, Europe, AllProducts), IncrSales(YR2, Europe, AllProducts), IncrSales(YR3, Europe, AllProducts) \quad (6.6)$$

where

$$IncrSales(YR1, Europe, AllProducts) = = \frac{1}{3}\frac{expGrowth(YR1)}{expGrowth(YR1) + expGrowth(YR2) + expGrowth(YR3)} * 1.02 * sales(now, Europe, AllProducts) \quad (6.7)$$

$$IncrSales(YR2, Europe, AllProducts) = = sales(now + 2YR, Europe, AllProducts) = \frac{1}{3}\frac{expGrowth(YR2)}{expGrowth(YR1) + expGrowth(YR2) + expGrowth(YR3)} * 1.02 * sales(now, Europe, AllProducts) \quad (6.8)$$

$$IncrSales(YR3, Europe, AllProducts) = = sales(now + 3YR, Europe, AllProducts) = \frac{1}{3}\frac{expGrowth(YR3)}{expGrowth(YR1) + expGrowth(YR2) + expGrowth(YR3)} * 1.02 * sales(now, Europe, AllProducts) \quad (6.9)$$

Analogously, we have normalized growth targets per year by using the expGrowth for each country criterion. Figure 6.4(c) shows the refinement by TIME, using equation (6.7), with the actual expGrowth criteria: *expGrowth(YR1)* = 0.6622%, *expGrowth(YR2)* = 0.6622% and *expGrowth(YR3)* = 0.6622% and current sales as: *sales(now, Europe, AllProducts)* = 20000. In this case, we have considered a uniform expected growth of the economy (expGrowth) each year and its value (0.6622%) has been calculated using the compound interest formula V = P*(1+expGrowth)^y. With this value in hands, in equation (6.7), IncrSales(YR1, Europe, AllProducts) has the target value of 2266.66, which is targeted increase of 11.33% in sales over the current value (considering that sales(YR1, Europe, AllProducts) = 20000)(depicted in Figure 6.4(c)).

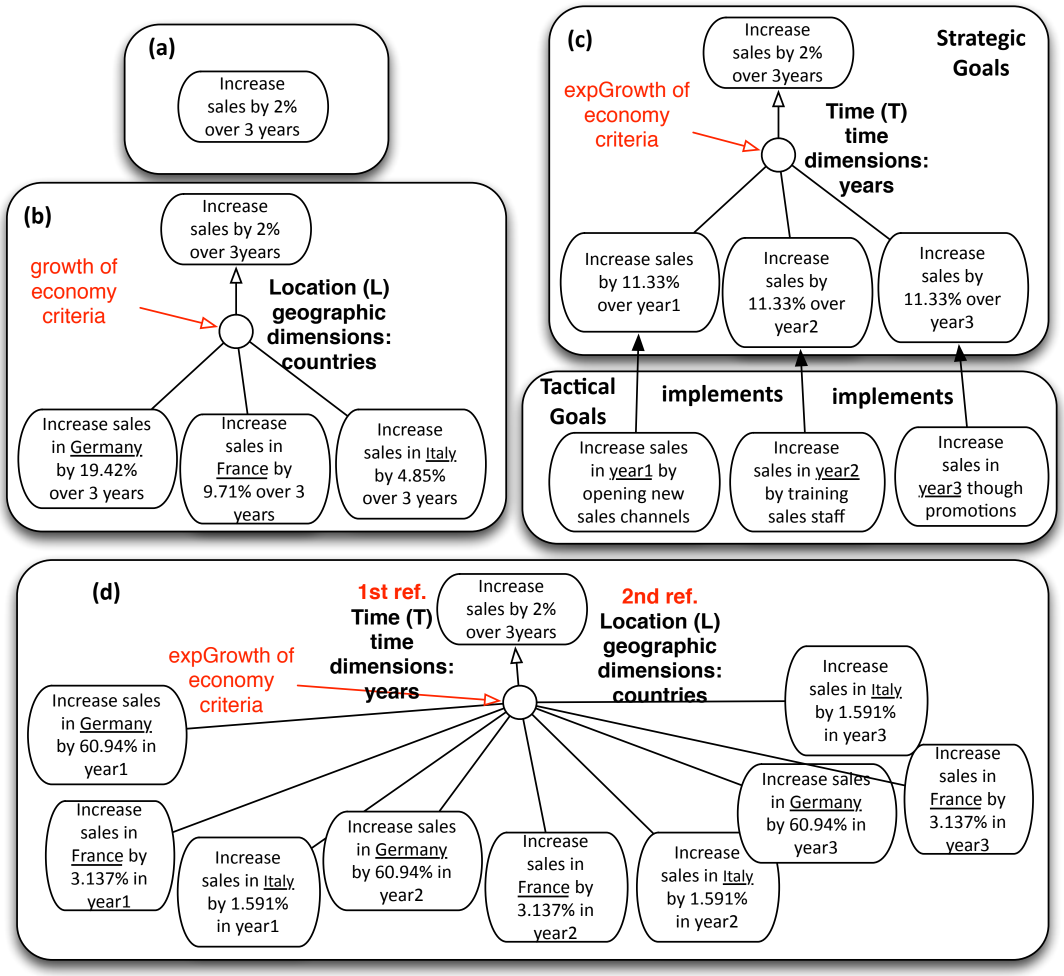


Figure 6.4: Strategic Goals, Dimensional Refinements and Properties Heritage

As presented in Section 5.1.1, besides d-refinements, strategic goals can also be *AND/OR-refined* and implemented by tactical goals. In this context, a problem that arises is how to accommodate d-refinements and AND/OR/implement relations for strategic goals. In SIENA, by d-refining a strategic goal G(X,Y,Z) into $G_1(X_1,Y_1,Z_1)$, ... , $G_N(X_N,Y_N,Z_N)$ and also AND-refining/OR-refining/implementing the same goal into G'_1, G'_2, ... , G'_M, the second refinement (i.e., G'_1,G'_2, ...,G'_M) inheres by default the properties of $G_1(X_1,Y_1,Z_1)$, ... , $G_N(X_N,Y_N,Z_N)$. More concretely, if we d-refine IncrSales into G_1 = "Increase sales by 11.33% in year 1", G_2 = "Increase sales by 11.33% in year 2", G_3 = "Increase sales by 11.33% in year 3", and also implement it into using (tactical goals) G'_1 = "Increase sales by opening new sales channels", G'_2 = "Increase sales by training sales staff", G'_3 = "Increase sales through promotions", then the sub-goals G'_1, G'_2 and G'_3 inhere the properties of G_1, G_2, G_3, unless the analyst chooses to override it for some reason. In this particular case, G'_1, G'_2 and G'_3 inhere the property "year 1", "year 2" and "year 3" from G_1, G_2 and G_3, respectively. Fig. 6.4(c) depicts this example of refinements inheritance with tactical goals inhering the properties of strategic goals. Further, the analyst chose to prune some tactical goals, thus applying a specific tactic for each year due to trends revealed by past real data. Therefore, "Increase sales in year 2 by opening new sales channels" and "Increase sales in year 3 by opening new sales channels" have been pruned (and similarly, the other tactics for the other years.)

The same rationale applies if two orthogonal d-refinements are performed successively, i.e., the subgoals of d-refinement1 inheres the properties of d-refinement2 and vice versa. Obviously, inheritance of a refinement includes the criterion associated with the refinement. For instance, since expGrowth is the criterion for the TIME d-refinement above, when it is inherited by IncrSales(3YR, Germany, AllProducts), IncrSales(3YR, France, AllProducts), IncrSales(3YR, Italy, AllProducts) respectively, it will generate three subgoals for each one of them and use expGrowth to normalize targets for these subgoals. Figure 6.4(d) illustrates the result of two successive d-refinements of "Increase sales by 2% over 3 years". In the first d-refinement by TIME (not depicted in 6.4(d)), equation (6.7) is used with the respective expGrowth criteria, thus generating the following sub-goals G_1 = "Increase sales by 11.33% in year 1", G_2 = "Increase sales by 11.33% in year 2", G_3 = "Increase sales by 11.33% in year 3". In the second d-refinement by LOCATION, the d-refinement inherited the expGrowth criterion from the first d-refinement and yielded the nine sub-goals depicted in Figure 6.4(d). In this case, in order to calculate the respective weights of each country in each year, each factor from the right side from equation (6.2) (i.e., IncrSales(3YR, Germany, AllProducts), IncrSales(3YR, France, AllProducts), IncrSales(3YR, Italy, AllProducts)) is substituted in each factor of the right side of equation (6.4).Therefore, this new equation (not depicted here) has nine factors like IncrSales(YR1, Germany, AllProducts), requiring one to estimate the expected economy growth (expGrowth criteria) of each country in each of the three years (i.e., expGrowth(YR1, Germany, AllProducts) should be estimated for all countries in each of the three years). For simplification purposes, in order to calculate IncrSales(YR1, Germany, AllProducts), we have considered that the increase in sales in Germany of 19.42% should be uniformly distributed across the three years. Therefore, using the compound interest formula $V = P^*(1+r)^y$, we have that: 1.1942 sales = sales*(1+expGrowth(YR1, Germany, AllProducts))^3, which yields expGrowth(YR1, Germany, AllProducts) = 60.94% (depicted in Figure 6.4(d)). Observe also that contrasting with previous example (Fig. 6.4(c)) in which some goals have been pruned based on trends of past real data, the two successive d-refinements here yielded nine sub-goals.

Observe that d-refinements in Figures 6.4(b) and 6.4(c) used growth and expGrowth criteria respectively for d-refinements, whereas in Figure 6.4(d), expGrowth has been used successively for the two d-refinements, with the second d-refinement inhering the expGrowth criterion from the first d-refinement. Note also that, since criteria are inherited along with their d-refinements, d-refining with respect to ref1then ref2 will result in the same sub-subgoals as when d-refining with respect to ref2 and then ref1. However, this is not the case if the analyst overrides inheritance of ref1 or ref2.

The refinement process (with defaults) can end when a given strategic goal has reached leaf elements for all points of interest of the dimensions spanned by the goal, or when this strategic goal spans regions that are sufficiently uniform so that they do not require further refinements. For example, if Italy is deemed sufficiently uniform to admit one tactical solution for all its subregions (provinces and stores)(e.g. "Train sales staff"), then the analyst does not need to drill further down in the planning. Consequently, the estimate given above of the number of tactical plans that need to be generated is a worst case bound, when there is too much variance from subregion to subregion for each dimension, so that planning has to reach the finest granularity supported by each dimension.

6.3 Optimization Goals

In order to differentiate among strategic plans, assigning them quantitative values and thus enabling the ranking of strategic plans, the framework for formal strategic goal models also includes optimization goals. For example, we may be interested in a strategic plan for achieving IncrSales that minimizes expenses, or maximizes profits. We express such goals as

$$OPT[cost, IncrSales(3YR, Europe, AllProducts)] \qquad (6.10)$$

OR

$$OPT[profits, IncrSales(3YR, Europe, AllProducts)] \qquad (6.11)$$

Therefore, each optimization goal is defined relative to an attribute (or a linear combination thereof) and a strategic goal. Note that in order to generate optimum plans, we need to have values for these attributes for every tactical goal used to realize a strategic goal for all leaf-level regions. For our example, this means that we know (estimated) costs and profits for opening new stores, training sales forces and having promotions. Further than the definition of optimization goals to enable an automated approach using CGMs, we also need to map strategic planning concepts into the CGM formalism which is described in next section.

6.4 Formal Reasoning with Strategic Goals using CGM

This section describes the approach for specifying SIENA's strategic planning concepts discussed in Sections 5.1.1, 6.2 and 6.3 using the CGM formalism. This specification enables us to use CGMs for automatically selecting the best strategic plans (in CGM terminology, *optimum* strategic plans) for strategic goal models.

6.4.1 Specify Strategic Planning Concepts in CGM

The approach starts by mapping the concepts mentioned in Sections 5.1.1 and 6.2 to the CGM formalism with the purpose of providing formal CGM semantics for such modeling constructs. Those concepts are strategic and tactical goals, dimensional refinements, AND/OR-refinements, positive and negative (+/++/-/–) contributions, implements relations, operations and situations.

Strategic, Tactical Goals, AND/OR/d-refinements, Implements Relation. In Section 6.2, the strategic planning process starts with the specification of strategic goals and their subsequent refinements in terms of AND/OR/d-refinements until finding tactical goals that implement points of each dimension. This idea is very similar to the progressive refinement of the CGM *root goal* into *intermediate goals* and therefore, every top strategic goal is specified as a CGM *root goal*, while its strategic subgoals and tactical goals correspond to *intermediate goals* in CGM. Both in SIENA and CGM, the process of goal refinement ends when no decomposition is required and the goal can be executed. Therefore, operations in SIENA correspond to a CGM *task* as both concepts refer to the lowest level of refinement in a goal tree. Regarding relations among goals, AND/d-refinements are mapped into one CGM *refinement*, while alternative AND/d-refinements (OR) of the same strategic goal are mapped into multiple CGM *refinements*. Each implements relation is mapped into a CGM *refinement* to depict that a strategic goal is implemented by a conjunction of different tactics.

Positive and Negative (+/++/−/−−) Contributions. Besides different types of refinements, SIENA's positive and negative (+/++/−/−−) contributions must be also specified accordingly in CGM. In this context, unlike AND/OR/d-refinements that can be straightforwardly mapped to CGM *refinements*, *positive and negative contributions* exist in both frameworks but carry slightly divergent semantics. In SIENA, *partial +/- contributions* encompass qualitative and quantitative relations that denote that one goal contributes positively/negatively towards the satisfaction of the other goal (respectively), whereas *full ++/−− contributions* denote respectively that one goal entails the satisfaction/denial of the other goal. This semantics is commonly found in many GORE frameworks [93, 65] for systematically reasoning about the satisfaction of goals based on label propagation algorithms [65] (See Section 4.5.5, Chapter 4). In contrast, GMG *contribution edges (++)* and *conflict edges (−−)* are constraints that state that if one goal is part of one solution, the other goal must be in the solution set (contribution edges) or must not be in the solution set (conflict edge).

Hence, for representing SIENA's full contributions in CGM, even with the aforementioned semantic differences, we directly mapped ++/−− in SIENA to ++/−− CGM relations by considering that in both frameworks the existence of one goal entails the full satisfaction of the other one. In contrast, the lack of partial relations in CGM does not allow us to directly map such relations in both frameworks. Therefore, we have considered three possible mappings from SIENA partial relations to CGM. These three possibilities are depicted in Table 6.1.

	SIENA	**CGM**
1	+	++
	−	++
2	+	++
	−	−−
3	+	Ignore the existence of +/- partial relations
	−	

Table 6.1: Mapping between SIENA and CGM partial relations

The first possibility basically maps both SIENA +/- relations to contribution links (++) in CGM, while the second possibility consists in applying the same semantic rules applied for the case of full contributions (i.e., + are translated to ++ while − are translated to −−). In the third type of tests, no translations have been considered.

In order to test the possible mappings of Table 6.1, we selected a number of SIENA models in which we have represented positive/negative contributions. With these models in hands, our intention was to investigate the

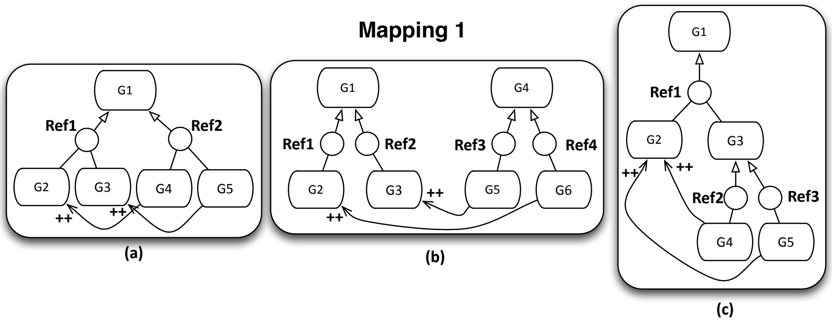


Figure 6.5: Mapping from SIENA Partial +/- Contributions to CGM ++ Links

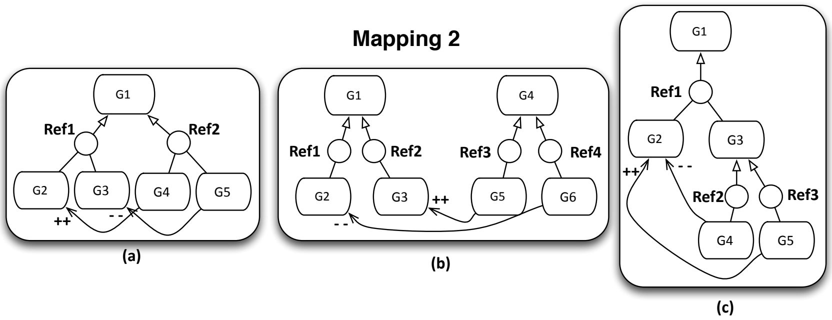


Figure 6.6: Mapping from SIENA Partial +/- Contributions to CGM ++/– Links (respectively)

possible different configurations in which positive/negative contributions may appear in SIENA models and the implications of such configurations in the reasoning results generated by CGM. Then, we started with the first mapping of Table 6.1 (line 1) by mapping SIENA partial contributions (+/-) to CGM contributions (++) for every model. After that, each CGM model has been executed in the CGM tool to test the consistency of reasoning results. On the basis of multiple tests with CGM models, we concluded that the three possible configurations of contributions that may exist in a CGM representation are depicted in Figs. 6.5 and 6.6 (or combinations thereof).

In order to actually test the possible mappings, we start with the three CGM models depicted in Figs. 6.5 and 6.6 and apply the three possible mappings of Table 6.1 in each model. More specifically, Fig. 6.5 shows

Model	Possible Realizations (SIENA +/- mapped to CGM ++)(mapping 1)	Possible Realizations (SIENA +/- mapped to CGM ++/−)(mapping 2)	Possible Realizations (SIENA + dropped)(mapping 3)
(a)	G1G2G3	G1G2G3	G1G2G3
	-	-	G1G4G5
(b)	G1G3 and G4G5	G1G3 and G4G5	G1G2 and G4G5
	-	-	G1G2 and G4G6
	-	-	G1G3 and G4G5
	-	-	G1G3 and G4G6
(c)	G1G2G3G4	G1G2G3G5	G1G2G3G4
	G1G2G3G5	-	G1G2G3G5

Table 6.2: Mapping between SIENA and CGM partial relations

the results of the application of the first mapping in the three models (i.e., SIENA +/- are mapped to CGM ++), while Fig. 6.6 shows the results of the application of the second mapping in the three models (i.e., SIENA +/- are mapped to CGM ++/−, respectively). Subsequently, we manually generate the CGM reasoning results that are expected according to the configuration of each model. The possible reasoning results (realizations) manually generated for the three mapping are depicted in Table 6.2. Finally, we test the expected results (realizations) in the CGM tool to verify whether the results of the manual generation are correct.

After we performed such process, we analyze the results of Table 6.2. As can be observed in this table, every model from Figs. 6.5 and 6.6 have more possible realizations when CGM contributions (++) and conflict (–) links are removed (evidenced by more realizations in the rightmost column of Table 6.2). Furthermore, CGM links constrain the number of realizations in such a way that most of the examples of Figs. 6.5 and 6.6 have just one possible realization in Table 6.2 for mappings 1 and 2. With such results in hands, we conclude that CGM contribution/conflict links constrain the number of possible realizations that can be selected by CGM in such way, that in most of the cases, there is just one possible realization. As the usage of CGM solver intends to select among *multiple solutions* and CGM links drastically reduce the number of feasible solutions, we lose one of the most prominent benefits of using the CGM approach. Consequently, we have opted for dropping SIENA +/- contributions in the translation to CGM, simply ignoring the existence of such relations in the mapping of SIENA to CGM.

Situations and SWOT Relations. The specification of situations requires a more careful treatment due to a divergence of design principles that drove the creation of SIENA and CGM frameworks. In SIENA models, situations are represented with the purpose of determining the impacts they have on strategic goals and strategic plans. In their representation, they are elements which are not part of any refinement goal tree, i.e., they are not further refined into other elements and they are not leaf levels of any tree, but rather, they are represented as elements linked to goals by SWOT relations. As they are not further refined into any other elements, an initial mapping considered situations as CGM *leaf goals* since leaf goals represent the lowest level of refinement in CGM. Although this mapping intuitively seems to be straightforward, the first attempt with formal reasoning with the CGM tool issued a compiling error, indicating that situations are leaf elements of an inexistent refinement tree. In fact, this decision is not admissible from the CGM point of view since requirements trees (goals) must always start by a root goal and the tool is responsible for finding alternative refinements to achieve mandatory root goals. Therefore, in a second attempt, we made a second mapping with situations as CGM *root goals* following a suggestion provided by the CGM tool. In principle, although the mapping seems to be counterintuitive from the SIENA point of view, CGM successfully accommodates such representation because root goals (situations) are the most basic level of a refinement tree and is subject to further refinement (although the modeler should not refine them in this case). Concerning the SWOT relations that situations may have towards goals, we restrict our interpretation to strengths and opportunities relations, leaving as future work weaknesses and threats. Therefore, strengths and opportunities relations are represented by means of CGM *contribution edges* of situations towards goals. As the semantics of *contribution edges* states that if the source goal is satisfied then the target goal must be also satisfied, this semantics is used to interpret that, if one situation is included in a given analysis, the goals targeted by that situation should be also in the realization.

For the purposes of the strategic planning activity, it is usually difficult to foresee how isolated situations may affect goals, and rather, managers are interested in determining how the whole business environment might evolve, especially in the presence of complex or rapid changes. In order to cope with uncertainty, managers usually carry out scenarios analyses by building detailed and plausible views (scenarios) about how the business environment of the company might develop in the future [34]. In our approach, scenarios are represented as a set of situations $\{s_1, s_2, ..., s_n\}$ that represent the company context similarly in [126]. Scenario analysis is carried out separately by marking the situations $\{s_1, s_2, ..., s_n\}$ of the corresponding scenario as true by means of CGM *user's assertions* to indicate they are active in a given analysis. For the situations that pertain to other scenarios, no further users' assertions have to be performed. Table 6.3 depicts the corresponding mapping between SIENA and CGM concepts.

SIENA	CGM
Top Strategic Goal	Root goal
Strategic Goal	Intermediate goal
Dimensional Refinement Operator	Refinement (for each dimension)
AND-refinement (among Strategic Goals)	Refinement
Implement (between Strategic and Tactical Goal)	Each implement relation is mapped to one Refinement
Tactical Goal	Intermediate goals
Operationalize relation (between Tactical Goal and Operation)	Refinement
Situation	Root goal
SWOT relations between Situations (S) and Goals (G)	Contribution Edge (S $\xrightarrow{++}$ G)

Table 6.3: Mapping between SIENA and CGM concepts

6.4.2 Specify *Objective Functions* and *Strategic Planning Constraints* into CGM

In the discussion about optimization goals (Section 6.3), we have argued that one may want to find optimum plans with respect to different attributes, e.g. strategic plans that minimize costs or maximize profits in order to increase sales. Therefore, we need to assign some quantitative values for the attributes of each strategic plan so that we can define the notion of optimum strategic plan. In our case, as CGM requires modelers to assign values to *leaf goals*, our strategic planning activity assigns values to operations to select the optimum realization.

A natural question concerns the sources of such numerical values. In Management literature [34], quality is usually expressed by an objective function to be maximized (profit, product quality, speed of service, utility) or minimized (cost, loss, risk, etc). In our approach, we assign the corresponding estimated cost of execution of operations and use the notion of "cheapest strategic plan" to execute as the optimum solution (i.e., minimize (cost of operations)). In order to implement such solution in CGM, we create a *numerical variable* $\{\text{cost}_1, \ldots, \text{cost}_k\}$ for each operation $\{\text{op}_1, \ldots, \text{op}_k\}$, respectively. As it is possible to create variables for other factors (loss, utility, etc.) and define multiple objective functions lexicographically organized, we also assigned the estimated execution time for execution of operations, thus creating the variables $\{\text{workingTime}_1, \ldots, \text{workingTime}_k\}$ for each operation, in the same fashion of costs. We also used the notion of "fastest strategic plan" to be executed. Fig. 6.7(a) depicts the cost of execution (in millions) and the time for duration (in years) of the operations of our strategic goal model.

In addition to objective functions, we also need to elaborate on the constraints our strategic plan is subject to. Constraints may be of several types like technical, physical, environmental and stem from a variety of sources, such as limited resources, contractual obligations, particularities of the domain, etc. In our case, we defined ranges for costs of operations (e.g. $1000 > \text{cost} > 200$) (in millions) and time limits ($\text{workingTime} < 8$) (in years) for operations execution (depicted in Fig. 6.7(a)).

6.5 Illustrative Example

In order to illustrate our strategic planning approach, Figures 6.7 and 6.8 shows two strategic goal models resulted from the application of the mapping rules described in Section 6.4 enriched with constraints and numerical values. With both strategic goal model in hands, our intention is to illustrate our strategic planning approach with the generation of different strategic plans on the basis of strategic goals. In order to depict the generation of strategic plans, we have performed a number of tests using

Scenario	Configuration	Constraints	Objective Functions (lexicographically organized)
1	Generation of Strategic Plans	(200 < cost < 1000) (in millions) (workingTime < 8) (in years)	minimize(cost) minimize(workingTime)
2	Generation of Strategic Plans in Specific Scenarios	(200 < cost < 1000) (in millions) (workingTime < 8) (in years) s_2, s_5, s_6 (scenario1) marked as true by means of user's assertions	minimize(cost) minimize(workingTime)

Table 6.4: Types of Tests Performed Using CGM

two configurations depicted in Table 6.4 and discussed along this section.

In the first configuration, operations in the strategic goal model have cost and duration (cost and workingTime variables). Further, as managers are interested in generating strategic plans that cost between 1000 and 200 million and last less than 8 years, we attached the respective constraints (200 < cost < 1000) (in millions) and time limits (workingTime < 8) (in years) for operations execution. Finally, managers are also interested in strategic plans with minimal cost and time duration (thus, we have used lexicographically organized objective functions like minimize(cost) AND minimize(workingTime)). Figure 6.7(a) depicts a strategic plan generated by CGM that satisfies all the constraints. An interpretation of such results states that there exists a strategic plan (in red) that satisfies all the constraints and also achieves the top strategic goal "Increase sales by 3% over 2 years". This strategic plan consists of increasing sales in Germany by 2.3% AND increasing sales in Italy by 5.8%. Figure 6.7(a.1) depicts the same strategic plan of Figure 6.7(a) as a CGM screenshot for illustrative purposes.

Still in the first configuration, we have used the same strategic goal model with stronger constraints. In this case, the ranges for cost of operations have been maintained but the duration of strategic plans have been reduced for workingTime < 5 (in years). Figure 6.7(a.2) depicts a screenshot from the CGM tool indicating the non-existence of strategic plans that satisfy the constraints. In this case, as it does not exist strategic plans that satisfy all the constraints, the tool highlights (in yellow) the strategic plans that could potentially be selected and indicates the encountered problems that prevent them to be chosen as a solution (UNSAT Core Extraction tab).

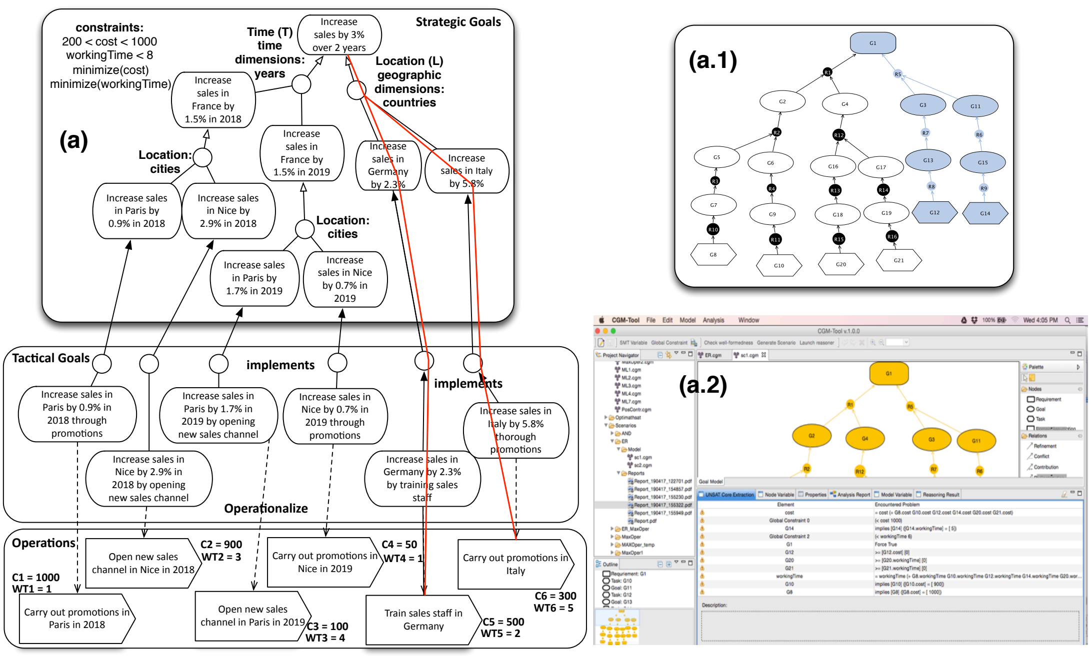


Figure 6.7: Generation of alternatives strategic plans in CGM

The first configuration revealed that imposition of strong constraints may lead to the generation of no strategic plans (e.g., no plans have been generated with the constraint workingTime < 5). In theory, the generation of strategic plans together with scenario analysis would be even more limited since not all strategic plans that satisfy the constraints can be part of the solution, but only those strategic plans for active scenarios can be generated. In face of such observations, we initially used the same strategic goal model of Figure 6.7(a) enriched with situations and tried to generate strategic plans for specific scenarios. Unfortunately, no strategic plans have been generated with such constraints.

Consequently, we elaborated a more complex strategic goal model with the same constraints used in the first configuration and enriched the model with a set situations s_1, ..., s_6. In the second configuration, we simulated a scenario of financial crisis (scenario1) that should be active during the generation of strategic plan by marking s_2 = “Sudden spike of metal cost”, s_5= “Low demand of steel in market”, s_6 = “Economic downturn” as true by means of user’s assertions. Figure 6.8(a) shows the strategic goal model with constraints, active situations (circled in red) and the generated strategic plan (in red). Tests with scenario analysis have been useful also for the selection of plans when there is more than one admissible plan according to the constraints. For example, with the constraints of Fig. 6.8(a), two plans are eliminated (one plan does not satisfy cost constraint, other does not satisfy the workingTime constraint and the third one does not satisfy the minimize(cost) objective function). In face of that, there are two possible plans. By marking s_2, s_5, s_6 as mandatory, the solver has just one plan (depicted in Fig. 6.8(a) in red).

Finally, when there exist multiple optimum strategic plans, the CGM solver does not depict all possible optimum plans, but rather just one admissible plan. Since there is potentially an exponential number of optimum solutions, the graphical representation of multiple optimum plans would not make sense. Therefore, our approach is bounded by such aspect of CGM to depict just one optimum strategic plan.

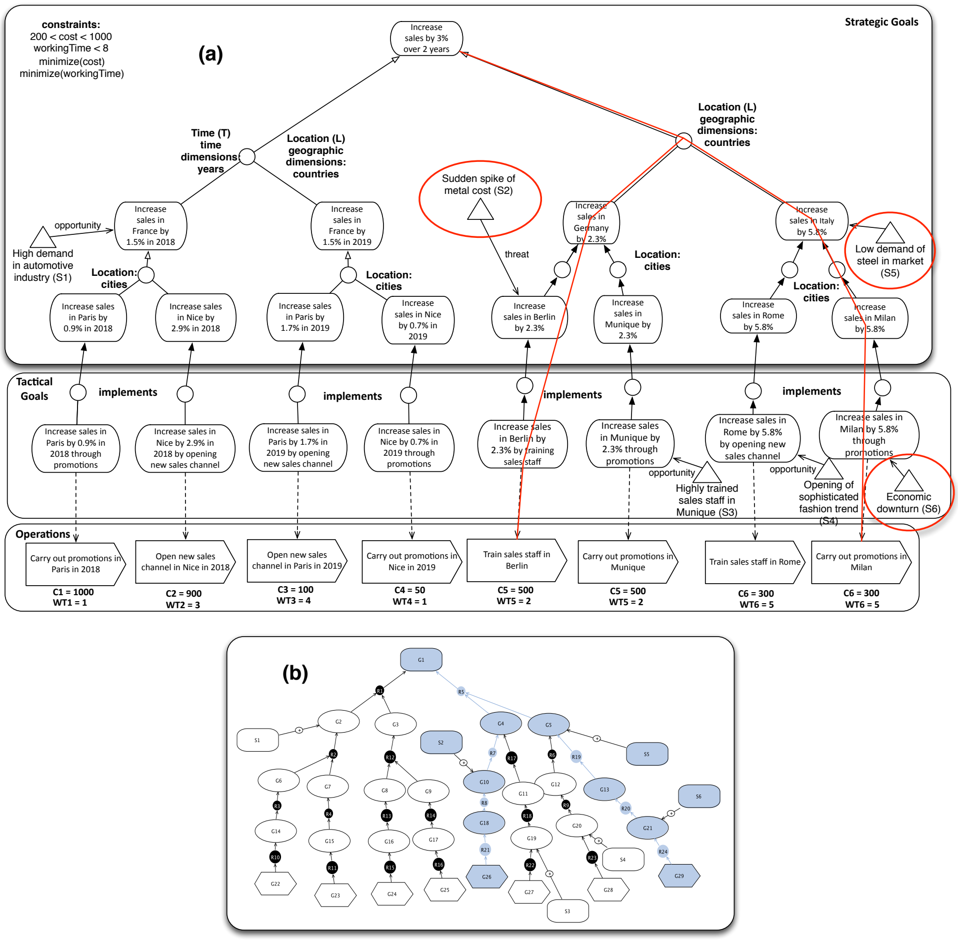


Figure 6.8: Generation of alternatives strategic plans in CGM in a scenario of financial crisis

6.6 Summary

This chapter proposed a formal representation of strategic goals and automated reasoning technique for strategic planning that consists of the generation of optimum strategic plans to achieve strategic goals on the basis of objective functions, constraints and scenarios.

In order to do that, based on the concepts of SIENA's *strategic goals* and their *dimensional refinement operators*, we have proposed a formalization of strategic goals, how to d-refine them in terms of refinement dimensions and how to accommodate d-refinements and other relations of strategic goals (AND/OR-refinements and implement-relations) with such d-refinements. Once strategic goals have been formally specified, strategic planning concepts are mapped into the CGM formalism in order to generate optimum strategic plans on the basis of objective functions, constraints and scenarios.

Regarding the achievement of the requirements for strategic enterprise architectures (Chapter 2), our strategic planning approach fully addresses support for automated reasoning with strategic enterprise architectures (R3) requirement, with an exception for control and evaluation of implemented strategic alternatives. We initially formalize strategic goals, their dimensional refinement operators, AND/OR refinements and implements relation, thus providing SIENA's specifications with formal rigor (R3.1). Further, our automated reasoning technique supports the execution of the several steps of the planning process (R3.2) in the following ways. First, the formalization of strategic goals and their dimensional refinement operators enables the reasoning with goals in multiple levels of abstraction (strategic and tactical goals)(R3.2.1). In its turn, our strategic planning approach allows the identification of strategic alternatives and generation of optimum strategic alternatives (strategic plans) that explore enterprise variability and constraints (R3.2.3). Second, the mapping of situations and domain assumptions to CGM modeling constructs enables us to perform realization of scenario analysis which corresponds to the assessment of environment and their impacts on the achievement of strategic goals (R3.2.2). Support for control and evaluation of implemented strategic alternatives (R3.2.4) is not achieved in this thesis.

In comparison with the current state of the art in goal-based reasoning techniques, GORE approaches (Section 4.5.5) represent stakeholders' motivations for a target software system as goals models that can be subsequently used as the starting point for the generation of system requirements. In terms of reasoning with such goal models, as described in Section 4.7, GORE forward techniques quantify the level of satisfaction of top system goals depending on alternative system designs [65, 86], whereas GORE backward reasoning techniques can recommend which designs to select [179, 142, 115] (in particular, the CGM approach used in this chapter is able to find the optimum set of subgoals to achieve a given root goal). Although GORE techniques allow one to perform advanced reasoning with goal models, their scope relies on the evaluation/generation of system designs, not strategic plans like our strategic planning approach in this chapter.

Enterprise modeling approaches like BIM, ARIS, EKD and ArchiMate (Section 4.5.5) allow one to define strategic concerns (e..g "Increase sales") using GORE goals and to infer the satisfaction of such goals by means of GORE forward/backward reasoning techniques. However, such approaches neither distinguish among multiple levels of abstraction for goals nor provide a modeling construct as our *dimensional refinement operator*. Consequently, they cannot explore such primitives in reasoning, entailing no automated support for the enterprise planning process. The only exceptions are backward reasoning techniques (BIM, i*) and [126] that provide partial support by the enterprise planning process. For instance, in backward reasoning techniques (BIM, i*), different strategies are generated using GORE backward algorithms [179], thus allowing the generation of strategic alternatives (requirement R3.2.3), but goals of different shades are not addressed (requirement R3.2.1). In [126], business strategies are systematically analyzed in different scenarios (set of situations) (requirement R3.2.3) in a framework for stress testing that builds on top of BIM, but strategic alternatives cannot be generated (requirement 3.2.3).

Similarly, motivational and behavioral approaches (Liaskos et al. [120], Bryl et al. [20, 21], Asnar et al. [6], Lapouchnian et al. [117], Greenwood et al. [72, 73], Lapouchnian et al. [116] and Morrison et al. [136]) also do not recognize the existence of an integrated hierarchy of goals and enterprise variability in terms of products/services, time or geographical distribution. Consequently, they also do not address such features in automated reasoning. In contrast, our strategic planning approach generates optimum strategic plans (according to some objective function) for achieving strategic goals, exploring different dimensions of the company like time, location and product/service.

The advent of SIENA's multiple levels of abstraction and dimensional refinement operators opens us the possibility of generating alternative strategic plans based on different dimensions of the enterprise. For example, in traditional goal analysis, for a "Increase sales" goal, one can represent two alternatives like "Increase sales by promotions" OR "Increase sales by training sales force", but no variability with respect to the enterprise dimensions like products/services, time or geographical distribution can be explored (e.g., "Increase sales by promotions in Italy" or "Increase sales of metal tables by training sales force"). In contrast, SIENA allows the selection of different tactics (promotions, salesforce training) for distinct enterprise dimensions (time, location and product/service) (e.g. promotions in Italy or salesforce training for metal products) to achieve strategic goals.

Other interesting SIENA feature is the possibility of performing scenario analysis (environmental analysis) during the enterprise planning activity (in this context, only the BIM framework and [126] reason with situations and domain assumptions as environmental factors). This is also an advantage as it allows one to generate strategic plans on the basis of the likelihood of certain business scenarios to happen. Finally, the usage of CGM formalism was also instrumental to our strategic planning approach, as it allows the generation of strategic plans that obey certain environmental constraints and properties (e.g. strategic plans with minimum cost). This feature is not addressed in reasoning by any enterprise architecture approach, to the best of our knowledge.

Chapter 7

Azzurra Modeling Language

This chapter introduces the Azzurra modeling language which is used as a sub-language for the representation of the control-flow of business processes within the *BPA - Level 1* Layer of the SIENA Modeling Framework. Furthermore, this chapter also completes the strategic planning approach by generating the BPA and business process' control-flow specified in Azzurra from SIENA's operational goals. The Azzurra modeling language is part of the published paper [36], in which the first author of this thesis re-wrote the full paper based on previous ideas and included some new content under the supervision of her co-authors (more specifically, the role of the first author of this thesis is explained in Section 1.8). The chapter is organized as follows: Section 6.1 further advances our strategic planning approach with the SIENA modeling language using the metal manufacturing example as a motivating scenario. Section 7.2 introduces the Azzurra modeling language, describing its abstract syntax, concrete syntax and runtime semantics, whereas Section 7.4 completes our strategic planning approach by generating the business process' control-flow specified in Azzurra from SIENA's operational goals.

7.1 The Strategic Planning Approach

In order to achieve traceability between motivational and behavioral perspectives requirement (R2) described in Section 2.3, this chapter presents the Azzurra Modeling Language (highlighted in Figure 7.1 by a red circle). In particular, we present Azzurra's abstract syntax (using the Extended Backus-Naur Form (EBNF) notation), concrete syntax (graphical notation) and runtime semantics. Furthermore, we finish our strategic planning approach by generating the BPA and business process' control-flow specified in Azzurra from SIENA's operational goals.

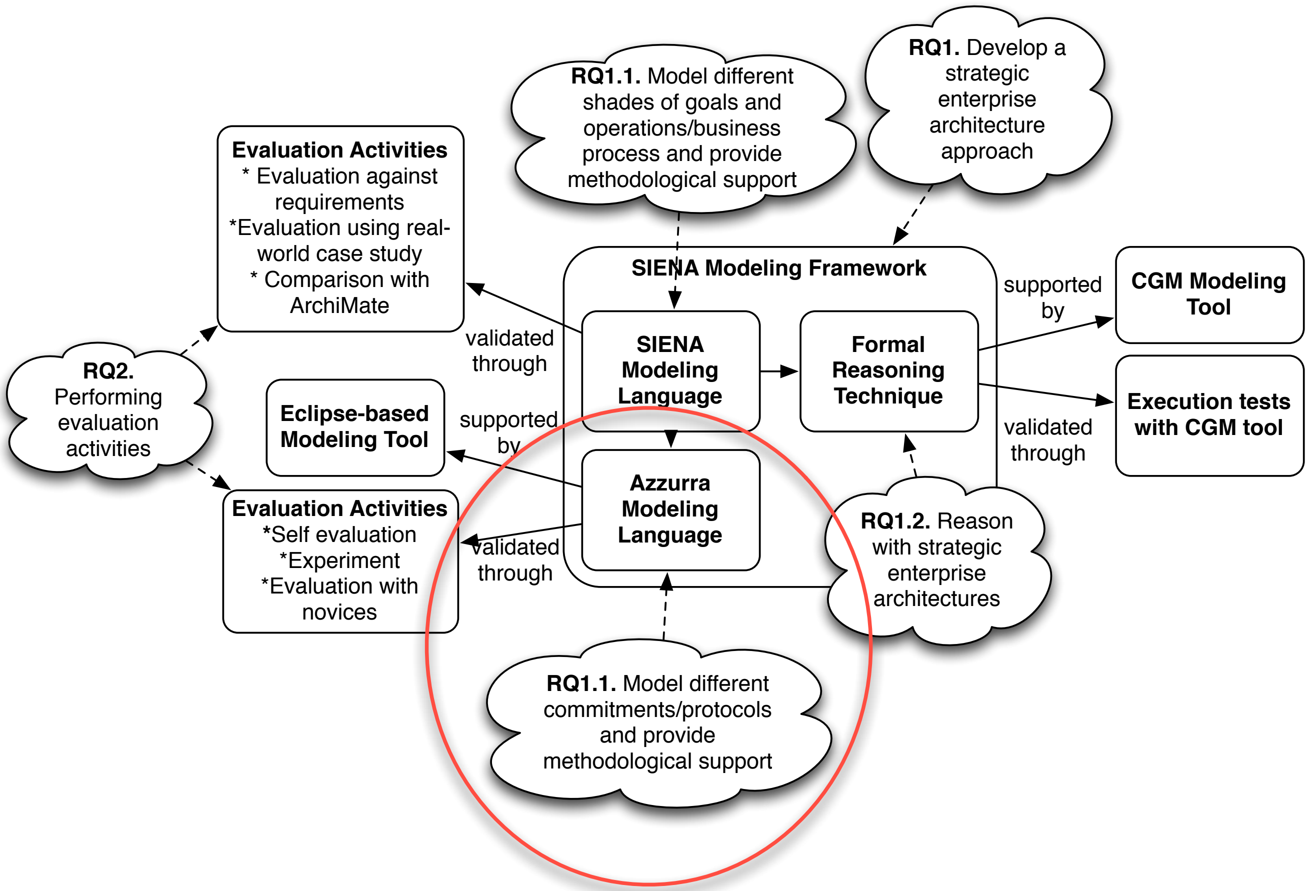


Figure 7.1: The Contribution of this Chapter in the Context of the Overall Thesis

While Chapter 5 has presented the overall SIENA Modeling Language together with its concrete syntax and semantics, Chapter 6 introduced our general approach to strategic enterprise architectures. In Chapter 6, our strategic planning approach started with the formal specification and refinement of strategic goals in terms of their AND/d-refinements. With such specification in hands, the approach generated optimum strategic plans to achieve strategic goals under certain constraints and likelihood of occurrence of business scenarios. Figure 7.2 shows a strategic plan in red to achieve the top strategic goal "Increase sales by 3% over 2 years". As can be noticed from this figure, although our automated reasoning technique can generate optimum strategic plans to achieve strategic goals, such strategic plans reach the level of operations. As a consequence, in order to ensure the achievement of strategic goals, business processes and the BPA have to modeled accordingly. In this context, our overall approach is interested in: How to derive business processes and the BPA starting from Operations?

In Chapter 5 (Section 5.2), our approach described the elaboration of operational goals by refining the tactical goal that corresponds to the final state to be achieved by a given operation into intermediate milestones that compose such operation. For example, starting with the "Carry out promotions in Italy" operation in Figure 7.2, the final state to be achieved by such operation corresponds to the tactical goal that should be refined (in this case, "Carry out promotions in Italy") to acquire the operational goals that compose such operation. Therefore, starting from "Carry out promotions in Italy" operation and applying the rules for operational goals elaboration from Section 5.2 would result in the operational goal model and BPA for promotions depicted in Figure 7.3. Although the guideline for operational goals elaboration state that their refinement finishes when it is possible to find a business process whose final state corresponds to the operational goal under consideration or when it is possible to assign roles for the satisfaction of Operational Goals, the guidelines refrains from

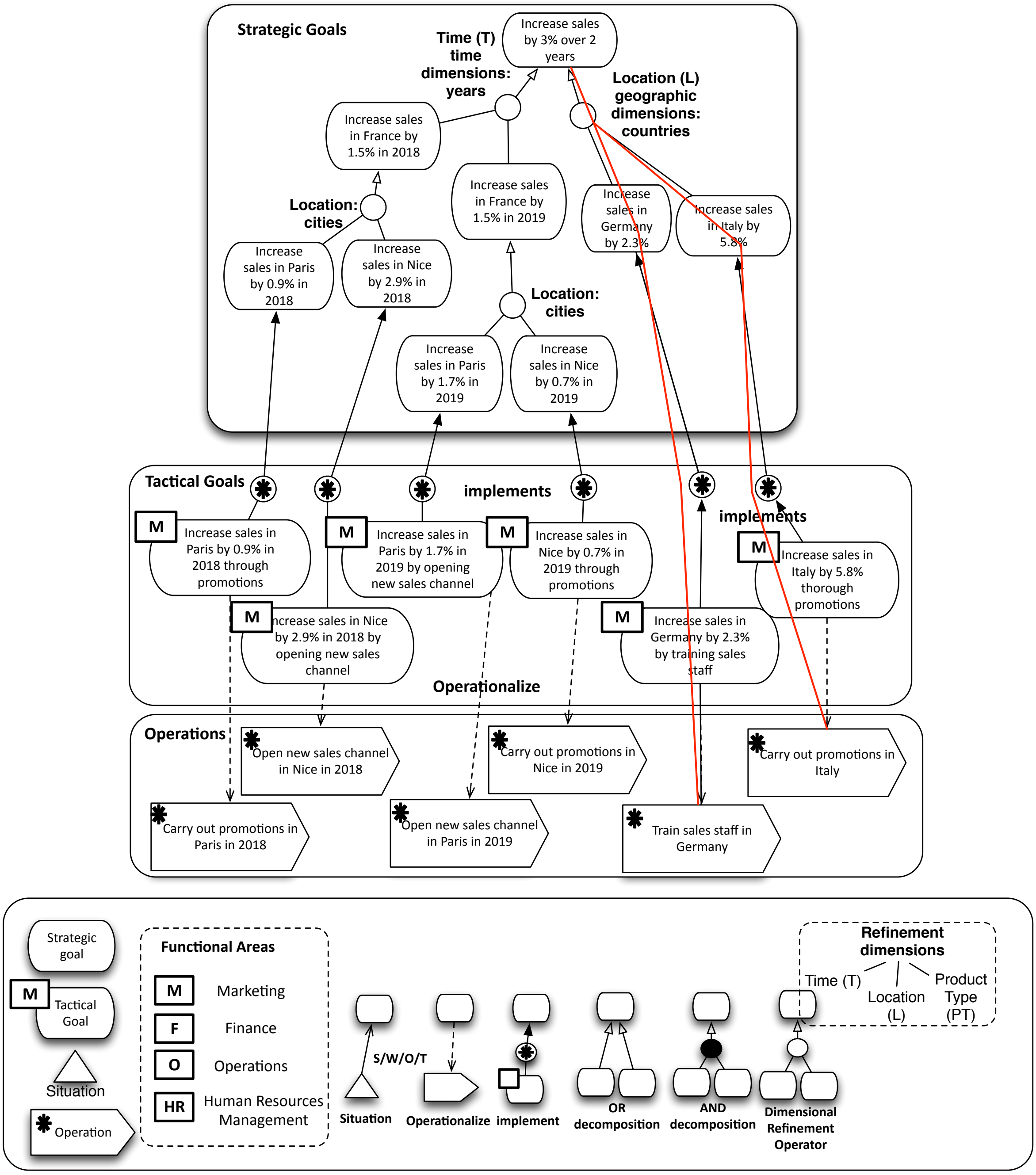


Figure 7.2: Strategic Plan Generated with Automated Reasoning Technique (Chapter 6)

specifying how the control-flow of business processes is derived from the operational goals.

In the remainder of this chapter, we refine the rules for the elaboration and operationalization of operational goals from Section 5.2 in order to

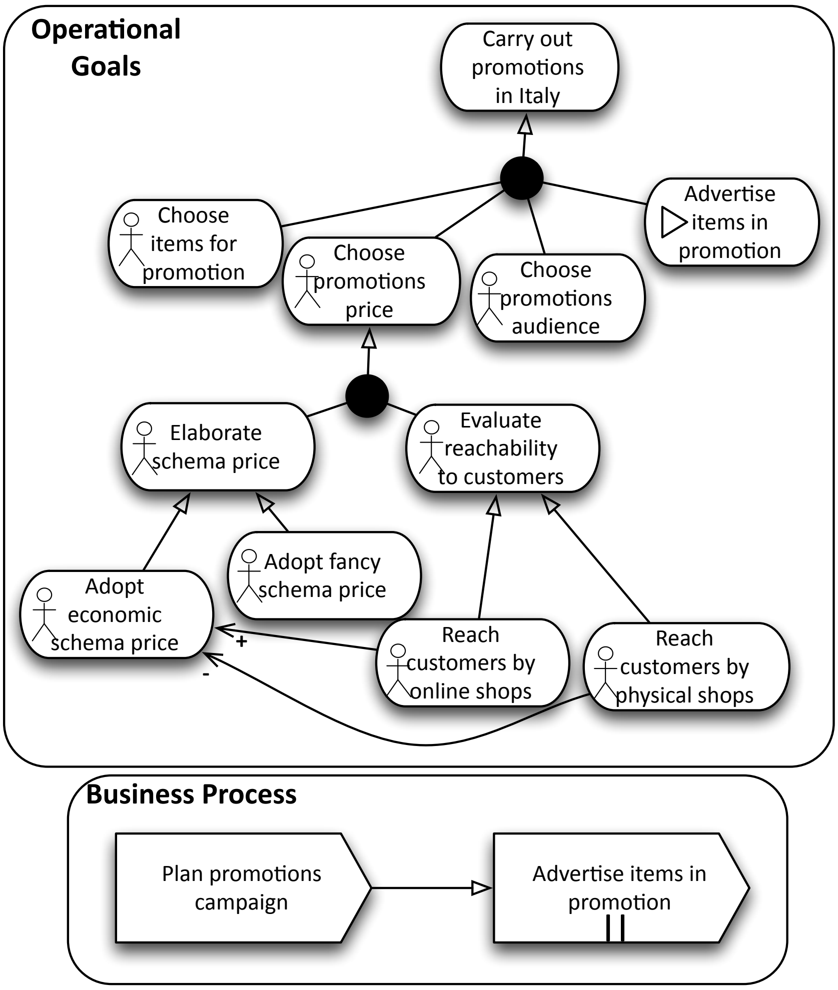


Figure 7.3: Operational Goals From Strategic Plan Generated with Automated Reasoning Technique (Chapter 6)

generate the control-flow of business processes in our strategic planning approach. For that, we present the Azzurra Modeling Language in Section 7, a modeling language for the representation of business processes in social terms that abstracts from the representation of operational and technical details. In order to perform such shift in the representation focus, Azzurra adopts the notion of *social commitment* [184] among actors in a business process as the fundamental business process abstraction. Commitment protocols [207] are then used to represent business processes as a protocol in which commitments explicitly capture the social responsibilities of actors towards each other.

Due to its representation focus on social primitives, Azzurra allows a seamless transition from goals to business process models and therefore, the language is used in the context of our strategic planning approach in order to represent the control-flow of business processes. Subsequently, we present our approach for deriving Azzurra models for the representation of process control-flow in the *BPA - Level 1* Layer from operational goals in SIENA in Section 7.4.

7.2 The Azzurra Modeling Language

This section presents Azzurra's syntax in terms of Extended Backus-Naur Form (EBNF) notation together with its runtime semantics. The syntax is presented in Table 7.1 and illustrated in Table 7.2. Figure 7.4 shows a graphical notation for visualizing the main elements of an Azzurra specification. The notation can be used via a prototype modeling tool built on top of Eclipse (see Section 7.3). The semantics is explained textually while describing the EBNF syntax.

In order to illustrate Azzurra's syntax and graphical notation, we have chosen the fracture treatment extracted from literature [198] which is also used in the original Azzurra publication. Although we are aware that follow using the manufacturing company scenario presented in previous chapters (Chapters 3, 5 and 6) could be more intuitive, we consider that the fracture treatment better illustrates the features of the modeling language and its advantages over the current state of the art in process modeling. We follow using the example of the metal manufacturing company to illustrate the derivation of Azzurra models from operational goals in SIENA in Section 7.4.

Notational conventions. We denote classes with identifiers that have a leading capital letter, and instances with identifiers that have a leading lowercase letter.

Protocol signature (1,3). A protocol (1) has an identifier p_{id} and a set of parameters (3): a "key" variable that is the unique identifier for the instances of that protocol, and a set of agent variables (two or more) associated with specific roles. Protocol designers are responsible for choosing a meaningful key for the protocol. The agent variables indicate those agents that play certain roles when a protocol is instantiated. The semantics of protocol instantiation is explained later in this section.

Example. In the treatment protocol in Table 7.2, the protocol name is Treatment, the key is the hospitalization number hospnr, the agent variables are patient pt and specialist sp.

Protocol body (2). It includes a set of typed agent variables (their type is a role), a set of commitment classes, a set of protocol refinements (optional), and a knowledge base that defines semantic relations between atomic propositions (optional).

Example. In Table 7.2, there are five agent variables, including rc (a rehab center) and ra (a radiologist), nine commitments (C_1–C_9), and two commitment refinements.

Commitments (5,6). The core of a protocol (5) consists of commitment classes. A commitment in Azzurra (6) extends the semantics presented in our baseline in different ways. First, we introduce the notion of a strong commitment (C^*), where the debtor commits to bring about the consequent only after the antecedent has occurred. Second, given that commitments belong to a specific Azzurra protocol, every state of affairs appearing in the antecedent and consequent of a commitment (e.g., Examined, Diagnosed) has an implicit parameter, i.e., the key of the protocol. This parameter enables relating commitment instances associated with one protocol instance (e.g., examined(121) and diagnosed(234) refer to two different protocol instances, each concerning a specific patient hospitalization). Third, Azzurra enriches the syntax of commitments with triggers and creation deadlines. A trigger—the expression before the $\twoheadrightarrow$ symbol—is an event that triggers a commitment creation. Triggers may have an associated precondition—[*prec*] in (5)—that indicates that, when the event occurs, the commitment shall be created only if the precondition evaluates to true. A deadline ($\leq time$) specifies that the commitment has to be created within a certain time period after the trigger event fires off. Finally, Azzurra supports two special types of commitments that relate to protocol instantiation and termination:

- *Initial commitments* are created when a protocol is instantiated. Their trigger is "init", an event that occurs when a protocol is instantiated. Debtor and creditor of initial commitments shall be agent variables in the parameters of the protocol. This way, initial commitments are created between couples of agents (debtor and creditor do not refer to unassigned agent variables).

- *Final commitments*: every protocol must contain at least one final commitment. A protocol instance terminates successfully when any of its final commitments are fulfilled, while it terminates unsuccessfully if all final commitments are violated (e.g., canceled by the debtor). Final commitments are also initial. When a protocol terminates, all debtors of active commitments are released from their responsibility towards the respective creditors.

The agent variables corresponding to debtor and creditor prescribe that:

- if an agent a is assigned to the agent variable, a shall be debtor (or creditor);

Table 7.1: EBNF syntax of Azzurra; terminals in bold, non-terminals in italics

prot→	**protocol** p_{id} (*params*) **{**	(1)
	[**ag-variables:** *vars*]	
	commitments: *comms crefn**	
	[**refinements:** (*id* : *refn*)*] [**kb:** *domain*$^+$] **}**	(2)
params→	**key** v, v **:** *role* (, v **:** *role*)$^+$	(3)
vars →	v **:** *role* (, v **:** *role*)***;**	(4)
comms →	(**init** $\twoheadrightarrow$ [$\leq_{time}$] *comm* **final;**)$^+$ (*ev* [[*prec*]] $\twoheadrightarrow$ [$\leq_{time}$] *comm*;)*	(5)
comm →	*id* : **C**[*****](v, v, *prop*, *prop*)	(6)
crefn →	**deadline**(*id*, *time*)| **can-deleg-ret-resp**(*id*)|	
	can-deleg-no-resp(*id*)|**can-assign-ret-cred**(*id*)|	
	can-assign-no-cred(*id*) | **can-cancel**(*id*)	(7)
refn →	**max-per-role**(*role*, *nr*) | **max-of-class**(*role*,*id*, *nr*) |	
	role-confl(*role*, *role*)|**comm-role-confl**(*role*, *id*, *id*) |	
	sep-duties(*id*, *id*)	(8)
prec →	*atom* | *cstate* | *pstate* | *prec op prec* | ¬*prec* | (*prec*)	(9)
prop →	*atom* | *cstate* | *pstate* | *prop op prop* | (*prop*)	(10)
op →	$\wedge$ | $\vee$ | $\oplus$ | $\cdot$	(11)
cstate →	**create**(*id*) | **deleg-no-resp**(*id* [**to** v]) | **deleg-ret-resp**(*id* [**to** v]) |	
	fulfil(*id*)|**cancel**(*id*) | **expire**(*id*) | **release**(*id*) |	
	assign-ret-cred(*id* [**to** v])|**assign-no-cred**(*id* [**to** v])	(12)
pstate →	**init-p**(p_{id} (, $v = v$)*) | **fulfil-p**(p_{id} (, $v = v$)*)	(13)
ev →	**init** | *atom* | *cstate* | *pstate*	(14)
atom →	$\top$ | $\bot$ | *staffairs* [(v (, v)*)]	(15)
domain →	**implies**(*staffairs*, *staffairs*) | **mut-excl**(*staffairs*(, *staffairs*)$^+$)	(16)

- if the agent variable is unassigned, any agent a' can be debtor (or creditor), and a' is assigned to the agent variable by participating in the commitment.

Example. In Table 7.2, C_1 is the only initial and final commitment. The protocol has two agent variable parameters (pt and sp), which are the

Table 7.2: Azzurra protocol for the fracture treatment scenario

protocol Treatment (**key** hospnr, pt : Patient, sp : Specialist) {
ag-variables: rc : RehabCentre, ra : Radiologist, or : Orthopedist,
su : Surgeon, nu : Nurse;
commitments:
init $\twoheadrightarrow$ C_1 : C(sp, pt, $\top$, Examined · Diagnosed · Dehospd) final
NoXRayNeeded $\twoheadrightarrow$ C_2 : C(or, sp, $\top$, SlingMade)
XRayRequested $\twoheadrightarrow$ C_3 : C(ra, sp, $\top$, XRayPerformed)
XRayRequested $\twoheadrightarrow$ C_4 : C^*(sp, ra, XRayPerformed, FractAssessed)
FractAssessed $\twoheadrightarrow$ C_5 : C(or, sp, $\top$, ((Fixated$\oplus$Plastered) $\vee$ fulfil(C_6) $\vee$ SlingMade))
FractAssessed $\twoheadrightarrow_{\leq 2h}$ C_6 : C^*(su, or, SurgeryRequested,Operated)
Operated [$\neg$fused] $\twoheadrightarrow$ C_7 : C(nu, pt, $\top$, RcChosen(rc))
RcChosen(rc) $\twoheadrightarrow$ C_8 : C(rc, pt, $\top$, fulfil-p(RehabGiven, key=hospnr, pat-id=pt, ref-sp=sp))
MedPrescribed(m) $\twoheadrightarrow$ C_9 : C(nu, sp, $\top$, MedApplied(m))
can-deleg-no-resp(C_3)
deadline(C_2, 2h)
protocol refinements:
role-confl(Radiologist,Orthopedist)
kb:
implies(XRayRequested, Diagnosed)
implies(NoXRayNeeded, Diagnosed)
implies(MedPrescribed(m), Diagnosed)
mutExcl(XRayRequested, NoXRayNeeded) }

debtor and the creditor of C_1. When an instance of the protocol is created, with agent frank assigned to sp and agent mel assigned to pt, an instance c_1 of C_1 shall be created with debtor frank and creditor mel. When c_1 is fulfilled (the patient is examined, then diagnosed, and finally dehospitalized), the protocol instance terminates successfully. If c_1 is violated, the protocol terminates unsuccessfully. The triggered commitment C_2 is instantiated only if x-rays are not needed, and it specifies that an or has to commit to sp to make a sling. C_4 shows strong commitments: a specialist commits to assess the fracture only after x-rays have been performed.

Agent variables (2,4). We support agent variables that are unassigned

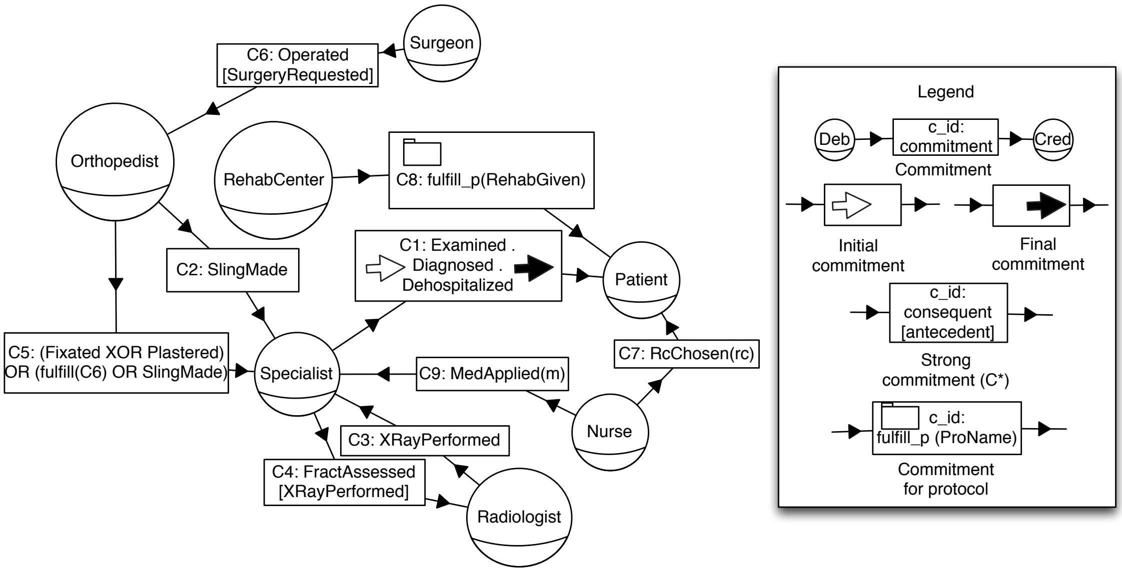


Figure 7.4: Graphical representation for the Azzurra protocol in Table 7.2

when the protocol is instantiated. They get assigned when an instance of a commitment where they appear is created, and, as an additional effect, the assigned agent adopts the specified role in the protocol instance. Azzurra employs assign-once variables: once an agent is assigned, no other agent can be assigned to that variable.

Example. In Table 7.2, there are agent variables for a rehab center, a radiologist, an orthopedist, a surgeon, and a nurse. Actual agents will be assigned to these variables as the protocol unfolds, i.e., when commitments are created. For example, an orthopedist will be assigned to or as soon as an instance of C_2 is created.

Commitment refinements (7). A deadline commits the debtor to bring about the consequent within a certain time after the antecedent occurs. The debtor can be authorized to delegate the commitment, either retaining (can-deleg-ret-resp) or releasing (can-deleg-no-resp) her responsibility. The creditor, similarly, can be authorized to assign the commitment, either retaining (can-assign-ret-cred) or releasing (can-assign-no-cred) her credit.

The debtor can be authorized to cancel her commitment (can-cancel).

Example. In Table 7.2, the radiologist can delegate instances of C_3, possibly to a colleague, without retaining responsibility. Without such authorization, delegations would correspond to a violation on part of the radiologist.

Protocol refinements (8). They constrain the agents that participate in a protocol instance. The maximum number of concurrent commitments for an agent playing a certain role can be limited (max-per-role), as well as the number of instances of a commitment class that an agent can make (max-of-class). Role conflicts (role-confl) prescribe that an agent cannot play two roles in the same protocol instance. Separation of duties (sep-duties) implies that an agent cannot be the debtor in instances of two commitment classes, and it can be restricted to agents playing a specific role (comm-role-confl).

Example. A role-confl refinement specifies that the same agent cannot play both radiologist and orthopedist, because their roles are incompatible in the same protocol instance.

Preconditions, propositions, and triggers (9–15). Azzurra supports different types of preconditions (9) and propositions types (10): atomic (atom), commitment states (cstate), protocol states (pstate), binary operators, and so on. The binary operators (11) are conjunction ($\wedge$), disjunction ($\vee$), exclusive disjunction ($\oplus$), and temporal precedence ($\cdot$). Atomic propositions (15) can be truth ($\top$), falsity ($\bot$), or states of affairs (e.g. FractAssessed). States of affairs may be parametric and, thus, have multiple instances. For example, MedPrescribed(med-id) has an instance for each medication the patient is given. The state of a protocol instance evolves because of the occurrence of events (14), as they trigger new commitment instances and change the state of existing commitment instances. Three event types are supported:

- An atomic proposition becomes true. This includes the occurrence of a state of affairs (e.g., the patient is diagnosed).
- The state of a commitment instance changes (see clause (12) below).
- The state of another protocol instance changes, i.e., it is instantiated (init-p) or fulfilled (fulfil-p). Optionally, one can specify constraints on the protocol instance parameters, e.g., to impose a certain key or that a specific agent in the current protocol instance shall be assigned to an agent parameter in the referenced protocol.

Example. The consequent of C_5 tells that the commitment is fulfilled if either an instance of Fixated or Plastered occurs (but not both), an instance of C_6 is fulfilled, or an instance of SlingMade occurs. The consequent of C_8 indicates that a successful instance of the protocol RehabGiven is expected, with the constraints that the patient identifier parameter (pat-id) corresponds to the patient in the instance of Treatment, and that the reference specialist (ref-sp) is the specialist who is responsible for the hospitalization of the considered patient.

Commitment states (12). Propositions may denote that a commitment is in or has changed to a specific state. Given a commitment class id:

- create(id): an instance of id is created;
- deleg-no-resp(id [to v]): an instance of id is delegated (to agent v) without retaining responsibility;
- deleg-ret-resp(id [to v]): an instance of id is delegated (to v); the delegator keeps responsibility;
- fulfil(id): an instance of id is fulfilled;
- cancel(id): an instance of id is canceled;

- expire(id): an instance of id has expired;

- release(id): an instance of id is released;

- assign-ret-cred(id [to v]): an instance of id is assigned (to v) retaining the credit;

- assign-no-cred(id [to v]): id is assigned, but the assignor does not retain the credit.

Knowledge base (16). It specifies semantic relationships, i.e., implications and mutual exclusions, between states of affairs. These relationships belong to the shared vocabulary of the participants in a protocol.

Example. Three states of affairs imply a diagnosis: XRayRequested, NoXRayNeeded, and MedPrescribed. XRayRequested is mutually exclusive with NoXRayNeeded.

7.3 Implementation

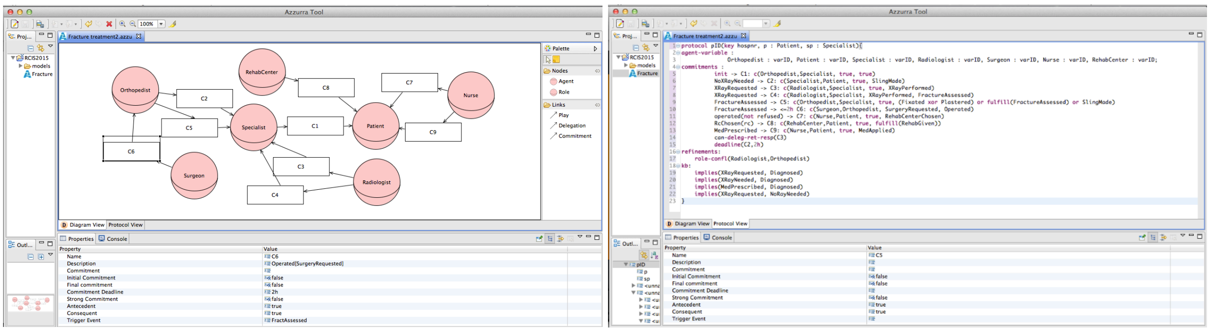
As a proof of concept, a prototype[1] that enable the creation of Azzurra textual and diagrammatic specifications has been developed. The Azzurra modeling tool is a standalone Eclipse application, built on top of the GEF (Graphical Editing Framework) and XText frameworks. The environment supports the modeling of business processes in terms of *views* that allows the modeler to focus on different aspects of the domain and, thus enables a better separation and representation of concerns during modeling time.

The *social view* (depicted in Fig. 7.5a) provides an intuitive interface for the modeler, by enabling designers to graphically represent the social relations among the several roles and agents, in terms of their commitments and commitment delegations. Using the Properties tab (below the

[1] `https://trinity.disi.unitn.it/azura/azura/`

graphical representation), it is possible to specify commitment's name, antecedent, consequent, triggering event and deadline. Further, one can also specify whether a given commitment is an initial, final or strong commitment.

The *protocol view* (textual view) also enables designers to enrich the specification by capturing other details like triggering events for commitments as well as commitment refinements and parts of the knowledge base of the protocol (depicted in Fig. 7.5b). Commitment refinements can be captured either by editing the Properties tab or by editing the textual protocol representation as depicted in Fig. 7.5b. Finally, the tool also enhances the modeling process by enabling the checking of well-formed Azzurra models, detecting invalid commitments and commitment delegations at modeling-time.



(a) *Social view* (graphical representation) (b) *Protocol view* (textual representation)

Figure 7.5: Views of fracture treatment scenario (Fig. 7.4) using the Azzurra modeling language

7.4 Formal Business Process Design from Operational Goals

Chapter 5 (Section 5.2) provided methodological guidelines for the specification of SIENA models in the *BPA - Level 0* Layer. This section presents the specification of guidelines for SIENA models, by prescribing guidelines for the specification of *BPA - Level 1* and *BPA - Level 2* Layers from the SIENA framework.

Guidelines for *BPA - Level 1* Layer describe a top-down approach to the design of business process's control-flow and their subsequent specification using the Azzurra language. Such top-down approach starts with Operational Goals (Section 5.1.1) and derives the elements required for the specification of business processes in Azzurra.

The task of promoting a transition from goal models to process models is usually challenging due to the nature of information captured in both types of models. On one hand, goal modeling involves the representation of "why" certain states need to be achieved in the course of performing the process, regardless "how" to achieve it. On the other hand, process models precisely capture "how" to satisfy such goals, also considering the temporal aspects of behavior like durations and deadlines. Therefore, as process models consist of detailed specifications on how goals are operationalized, the transition from goal models to process models requires the specification of additional information, such as ordering between the achievement of goals or deadlines for their achievement.

In order to facilitate the specification of this additional information about the operationalization of goals in the process model, our approach is inspired by the KAOS requirements specification and operationalization process [118, 38, 10] (described in Section 3.6). In this context, the KAOS approach has been chosen for two reasons. First, the approach provides an operationalization process in which a model of software operations is incrementally built from goal formulations, thus ensuring correctness by design of software specifications [118, 10]. Second, the framework also provides a formal goal refinement and operationalization approach based on refinement patterns that guides the goal refinement, helping to identify mistakes and missing elements in goal refinements [38]. Therefore, we take advantage of both approaches in order to ensure the derivation of correct and complete Azzurra specifications. Further, as we intend to depict an integrated approach for the generation of SIENA's goal hierarchy, we follow with the example of the metal manufacturing company of Chapter 5 (more precisely, we start with Figure 5.7 that represents Operational Goals).

In the remainder, we describe a mapping from SIENA *Operational* Level concepts to the KAOS formalism (Section 7.4.1), followed by the derivation approach from SIENA *Operational* Level concepts to Azzurra specifications (Section 7.4.2).

7.4.1 Specify Operational Goal Models using KAOS Semantics

Our approach starts by mapping the concepts from SIENA *Operational* Level to the KAOS goal language formalism in order to use the KAOS operationalization approach as a source of inspiration for the specification of business processes in Azzurra. The concepts from *Operational* Level here used are operational goals and their AND/OR refinements, positive/negative contributions, situations and domain assumptions, whereas KAOS concepts are goals (their natural language description and its formal counterpart in LTL), goal patterns, AND/OR refinements and domain assumptions.

Starting with the mapping of concepts, the SIENA distinction between *(Operational) Role Goals* and *(Operational) Business Process Goals* intends to represent different levels of assignment for Operational Goals, i.e., business process goals represent goals to be achieved by multiple roles. Such differences are reflected in their specification methodology and thus, one needs to further refine business process goals in order to reach roles goals, thus having a finer-grained perspective of Operational Goals. As a consequence of such differences, both types of SIENA Operational Goals are mapped into a KAOS *goal* and their differences are equally reflected in different steps in business process design. For the mapping of other SIENA concepts, *domain assumptions* are trivially mapped into KAOS *domain assumptions* and *AND/OR refinements* are trivially mapped into KAOS *AND/OR refinements*.

Although some SIENA concepts can be straightforwardly mapped into KAOS concepts, both languages contain concepts that do not overlap. In KAOS, such concepts are LTL goal assertions and goal patterns while in SIENA, situations, their SWOT relations and positive/negative contributions among Operational Goals. In order to deal with such non-overlapping concepts, we first investigated their semantics accordingly and tried to find a correspondence between both languages.

Regarding KAOS goals patterns and LTL assertions, our intention is to use KAOS operationalization approach as the source of inspiration and thus we start by formalizing the root SIENA *Operational Goal* in terms of KAOS *LTL goal assertions* and *goal patterns*. For example, the root Operational Goal "Carry out promotions" (**Goal** *Achieve* [*PromotionsCarriedOut*]) is formalized in terms of LTL assertions and goal patterns as follows:

$$\begin{aligned}&\textbf{Goal} Achieve[PromotionsCarriedOut]\\&\quad\textbf{FormalDef}\quad \exists i : Item\\&\qquad IdentifiedNeedPromotion(i)\\&\qquad\quad \implies \Diamond(\exists p : Promotion)(Advertised(i,p) \wedge Run(p))\end{aligned}$$

in which the predicate $IdentifiedNeedPromotion$ checks whether some items remain in the stock and promotions need to be carried out to sell them. In our business process design approach, after formalizing the root Operational Goal by means of LTL assertions and goal patterns, such goals need to be formally decomposed accordingly. This step is further described in Section 7.4.2 (step 2).

For the mapping of SIENA concepts (positive/negative contributions among Operational Goals, situations and their SWOT relations) into KAOS concepts, although positive/negative contributions do not have any correspondence in KAOS, they are still useful for our business process design approach described next section (Section 7.4.2). In contrast, although situations and their relations are very useful for representing the factors that impact goal achievement, they do not impact our business process design approach and consequently, they are dropped for our analysis here.

Table 7.3 depicts a mapping of concepts between SIENA and KAOS languages. For those concepts that do not present a direct correspondence in both languages like goal patterns and formal LTL goal assertions, we explain how a correspondence is established in Section 7.4.2. For this reason, the third and sixth rows of Table 7.3 point out to Section 7.4.2. Further, as situations and SWOT relations are not used in our business process design approach, the - sign in Table 7.3 is used to denote that both concepts are not mapped.

<table>
<tr><th>SIENA</th><th>KAOS</th></tr>
<tr><td>(Operational) Role Goal</td><td rowspan="2">Goal</td></tr>
<tr><td>(Operational) Business Process Goal</td></tr>
<tr><td rowspan="2">Section 7.4.2</td><td>Goal Pattern</td></tr>
<tr><td>Formal Goal Assertions in LTL</td></tr>
<tr><td>Domain Assumption</td><td>Domain Assumption</td></tr>
<tr><td>AND/OR refinement</td><td>AND/OR refinement</td></tr>
<tr><td>Positive/Negative Contribution</td><td>Section 7.4.2</td></tr>
<tr><td>Situation</td><td rowspan="2">-</td></tr>
<tr><td>SWOT relations between Situations (S) and Goals (G)</td></tr>
</table>

Table 7.3: Mapping between SIENA and KAOS concepts

7.4.2 Deriving Azzurra Models from Formal Operational Goals

Once we have mapped SIENA *Operational* Level concepts into KAOS semantics, this section describes the steps of our approach for business process design by deriving Azzurra models from Operational Goals at the SIENA *Operational* (Goal) Level.

1. Derive Roles and Agents from Operational Goal Models. Within the *Operational* Level in SIENA, the roles responsible for the achievement of Operational Goals are specified during their elaboration (Guideline G4, Section 5.2). Therefore, our approach starts by deriving the roles $\{R_1, \ldots, R_N\}$ responsible for the achievement of Operational Goal and specifying them accordingly in Azzurra models.

Other details regarding the specification of roles can be also specified at this stage. First, as Azzurra requires the specification of roles as debtor and creditors, roles responsible for the achievement of Operational Goals become the debtor to other roles to achieve the Operational Goal under consideration. Similarly, protocol refinements may be also specified at this stage, namely, the maximum number of concurrent commitments for an agent (max-per-role), the number of instances of a given commitment class that an agent can make (max-of-class), conflicts between roles (role-confl) and separation of duties (sep-duties and comm-role-confl).

2. Formally Decompose Operational Goals. Subsequently, the approach proposes to formally decompose SIENA Operational Goals in terms of AND/OR refinements with the support of KAOS goal refinement patterns. Such refinement patterns [38] have the purpose of supporting goal refinement, helping to ensure consistency and completeness of refinements and allowing correct operationalization of Operational Goals.

In Section 5.2.4, methodological guidelines for informal AND/OR refinements of Operational Goals have been proposed. More specifically, the guideline for AND decomposition states that the final state of each Operation becomes a root Operational Goal that needs to be structurally decomposed into intermediate Operational Goals (milestones) necessary for the execution of some tactics, whereas OR decomposition can be used to represent the existence of different alternatives for achieving an Operational Goal. However, the informal nature of such guidelines may lead to incomplete or inconsistent goal refinements [38], thus indirectly leading to incomplete Azzurra specifications. Therefore, this formal refinement step proposes to formally refine Operational Goals with the support of KAOS goal refinement patterns [38] (Section 3.6).

In order to do that, the approach starts by formally decomposing the LTL specification of the root Operational Goal into sub-goals using the same guidelines from Section 5.2 and KAOS refinement patterns. For example, the root Operational Goal "Carry out promotions" (**Goal** *Achieve* [$Promotions\ CarriedOut$]) has been informally AND-refined in Chapter 5 (Figure 5.7) in terms of "Plan promotions campaign" and "Advertise items in promotion" sub-goals. A formalization of such sub-goals may be done as follows:

$$\begin{array}{l}\textbf{Goal} Achieve[PromotionsCampaignPlanned]\\ \quad \textbf{FormalDef} \quad \exists i : Item\\ \quad\quad IdentifiedNeedPromotion(i)\\ \quad\quad\quad \Longrightarrow \Diamond(\exists p : Promotion)(Planned(p))\end{array}$$

$$\begin{array}{l}\textbf{Goal} Achieve[ItemsAdvertisedPromotion]\\ \quad \textbf{FormalDef} \quad \exists i : Item, p : Promotion\\ \quad\quad IdentifiedNeedPromotion(i) \quad \wedge \quad Planned(p)\\ \quad\quad\quad \Longrightarrow \Diamond(Advertised(i,p) \wedge Run(p))\end{array}$$

However, the application of the *Achieve* KAOS refinement pattern (*Refinement Pattern (RP1)*) [38] (Figure 3.5) to the "Carry out promotions" goal evidences an incompleteness of this informal refinement as the RP1 suggests three sub-goals for an *Achieve* goal. More precisely, the RP1 states that an *Achieve* [$P \Longrightarrow \Diamond\ Q$] goal should be AND-decomposed into three sub-goals: *Achieve* $[P \implies \Diamond R]$, *Achieve* $[P \wedge R \implies \Diamond Q]$ and *Maintain* $[P \implies P\ W\ Q]$. In this case, the patterns helps analysts to find a third missing goal that can be formalized as follows:

$$\textbf{Goal}Maintain[NormalPricesKept]$$
$$\textbf{FormalDef} \quad \exists i : Item$$
$$IdentifiedNeedPromotion(i)$$
$$\implies \Diamond IdentifiedNeedPromotion(i) \quad W \quad (\exists p : Promotion)$$
$$(Advertised(i, p) \wedge Run(p))$$

therefore, the missing goal can be added to Operational Goal model, what indirectly would impact the Azzurra specification. Alternatively, we could have applied RP3 (Figure 7.6) that states that an *Achieve* $[P \implies \Diamond Q]$ goal should be AND-decomposed into two sub-goals: *Achieve* $[P \implies \Diamond R]$ and *Achieve* $[R \implies \Diamond Q]$. This would yield the following sub-goals:

$$\textbf{Goal}Achieve[PromotionsCampaignPlanned]$$
$$\textbf{FormalDef} \quad \exists i : Item$$
$$IdentifiedNeedPromotion(i) \implies \Diamond(\exists p : Promotion)Planned(p)$$

$$\textbf{Goal}Achieve[ItemsAdvertisedPromotion]$$
$$\textbf{FormalDef} \quad \exists p : Promotion$$
$$Planned(p) \implies \Diamond(\exists i : Item)(Advertised(i, p) \wedge Run(p)).$$

As can be seen, the use of one pattern instead of another may lead to different goal models, indirectly impacting in the Azzurra specification. Therefore, the choice of the refinement pattern to be used depends on different design purposes. For example, by using RP1, three sub-goals are generated, whereas the use of RP3 yields two sub-goals. As each Operational goal is translated to a commitment in the Azzurra specification (see

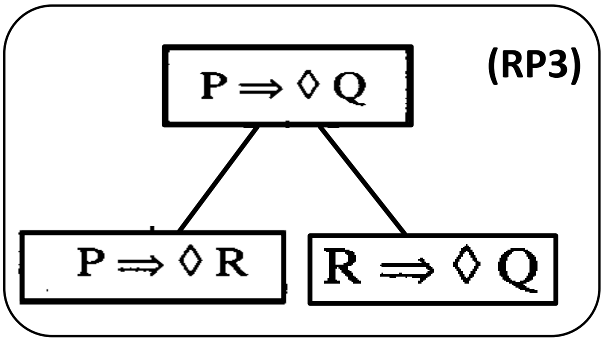


Figure 7.6: Example of KAOS Refinement Patterns Extracted From [38]

step 4), the choice of RP1 may lead to more commitments to be achieved by process participants. Observe also that the previous example formally AND decompose the root Operational Goal, but the alternatives for achieving Operational goals expressed by means of OR refinements should be also checked for consistency at this stage. Such AND/OR refinements finish when it is possible to find either a business process whose final state corresponds to the goal or roles (process participants) that are responsible for achieving such goals (similarly to Section 5.2).

3. Select Leaf Goals for Operationalization and "Best" Alternatives of OR-refinements. By (formally) decomposing goals, an AND-refinement determines the structure of the business process in terms of a series of intermediate goals (milestones) to be achieved [117, 45], while an OR-refinement specifies alternatives ways to achieve such goals. Therefore, the next step consists of selecting which milestones will compose the structure the business processes, by selecting the leaf Operational goals stemmed from the AND-refinements and by selecting the "best" alternatives among the multiple goals specified by OR-refinements. This step can be performed using the CGM formalism in a similar fashion as our strategic planning approach in the previous chapter (Chapter 6). In order to do that, the same steps of our strategic planning approach (Section 6.4) should be adopted, i.e., first Operational Goals are mapped to the CGM formalism and then the model is used by CGM to select the optimum alternatives.

Regarding the mapping of concepts of the Operational Layer, the root Operational Goal must be mapped to a CGM *root goal*, while the other Operational Goals are mapped to CGM *intermediate goals* or *tasks*. AND/OR-refinements of Operational goals and their positive/negative contributions have also the same mapping from AND/OR-refinements and positive/negative contributions from Strategic Goals, i.e., AND-refinements are mapped to CGM *refinement*, OR-refinement is mapped to multiple CGM *refinements* and positive/negative contributions are dropped from the analysis. Subsequently, the CGM formalism may be used to select the best (optimum) alternatives.

4. Derive Commitments from Operational Goals. In KAOS [118, 10], the essence of goal specifications is to decompose goals so that they can be assigned to *agents* responsible for their satisfaction. After assigned to agents, goals need to be *operationalized* accordingly by prescribing pre-, trigger- and postconditions on *operations* in order to achieve goal specifications (Section 3.6). Our approach follows the same rationale by first assigning the Operational Goals selected by CGM to process participants, thus transforming them into Azzurra commitments. Subsequently, these Operational Goals (or commitments) need to be *operationalized* by prescribing commitments' triggering event, pre- and post-conditions. In Table 7.4, we start with KAOS formal goal definitions and provide operationalization patterns for $Achieve$, $Cease$, $Maintain$ and *Avoid* goal patterns.

In Table 7.4, P and Q denote first-order logical formulae in terms of propositions and binary operators. Observe that OP2 and OP5 lines are time-bounded LTL assertions, thus reflecting in its corresponding Azzurra operationalizations. In this context, Azzurra language allows the specification of two types of deadlines (i.e., a deadline for commitments creation ($ev \twoheadrightarrow_{\leq time} \mathrm{C}_n$) and a deadline for commitment fulfillment (deadline(C_n,time)).

Goal Pattern		Goal Formal Definition (from KAOS)	Azzurra Operationalization
Achieve	OP1	P $\Longrightarrow \Diamond$ Q	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, Q)
	OP2	P $\Longrightarrow \Diamond_{\leq d}$ Q	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, Q) deadline (C, d)
	OP3	P $\Longrightarrow \Diamond$ Q	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, Q)
Cease	OP4	P $\Longrightarrow \Diamond \neg$ Q	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, ¬Q)
	OP5	P $\Longrightarrow \Diamond_{\leq d} \neg$ Q	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, ¬Q) deadline (C, d)
	OP6	P $\Longrightarrow \Diamond \neg$ Q	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, ¬Q)
Maintain	OP7	$P \Longrightarrow \Box Q$	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, Q) $\neg Q \twoheadrightarrow$ C (debtor, creditor, antecedent, Q)
Avoid	OP8	$P \Longrightarrow \neg \Box Q$	$P \twoheadrightarrow$ C (debtor, creditor, antecedent, ¬Q) $Q \twoheadrightarrow$ C (debtor, creditor, antecedent, ¬Q)

Table 7.4: Patterns for Operationalizing Goals into Azzurra Specifications

As the LTL assertion states that a proposition Q should hold in some future state, this corresponds to a deadline for the achievement of commitment's consequent and thus, the Azzurra operationalization pattern includes this deadline $deadline(C, d)$. For Maintain and Avoid goal patterns, the LTL expressions states that some property must hold "at all times in the future" for a maintain pattern, while a property must not hold "at all times in the future" for an avoid pattern. In this case, operationalizations pattern OP7 (maintain pattern) states that two commitments must be created. The first one ensures that if the event P happens, then the proposition Q should be brought about. As proposition Q should hold in all future states according to the maintain pattern, if the event ¬Q happens at a certain point in future time, then proposition Q should be brought about again. The same rationale is applied regarding operationalizations pattern OP8.

5. Complete Azzurra Specifications. Once roles and commitments have been derived from Operational Goal models, commitments still lack a number of details to be specified. Such details correspond to commitment's antecedents and other commitment refinements, such as deadlines, strong commitments, initial/final commitments and delegations and their types. Concerning initial/final commitments, Operational Goals in the goal model may be used to derive the commitments which start and finish the protocol, respectively.

6. Specify Commitment Operationalizations (optional). While Azzurra specifies correctness criteria as the commitments' consequent to enable one to determine whether commitments have been fulfilled or not, it refrains from specifying particular operationalizations to achieve such commitments. Therefore, the last stage of our approach concerns the (optional) specification of particular operationalizations (activities and routing connectors) to fulfill such commitments. In this context, given one commitment C (debtor, creditor, antecedent, consequent) which is operationalized in terms of an ordered set of activities $\{a_1, a_2, \ldots, a_n\}$, the commitment's antecedent is the condition that triggers the first activity a_1, whereas the effect of the last activity a_n represents the commitments' consequent.

7.4.3 Illustrative Example

In order to illustrate our approach for the derivation of Azzurra models from formal Operational Goals, Figure 7.7(a) shows the Operational Goal Model from Chapter 5 (Figure 5.7(b)) together with its respective Azzurra specification (Figure 7.7(b)). The approach starts with the SIENA Operational Goal model from Figure 5.7(b) and successively applies the mapping rules described in Section 7.4.1 and 7.4.2, thus generating the corresponding Azzurra specification of Figure 7.7(b).

In this context, a natural question that arises regards the Operational Goal Model to be used as the starting point in our derivation approach. On one hand, as the CGM approach generated an optimum strategic plan (for a particular business scenario), this strategic plan can be used as the starting point in the implementation of strategic alternatives. On the other hand, the whole model (i.e., Operational Goal models that belong to all strategic plans) can be used as the starting point for the implementation of strategic alternatives. The selection of which solution to adopt depends on the managers' purpose, i.e, while the first solution is faster as it implements just one strategic plan, the second solution is more complete, allowing managers to gain a holistic view about the implementation of all strategic alternatives. In the remainder of this section, we take the first solution, i.e., we use the Operational Goal model that corresponds to the promotions tactics (Figure 5.7(b) that starts with the "Carry out promotions" operational goal) and detail each step of our approach that led to the generation of the Azzurra specification of the promotion tactics.

1. Derive Roles and Agents from Operational Goal Models: The derivation of roles starts with Operational Goals from Chapter 5 (Figure 5.7) and derives the roles of "Marketing analyst" (debtor) and "Marketing manager" (creditor) from the Operational Goals. Observe that, as Operational Goals in Figure 5.7 have been refined until reaching the level of role goals, this enabled the straightforward derivation of roles in our approach. In contrast, if the refinement of Operational Goals had only reached business process goals, an extra step would be required to refine Operational Goals until reaching the level of granularity of roles. Further, although no protocol refinements were necessary in this example, they could have been represented in the *protocol view* (see Figure 7.5b), if required.

2. Formally Decompose Operational Goals: The root Operational Goal "Carry out promotions" (Figure 7.7(a)) has been formalized in terms of KAOS LTL assertions and goal patterns and subsequently refined with the support of KAOS refinement patterns. More specifi-

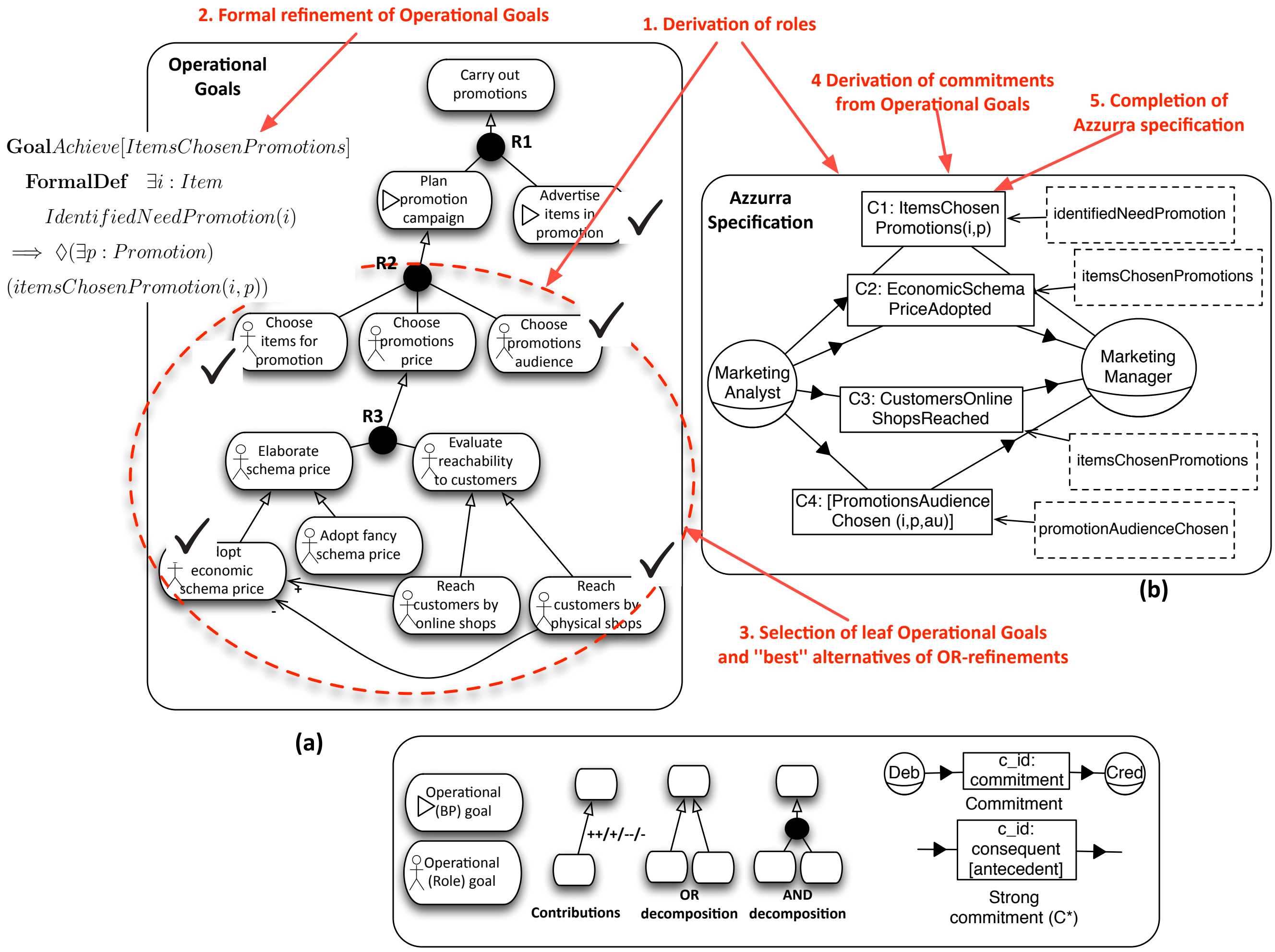


Figure 7.7: Derivation of Azzurra Specification from Operational Goals Model in SIENA

cally, starting from "Carry out promotions", RP3 has been applied within the refinement R1, thus generating "Plan promotion campaign" (**Goal** *Achieve* [*PromotionsCampaign Planned*]) and "Advertise items in promotion" (**Goal** *Achieve* [*Items Advertised Promotion*]) as sub-goals (Figure 7.8). RP3 has been also used to decompose "Plan promotions campaign" (refinement R2)(Figure 7.9) and "Choose promotions price" (refinement R3)(Figure 7.10). The formalization of **Goal** *Achieve* [*Items ChosenPromotions*], one of the sub-goals of "Plan promotions campaign" is also depicted in Figure 7.7(a).

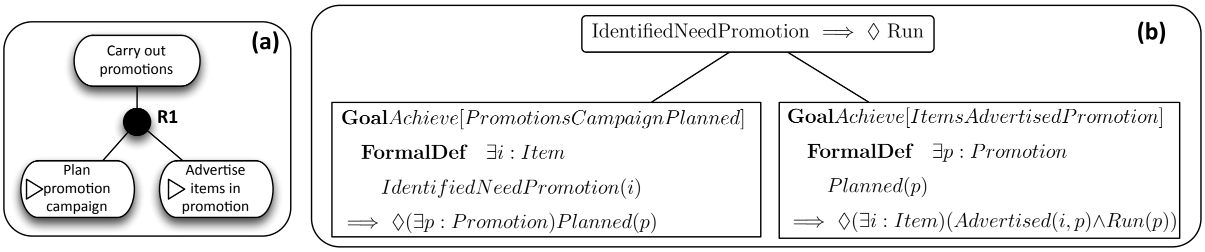


Figure 7.8: Formalization and Refinement of “Carry out promotions” Goal

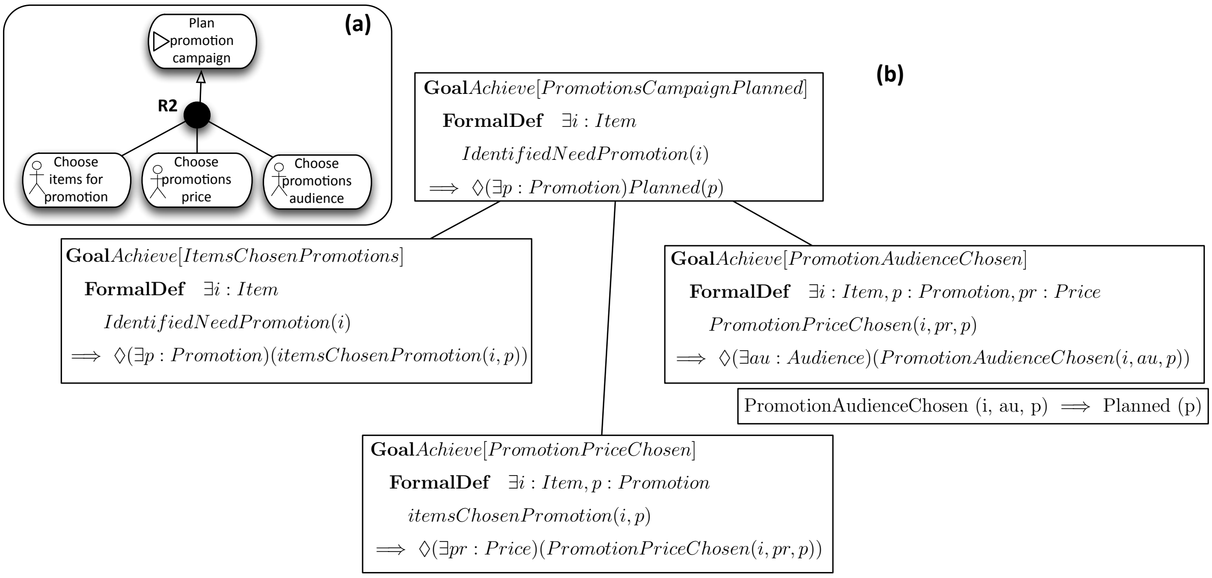


Figure 7.9: Formalization and Refinement of “Plan promotions campaign” Goal

3. Select Leaf Goals for Operationalization and “Best” Alternatives of OR-refinements. After formally refined, goals need to be operationalized. In Figure 7.7(a), the goals selected for operationalization by CGM (circled in red, signed with black check marks) consists of the leaf goals stemmed from AND-refinements and the “best” options from the OR-refinements. In this context, “Adopt economic schema price” and “Reach customer by physical shops” have been selected by CGM as the “best” options in the OR-refinements. In order to perform the selection of goals for operationalization by CGM, the mapping suggested in step 3 from the Operational Goal model (Figure 7.7(a) and 7.11(a)) to CGM has been considered. The corresponding CGM model is depicted in Figure

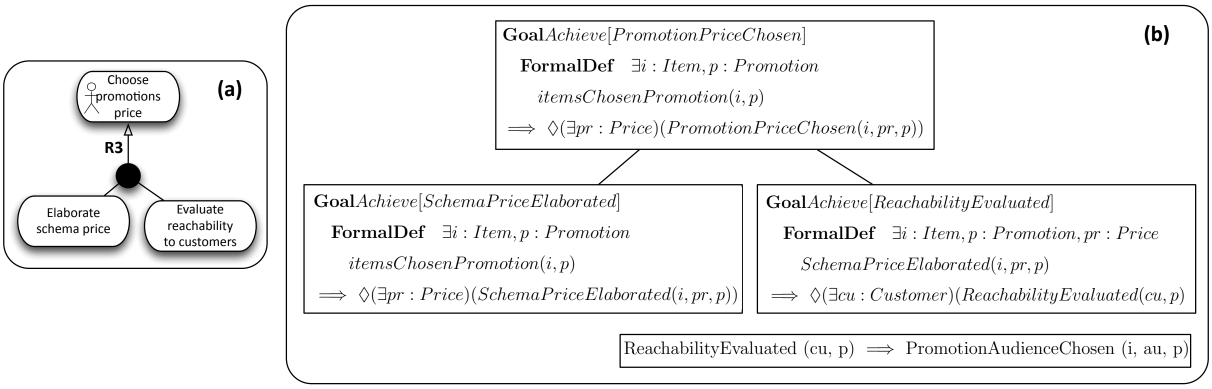


Figure 7.10: Formalization and Refinement of "Choose promotions price" Goal

7.11(b) together with the optimum (best) realizations generated by CGM (in blue). Observe that the best realizations generated by CGM in Figure 7.11(b) correspond to the "Adopt economic schema price" and "Reach customer by physical shops" Operational Goals depicted in Figure 7.11(a).

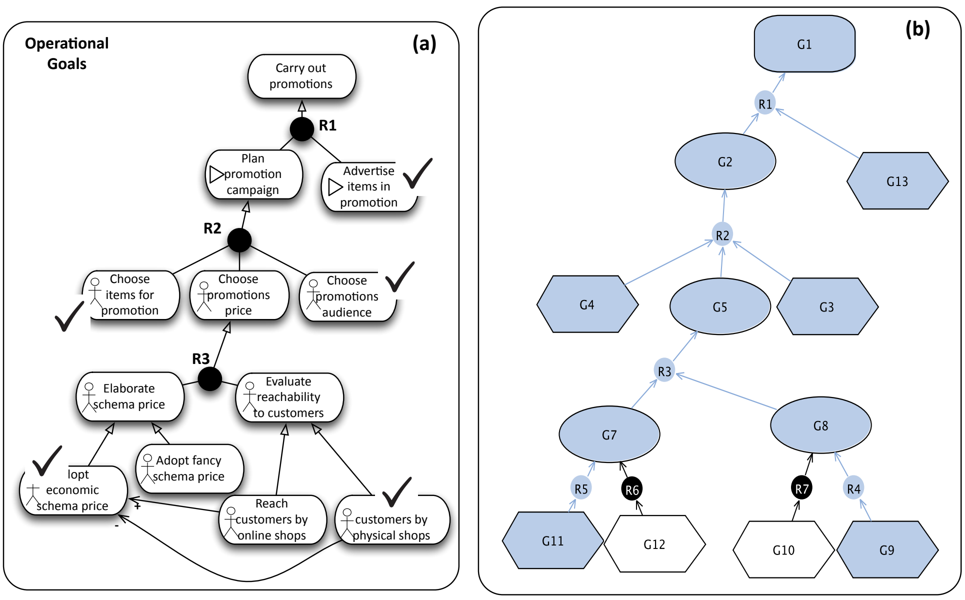


Figure 7.11: Selection of Goals for Operationalization by CGM (step 3)

4. Derive Commitments from Operational Goals. After Operational goals have been selected, they need to be to be operationalized accordingly. Therefore, the **Goal** *Achieve* [*ItemsChosenPromotions*] goal (Figure 7.7(a)) has been operationalized into commitment C_1 (Figure 7.7(b)) using operationalization pattern OP3 from Table 7.4. The other goals with check marks have been operationalized similarly.

5. Complete Azzurra Specifications. Other commitments details (e.g. triggering events, deadlines, strong commitments, initial/final commitments, delegations and their types) are also specified within the *protocol view*. Observe also that, as Azzurra language does not represent the triggering events in its graphical representation (*social view*), we have denoted them as dashed squares in Figure 7.7(b) in order to illustrate our approach.

6. Specify Commitment Operationalizations (optional). Finally, operationalizations for the commitment C_1 that concerns the choice of items to be sold in promotions are also specified (Figure 7.12) in terms of the operational activities required to fulfill the commitment. In Figure 7.12, different items are chosen to be sold in promotions, depending on the goal of such promotion.

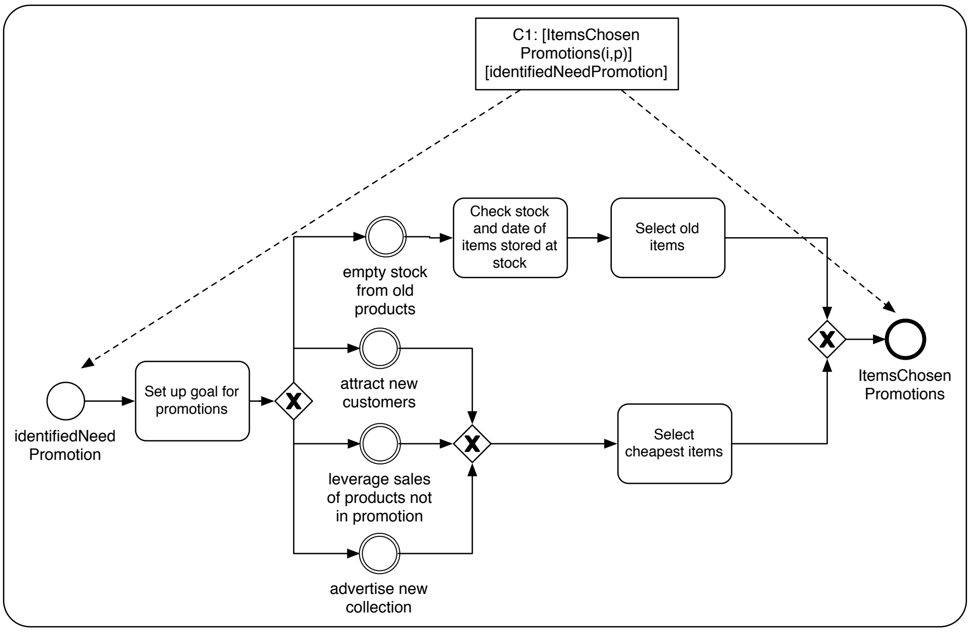


Figure 7.12: Operationalization of commitment C_1

7.5 Summary

In this chapter, we have presented Azzurra, a specification language for business processes founded on the concepts of social commitments and protocols. Besides both modeling constructs, Azzurra contains other business primitives that focus on the obligations that process participants have towards each other, including delegations, deadlines and role adoption constraints. By centering the representation on commitments, Azzurra enables the specification of correctness criteria to be achieved as commitments' consequents, rather than specific operationalizations (activities) that must be carried out. In order to unambiguously specify the language, Azzurra has been presented in terms of its syntax, semantics, graphical notation and prototypical implementation of the language.

The second contribution of this chapter concerns the derivation of business process's control-flow from operational goals. In this context, we have derived Azzurra specifications from SIENA's operational goals, inspired by KAOS's refinement patterns and operationalization approaches. Within the overall context of strategic enterprise architecture approach, operational goals are specified within the *Operational Level* from the Goal View, whereas Azzurra is used to represent the control-flow of business processes within *BPA - Level 1* and *BPA - Level 2* layers of the SIENA modeling framework.

Regarding the achievement of the requirements for strategic enterprise architectures (Chapter 2), the Azzurra language partially achieves expressiveness in behavioral perspectives (R1.2) and traceability between motivational and behavioral perspectives requirements (R2). More specifically, with the specification of business processes in terms social commitments and protocols, Azzurra achieves the needs of providing a social perspective of business processes (1.1). Furthermore, the possibility of specifying activities that operationalize commitments address the needs of providing operational perspective for business processes (1.2). The specification of the overall set of business processes executed by the company together with the relations among such processes (i.e. the BPA) has been addressed in Chapter 5.

Regarding traceability between motivational and behavioral perspectives requirement (R2), traceability in the representation has been partially achieved by the SIENA language (Chapter 5) with the interconnections between tactical goals, operations, operational goals and business processes. In this chapter, traceability in methodological consistency between motivational and behavioral perspectives is also achieved with our derivation approach of Azzurra specifications from SIENA's operational goals. Furthermore, in order to derive Azzurra behavioral specifications, goal patterns (1.4) (motivational perspective) have also be specified for operational goals as means of informing the type of behavior required for operational goals to be achieved.

In comparison with the current business process modeling approaches (Sections 4.3.1 and 4.5.3), Azzurra advances the state of the art by providing a social perspective for the representation of business processes. This social perspective is not recognized by traditional process modeling language like procedural languages (e.g., Petri-nets, BPMN, BPEL, EPCs, workflow nets), declarative (Declare) and artifact-centered approaches (case handling, object-aware processes, GSM). Therefore, as Azzurra captures business processes in terms of intentional agents and the expectations of these agents towards each other (i.e., their commitments), instead of expressing how to achieve a determined business goal through a prescription of a number of steps (activities), Azzurra specifies the constraints that have to be respected and gives the participating agents the autonomy to decide the best operationalizations to achieve the outcomes during runtime. This shift in the modeling paradigm opens up the possibility of providing more flexible specifications for business processes.

Regarding other approaches that address the representation of interactions among actors (e.g. choreographies, commitment-based, resource management, compliance management and L/A perspective approaches), although they recognize the need of leveraging business process specifications to the social level, they do not provide *commitments* as first-class citizens for process representation. In particular, besides using commitments as the first-class citizen in the language, Azzurra also provides the following novelties in relation to commitment-based approaches ([43, 207, 124, 42, 164, 139, 127]): (i) advanced primitives for expressing business patterns such as separation of duties, compensations, workload limits; also lifetime support for protocol instances, from initiation to termination; (ii) a graphical notation to visualize the main elements of a protocol.

In comparison with the motivational and behavioral approaches (e.g. [72, 1, 141, 112, 17, 117, 116, 165, 136, 69, 5, 107, 108, 185, 173, 120, 20, 6]) (Section 4.4 and 4.5.4) that generate process designs from business goals, as such approaches neither recognize goals of different shades nor provide abstractions for capturing the social perspective of business processes, our derivation approach can be considered more expressive then those approaches, by enabling the generation of strategic plans to achieve strategic goals and subsequently implementing such plans by deriving the social perspective of processes from operational goals.

Chapter 8

Evaluation of SIENA Modeling Framework

After a new modeling framework is created, its artifacts (e.g. modeling languages and reasoning techniques) need to be evaluated accordingly. This chapter describes the evaluation of the SIENA modeling framework which is performed in three phases. In particular, the first phase evaluates the SIENA framework against the requirements of Chapter 2 in order to check whether the languages and automated reasoning technique meet all the requirements stipulated for strategic enterprise architectures (Section 8.1). The second phase evaluates SIENA expressiveness and suitability for modeling a real use case and its real-world applicability. In particular, we use the hospital scenario of our previous effort [24](Section 8.2). Finally, the third phase compares the SIENA modeling framework with the ArchiMate framework in order to check whether SIENA advances the state of art in the representation and analysis of strategic enterprise architectures (Section 8.3). In each evaluation stage, we first describe how it has been conducted and the conclusions acquired from each stage.

8.1 Achievement of Requirements for Strategic Enterprise Architectures

The first phase intends to evaluate whether the SIENA modeling framework achieves the requirements for strategic enterprise architectures stated in Chapter 2 using the example from the metal manufacturing company used throughout this thesis. Although such evaluation has been partially carried out in each chapter of this thesis (Chapters 5, 6 and 7), here we want to comprehensively evaluate the overall framework. Subsequently, this comprehensive evaluation allows us to draw some considerations about the process of creating the SIENA language with the conceptualization provided by Management literature, also reflecting about some of our modeling decisions.

In order to remind the reader about the requirements for strategic enterprise architectures and facilitate our evaluation, Table 8.1 summarizes the evaluation of the SIENA Modeling Framework with respect to the achievement of requirements for strategic enterprise architectures from Chapter 2. In this table, each row depicts the (sub)-requirements from Chapter 2, while each column depicts whether SIENA meets or not the requirement under consideration and the chapter where the (sub)-requirement is met. Regarding the conventions for the assessment, a ✓sign indicates that SIENA fully addresses the requirement, whereas a − sign indicates that SIENA does not address the requirement.

8.1.1 Results and Discussion Regarding Achievement of Requirements for Strategic Enterprise Architectures

Figure 8.1 depicts the comprehensive example of the metal manufacturing company modeled using the SIENA modeling framework. The figure points out in the example how (sub)-requirements R1 (expressiveness in

Requirements for Strategic Enterprise Architectures (Chapter 2)			
Evaluation of SIENA Modeling Framework			
			Where?
Expressiveness mot. perspective	Description (1.1)	✓	Chapter 5
	Ownership (1.2)	✓	
	Time Frame (1.3)	✓	
	Goal Pattern (1.4)	✓	Chapter 7
	Targets (1.5)	✓	Chapter 5
	Multiple Levels of Abstraction (1.6)	✓	
	Goal Relations (2)	✓	
	Factors that Impact Goal Achievement (3)	✓	
Expressiveness beh. perspective	Social Perspective (1.1)	✓	Chapter 7
	Operational Perspective (1.2)	✓	
	Business Process Architecture (BPA) (1.3)	✓	Chapter 5
Traceability mot/beh persp.	Representation (2.1)	✓	Chapter 5
	Methodological Consistency (2.2)	✓	Chapter 7
Support for Automated Reasoning	Formal rigor in specifications (R3.1)	✓	Chapter 6
	Support for Execution of Planning Process (R3.2)	✓	
	Reason with Different Shades of Goals (R3.2.1)	✓	
	Reason with Environmental Factors (R3.2.2)	✓	
	Support Selection of Best Strategic Alternatives (R3.2.3)	✓	
	Support for Control of Implemented of Strategic Alternatives (R3.2.4)	–	–

Table 8.1: Summary of Assessment of SIENA Modeling Framework With Respect to Achievement of Requirements for Strategic Enterprise Architectures (Chapter 2)

motivational and behavioral perspectives) and R2.1 (support for representation in the traceability between perspectives) from Chapter 2 are met in terms of the modeling languages of the framework. As requirements R2.2 (support for methodological traceability between motivational and behavioral perspectives) and R3 (support for automated reasoning) concern (respectively) methodology and reasoning with SIENA models, they are not graphically represented, but just discussed in the next sub-section.

In the remainder of this section, we provide a discussion about the achievement of requirements of strategic enterprise architectures within the SIENA modeling framework.

Expressiveness in motivational perspective (R1.1). Regarding expressiveness in motivational perspective, as can be seen in Figure 8.1 and Table 8.1, the SIENA language (Chapter 5) addresses all the motivational requirements. As argued in Section 5.3, SIENA's goals are labeled descriptions (1.1) segmented into multiple levels of abstraction (strategic, tactical and operational levels) (1.6). Each goal layer has a target (1.5) to be achieved and distinct time frames (1.3) in which they need to be accomplished, i.e., strategic goals are usually long-term (between two and five years), whereas tactical goals have typically shorter time horizons (usually from one to three years) and operational goals consists of daily results. Each goal category from the hierarchical level is assigned to some member of the organizational structure (1.2) responsible for its achievement. Strategic goals are assigned to the overall organization, while tactical goals are assigned to functional areas and organization units. In its turn, operational goals are assigned either to a role or to multiple roles (business process operational goal). Furthermore, SIENA also captures a number of relations among goals, such as refinements (AND-relations), alternatives (OR-relations) and partial, qualitative relations (positive and negative contributions) (2) among the goals inside of all layers. Factors that impact the

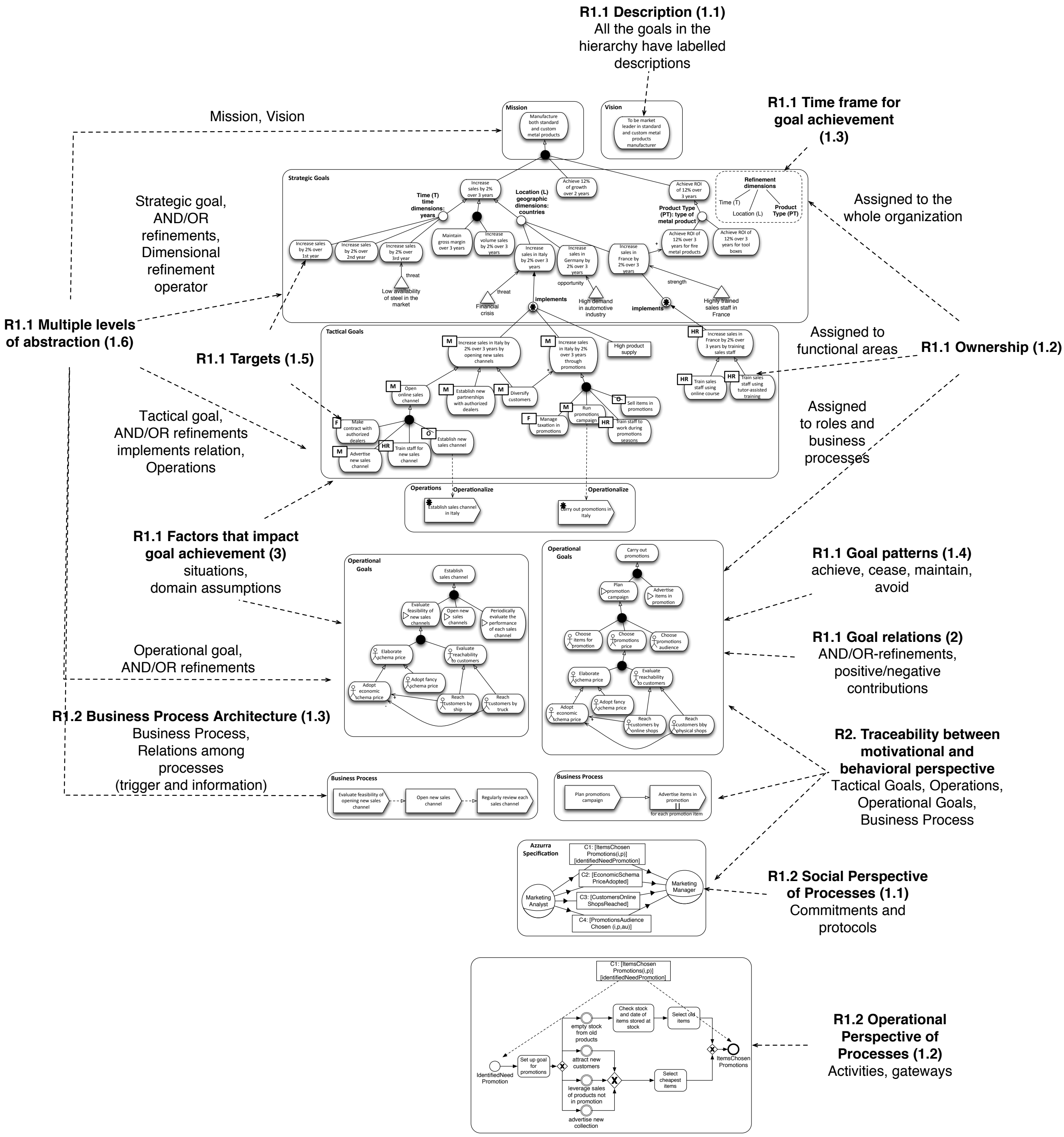


Figure 8.1: Assessment of SIENA Modeling Framework With Respect to Achievement of Requirements for Strategic Enterprise Architectures (Chapter 2)

achievement of goals are also captured as situations and domain assumptions (3). In order to specify the type of behavior required for operational goals to be achieved, goal patterns (1.4) have also be attached to operational goals in our derivation approach of Azzurra behavioral specifications from operational goals (Section 7.4).

In order to build the SIENA model, conceptualization from Management literature described in Chapter 3 (Section 3.7) has been used to refine the BIM and BMM notion of *goal*. In this context, the biggest challenge was to find precise criteria to distinguish among goals of different shades due to the lack of clear semantics of goals in Management literature. In the remainder of this section, we describe some of these challenges and some modeling decisions to cope with them.

Starting at the *Strategic Level*, Management Literature defines company's mission as the reason why the organization exists, i.e., to deliver some aggregate value to the external world by means of products/services. In this context, strategic goals represent competitive requirements to be achieved to measure the achievement of mission statements. Simultaneously, Management literature also distinguishes among corporate and business unit strategies (strategic goals), with a business unit strategy defining requirements on how business units (product/services) intend to compete and the corporate strategy defining which different business units (products/services) aggregate value to the overall company's scope. With such definitions in hand, their careful examination reveals a great similarity between the concepts of mission and corporate strategy and between strategic goals and business unit strategy.

In order to disambiguate such definitions, SIENA provides the concepts of *mission*, *strategic goals* and their *dimensional refinement operators*. A mission defines which products/services the company delivers to the external world (e.g. manufacture metal products) and strategic goals define competitive requirements for measuring the achievement of *one specific mission* (e.g. increase sales (of metal products) by 2% over 3 years, achieve 12% of growth over 2 years). For modeling multiple business units for the same company in SIENA, it suffices to represent them as multiple missions and the strategic goals relative to each mission define the competitive requirements that measure the achievement of their respective missions. To exemplify this situation, Figure 8.2 shows a hypothetic situation in which the metal company has two business units (metal products and wood products). The competitive requirements of each of them are represented as strategic goals resulting from the refinement of the missions. For instance, "Increase sales by 2% over 3 years" and "Achieve ROI of 12% over 3 years" strategic goals refers to a sales increase of metal products. The same rationale can be applied for the other strategic goals (e.g. "Achieve 12% of growth over 2 years", "Achieve ROI of 12% over 3 years" and "Improve market share in 5% over 3 years" strategic goals refers to a return of investment relative to wood products.

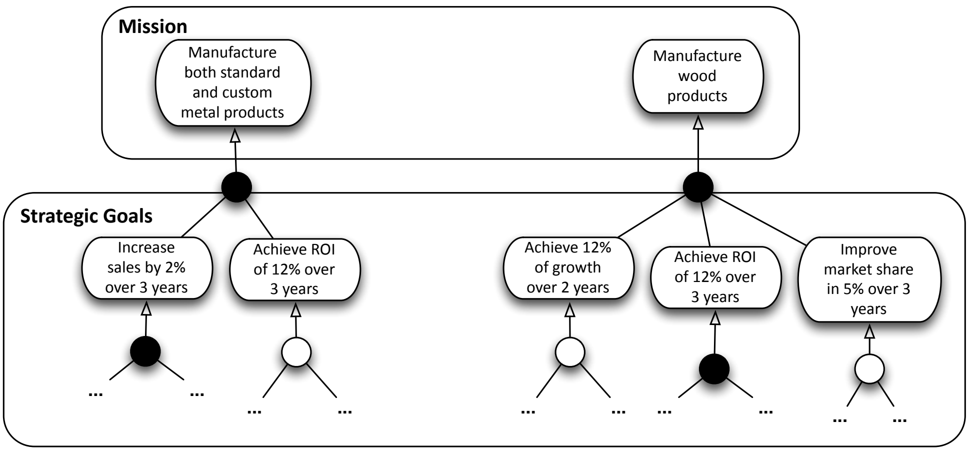


Figure 8.2: Representation of Multiple Business Units in SIENA Modeling Language

Within the *Tactical Level*, SIENA aggregates the dual perspective of Management literature that tactical goals refer to responsibilities of functional areas or tactics to achieve strategic goals. In order to conciliate both views, the definition of tactical goal encompasses both perspectives (Section 5.1.1) and the methodological guidelines of tactical goals prescribe in which order to specify them (Section 5.2.3). Initially, tactics (e.g., create promotions, open new sales channels, manufacture products) are specified by means of implements-relations for each point of a refinement dimension of a strategic goal. Subsequently, each tactic must be AND/OR refined in terms of the responsibilities of functional areas (e.g., the responsibilities of operations, finance, human resources and marketing areas must be specified in the context of promotions tactics (or open new sales channels tactics)). Finally, tactics must be differentiated in terms of established routines vs. singular initiatives (Section 5.2.3). For example, "create promotions" is classified as an initiative, since it is executed once in order to operate changes in the organizational context. In contrast, the "manufacture products" consists of an established responsibility as it needs to be repeatedly executed.

By conceptualizing tactical goals ("tactics") as established routines or singular initiatives, we bundle goals of quite different flavors into the concept of tactical goals. This modeling decision has been taken after careful analysis of Management literature.

In Management literature, tactical goals have a dual definition by either focusing on the specification of responsibilities of functional areas in the context of overall strategic goals or on the specification of tactics to achieve strategic goals. Consequently, after analyzing this literature, we faced three options. The first option consisted in simply ignoring the different notions of tactical goals presented in Management literature. In this case, it would suffice to opt for the definition we considered most adequate. The second option was to adopt the different notions of tactical goals and create two concepts for capturing them in our language, while the third option was to bundle the different notions into the concept of tactical goals (option adopted by Management literature).

In facing such situation, we have adopted the third option for three reasons. First, as we argued in Section 2.3, our main source for requirements for strategic enterprise architecture has been acquired from Management literature, with the incorporation of some insights from the previous exploratory case study in the hospital environment as well as from the literature review in Computer Science (BPM, EM, GORE and multi-agent approaches). As Management literature consisted of the main source of conceptualization in our work, we decided to incorporate as many as possible notions of this literature (including this definition of tactical goals). This decision is based on the fact that distinct bodies of literature provide definitions based on different assumptions and aggregating conceptualization from different sources into the same framework might cause semantic inconsistencies.

Second, capturing the different notions of tactical goals is important due to its implications on scheduling constraints at runtime of operations/business processes that operationalize the tactical goals. For example, a business process that operationalizes an *initiative* (e.g. "Advertise items in promotions) tends to be executed with a lower frequency in relation to a business process that operationalizes an *established routine* (e.g. "Manufacture products). These differences in terms of scheduling constraints of processes will certainly imply in the workload of the overall architecture at runtime and on the development of automated reasoning techniques that use data from process execution to evaluate whether goals are being achieved. Although this thesis does not address automated reasoning at runtime, we would like the framework to be easily extensible in future research efforts.

Third, during our second evaluation phase within the hospital environment (Section 8.2), these distinct notions of tactical goals have been also noticed. As this evaluation phase uses a real-world case study to acquire conceptualization, this confirms the existence of such distinctions in practice and for this reason, they have been included here. Observe also that we might have taken the second option (i.e., to create two distinct concepts for denoting the two types of tactical goals). However, we opted for joining the two notions of tactical goals to stick to the simplest option, since this distinction does not have any further implications in terms of automated reasoning in the current state of the framework.

Notice that, while we argue that our decision may prevent semantic inconsistencies and allow future extensions from the framework, we also recognize that such aggregation may potentially cause drawbacks in our enterprise models. In the current version of the SIENA framework, we perform strategic planning using strategic planning concepts (strategic, tactical goals, operations, dimensional refinement operators, etc.)(Chapter 6). Our automated reasoning technique generates strategic plans to achieve strategic goals. In a similar rationale, if we target future efforts for the development of an automated technique for performing tactical planning, we envision that the aggregation of goals of different flavors into tactical goals would potentially affect such technique in a negative way. This would happen because such aggregation of goals of different flavors into the same concept causes a lack of expressiveness in the language. In the latter instance, this lack of expressiveness in the language may affect the reasoning results with tactical models. Currently, we do not address tactical planning and therefore, we cannot fully estimate the consequences of such modeling decision at the moment.

In course of providing a definition for tactical goals, besides the need of aggregating two distinct views from Management literature, we had to define an AND-semantics for the implements-relations to denote that one strategic goal is AND-refined into tactics that implement it under some domain assumptions. Furthermore, as tactical goals implement strategic goals differently according to different points of a given refinement dimension (e.g. "Manufacture 900000 products at average cost of $19 in Italy" and "Manufacture 1200000 products at average cost of $21 in Germany"), SIENA opens up the possibility of setting up customized targets to be achieved by functional areas, depending on different points of time, location and products/services.

Regarding the *Operational Level*, the lack of clear semantic distinctions among tactical and operational goals led us to distinguish tactical goals as goals to be achieved by organization units and operational goals to be achieved by roles or business processes.

Regarding other motivational sub-requirements, time frames (1.3) can be specified explicitly through *dimensional refinement operators* when there is a need of refining strategic goals in terms of time. They can be also specified by following the methodological guidelines for SIENA in which each layer (strategic, tactical, operational) has its own time frame (Section 5.2). As explained in Section 5.2, strategic goals are usually long-term (lasting from two to five years), while tactical goals have shorter time horizons (usually from one to three years) and operational goals consist of daily, weekly and monthly results to be achieved.

Regarding goal patterns (1.4), their specification is instrumental for distinguishing the different types of behavior required for satisfying goals. For example, some goals need to be achieved (and then, an action has to be adopted), while others need to be avoided (then, an action has not to be adopted). In this context, goal patterns may be specified for operational goals in Chapter 7. Our strategic planning approach (Chapter 6) also allows the specification of optimization goals in order to allow the selection of optimum strategic plans to achieve strategic goals.

Expressiveness in behavioral perspective (R1.2). Regarding expressiveness in behavioral perspective, the adoption of commitments and protocols by the Azzurra language (Chapter 7) enables the specification of business processes in social terms (1.1), by focusing on the social interactions among process participants, rather than the operational activities executed by them. In its turn, the association of commitments with activities and gateways in Azzurra enables the specification of the operational steps required to fulfill commitments, thus providing an operational representation of processes (1.2). Finally, the concept of business processes and their relations provided by the SIENA language (Chapter 5) characterizes the concept of business process architecture (1.3) as the set of business processes necessary to realize company's strategy. In this context, the concept of value added chains from Management literature enabled us to characterize the set of business processes as a chain of processes that interact by means of the trigger and information relations to deliver company's products/services. Figure 8.1 and Table 8.1 depicts the achievement of all expressiveness in behavioral perspective sub-requirements.

Traceability between motivational and behavioral perspectives (R2). Regarding the traceability between motivational and behavioral perspectives (R2) requirement, traceability in the representation of the interconnection between both perspectives is achieved by connecting tactical goals with operations and operations with operational goals and business processes in the SIENA language (Chapter 5). In fact, the creation of such interconnections was challenging due to a lack of clear connections of both perspectives within Management literature. In this context, inspired by the need of providing support for the enterprise's planning process (R3), operations have been created with the purpose of planning the achievement of enterprise's tactical goals. Also, in order to provide an integrated perspective of the tactics to implement strategic goals and the business processes that indeed fulfill such strategic goals, operations are refined in terms of operational goals, business processes and their relations.

In its turn, traceability in methodological consistency between motivational and behavioral perspectives is achieved in Chapter 7 with the derivation of Azzurra specifications (process specifications) from SIENA's operational goals.

Support for automated reasoning with strategic enterprise architectures (R3). Regarding the achievement of the requirements for strategic enterprise architectures (Chapter 2), our strategic planning approach fully addresses support for automated reasoning with strategic enterprise architectures (R3) requirement, with an exception for control and evaluation of implemented strategic alternatives. We initially formalize strategic goals, their dimensional refinement operators, AND/OR refinements and implements relation, thus providing SIENA's specifications with formal rigor (R3.1). This formalization is an essential step of our approach to enable the execution of automated reasoning with SIENA models.

Our automated reasoning technique supports the execution of the several steps of the planning process (R3.2) in the following ways. First, the formalization of strategic goals and their dimensional refinement operators enables our approach to perform automated reasoning with goals in multiple levels of abstraction (strategic and tactical goals)(R3.2.1). In its turn, such automated reasoning allows the identification of strategic alternatives and generation of optimum strategic alternatives (strategic plans) that explore enterprise variability and constraints (R3.2.3). Second, the mapping of situations and domain assumptions to CGM modeling constructs enables us to perform scenario analysis which corresponds to the assessment of environment and their impacts on the achievement of strategic goals (R3.2.2). Support for control and evaluation of implemented strategic alternatives (R3.2.4) is not achieved in this thesis.

8.2 Real-World Case Study in Rheumatology Department of University Hospital

The second evaluation phase uses the real-world case study from our previous experience [24] developed in the Rheumatology Department of Cassiano de Moraes University Hospital (HUCAM Hospital) which is part of the Federal University of Espírito Santo in Vitória, Brazil. This case study has been used to evaluate the expressiveness of the SIENA framework for the representation of a real-world scenario and to demonstrate its applicability in practice.

At the time of our previous work, the Rheumatology department mainly accumulated the following functions: (i) provide educational training to form specialists in rheumatology; (ii) provide outpatient medical care and (iii) develop research to investigate the incidence of rheumatology conditions in population. It was composed of six specialists in rheumatology, two nurses and two physiotherapists, among other professionals to help to host patients. Rheumatology residents and interns temporarily join the department for educational purposes, also assisting in the daily routine. The department performed fifteen business processes, such as outpatient care, drugs infusion, among others and had an average rate of five thousand and seven hundred outpatient medical care instances per year.

For conducting the enterprise modeling approach, the project team was composed of: (i) enterprise modelers: one analyst (junior researcher), two consultants (senior researchers); and (ii) hospital clients: one doctor, one resident, one member of administrative staff, and a few patients. The junior researcher of the project is now the first author of this thesis.

8.2.1 The Modeling Process

At the project, we have focused on seven business processes related to outpatient medical care functions and one business process referring to the High-Cost Drug Assessment Commission. These eight business processes have been selected from a total of fifteen business processes that corresponded to the total number of business processes performed by the department to deliver healthcare services to the public.

We have produced eight goal models (represented in the Tropos modeling language and methodology) and their respective business processes (represented in the ARIS/EPCs methodology). An additional goal model has been developed to capture organizational issues which were not related with a specific business process (but with a set of business processes) or with other organizational concerns, such as infrastructure, policies, management, among others. Many draft models had been elaborated in several cycles (involving elicitation, analysis and modeling) before these resulting models were finalized.

All goals and process models have been fully validated by the doctor who was the head of the department (responsible for the project) and other department members. The validation with the head of the department was particularly important due to her broad knowledge of the department, the functioning, connections with the other departments and the overall hospital. For the interested reader, the annex of our previous work [24] depicts a complete goal and business process model of two business processes which have been developed in the case study (Diagnose patient's health state and Realize procedures business processes).

The usage of Tropos and ARIS/EPCs for capturing the strategic enterprise architecture entailed some limitations in this approach. Regarding the motivational perspective, the lack of goals of different shades in Tropos prevented us to distinguish among goals in our models and led us to propose a goal categorization in [27] which was captured externally to the model. Second, as the Tropos language is structured in terms of actors' perspectives and their dependencies, goals have been captured inside actor's personal goals (as examples of actors, we have doctors, receptionist, resident, patient, among others). As a consequence, organization's goals (goals that pertained to the overall department and not to a unique actor) and emergent goals (goals that arise due to interaction of multiple actors) have been also captured inside some actor's perspective and no further distinction among agent's personal goals, organization's goals and emergent goals have been made.

Regarding the interconnection between motivational and behavioral perspective, the absence of a unique language for modeling goals and processes in a single strategic enterprise architecture hindered our efforts to properly connect both perspectives. As a consequence of that, we have proposed an approach to establish a connection between the motivational domain and the enterprise architecture elements responsible for the satisfaction of goals using foundational ontologies [26]. Overall, all the aforementioned limitations also led us to capture information about the goals and their relations with processes as natural language in documents.

8.2.2 Results With SIENA Language in Hospital Case Study

In the present evaluation, we have used the Rheumatology Department models and documentation to elaborate goals using the SIENA modeling framework. SIENA models of the Rheumatology Department are depicted in Figures 8.3 - 8.9.

Given that the hospital scenario consists of an example of a real-world, public enterprise, it has some additional features that increase its complexity compared with the metal manufacturing example used throughout this thesis. In the remainder of this section, we enumerate those features for each SIENA's Layer.

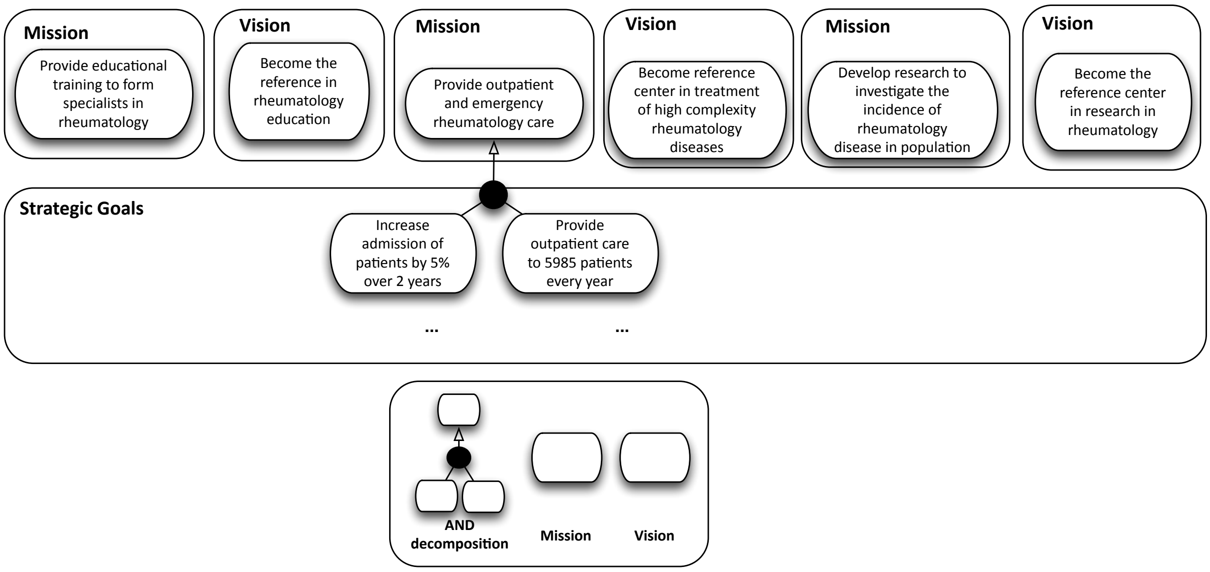


Figure 8.3: Three Mission Statements from Rheumatology Department

Strategic Layer. Contrasting with the metal manufacturing company that contains a unique mission statement, the three hospital functions (i.e., provide educational training to form specialists in rheumatology, provide outpatient care and develop research in rheumatology) can be considered three different services to the external world. For this reason, we elaborated these three services as three different mission statements with their associated vision statements (Figure 8.3). The existence of three distinct mission statements entails the need of representing strategic, tactical and operational goals for each of them. However, for the sake of simplicity, we proceed with the modeling effort solely for rheumatology outpatient services due to its highest importance for the department. Another important characteristic of the hospital case study lies on the fact that being the hospital a public enterprise, its mission and vision statements do not contain financial terms (See Section 5.2.1, regarding the discussion about the development of mission statement for public and private companies).

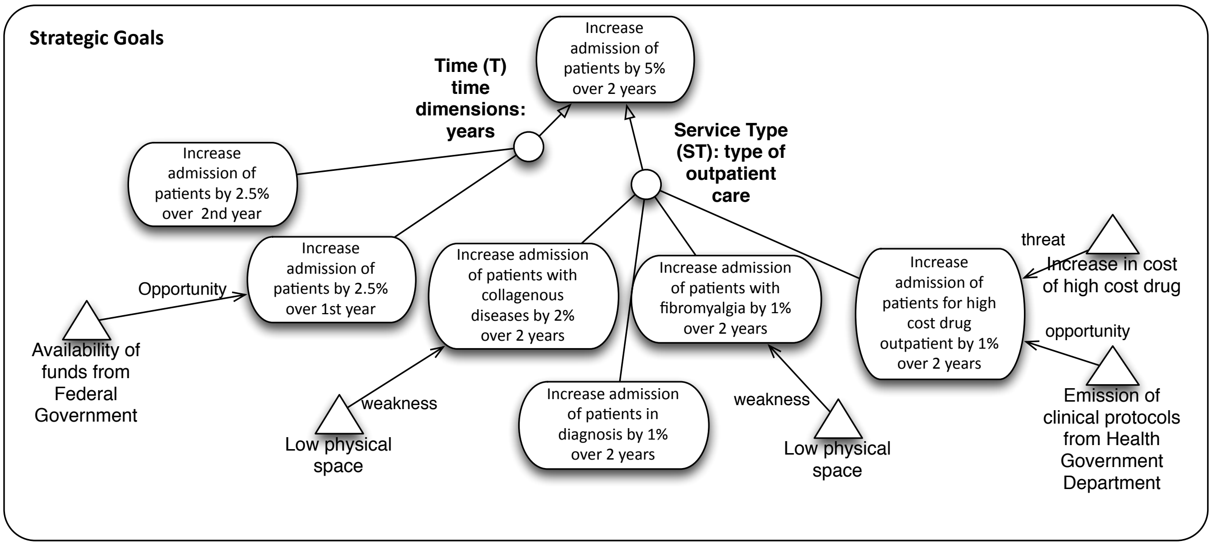


Figure 8.4: Strategic Goal "Increase admission of patients by 5% over 2 years" and Its Refinements

Regarding the elaboration of strategic goals, being the hospital a public enterprise, strategic goals also do not reflect competitive requirements, but only focus on concrete outcomes that measure the achievement of the mission by the *overall organization* (see Section 5.2.2 regarding the elaboration of strategic goals). In this context, two strategic goals regarding expected customer service levels have been elaborated for the department ("Increase admission of patients by 5% over 2 years" (Figure 8.3 and 8.4) and "Provide outpatient care to 5985 patients every year" (Figure 8.3 and 8.5)). We opted for distinguishing among two strategic goals regarding desired customer level in order to subsequently highlight the existence of *initiatives* and *established responsibilities* at the tactical level (Section 5.2.3).

Both strategic goals have been elaborated following the guidelines from Section 5.2.2. "Increase admission of patients by 5% over 2 years" consists of a need of the department to extend the provisioning of the rheumatology services to population, while "Provide outpatient care to 5985 patients every year" reflects a future target (5985 patients) to be achieved by the

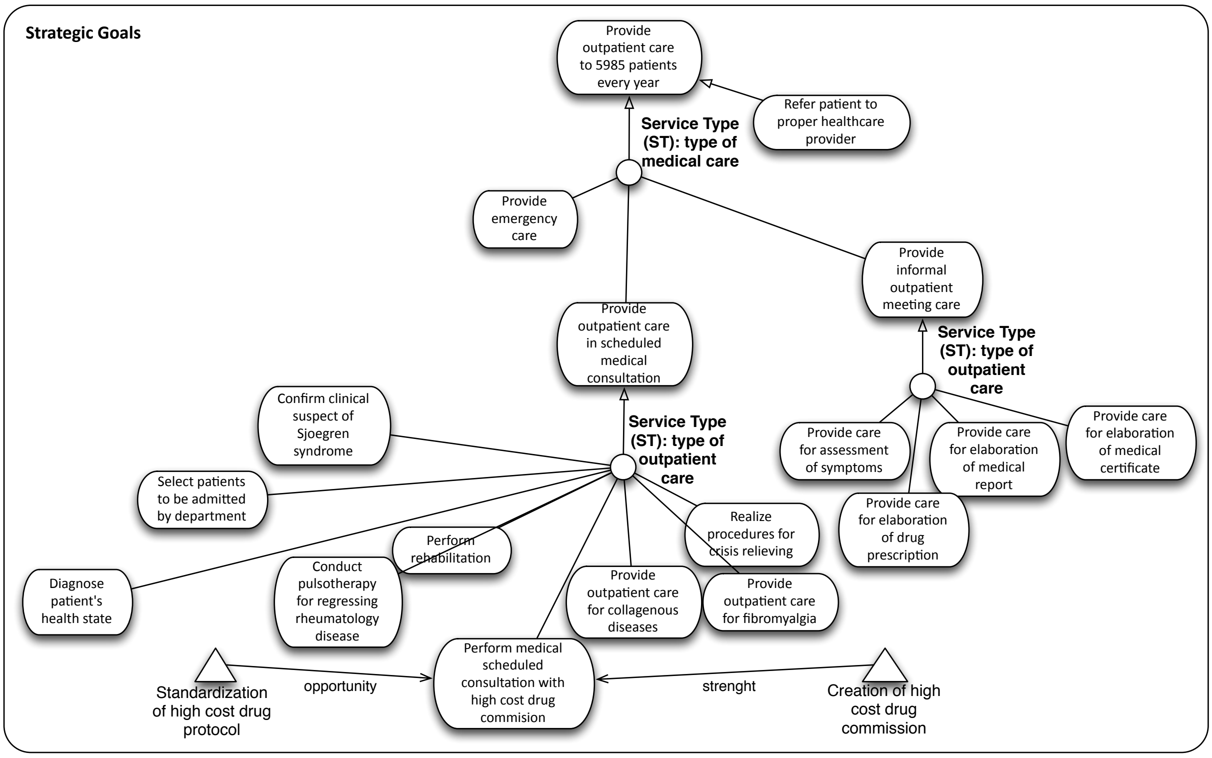


Figure 8.5: Strategic Goal “Provide outpatient care to 5985 patients every year” and Its Refinements

department, which is a result of an increase of 5% in the number of attended patients (since the current number of patients at the time of the study was 5700 patients, as highlighted before).

Both strategic goals have been refined using *dimensional refinement operators*. “Increase admission of patients by 5% over 2 years” has been refined by time (years) and by service type (type of outpatient care) (Figure 8.4), while “Provide outpatient care to 5985 patients every year” has been initially refined in terms of service type (type of medical care). Subsequently, two sub-goals from “Provide outpatient care to 5985 patients every year” have also been refined using dimensional operators in terms of service type (both in terms of the type of outpatient care)(Figure 8.5).

Our series of interviews were mainly focused on the operational (with roles like doctors, nurses, patients) and tactical levels (with the head of the department) and stakeholders at higher levels of the hospital hierarchy (e.g. public administrators or physicians of other public health services) have not been covered. This fact has limited the identification of higher-level goals of the overall system. Although we firmly believe to exist a number of other strategic concerns, they are not reflected in our models that solely focus on strategic goals we captured by means of interviews.

Tactical Layer. Within the tactical layer, two branches of tactical goals have been elaborated, one relative to the "Increase admission of patients by 5% over 2 years" strategic goal and other relative to the "Provide outpatient care to 5985 patients every year" strategic goal. In the branch of the first goal, our intention is to focus on the tactical measures that implement changes within the hospital environment (i.e. the *initiatives*), while in the second strategic goal, we intend to depict tactical measures that correspond to the *established routines* of the hospital.

Starting with the refinement of "Increase admission of patients by 5% over 2 years" (Figure 8.6), different tactical measures (*initiatives*) are adopted for implementing changes in the department that will enable the admission of more patients. In Figure 8.6, tactical measures relative to each outpatient service are delimited by dashed lines in order to facilitate the understanding. For example, in order to increase admission of patients by time ("Increase admission of patients by 2.5% over 1st year"), the department has to acquire a new radiography equipment and expand the project for education to rheumatology patients (education about how to follow the medical recommendations correctly in order to treat the rheumatology diseases). Both tactics are not represented in the scope of any outpatient service since their adoption is beneficial to all outpatient services.

In the scope of the outpatient service for treating collagenous diseases, availability of physical space ("Increase admission of patients with collage-

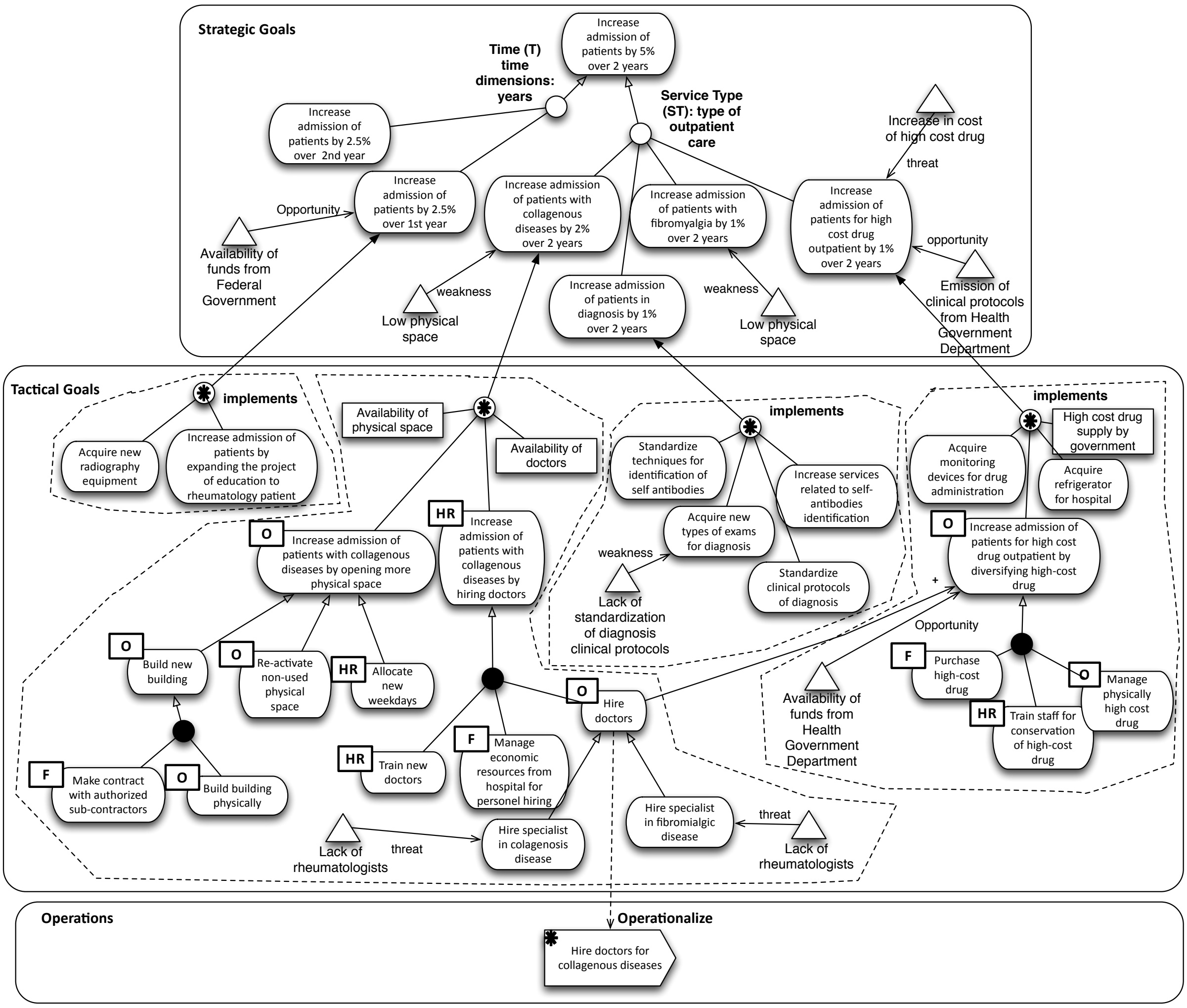


Figure 8.6: Tactical Goals Relative to the "Increase admission of patients by 5% over 2 years" Strategic Goal

nous diseases by opening more physical space") and acquisition of new rheumatologists ("Increase admission of patients with collagenous diseases by hiring doctors") are the required tactical measures.

Such tactical measures have been subsequently refined in terms of the responsibilities of the functional areas. Observe that, among the responsibilities of the functional areas, the tactics for opening more physical space ("Increase admission of patients with collagenous diseases by opening more physical space") has responsibilities for Finance ("Make contract with authorized sub-contractors") and for Operations ("Build building physically"), but no responsibilities for Marketing and Human Resources. Observe also that the responsibility from operations corresponds to a responsibility for other operational sectors of the hospital or sub-contractors (build the building), and not for the operational roles of the rheumatology department (e.g. doctors, nurses, etc.). Furthermore, we assume an availability of physical space and rheumatologists for both tactics to work (modeled as domain assumptions), otherwise, they could not be applicable.

Similar considerations regarding tactical measures can be made by other outpatient services (Figure 8.6). For example, admission of patients in the diagnosis outpatient service ("Increase admission of patients in diagnosis by 1% over 2% years" strategic goal) requires four tactical measures to be adopted: (i) standardization of techniques of identification of self-antibodies, (ii) acquisition of new types of diagnosis exams, (iii) standardization of clinical protocols of diagnosis and (iv) increase of services related to self-antibodies identification. For the high-cost drug outpatient service ("Increase admission of patients for high-cost drug outpatient by 1% over 2 years" strategic goal), the tactical measures consists of: (i) acquire devices for monitoring the injection of high-cost drug, (ii) diversify the types of high-cost drugs and (iii) acquire new refrigerator for hospital to store the high-cost drug in the department. Regarding operations, we have created just an operation for planning the hiring of doctors ("Hire doctors for collagenous diseases") for simplifying the model and the other operations still remain to be modeled.

Within the branch of "Provide outpatient care to 5985 patients every year" strategic goal, our main intention lies on the identification of *established routines* that are periodically executed for the department to provide rheumatology services to the population. These established routines correspond to the responsibilities of each functional area in the scope of the overall hospital's strategy. For that, we take two leaf strategic goals from Figure 8.5 ("Diagnose patient's health state" and "Perform medical scheduled consultation with high-cost drug commission" strategic goals) and depict them as strategic goals in Figure 8.7.

Starting from both strategic goals, Figure 8.7 depicts the tactical goals relative to each of them. For example, in the scope of diagnosis outpatient service ("Diagnose patient's health state" strategic goal), we have elaborated the responsibilities for the roles within the rheumatology department (e.g. "Manage patient's access to hospital service" (receptionist), "Diagnose rheumatology diseases" (rheumatologist)) and external departments ("Help in differential diagnosis" (ophthalmologist, dermatologist and cardiologist)), but not in the functional areas (please remember that tactical goals might refer to responsibilities of functional areas or organization's units (see Section 5.2.3 for tactical goal elaboration)). The planning of the execution of such responsibilities is then performed in the scope of the "Plan diagnosis process" operation.

Similar considerations can be made in the scope of the high-cost drug outpatient service ("Perform medical scheduled consultation with high-cost drug commission" strategic goal). In this context, core responsibilities of the high-cost outpatient service are enumerated (e.g. "Administer high-cost drug" (nurse), "Report to Anvisa reactions and adverse events during drug administration" (rheumatologist), "Manage results of laboratory exams" (receptionist)) and supporting responsibilities such as "Obtain data about patient's health state during drug administration" (receptionist), "Manage patient's access to hospital service" (receptionist) and "Manage drug for high-cost and pulsotherapy outpatient services" (nurse). Such functional responsibilities are then planned in the scope of the "Plan administration of high-cost drug" operation.

Operational Layer. The "Plan administration of high-cost drug" oper-

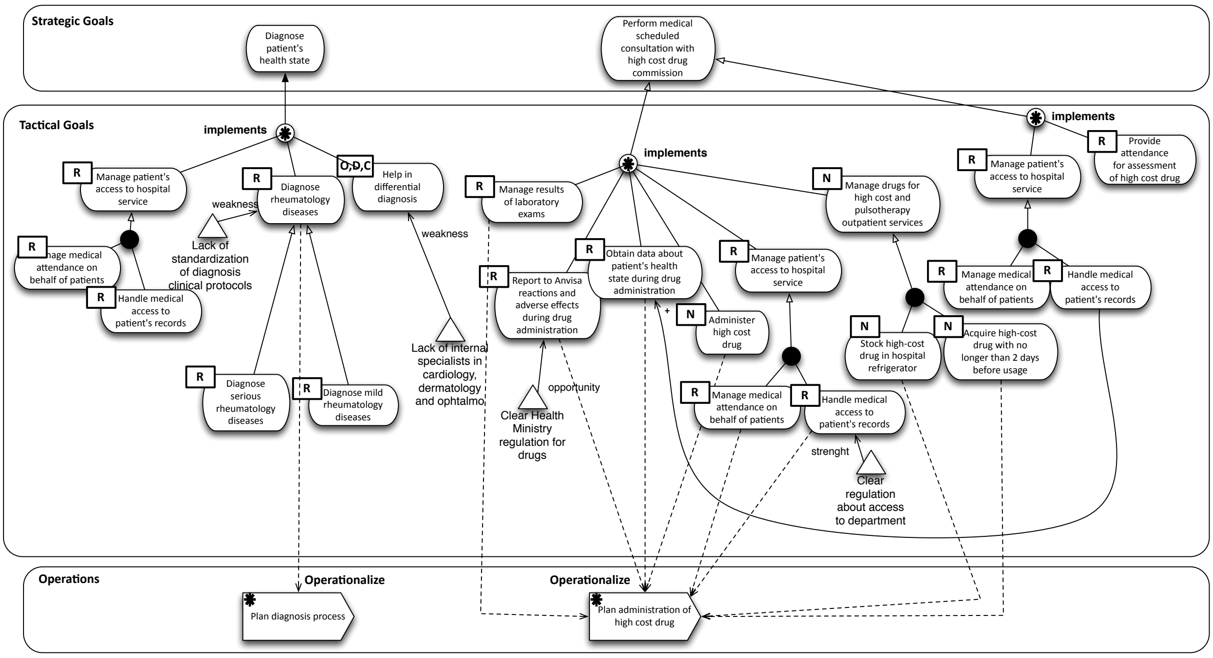


Figure 8.7: Tactical Goals Relative to the "Provide outpatient care to 5985 patients every year" Strategic Goal

ation is responsible for planning the execution of all operational steps of the high-cost drug outpatient service. This planning corresponds to the elaboration of the operational goals and business processes relative to the high-cost drug outpatient service (as explained in Section 5.2.4). Operational goals and business processes relative to the high-cost drug outpatient service are depicted in Figures 8.8 and 8.9.

The final state of the "Plan administration of high-cost drug" operation in Figure 8.7 corresponds to the root operational goal in Figure 8.8. This operational goal is then refined into three operational goals ("Schedule patients for high-cost drug administration", "Administer high-cost drug" and "Manage results of laboratory exams") which correspond to the final state of business processes (Figure 8.8). In particular, the "Schedule patients for high-cost drug administration" operational goal is refined into its sub-goals, depicting the milestones (operational goals assigned to roles) necessary to execute the "Schedule patients for high-cost drug administration" business process. In Figure 8.9, the "Administer high-cost drug" (business process) operational goal is also refined in terms of its sub-goals (role operational goals) and their contributions.

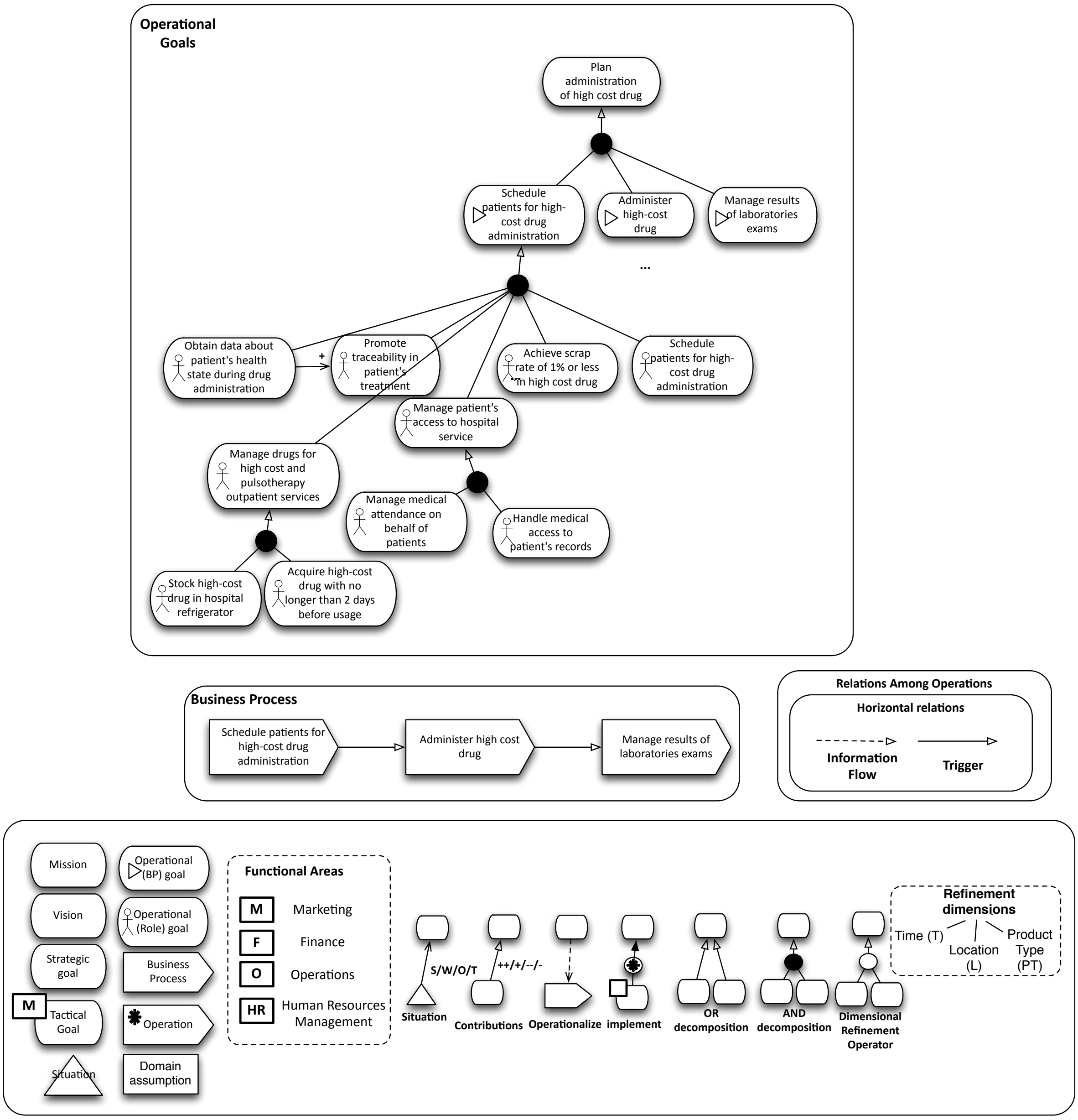


Figure 8.8: Operational Goals Relative to the "Plan administration of high-cost drug" Operation (Part 1)

Situations and Domain Assumptions. Situations and domain assumptions are captured within the three SIENA layers of goals, as for example

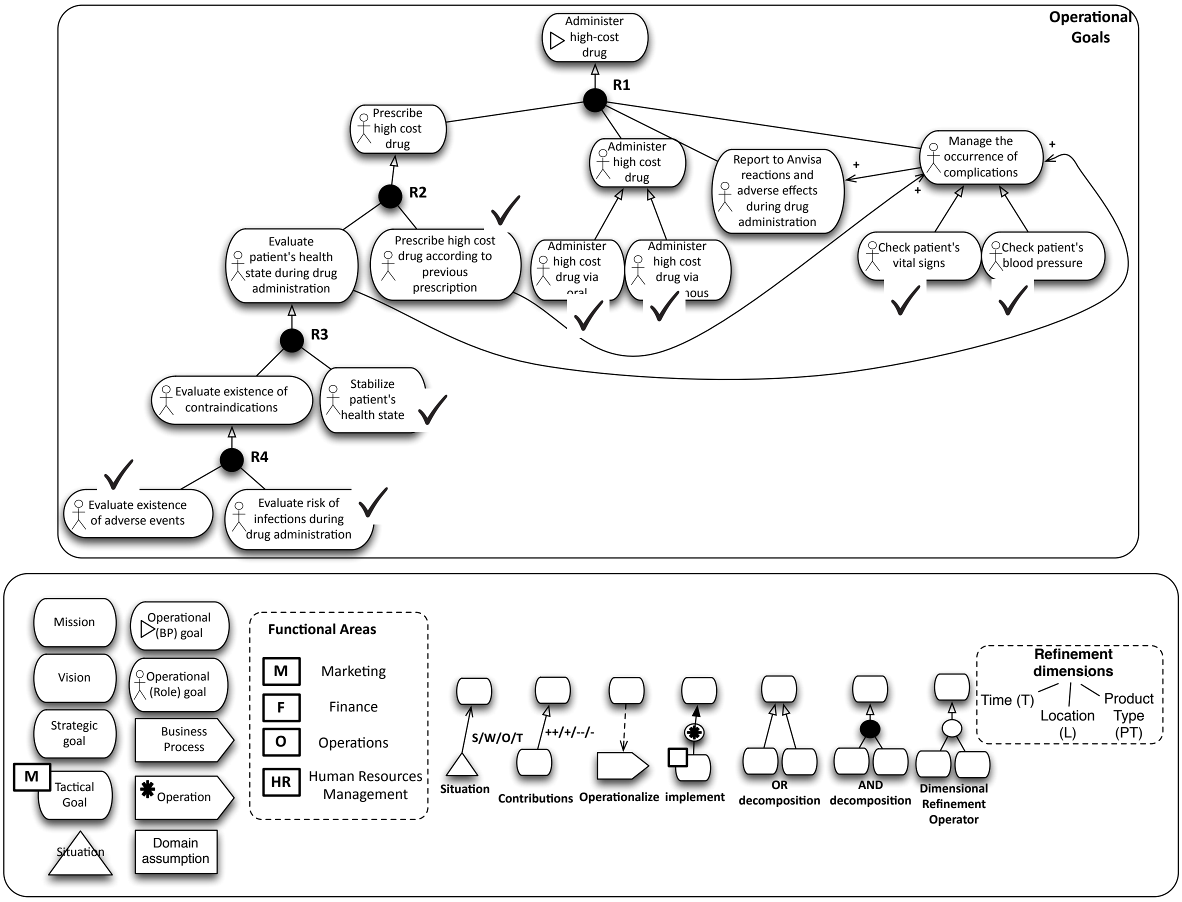


Figure 8.9: Operational Goals Relative to the "Plan administration of high-cost drug" Operation (Part 2)

in Figure 8.6 that assumes an availability of space and doctors for the achievement of the tactics of opening more physical space and hiring doctors ("Increase admission of patients with collagenous diseases by opening more physical space" and "Increase admission of patients with collagenous diseases by hiring doctors"). Regarding situations (still in the same figure), the lack of standardization of diagnosis clinical protocols in the department leads to no clear criteria for the identification of rheumatology diseases and therefore, this is a weakness in the acquisition of new types of diagnosis exams, since new diagnosis techniques can be only acquired if there is standardization in the way how they are supposed to be used. This lack of standardization also consists of a weakness in the diagnosis of new rheumatology diseases in patients (depicted as a weakness relation from the "Lack of standardization of diagnosis of clinical protocols" situation to the "Diagnose rheumatology diseases" tactical goal in Figure 8.7).

8.2.3 Results With Azzurra Language in Hospital Case Study

In order to evaluate the overall expressiveness of the SIENA modeling framework, we also built an Azzurra specification to test this language. Therefore, we have used our formal approach for business process design in order to derive an Azzurra specification from operational goals (Section 7.4). By using our business process design approach, we intended to check Azzurra expressiveness and the overall feasibility of our process design approach.

In order to apply our business process design approach, we chose to detail the "Administer high-cost drug" operational goal and business process from Figure 8.9. Therefore, Figure 8.10 depicts this goal as its root operational goal together with its sub-goals, resulted from AND/OR refinements and contributions among them. This goal structure is then used as the starting point for our business process design approach. In the remainder, we follow the steps of our business process design approach which have been proposed in Section 7.4.

1. Specify Operational Goal Models using KAOS Semantics: In order to perform this step, the root Operational Goal "Administer high-cost drug" (**Goal** *Achieve* [*HighCostDrugAdministered*]) from Figure 8.10 is formalized in terms of LTL assertions and goal patterns in Equation 8.1 as follows:

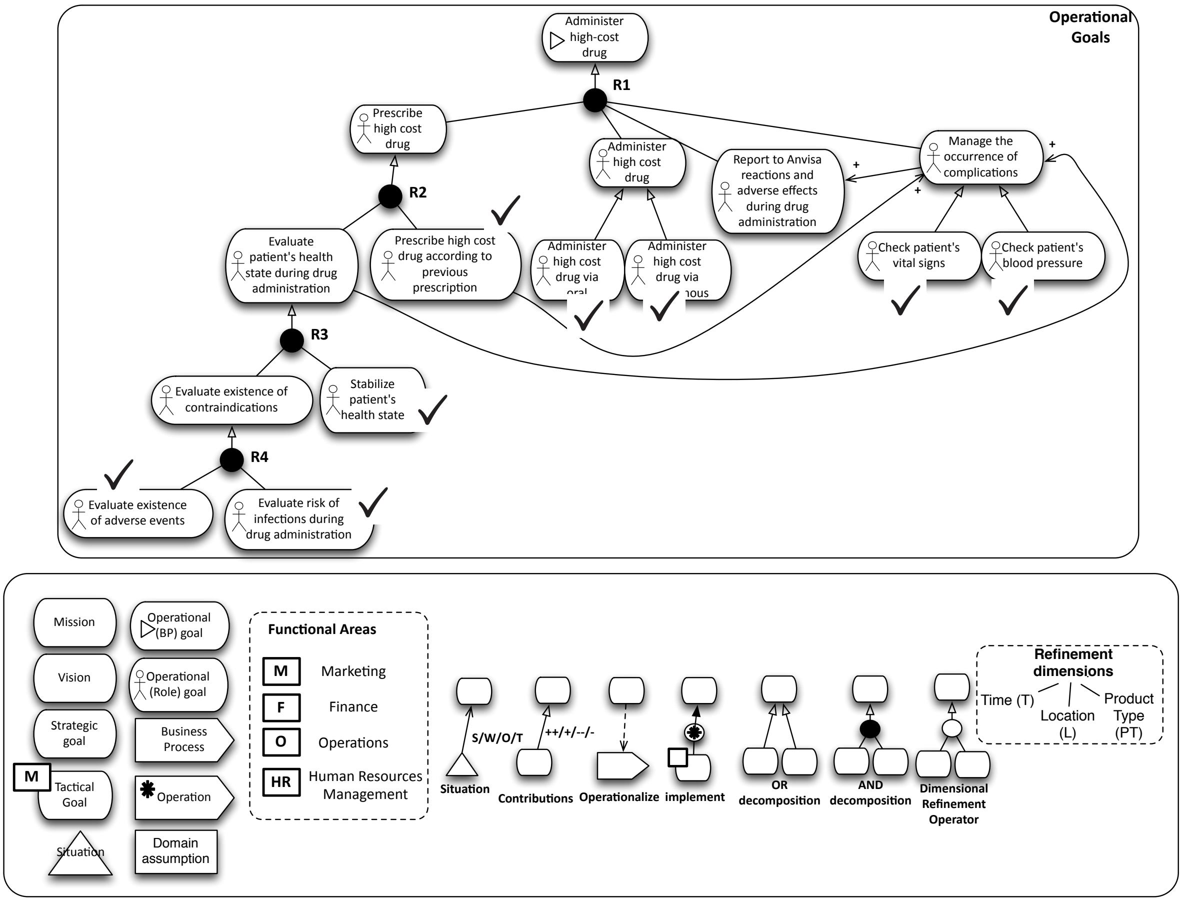


Figure 8.10: (Role) Operational Goals Relative to the "Administer high-cost drug" Business Process

$$
\begin{aligned}
&\textbf{Goal}\, Achieve[HighCostDrugAdministered] \\
&\textbf{FormalDef} \quad \exists d : Drug, p : Patient \\
&\text{IdentifiedNeedHighCostDrug (d,p)} \\
&\implies \Diamond \text{Administered (d,p)}
\end{aligned} \tag{8.1}
$$

2. Derive Roles and Agents from Operational Goal Models: From Figure 8.10, we have derived three roles (patient, nurse and rheumatologist) which are depicted in Figure 8.15.

3. Formally Decompose Operational Goals: Equation 8.1 shows the root operational goal "Administer high-cost drug" (**Goal** *Achieve*

$[HighCostDrugAdministered]$) formalized in LTL and goal patterns. In order to formally refine this goal (refinement R1 in Figure 8.10), we have to find an appropriate pattern according to the semantics of the domain. Darimont et al. [38] mention that patterns whose root goals match with the goal assertion to be refined are suitable candidates. In this case, we have chosen refinement pattern RP1 which has been originally introduced in Chapter 3 (Figure 3.5) and is also depicted in Figure 8.11.

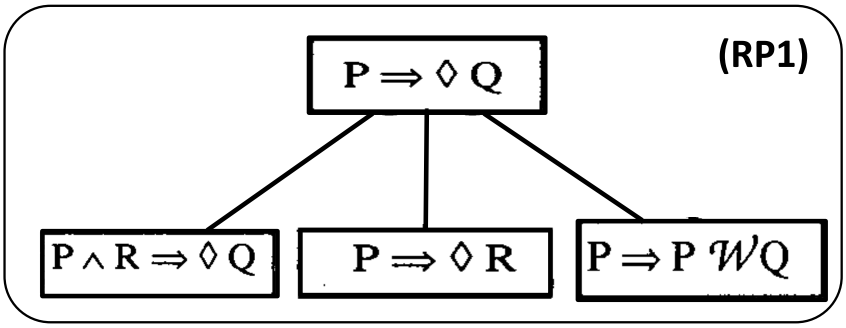


Figure 8.11: Example of KAOS Refinement Patterns Extracted From [38]

In order to apply RP1, we have to start from the "Administer high-cost drug" formal assertion (**Goal** *Achieve* $[HighCostDrugAdministered]$) and formally refine it using RP1. In this case, the RP1 requires us to start with the assertion $P \implies \Diamond Q$ (in our case, IdentifiedNeedHighCostDrug $\implies \Diamond$ Administered) and then, find the three sub-goals defined in Figure 8.11. By analyzing the refinements of the "Administer high-cost drug" operational goal in Figure 8.10 and following a suitable semantics for the refinement according to the domain, we found the following sub-goals (depicted as Equations 8.2, 8.3 and 8.4):

$$\begin{gathered}\textbf{Goal}Achieve[HighCostDrugPrescribed]\\ \textbf{FormalDef}\quad \exists d : Drug, p : Patient\\ \text{IdentifiedNeedHighCostDrug (d,p)}\\ \implies \Diamond\text{HighCostDrugPrescribed (d,p)}\end{gathered} \tag{8.2}$$

$$\textbf{Goal}\,Achieve[HighCostDrugAdministered]$$
$$\textbf{FormalDef}\quad \exists d : Drug, p : Patient$$
$$\text{IdentifiedNeedHighCostDrug (d,p)} \land \text{PrescribedHighCostDrug (d,p)}$$
$$\implies \Diamond\text{Administered (d,p)} \tag{8.3}$$

$$\textbf{Goal}\,Maintain[AdministrationOccurring]$$
$$\textbf{FormalDef}\quad \exists d : Drug, p : Patient, c : complication$$
$$\text{IdentifiedNeedHighCostDrug (d,p)} \implies$$
$$\text{IdentifiedNeedHighCostDrug (d,p)}\,W$$
$$(\text{OccurrenceComplicManaged (d,p,c)} \land \text{OccurrenceReported (d,p,c)}) \tag{8.4}$$

Equation 8.2 states that if a need of high-cost drug is identified, then the high-cost drug must be prescribed, while Equation 8.3 states that if the need of high-cost drug is identified and the drug is prescribed, then it needs to be administered in the patient. Equation 8.4 states that a need of high-cost drug is identified (and therefore the drug continues to be administered) until a complication with the drug administration occurs. In this case, if a complication occurs, it needs to be managed and reported to Anvisa. The ability of correctly applying RP1 in the root operational goal "Administer high-cost drug" confirms the correctness of the informal refinement previously performed.

Finally, due to the existence of other AND-refinements among the operational goals of Figure 8.10 (AND-refinements of the "Prescribe high-cost drug" (R2), "Evaluate patient's health state during drug administration" (R3) and "Evaluate existence of contraindications" (R4) operational goals), refinement patterns need to be applied again. In the three refinements R2, R3 and R4, we have applied RP3. Figures 8.12, 8.13 and 8.14 show the formal refinements for R2, R3 and R4, respectively.

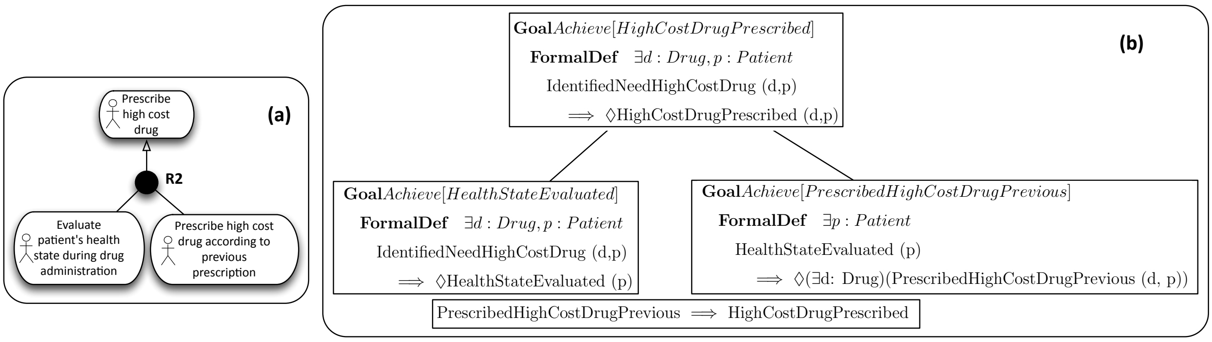


Figure 8.12: Formalization and Refinement of “Prescribe high-cost drug” Goal

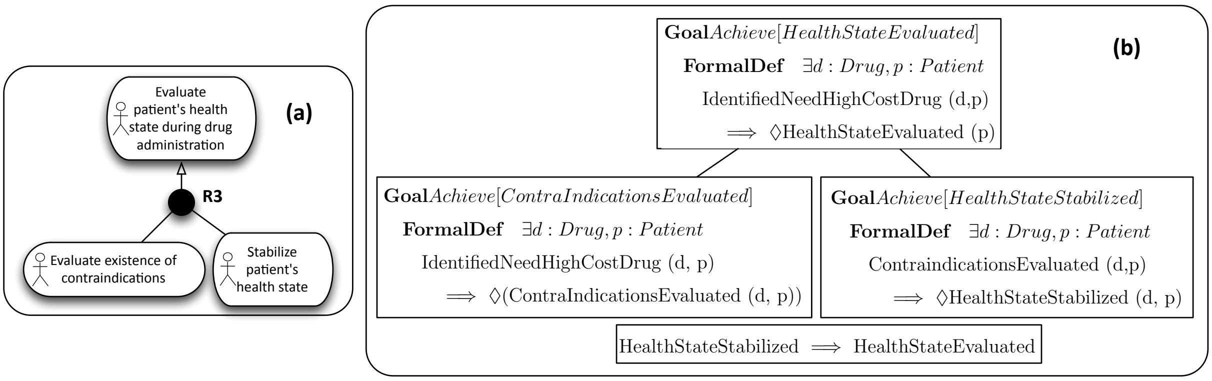


Figure 8.13: Formalization and Refinement of “Evaluate patient’s health state during drug administration” Goal

4. Select Leaf Goals for Operationalization and “Best” Alternatives of OR-refinements. After an analysis of Figure 8.10, we conclude that OR-refinements in the model do not represent alternatives that need to be chosen, but rather, alternatives that might happen. For example, during drug administration, the nurse decides whether the drug will be administered via oral or intravenous, depending on multiple factors (e.g., the patient’s health state, the type of drug, among others). In this sense, although the OR-refinement expresses an alternative, the nurse does not have choices to administer the drug in one way or another. In this

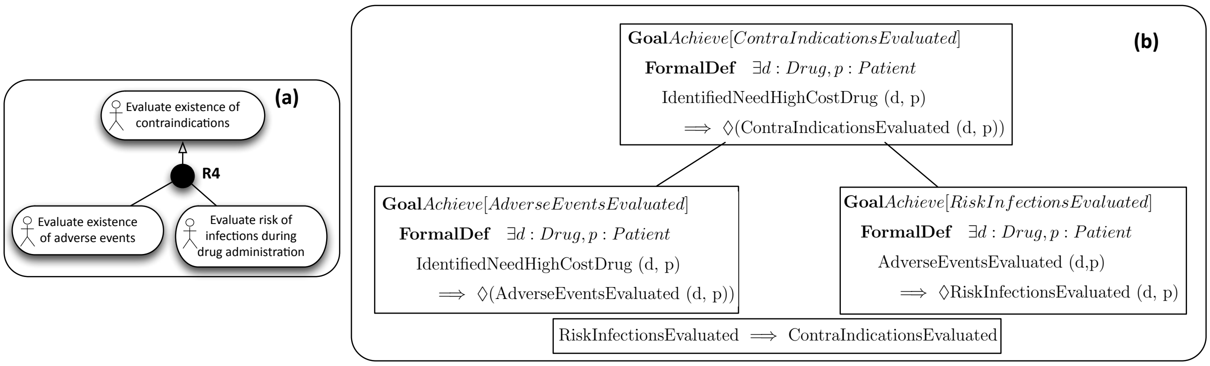


Figure 8.14: Formalization and Refinement of "Evaluate existence of contraindications" Goal

case, we leave as future work the refinement of methodological guidelines for operational goals in order to cope with similar cases. For practical purposes, we simply skip the choice of OR-refinements and use all leaf goals to proceed to step 5 (in Figure 8.10, all goals with check marks are used for subsequent operationalization in step 5).

5. Derive Commitments from Operational Goals. In order to operationalize the goals with check marks from Figure 8.10, we started with Equations 8.2, 8.3 and 8.4) and formal goals from Figures 8.12, 8.13 and 8.14 and applied operationalization patterns from Table 7.4. Operationalization pattern OP3 has been used for all goals, except for $Maintain[AdministrationOccurring]$ goal in which operationalization pattern OP7 has been applied. Figure 8.15 depicts the commitments resulted from the application of operationalization patterns.

6. Complete Azzurra Specifications. Finally, triggering events have been also specified in Figure 8.15. Although Azzurra language does not represent the triggering events in its graphical representation (*social view*), we have denoted them as dashed squares in this figure in order to illustrate our approach.

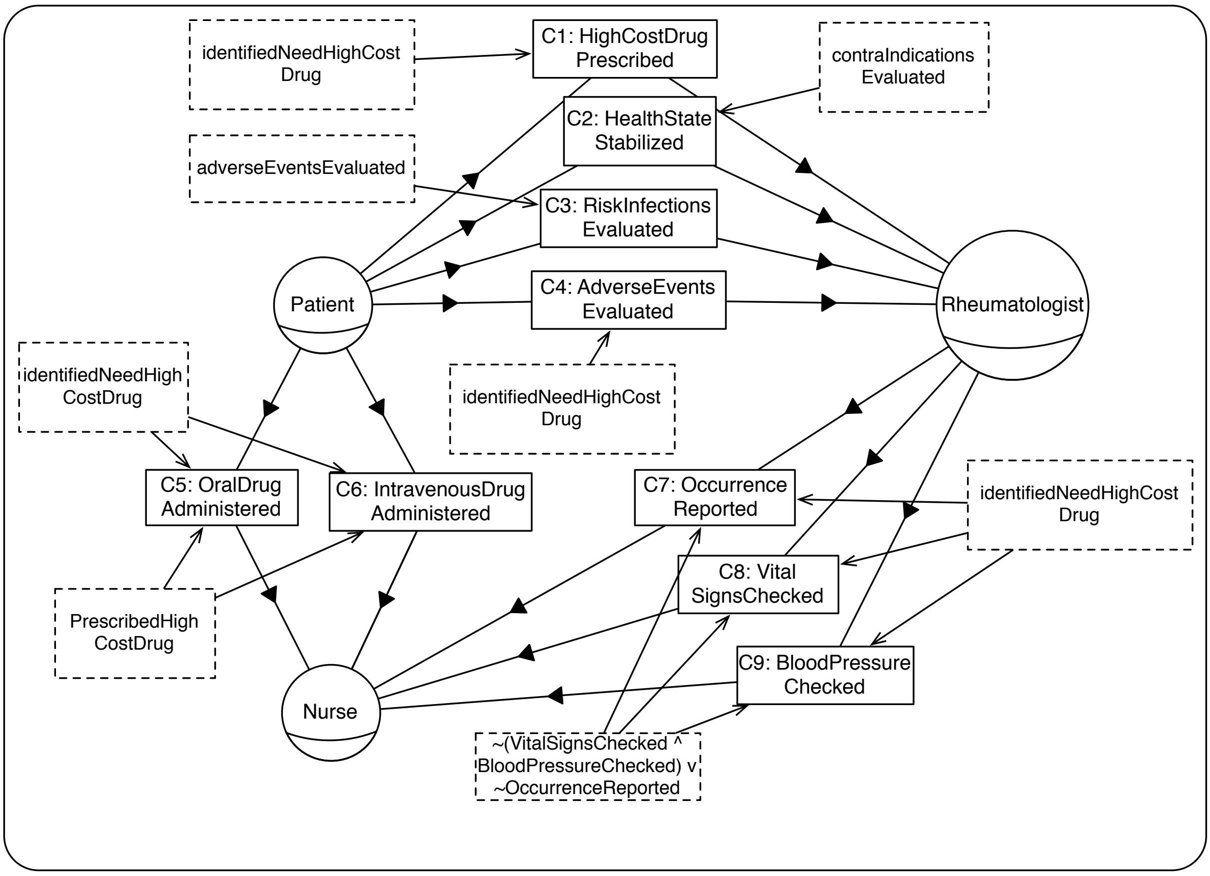


Figure 8.15: Azzurra Specification Relative to Operational Goals from Figure 8.10

8.2.4 Considerations about the Hospital Case Study

In this second evaluation phase, we have modeled the hospital scenario of our previous effort in order to check the applicability of the SIENA and Azzurra modeling languages in a real-world example. In the remainder, we make some reflections about this modeling effort and perform a comparison with our previous modeling effort using Tropos and ARIS frameworks.

The ontology of the Tropos language only supports the representation of goals and soft-goals in the scope of software engineering activities (Section 4.2.1). Consequently, the lack of support for different shades of goals in enterprise modeling (mission, vision, strategic, tactical and operational goals) leads to no hierarchical structure for capturing goals in Tropos. In order to cope with the lack of hierarchical structure, each outpatient service has been captured in terms of a goal and process model in our previous modeling effort. Besides a goal and process model for each service provided by the rheumatology department, we have also elaborated a goal model for the overall rheumatology department to capture issues common to many processes and overall organization.

However, this lack of structure and the strategy of dividing goals and process models according to outpatient service was a serious shortcoming faced in the course of our previous case study due to some reasons. First, the separation of models entails no control in terms of redundant goals, i.e., we noticed some repeated goals in goal models of different outpatient services. This fact raised a second issue that regards the correct placement of goals in goal models of different outpatient services. Third, relations among goals of different outpatient services and relations among goals of the goal model of the overall department and the outpatient services could not have been documented due to a lack of modeling construct in the Tropos language. Furthermore, inside each goal model of the study, several small goal graphs that presented no relations with other goal graphs inside the same goal model have also been captured.

In contrast, SIENA's hierarchical structure leads to an organized representation of the relations among the goals of different outpatient services and between outpatient services and the overall department. Goals from the overall department were usually captured as strategic and tactical goals, whereas goals belonging to different outpatient services were mainly captured as operational goals. This integrated goal representation thus provides criteria for goal completeness, i.e., we can achieve completeness in goal specification by simply following the SIENA's methodological guidelines and goal ontology, as all goals that exist are somehow connected along the integrated structure.

Regarding conclusions in each SIENA's layer, SIENA's guidelines were particularly useful for the elaboration of strategic goals since strategic goals have not been elicited in depth in the study. For example, the "Increase admission of patients" existed before, but have been refined in terms of quantifiable metrics. In this context, we estimated that an increase of 5 % over 2 years would be a reasonable target. The advent of *dimensional refinement operator* was also very beneficial for our modeling purposes as it enabled us to express the connection of an overall goal of the department ("Provide outpatient care to 5985 patients every year") with the corresponding different types of medical care (i.e., emergency care, scheduled medical consultation and informal meeting). In its turn, the goals of each type of medical consultation could be also derived from the "Provide outpatient care in scheduled medical consultation" strategic goal by means of the dimensional refinement operator in terms of the type of outpatient care.

Within the tactical layer, as SIENA aggregates the dual view from Management Sciences in which tactical goals are tactics that implement strategic goals and responsibilities of functional areas/organization units, the guidelines drove us to elaborate tactical goals following this rationale. Such tactical goals existed only in natural language in our previous modeling effort and therefore, they proved to be fruitful in the current modeling effort. However, with this real case study, we have discovered that not all functional areas/organization units have responsibilities in the context of some tactics. For example, in the scope of the "Increase admission of patients with collagenous diseases by opening more physical space" tactics (Figure 8.6), human resources have no responsibilities. This led us to the insight that methodological guidelines could be made more flexible, thus adapting to the reality of the company under consideration.

Furthermore, along this modeling effort, we have realized that *estab-*

lished routines (Section 5.2.3) (in tactical layer) might also refer to core responsibilities of the company (e.g. "Administer high-cost drug"), but also responsibilities that support the execution of the core ones (e.g. "Obtain data about patient's health state during drug administration")(Figure 8.7). This realization could be used in future work for distinguishing the concept of operations into *core operations* and *supporting operations.* Furthermore, the existence of operations, situations and domain assumptions in SIENA was also very useful for this modeling approach as it allowed us to represent details that were solely documented in natural language in the previous case study.

Regarding the interconnection between motivational and behavioral domains, the existence of operations in charge of planning the execution of some tactics was also a very useful concept in our modeling effort. As the Tropos language does not distinguish among operations and business processes, we have not modeled the planning of tactics in the previous study [24]. Now, we had made an extra effort to extract the information from the previous documentation. For example, the planning of the execution of the high-cost drug outpatient service (Figure 8.7) was fully elaborated in the present evaluation, but not in our previous effort.

From a methodological point of view, we had to think carefully in which goal categories each goal is inserted into and the methodological guidelines from Section 5.2 were helpful.

While this section presented some reflections about the modeling effort in a real use case using the SIENA language, it is important to highlight some aspects of such case study that were not addressed by our approach. Since the case study has been conducted in a hospital environment, the knowledge-intensive characteristic of the healthcare domain led the identification of many issues that can be considered goals as they need to be achieved. However, although their achievement is desirable, the organization has no control over their achievement. In the previous study, many of such issues have been modeled as soft-goals, as the absence of controlling mechanisms for their achievement leads to no clear criteria to determine whether the company is performing something to achieve them. For example, the "Reduce difficulties in the diagnosis process" soft-goal is an intrinsic feature of the domain that the doctor would like to achieve, but no special actions can be adopted towards that, since these difficulties are intrinsic to the domain. Further, although the doctor would like to reduce patient's suffering and symptoms ("Reduce patient's suffering and symptoms" soft-goal), there are no further actions that can be performed (besides providing the treatment) that would make this goal to be fulfilled. In the context of our approach, such issues have not been captured due to the absence of modeling constructs to capture them.

8.3 Comparative Evaluation between SIENA and ArchiMate Modeling Language

The third evaluation phase compares the SIENA modeling framework developed along this thesis with the ArchiMate modeling framework [113, 78].

The ArchiMate framework has been chosen due to its widespread applicability as a standard framework for enterprise modeling. As explained in Section 4.2.2 (Enterprise Modeling Frameworks), the core ArchiMate language has been initially extended with common GORE concepts, originating the ArchiMate Motivational Extension (AME) [160]. Subsequently, authors analyze strategic planning literature to extend AME with other finer-grained GORE concepts [8]. This second ArchiMate extension (i.e., the AME strategic planning extension [8]) is currently considered the most advanced approach for the representation of strategic enterprise architectures (please refer to Section 4.5.2 for a comparative assessment of enterprise modeling approaches regarding their support for requirements of strategic enterprise architectures).

In this context, our comparison starts with the modeling constructs from SIENA and Azzurra modeling languages and strives to match them with the concepts provided by the ArchiMate language. In the course of performing such comparison, we extracted concepts from the core ArchiMate standard [78], ArchiMate AME [160] and ArchiMate AME strategic planning extensions [8]. Such comparison has two advantages for our work. First, the ArchiMate strategic planning extension uses strategic planning literature to acquire its conceptualization, thus resembling our approach, as SIENA's modeling primitives are grounded in ontologies from Management literature. In this sense, such comparison is an opportunity to check our coverage with respect to the concepts acquired from Management literature. Second, as the AME strategic planning extension consists of the most advanced strategic enterprise architecture approach, we intend to compare both frameworks with respect to their expressiveness to determine whether SIENA advances the state of the art in the representation of strategic enterprises architectures. Figure 8.16 summarizes the steps of the evaluation process:

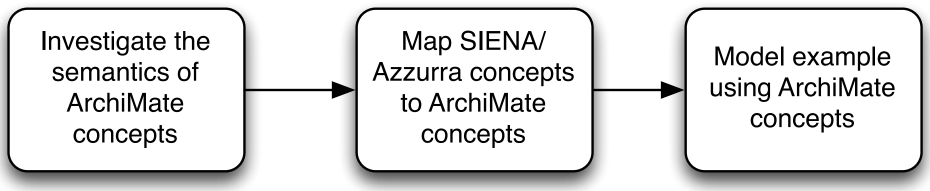


Figure 8.16: Steps of Third Evaluation Phase with ArchiMate Strategic Planning Concepts

In the remainder of this section, we detail the execution of each step of Figure 8.16.

8.3.1 Investigate Semantics of ArchiMate Concepts

Our third evaluation phase in SIENA starts with the initial investigation of the semantics of ArchiMate concepts as means of understanding them. These concepts are: *mission*, *vision*, *strategic goal*, *goal*, *planned goal*, *refinements*, *strategy* and *strategy bundles*, *target*, *time points* and *time intervals* from ArchiMate strategic planning extension [8]). From ArchiMate AME [160], we have extracted the concepts of *goals AND/OR refinements* and *positive/negative contributions* among goals. From the core ArchiMate language [113, 78], *realization relations*, *business process* and *business events* have been used. Table 8.2 depicts such concepts together with their corresponding definitions.

Concepts From ArchiMate Strategic Planning Extension [8]		
	Concepts	**Semantics**
1	Mission	Consists of a statement of organization's purpose, commonly defining in which business the organization is involved, its core beliefs about how business should be conducted, the markets and customers it serves, and the unique value to deliver to overall society
2	Vision	Consists of a description of company's future which is typically more attractive than the present
3	Strategic Goal	A strategic goal is either a mission or vision
4	Goal	Consists of an agent's intention. The goal concept is an abstract concept and does not present a concrete instantiation. Goals are refined into *strategic goals* and *planned goals*
5	Planned Goal	Consists of an agent's intention to achieve some strategic concern for the company
6	Refinements (among planned goals)	Consists of a type of refinement relation among an agent, a parent goal G_P and its refinements $\{G_1, G_2, \dots, G_N, G'_1, G'_2, \dots, G'_N\}$. In a refinement relation, the agent refines the parent goal G_P into sub-goals $\{G_1, G_2, \dots, G_N\}$. In the course of pursuing such sub-goals, the agent may decide to pursue new goals $\{G'_1, G'_2, \dots, G'_N\}$, motivated by its original intention of achieving the parent goal G_P. By achieving the newly defined goals, the agent believes his original goal G_P would be easier achievable

7	Strategy and Strategy Bundle	A strategy consists of all agent's intentions $\{G_1, G_2, ..., G_N, G'_1, G'_2, ..., G'_N\}$ to achieve one or more parent goals G_P in a *refinement relation*. Within concrete ArchiMate syntax, a strategy bundle captures a strategy
8	Target	Consists of an agent's intention that states sufficiently objective criteria to consider the intention achieved
9	Time Point	Consists of a certain point in time (date) in which a goal needs to be accomplished
10	Time Interval	Consists of a time window in which a goal should be accomplished
		Concepts from ArchiMate AME [160]
	Concepts	**Semantics**
	AND/OR Decomposition	AND decompositions allow analysts to model a goal G being decomposed into a series of sub-goals required for the achievement of goal G, while an OR decomposition allows analysts to express alternative ways of achieving a goal G
	Positive/Negative contributions	Positive/Negative contributions express the influences on the satisfaction of goals with the purpose of facilitating the evaluation of alternative goal refinements
		Concepts from Core ArchiMate Language [78]
	Concepts	**Semantics**
	Realization relation	Realization relations are used to denote that a goal is implemented by some artifact (e.g. business process)
	Business Process	"A business process represents a sequence of business behaviors that achieves a specific outcome such as a defined set of products or business services" [78]
	Business Event	"A business event is a business behavior element that denotes an organizational state change. It may originate from and be resolved inside or outside the organization" [78]
	Triggering Relationship	"The triggering relationship describes a temporal or causal relationship between elements" [78]
	Flow Relationship	"The flow relationship represents transfer from one element to another" [78]
	Contract	"A formal or informal specification of an agreement between a provider and a consumer that specifies the rights and obligations associated with a product and establishes functional and non-functional parameters for interaction" [78]

	Principle	"A principle represents a qualitative statement of intent that should be met by the architecture" [78]

Table 8.2: Summary ArchiMate Concepts Used for Comparison with SIENA/Azzurra Modeling Languages

Concepts from Table 8.2 are used for mapping to SIENA/Azzurra concepts, except the concept of "goal" which is an abstract concept and is used solely for organization purposes within the language meta-model.

8.3.2 Map SIENA/Azzurra Concepts to ArchiMate Concepts

On the basis of the semantics of ArchiMate modeling constructs discovered in the previous section (Section 8.3.1), we start with SIENA/Azzurra concepts and strive to find a corresponding concept in ArchiMate. In the remainder of this section, we describe and justify this mapping.

SIENA Mission and Vision. Starting with *mission* and *vision* concepts in SIENA, we find a straightforward mapping to respectively mission and vision concepts in ArchiMate strategic planning extension. The reason for this mapping can be justified by the definition of ArchiMate mission as a "statement of organization's purpose" that coincides with SIENA's mission definition. Similarly, vision in ArchiMate is defined as the "description of company's future" that also correspond to the same definition in SIENA.

In order to refine the mission in its respective strategic goals, unfortunately, ArchiMate strategic planning extension refrains from providing a deeper discussion about the topic. In this case, we simply map a SIENA AND decomposition to ArchiMate AME AND decomposition.

SIENA Strategic Goals, AND/OR decompositions, Dimensional Refinement Operators and Positive/Negative Contributions. For modeling strategic concerns of a given company, ArchiMate strategic planning extension recognizes the existence of *planned goals* that consists of an agent's intention to achieve some strategic concern for the company (e.g. "Increase sales"). Therefore, SIENA's strategic goals are mapped to ArchiMate *planned goals*.

Regarding the relationships among strategic goals, SIENA offers three types of refinement relations, i.e., *AND/OR* decompositions and *dimensional refinement operators*. In the context of AND/OR refinements, as ArchiMate also captures AND/OR relations among goals within the ArchiMate AME, both relations can be directly mapped. The mapping of SIENA *dimensional refinement operators*, however, presents some challenges due to the lack of similar modeling construct in ArchiMate. In this case, we can either skip the mapping of such relations or try to investigate the semantics of other types of refinements in ArchiMate. We opted for the second approach.

For the refinement of *planned goals*, the ArchiMate language offers two possibilities, i.e. either *AND decompositions* (from ArchiMate AME) or *refinement relations*[1] (from ArchiMate strategic planning extension). In an *AND decomposition*, the achievement of the sub-goals entails the achievement of the parent goal (similar to SIENA's semantics). In contrast, the achievement of the sub-goals $\{G_1, G_2, \ldots, G_N\}$ does not entail the achievement of the parent goal G_P in a *refinement relation*. In this type of relation, whenever an agent needs to achieve a parent goal G_P, it refines this parent goal into sub-goals $\{G_1, G_2, \ldots, G_N\}$. In the course of pursuing such sub-goals, the agent may decide to pursue new goals $\{G'_1, G'_2, \ldots, G'_N\}$, motivated by its original intention of achieving the parent goal G_P. By achieving the newly defined goals, the agent believes his original goal G_P

[1] Besides AND decompositions and refinements, the ArchiMate strategic planning extension also contains *aggregation* relations whose semantics states that "the enterprise believes that achieving the goals on the aggregation relation entails the achievement of the aggregated goal". With this semantics in hands, we interpret that aggregation relations have the same semantics of AND-refinements and therefore, they are not considered in our mapping effort.

would be easier achievable. The newly created goals $\{G'_1, G'_2, \dots , G'_N\}$ then present a special "bond" with the parent goal G_P and its first refinements $\{G_1, G_2, \dots , G_N\}$. With the creation of *refinement relations*, the designers of ArchiMate intended to capture the notion of *strategy* to achieve a parent goal G_P. In this context, a *strategy* denotes all the set of agent's intentions (i.e., the set of all sub-goals $\{G_1, G_2, \dots , G_N, G'_1, G'_2, \dots , G'_N\}$) created with the purpose of achieving one or more parent goals G_P. In ArchiMate's concrete syntax, the strategy concept is represented as a *strategy bundle*.

In face of the existence of *AND-decompositions* and *refinement relations*, we decided to map SIENA *dimensional refinement operators* to *refinement relations* given the similarity of the semantics of both operators. In SIENA, each *dimensional operator* intends to capture different *strategies* to achieve a given strategic goal, similarly to ArchiMate *refinement relations.* These strategies may encompass the achievement of sub-goals along time (by using the time dimension in the dimensional refinement operator), across different locations (by using the location dimension) or may follow domain specificities (by using the products/services dimension). However, although the ArchiMate refinement operator captures a notion of strategy which is slightly similar to SIENA dimensional refinements, the semantics is not exactly the same as in dimensional operators the achievement of sub-goals entails the achievement of the parent goal. Even with these slight divergences among the semantics of both operators, we opted for this mapping in order to make full usage of Archimate modeling constructs, as the AND-refinement has been already trivially mapped.

Besides refinements, the last concept to be considered within SIENA strategic layer consists of the SIENA *positive/negative contributions.* In this context, such relations are trivially mapped to positive/negative contributions in Archimate AME [160].

SIENA Tactical Goals, Implements Relation, Operationalizes Relation, Operations. Regarding the representation of concepts of SIENA *tactical layer*, ArchiMate does not consider the distinctions among goals of different shades (e.g. strategic and tactical goals), the implementation of strategic goals by tactical goals using *implement relations* and the operationalization of tactical goals by operations using *operationalize relations.*

To tackle this absence of concepts, we opted for mapping ArchiMate *planned goals* to SIENA *tactical goals.* Given that ArchiMate does not also distinguish among tactical goals, it is not possible to represent tactical goals that belong to different functional areas. Therefore, all tactical goals are simply represented as ArchiMate *planned goals.* For the refinements among tactical goals, we use the same mapping of SIENA *AND/OR refinements* within the strategic layer to *AND/OR refinements* in ArchiMate AME. For SIENA *positive/negative contributions* in the tactical layer, such relations are also trivially mapped to positive/negative contributions in Archimate AME.

Regarding the implementation of strategic goals by tactical goals using SIENA *implementation relations*, such relations are mapped either to *AND/OR refinements* or to the *realize relations* (from ArchiMate AME). *AND/OR refinements* are used when the implementation of a given strategic goal is performed by multiple tactical goals and a *realizes relation* is used when the implementation of the strategic goal is performed solely by one tactical goal.

After tactical goals have been refined accordingly, they need to be operationalized by operations. Unfortunately, ArchiMate does not recognize the linkage of goals with a concept similar to operations that appears to plan the execution of tactical goals. Therefore, we operations have no matching concept in the ArchiMate language.

SIENA Operational Goals, Business Processes and Relations.

Similar mapping applied to SIENA *tactical layer* concepts can be applied to the *operational layer*. In other words, SIENA operational goals (role and business process operational goals) are mapped to ArchiMate *planned goals*, SIENA *AND/OR refinements* are trivially mapped to ArchiMate AND/OR refinements. For SIENA *positive/negative contributions* in the operational layer, such relations are also trivially mapped to positive/negative contributions in Archimate AME.

Regarding the behavioral domain within the operational layer, SIENA *business processes* can be represented by means of ArchiMate *business processes*. Trigger and information relations among business processes in SIENA can be respectively represented by ArchiMate *triggering* and *flow relationships*. ArchiMate business processes, triggering and flow relationships are extracted from the core ArchiMate language.

Goal Ownership in SIENA. In SIENA, goals from all layers are assigned to agents within the organizational structure responsible for their achievement. Mission, vision and strategic goals are assigned to the overall organization, while tactical goals are assigned to functional areas or organizational units. In its turn, operational goals are assigned to roles (operational role goals) or a set of roles that perform a given business process (operational business process goals). In ArchiMate AME, the *stakeholder* concept is used to denote members that are concerned or interested in the enterprise architecture. Such stakeholders may be internal enterprise members (e.g., individuals, teams or the organization), but can also include external members (e.g., customers, non-organizational entities, etc.) [160].

A careful analysis of ArchiMate and SIENA semantics reveals slight divergences in terms of the assignment of goals to agents in both languages. In SIENA, the assignment of goals to agents takes into account that such agents are responsible for goal achievement, whereas in ArchiMate, a goal may belong to any stakeholder which is somehow interested in the enterprise architecture (e.g., even including external members like customers or external organizations). Hence, even with such slight divergences, goal owners in SIENA (organization, functional areas/organization units, roles) are mapped to ArchiMate *stakeholders*.

Situations, SWOT Relations and Domain Assumptions. In the core ArchiMate language, we have noticed the existence of principles that could be mapped to domain assumptions. However, although the ArchiMate AME and strategic planning extension are involved in the representation of goals of a given enterprise architecture, both of them refrain from capturing situations/SWOT relations of the enterprise environment that may affect the achievement of goals. The representation of such concepts is even more critical in the ArchiMate strategic planning extension, given that the language needs to cope with the representation of future uncertainties that naturally arise during the enterprise planning process.

Commitments, Activities, Events and Connectors. SIENA refines the internal logic of business processes in terms of process participants, their commitments and the triggering events that activate such commitments. Such design decision had the purpose of capturing the social interactions (i.e., commitments) among process participants in the execution of business process and indirectly in the achievement of operational goals. Commitments may be also optionally refined in terms of activities and gateways to depict their operationalizations.

In this context, Azzurra triggering events may be directly mapped to ArchiMate *business event*. For Azzurra commitments, we have found two candidate concepts that could potentially be mapped to (*business interaction* and *contract*). In ArchiMate, a business interaction consists of "a unit of collective business behavior performed by (a collaboration of) two or more business roles" [78]. Commitments cannot be mapped to ArchiMate interaction since a commitment is more than a mere interaction between roles in Azzurra, but rather, it also contains an involved contractual perspective. In this sense, an Azzurra commitment could be interpreted as an ArchiMate contract that consists of "a formal or informal specification of an agreement between a provider and a consumer that specifies the rights and obligations associated with a product and establishes functional and non-functional parameters for interaction" [78]. As Azzurra's commitments have a contractual nature, this is mapped to an ArchiMate contract.

Other ArchiMate strategic planning extension concepts. As we started with SIENA conceptualization and strived to find suitable modeling concepts in ArchiMate to evaluate SIENA's expressiveness, not all modeling constructs from ArchiMate have been explicitly used along with this mapping. These ArchiMate constructs are: (*target*, *time point* and *time interval*.

Although they are not explicitly mentioned along with our mapping, SIENA's conceptualization can cope with their representation. More specifically, in Section 8.1 in which we evaluate the achievement of requirements for strategic enterprise architectures, we concluded that SIENA addresses requirements 1.3 (time frame) and 1.5 (target) within the motivational perspective of strategic enterprise architectures. Therefore, as all shades of SIENA's goals need to be defined in terms of measurable criteria using targets, SIENA addresses the representation of ArchiMate *target* concept. Further, SIENA strategic goals may be refined in terms of the time dimension, thus forcing the modeler to define an ArchiMate *time point* and *time interval* for the achievement of strategic goals. Moreover, different goal layers in SIENA have implicit time frames, as discussed in Sections 5.3 and 8.1.

Table 8.3 summarizes the semantic correspondences between SIENA/Azzurra and ArchiMate concepts.

	SIENA Modeling Framework	ArchiMate Strategic Planning
1	Mission	Mission
2	Vision	Vision
3	Decomposition of Mission into Strategic Goals	AND Decomposition
4	Strategic Goal	Planned goal
5	Tactical Goal	
6	Operational Goal	
7	Dimensional Refinement Operator	Refinement(also denoted as Aggregation)
8	AND/OR Decomposition (in every Goal layer)	AND/OR Decomposition
9	Positive/Negative contributions (in every layer of Goal View)	Positive/Negative contributions
10	Implements relation	AND/OR Decomposition or Realization relation (from ArchiMate AME [160])
11	Operationalize relation	Realization relation
12	Goal Ownership (Entire organization)	Stakeholder
13	Goal Ownership (Functional areas (or organizational units))	
14	Goal Ownership (Roles)	
15	Target	Target
16	Time Frame	Time point
17	Time Frame	Time interval
18	Operations	-
19	Business Processes	Business Process
20	Relations among Business Processes (information and trigger)	Flow/Triggering relationships
21	Situations and SWOT relations	-
22	Domain Assumptions	Principles
23	Commitment	Contract
24	Activity, connector	-
24	Triggering Event	Business Event

Table 8.3: Mapping Between SIENA and ArchiMate Modeling Concepts

The mapping between SIENA to ArchiMate modeling frameworks enabled us to discover semantic overlaps between both frameworks by enabling the direct mapping of the following SIENA's concepts: mission, vision, AND/OR decompositions, positive/negative contributions, target, time frame, business processes, relations among business processes (information and trigger). However, this mapping also revealed some semantic gaps in ArchiMate, thus leading us to infer a superiority of SIENA over ArchiMate in terms of expressiveness. Below, we enumerate these ArchiMate semantic gaps, using Table 8.3:

1. ArchiMate refrains from capturing different shades of goals like SIENA. This can be corroborated by analysis of Table 8.3 in which lines 4-6 (strategic, tactical and operational goals) are all mapped to planned goals;

2. A direct consequence of the lack of expressiveness in the goal ontology of ArchiMate is reflected in a lack of expressiveness in the relations among goals. For example, implements and operationalize relations (lines 10-11) are mapped to a realization relation from the core ArchiMate language (sometimes implements relations can be also mapped to AND/OR decompositions, when multiple tactical goals implement a given strategic goal);

3. Distinct goal owners in SIENA (entire organization, functional areas or organization units and roles) (lines 12-14) are all assigned to ArchiMate stakeholders;

4. ArchiMate refrains from addressing the concepts of operations, situations (and their SWOT relations), activities and connectors;

5. Although the semantics of ArchiMate refinements does not totally coincide with the semantics of dimensional refinement operators in SIENA, and in this sense, we could consider this feature a gap in ArchiMate, we have opted for mapping both operators in order to

verify in practice the usage of Archimate refinements. Since next section (Section 8.3.3) models the metal manufacturing example using ArchiMate constructs, this mapping enables us to stress out the use of ArchiMate refinements and verify their usage in a practical example.

8.3.3 Model Metal Manufacturing Example using ArchiMate Concepts

The mapping between the concepts of SIENA and ArchiMate modeling frameworks presented in previous section (Section 8.3.2) enabled us to understand the correspondences between SIENA and Archimate conceptualizations, their overlaps and gaps. This understanding enabled us to properly select the modeling constructs from ArchiMate in order to model the example of the metal manufacturing company used throughout this thesis. With the representation of the metal manufacturing example, we intend to stress out the use of ArchiMate modeling constructs by verifying their usage in a practical example, to demonstrate the overlaps of ArchiMate with SIENA and to highlight ArchiMate gaps in the representation of motivational aspects within strategic enterprise architectures.

In order to represent the strategic planning concepts from ArchiMate [8, 160, 78], we used OmniGraffle to draw goal models following ArchiMate's visual syntax. In order to overcome the lack of some ArchiMate concepts and highlight their need, inexistent concepts are represented using SIENA's visual syntax in red. From this moment on, we present the example of metal manufacturing company modeled using ArchiMate constructs.

SIENA Mission and Vision. Figure 8.17 depicts the modeling concepts from SIENA strategic layer using ArchiMate constructs. The direct mapping of the concepts of *mission* and *vision* in both languages allowed us to directly represent the example from metal manufacturing using ArchiMate concepts and concrete syntax. Therefore, Figure 8.17 depicts the metal company's mission and vision modeled using ArchiMate's concrete notation.

SIENA Strategic Goals, AND/OR decompositions, Dimensional Refinement Operators and SIENA Positive/Negative Contributions. Figure 8.17 also depicts the metal company's mission AND-refined into the strategic goals from the company ("Increase sales by 2% over 3 years", "Achieve 12% of growth over 2 years" and "Achieve ROI of 12% over 3 years"). As can be observed in this figure, strategic goals are represented as ArchiMate *planned goals* following our mapping summarized in Table 8.3.

Regarding the relations among strategic goals, AND/OR refinements and positive/negative contributions have trivial mappings to ArchiMate (Section 8.3.2, Table 8.3). In this context, Figure 8.17 shows the AND-refinement of the "Increase sales by 2% over 3 years" strategic goal into "Increase volume sales by 2% over 3 years" and "Maintain gross margin over 3 years" using an ArchiMate AND-refinement. This figure also depicts a positive contribution from the "Increase sales in France by 2% over 3 years" strategic goal towards the "Achieve ROI of 12% over 3 years for fire metal products" strategic goal using ArchiMate positive contributions.

The absence of a direct construct to represent SIENA *dimensional refinement operators* led us to adopt *refinement relations*, *strategy* and *strategy bundles* from ArchiMate strategic planning extension (as described in Section 8.3.2, Table 8.3) to model dimensional refinements in SIENA. Therefore, Figure 8.17 depicts the parent goal "Increase sales by 2% over 3 years" refined by two *refinement relations* and *strategy bundles*, each of them referring to a dimensional refinement operator in the example of the metal company (Figure 5.5). In order to highlight the absence of SIENA *dimensional refinements operators*, time and location dimensions belonging to each dimensional refinement in the ArchiMate refinements are highlighted in red in the graphical model, just to depict that each ArchiMate refinement belongs to a dimensional operator in SIENA.

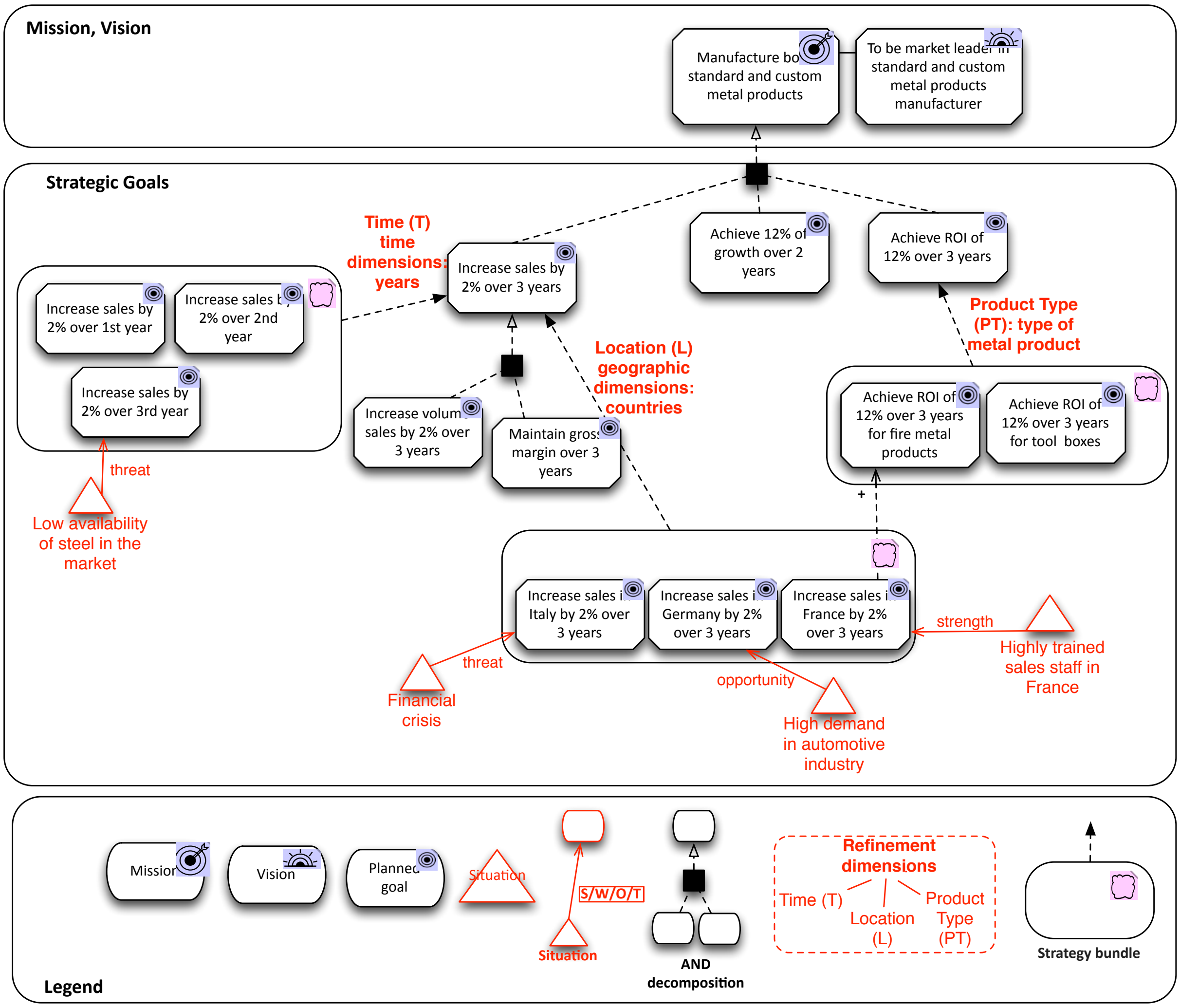


Figure 8.17: SIENA Strategic Layer Concepts Modeled in ArchiMate (concepts and relations in red belong to SIENA, but do not exist in ArchiMate)

SIENA Tactical Goals, Implements Relation, Operationalizes Relation, Operations. Figure 8.18 depicts the strategic and tactical goals from the metal manufacturing company modeled using ArchiMate *planned goals*. In this context, in order to "Increase sales in Italy by 2% over 3 years" (strategic goal), the company decided to open new sales channels and adopt promotions. To model this implementation relation in Archi-

Mate, we used an AND-refinement to denote that the achievement of both tactical goals and the domain assumption implements the strategic goal ("Increase sales in Italy by 2% over 3 years by opening new sales channels" and "Increase sales in Italy by 2% over 3 years through promotions", under the assumption that there will be a "High product supply"). The AND-refinement has been used in this case to denote that the strategic goal may be implemented by a conjunction of tactical goals and domain assumptions. In contrast, for the "Increase sales in France by 2% over 3 years" strategic goal, we have used a realization relation to denote that this strategic goal is implemented by training sales staff ("Increase sales in France by 2% over 3 years by training sales staff" tactical goal) implements the strategic goals. In this case, the realization relation has been used as just one tactical goal implements the strategic goal.

Once strategic goals are implemented by tactical goals, such tactical goals need to be refined accordingly. Figure 8.18 also depicts tactical goals refined in terms of ArchiMate AND/OR refinements. For example, the "Increase sales in Italy by 2% over 3 years by opening new sales channels" tactical goal is OR-refined in "Open new sales channels" or "Establish new partnerships with authorized dealers" or "Diversify customers". In its turn, the "Increase sales in Italy by 2% over 3 years through promotions" tactical goal is AND-refined in terms of the responsibilities of its functional areas ("Manage taxation in promotions" (finance), "Run promotions campaign" (marketing), "Train staff to work during promotions seasons" (human resources) and "Sell items in promotions" (operations)). In this context, as it is not possible to represent tactical goals associated with different functional areas, we have used SIENA notational construct in red to denote its absence (see **Goal Ownership in SIENA** discussion for a detailed discussion about the topic).

After tactical goals have been refined accordingly, they need to be operationalized by operations. The absence of a matching concept for operations led us to use SIENA's operations in red in Figure 8.18. The operationalizes relations from SIENA have been represented as ArchiMate realization relations. This rationale highlights the need for finding a construct in ArchiMate to operationalize tactical goals with a behavioral element that plans the execution of tactics (in line with SIENA's conceptualization).

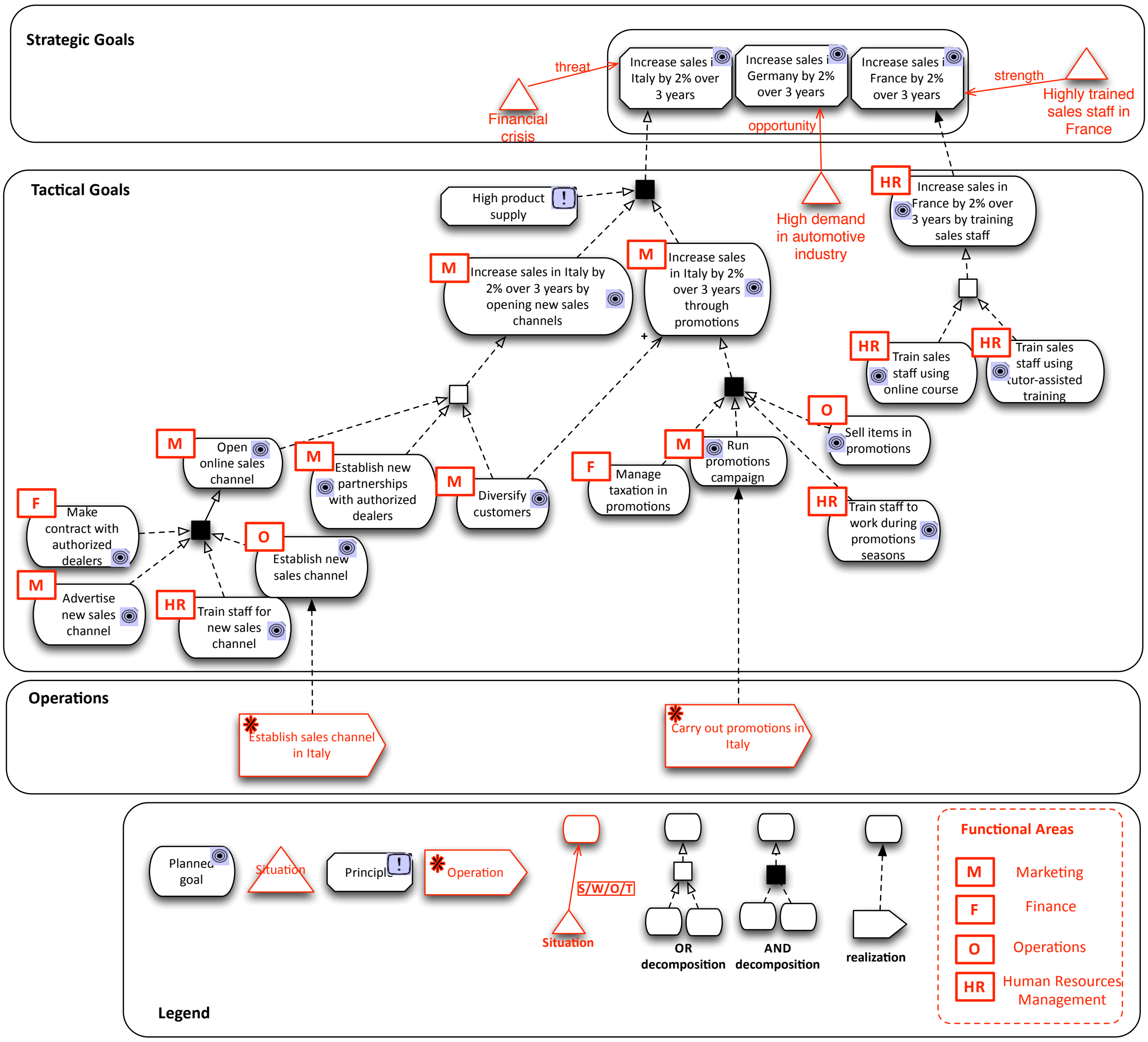


Figure 8.18: SIENA Tactical Layer Concepts Modeled in ArchiMate (concepts and relations in red belong to SIENA, but do not exist in ArchiMate)

SIENA Operational Goals, Business Processes and Relations.

Figure 8.19 depicts the metal company's operational goals using ArchiMate *planned goals*. In this context, such ArchiMate planned goals are refined by AND/OR ArchiMate AME relations as described in our mapping from Section 8.3.2.

In SIENA, Section 5.2.4 describes the elaboration and refinement of operational goals. In this context, the root Operational Goal "Carry out promotions" corresponds to the final state of the "Carry out promotions in Italy" operation (Figure 5.12). Figure 8.19 follows the same rationale in ArchiMate, in which the root operational goal "Carry out promotions" (modeled as ArchiMate planned goal) corresponds to the final state of the "Carry out promotions in Italy" operation. This root operational goal ("Carry out promotions") is then AND/OR-refined into operational goals using AND/OR ArchiMate constructs. For the representation of business processes and their relations from SIENA, we use the concepts of *business processes*, *triggering* and *flow relationships* from ArchiMate core language (as described in Section 8.3.2). Figure 8.19 also depicts such mapping using ArchiMate modeling constructs.

Goal Ownership in SIENA. In Section 8.3.2, we have mapped the different agents responsible for the achievement of goals in SIENA (e.g. the overall organization, organizational units and roles) to the *stakeholder* concept in ArchiMate. This mapping can be accounted by the fact that Archimate does not perform a distinction among the different owners for achieving goals. Although the ArchiMate language makes available a notational construct for capturing stakeholders, we have not included them in our graphical models as it would be required a number of additional stakeholder elements linked to every goal in each SIENA layer. This decision would have clearly increased the graphical complexity of our models. Instead, we have opted for sticking to SIENA notational constructs for denoting the agents assigned to goals. Hence, SIENA notational constructs

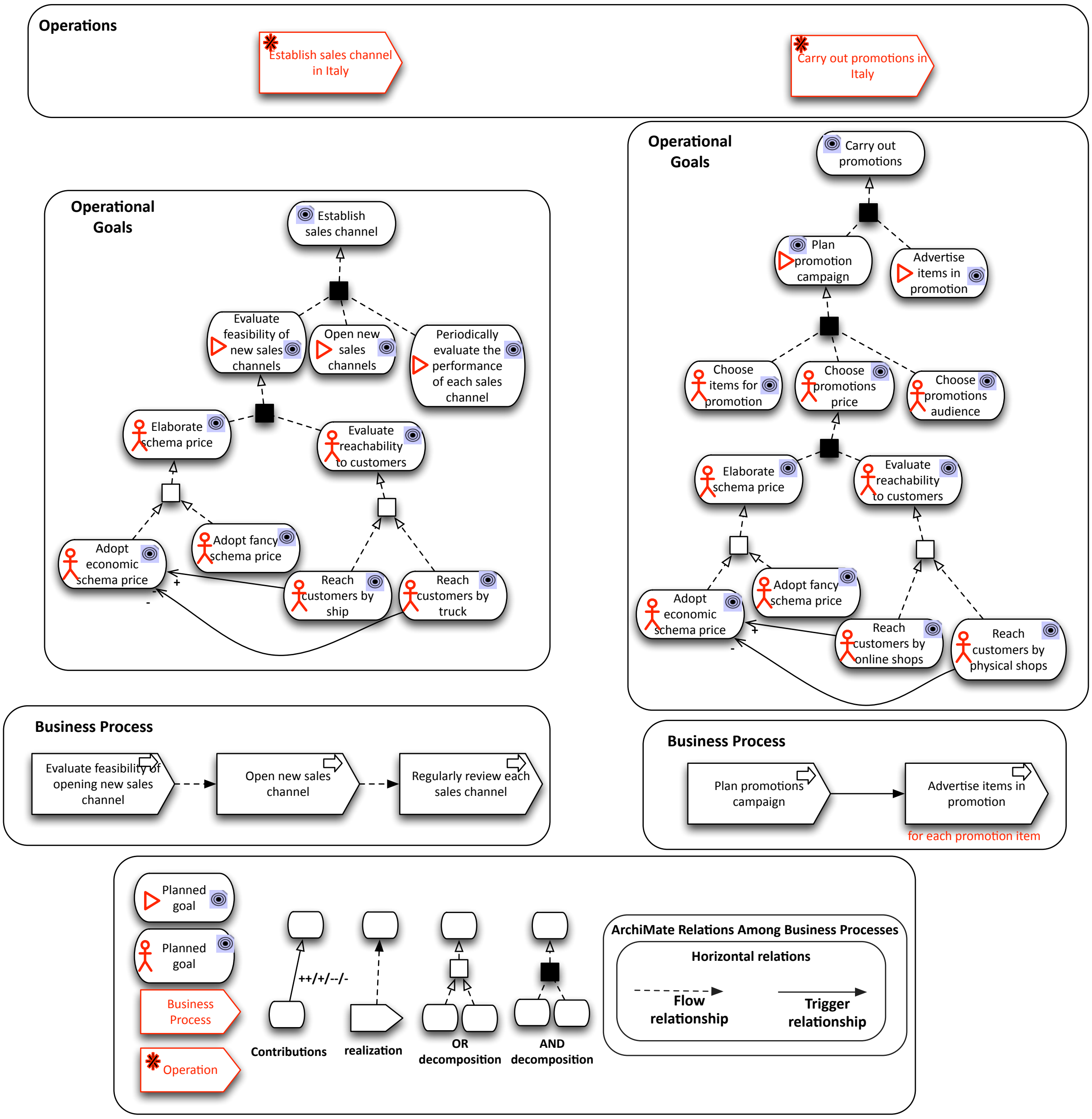


Figure 8.19: SIENA Operational Layer Concepts Modeled in ArchiMate (concepts and relations in red belong to SIENA, but do not exist in ArchiMate)

regarding the owners responsible for achieving goals are depicted in Figures 8.17, 8.18 and 8.19 in red to denote that they do not belong to the ArchiMate language.

Situations, SWOT Relations and Domain Assumptions. The absence of concepts that capture situations and SWOT relations in ArchiMate led us to represent those concepts using the same SIENA notational constructs highlighted in red in Figures 8.17, 8.18 and 8.19 to denote their lack in ArchiMate.

Commitments, Activities, Events and Connectors. The absence of modeling constructs for capturing business processes' control-flow (activities and connectors) in ArchiMate led us to omit the corresponding business process specification represented in ArchiMate. Although Azzurra commitments are mapped to ArchiMate contracts and Azzurra triggering events are mapped to ArchiMate business event, this mapping is straightforward and therefore, there are no practical differences of the diagrams built using the Azzurra language (and thus representing Azzurra triggering events and commitments) or using ArchiMate business events and contracts. Therefore, this ArchiMate model is omitted here.

8.3.4 Discussion About SIENA and ArchiMate Comparison

In this third evaluation phase, we have first studied the semantics of ArchiMate concepts with the purpose of understanding them to enable a subsequent mapping to SIENA concepts (Section 8.3.1). Such mapping intended to compare SIENA and ArchiMate conceptualizations with respect to SIENA's coverage of Management Literature and expressiveness (Section 8.3.2). The mapping has been illustrated with the representation of the metal manufacturing example using ArchiMate constructs (Section 8.3.3).

The mapping between SIENA and ArchiMate concepts described in Section 8.3.2 enabled us to find some overlaps between SIENA and ArchiMate frameworks. Such overlapping concepts are: mission, vision, AND/OR decompositions, positive/negative contributions, target, time frame, business processes, relations among business processes (information and trigger). As a result of this direct mapping, we could use ArchiMate's concrete syntax to perform the representation of such concepts in Section 8.3.3.

Comparison Between SIENA and ArchiMate Modeling Primitives. Besides identifying overlapping concepts between SIENA and ArchiMate frameworks, the mapping effort also allowed us to fully find corresponding concepts for ArchiMate in SIENA. However, not all SIENA concepts had a matching concept in ArchiMate. In order to tackle the absence of concepts in ArchiMate, two workarounds have been adopted. Within the first case, SIENA concepts have been mapped to the same Archimate concept, when it existed some concept with similar semantics. Alternatively, concepts have been borrowed from SIENA and represented using SIENA concrete syntax in red color, when no concepts with slightly similar semantics could be found in ArchiMate.

Within the first workaround, the absence of goal of different shades (strategic, tactical and operational) led us to represent all goal categories as ArchiMate *planned goals*. The same workaround has been adopted for SIENA implement relations that have been mapped to ArchiMate AND/OR refinements and realizations and SIENA operationalize relations that have been mapped to ArchiMate realizations. For the representation of SIENA dimensional refinement operators, their absence led us to map them to ArchiMate refinement relations and strategy bundles.

In this context, a direct conclusion acquired in the mapping of SIENA dimensional refinement operators to ArchiMate refinement relations and strategy bundles is the lack of ability of natively performing different types of decompositions based on time, location and product/service properties in ArchiMate. We have added refinement dimensions to the refinement relations in our model, but they do not natively belong to the language (i.e., they are just labels in the model). Consequently, it is not possible to reason with dimensional refinement operators, like in our strategic planning approach described in Chapter 6. The second conclusion refers to insights acquired with the practical usage of refinement relations. The semantics of refinement relations in ArchiMate states that a given parent goal can be refined using refinement relations to capture different ways (strategies) to achieve this goal. The achievement of the sub-goals stemmed from such refinement is not a sufficient condition to achieve the parent goal and the agent may create sub-goals at runtime to facilitate the achievement of the parent goal. However, our practical experience and careful analysis reveal that refinement relations have the same semantics of AND-refinement, i.e., a parent goal must be broken into a number of sub-goals in order to make the achievement of the parent goal more manageable. With the refinement semantics, the only difference between AND-refinement and refinements is the fact that the achievement of sub-goals represented in the model does not imply the achievement of the parent goal and it might exist other sub-goals whose satisfaction contribute to the satisfaction of the parent goal. However, opening the possibility of creating new goals on demand implies in the pursuit and achievement of goals which are not captured within the model and thus, it is not possible to reason with such goals.

Within the second workaround, when SIENA concepts had no concepts with slightly similar semantics to ArchiMate concepts, SIENA concrete syntax has been used in red color to highlight the absence of concepts in ArchiMate. In this context, different SIENA agents (owners) responsible for the achievement of goals have been mapped to ArchiMate stakeholders. However, although it is possible to map different SIENA agents to ArchiMate stakeholders, the inclusion of stakeholders for all goals in SIENA would increase the size and complexity of our goal models. Therefore, we have stuck to SIENA visual syntax in red for specifying them. Further, among other SIENA concepts, as the ArchiMate language contains a number of gaps, we have borrowed SIENA modeling constructs for capturing environmental factors (situations and their SWOT relations) as well as concepts for the representation of the behavioral domain (operations).

Comparison Between SIENA and ArchiMate Methodology. Along the present modeling effort, we have focused on the comparison of modeling primitives of SIENA and ArchiMate frameworks, but some considerations about the methodology and reasoning technique in both frameworks must be also made.

Regarding the methodology, both frameworks intend to use enterprise architecture models to perform the planning of the enterprise architecture, although they diverge in the ways how to perform that. In SIENA, the methodology intends to support the several steps of the enterprise planning process (Figure 2.1) by: (i) setting up mission and vision, (ii) setting up goals along multiple levels, (iii) environmental factors that impact their achievement, (iv) the business processes that realize such goals and (v) business process control-flow in terms of social expectations and operational steps. The main SIENA aim is to support the operationalization of the enterprise's mission and goals by the creation of its realizing business processes, as performed in the enterprise planning process.

In contrast, ArchiMate uses motivational models to drive changes in an AS-IS enterprise architecture to a future TO-BE enterprise architecture. In this sense, the ArchiMate framework is used as an instrument for planning the realization of the enterprise architecture by means of applications, services and processes. However, the language does not support the realization of the enterprise planning process as described in Management literature (according to described in Figure 2.1) due to the absence of many concepts inherent to the enterprise planning process. In order to precisely explain how ArchiMate supports such planning process, we summarize the ArchiMate methodology as follows: 1. The methodology starts with the enumeration of stakeholders' concerns (e.g. "profit", "customer satisfaction") for each stakeholder (e..g CEO, IT department). 2. Then, such stakeholders have to make assessments about what might help/harm such concerns. For example, "dropping sales" and "leaving customers" are two assessments that represent threats and weaknesses for the "profit" concern. 3. On the basis of such concerns and assessments, the company then elaborates its goals, their AND/OR-refinements and contributions for each concern. For example, in order to address the "profit" concern, the company elaborates the "Increase sales" sales goal and refine this goal in terms of sub-goals, alternatives and contributions. 4. Finally, the company derives use cases and (system) requirements from goals and link them to their realizing business services and processes. As can be seen from this methodology, the absence of goals of different shades, situations and their SWOT relations, operations and activities does not allow one to perform the steps of the enterprise planning process. The only planning process step which can be performed in ArchiMate is the goal setting (step 1), but goals of different shades cannot be specified due to the absence of such modeling constructs in the language. Furthermore, although the ArchiMate strategic planning extension provides a more expressive goal ontology, the approach refrains from presenting a methodology to specify such goals.

Comparison Between SIENA and ArchiMate Reasoning Techniques. Regarding reasoning, although the ArchiMate AME language and methodology [160] argues that goal analysis techniques can be used to evaluate architectural alternatives, the paper refrains from providing such contribution. Consequently, the ArchiMate language cannot be used for selecting strategic planning alternatives within the enterprise planning process, as done in Chapter 6.

Overall, as stated at the beginning of such evaluation, our ultimate goal is to determine whether the SIENA framework advances the state of art in strategic enterprise architectures. After we have analyzed the comparison between SIENA and ArchiMate modeling frameworks in terms of their modeling primitives, methodology and automated reasoning technique, we can conclude that:

1. The comparison between SIENA and ArchiMate frameworks in terms of their modeling primitives has revealed SIENA to have higher coverage for Management conceptualization, thus leading us to infer a superiority of SIENA over ArchiMate in terms of expressiveness for the representation of strategic enterprise architectures. Therefore, we conclude that SIENA advances the state of the art for the representation of strategic enterprise architectures;

2. The comparison between SIENA and ArchiMate frameworks in terms of their methodologies has revealed both frameworks to use enterprise models for enterprise planning. However, ArchiMate cannot support the enterprise planning process proposed by the SIENA methodology. Therefore, as SIENA proposes a different methodology for the use of enterprise models, we conclude that SIENA advances the state of the art for the methodological use of strategic enterprise architectures models;

3. An investigation of SIENA and ArchiMate frameworks in terms of reasoning techniques revealed a lack of ArchiMate reasoning techniques to the best of our knowledge. Therefore SIENA advances the state of the art for the analysis of strategic enterprise architectures models;

In face of the complementary characteristics of SIENA and ArchiMate frameworks, we consider both frameworks can learn from their experience. In particular, ArchiMate could be enhanced to incorporate different shades of goals, dimensional refinement operators, operations, situations and their SWOT relations and representation of process' control-flow. This would enrich the language, opening up the possibility of finding a complementary usage of such modeling concepts with the already existent concepts of the language (i.e., the concepts not covered by our mapping). This would allow one to get interesting insights from the enterprise architecture modeled in ArchiMate. In relation to SIENA, we initially thought about incorporating assessment and concerns, but difficulties with their understanding in real-world projects reported in [56] may hinder their practical usage. Due to our ultimate goal of supporting the enterprise planning process in SIENA, the incorporation of resource and capabilities from ArchiMate might represent a benefit for the language. Further, as ArchiMate also allows the representation of how applications may support the achievement of business goals (with the representation of use cases and system requirements), SIENA could also be enhanced to incorporate the same idea.

8.4 Summary

This chapter reports three evaluation phases of the SIENA modeling framework. More specifically, the first phase evaluates the achievement of the requirements for strategic enterprise architectures by the SIENA modeling framework. On the basis of such evaluation, we conclude that the framework achieves all requirements stipulated for strategic enterprise architectures. Table 8.1 depicts such desired requirements and in which chapter they are achieved.

The second phase evaluates the feasibility of the SIENA modeling framework for capturing a real-world strategic enterprise architecture from Rheumatology department of the university hospital. On the basis of the reported modeling effort, we demonstrate the real-world applicability of the SIENA modeling framework. Further, the experience with our integrated hierarchical architecture also demonstrates to be a more expressive solution for the hospital representation in comparison with our previous approach (using Tropos and ARIS modeling languages).

Finally, the third evaluation phase compares the SIENA modeling framework with the ArchiMate framework in order to check whether SIENA advances the state of art in the representation and analysis of strategic enterprise architectures. With this evaluation in hands, we concluded that SIENA advances the representation, methodology and reasoning in strategic enterprise architectures.

.

Chapter 9

Evaluation of Azzurra Modeling Language

This chapter describes the evaluation of the Azzurra modeling language which is performed in three phases, similar to the evaluation of the SIENA framework. The first evaluation phase (Section 9.1) reports on evaluation for the Azzurra language using two real-world scenarios from the medical domain. The first self-evaluation intends to compare Azzurra's representational features with the current state of the art of process modeling languages, whereas the second self-evaluation highlights certain domain features of the scenario that could be better supported by a commitment-based representation. The second phase (Section 9.2) reports on an experiment conducted with master students to investigate the suitability of Azzurra and BPMN for the representation of structured and unstructured processes. This experiment has been published in [28]. Finally, the third phase (Section 9.3) reports on a modeling effort in which the first author of this thesis has supervised three master students [188, 193, 190] in conjunction with her supervisors in a quality comparison between Azzurra and BPMN modeling frameworks in terms of expressiveness, usability and comprehensiveness criteria. With this third evaluation phase, our intention is to check the experience of other modelers with Azzurra to reduce the evaluation bias, by establishing a comparison of BPMN and Azzurra in terms of certain criteria. Further, we also intend to investigate the feasibility of Azzurra for conducting a real-world modeling effort with processes from the medical domain (clinical guidelines).

9.1 Comparison Between Azzurra and Process Modeling Languages

Within the first phase, we conducted an evaluation of Azzurra's applicability by modeling two scenarios that have been extracted from two different real-world cases from the medical domain. The healthcare domain has been selected due to the recognition of being one of the most promising, but still challenging domains for the adoption of process-oriented solutions due to complex needs stemming from the business domain [44].

The first scenario (Section 9.1.1) compares Azzurra's representational features to those of the three main types of process modeling languages: imperative, declarative and artifact-centered paradigms (for more details regarding process modeling languages, please refer to Section 4.3.1 (Business Process Modeling Approaches)). In order to conduct such comparison, we have selected the fracture treatment scenario from [198] which is represented in this approach using the DECLARE modeling language. Subsequently, we have selected BPMN, DECLARE and artifact-centered modeling languages to establish our comparison. Such languages have been selected since they are the most prominent representative languages of each paradigm. In this first scenario, our goal is to demonstrate in which aspects Azzurra conceptualization differs from the other representational methods.

The second scenario (Section 9.1.2) emphasizes certain domain characteristics of the scenario of clinical guidelines (CGs) that could be better supported by a commitment-based representation. For that, we select the

Transient Ischemic Attack (TIA) clinical guideline from [203] which is represented in this approach in an imperative style language. Subsequently, we depict the TIA guideline imperative process model as a means of presenting the domain characteristics of CGs and then, we contrast this imperative representation with the corresponding commitment-based representation. In this second scenario, our goal is to demonstrate that CGs could be better supported by a commitment-based representation.

9.1.1 Fracture Treatment Scenario

In this first scenario, we selected the fracture treatment scenario from [198] which is represented in this approach using the DECLARE modeling language. In this context, we have used this process model representation in DECLARE to learn about the domain of fracture treatments and to elaborate process models of this domain using BPMN and artifact-centered modeling languages. Figure 9.1 depicts the alternative models of the fracture treatment example, each of them modeled using a type of process modeling language and compare them with the Azzurra model presented earlier (Figure 7.4) in Chapter 7.

Results. Figure 9.1(a) depicts an operational model for the running example using the BPMN modeling language. Imperative languages represent business processes in terms of activities to be executed as well as the exact sequence between these activities. Here, the model consists of activities (e.g., "Examine patient" and "Verify need of medication") and the control flow among them. Since activities must be explicitly activated for enactment, this type of representation requires an explicit (... and exhaustive) specification of all possible enactment paths. For instance, a recurrent enactment path for our example is "Examine patient" and then "Verify need of medication", but there are many others as well (not represented in Figure 9.1(a)). Azzurra models enable more flexible specifications of process

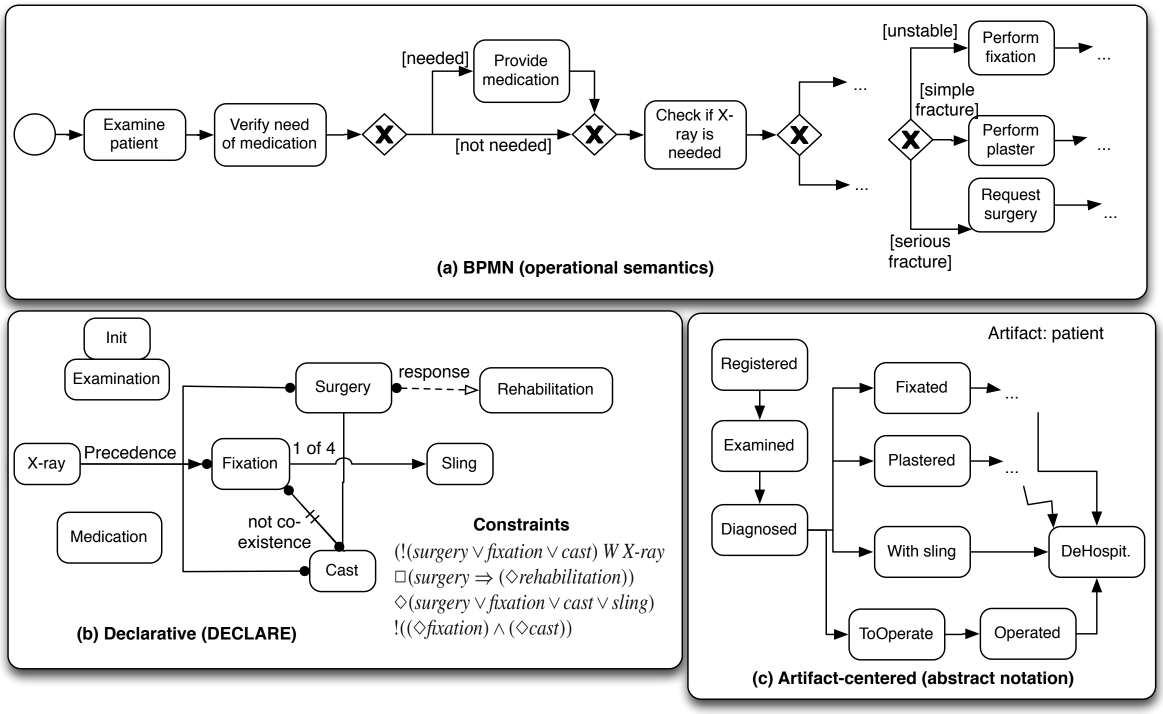


Figure 9.1: Snippets of the fracture treatment process using (a) an operational workflow language; (b) a declarative language; (c) an artifact-centered notation

models because it only requires the specification of essential ordering constraints between commitments. For instance, in Table 7.2 and Figure 7.4, only C_1, C_4 and C_6 include temporal constraints. Further, as commitments can be satisfied by different activities, the “Examine patient” commitment could be fulfilled through different operationalizations as for instance, the doctor could first “Perform a physical evaluation” and subsequently “Examine patient’s family history” or alternatively, s/he could perform the same activities in the inverse order.

Conclusions. Unlike imperative languages, declarative ones require only the minimal set of constraints between activities. By default, all execution paths are allowed and prohibited execution paths are specified by constraints on the execution order between activities. Figure 9.1(b) (extracted from [198]) presents the declarative specification of our running example using DECLARE. Azzurra is also declarative like DECLARE, but it does not focus on activities for expressing business processes, rather emphasizing their social nature by capturing agents and commitments between them. The approach of modeling business processes in terms of commitments among process participants also increases flexibility in the specification as in the imperative paradigm, once it does not constrain process participants to execute particular activities during runtime, but instead, it expands the number of operational choices as long as these activities satisfy the commitments among agents.

In contrast with its activity-centered cousins, the artifact-centered paradigm promotes data objects to first-class citizens in modeling a process, by describing the lifecycle of each object. Here, activities that change/update the state of an object are also represented. In our example, fracture treatment is represented as a data object called "Patient" with several interconnected states. The control flow of the business process does not have to be exhaustively modeled, relying instead on the lifecycle model of the data objects: "registered", "examined", . . . , "de-hospitalized" (see Figure 9.1(c)). The states of the data object are similar to the propositions in the Azzurra version of the process (e.g., "examined", "diagnosed" in Table 7.2). However, by centering the representation in artifacts, the business process has an operational perspective. Differently, Azzurra's commitment-based representation highlights the social nature of business processes, representing who is responsible for advancing the state (the debtor in a commitment). Further, while the artifact-centered paradigm focuses on the activities that change the states of data objects, Azzurra focuses on correctness criteria rather than specific operationalizations. This approach favors flexibility as different activities are admissible at runtime, as long as they satisfy the correctness criteria stipulated by the commitments.

9.1.2 Clinical Guidelines Scenario

In our second scenario, we also consider a business process from the medical domain that concerns Clinical Guidelines (CGs). CGs consists of "systematically developed statements to assist practitioner and patient decisions about appropriate health care for specific clinical circumstances" [70]. In the context of a CG, every activity in the process model corresponds to a recommendation that supports healthcare providers (doctors, nurses, etc.) to develop care actions for patients. Therefore, every activity within the process model can be understood as an abstract recommendation (abstract activity) to be adapted at runtime according to a specific patient by the healthcare provider executing the CG. Given the abstract nature of CGs that require extensive adaptation of abstract activities at runtime, we say that CGs are inherently *decision-intensive* business processes.

Results. Figure 9.2(a) depicts an example of an executable clinical guideline for transient ischemic attack (TIA) (an episode of neurological occurrence) from the literature [203, 214]. In this approach, the TIA guideline is represented in an imperative style language in which each activity represents a healthcare recommendation and the control-flow links from the imperative language represent executing constraints between these recommendations.

To exemplify the decision-intensive nature of a CG, consider the "Treat for stroke" recommendation/activity. During process execution, this recommendation has to be personalized for a specific patient, considering (i) the execution context (ii) doctor's expertise and (iii) patient's clinical circumstances. For instance, assuming that there are two procedures for treating stroke ("surgery" and "endovascular procedure"), the doctor has to select the best alternative for the patient by considering environmental constraints, such as the availability of procedures and/or costs of each of them.

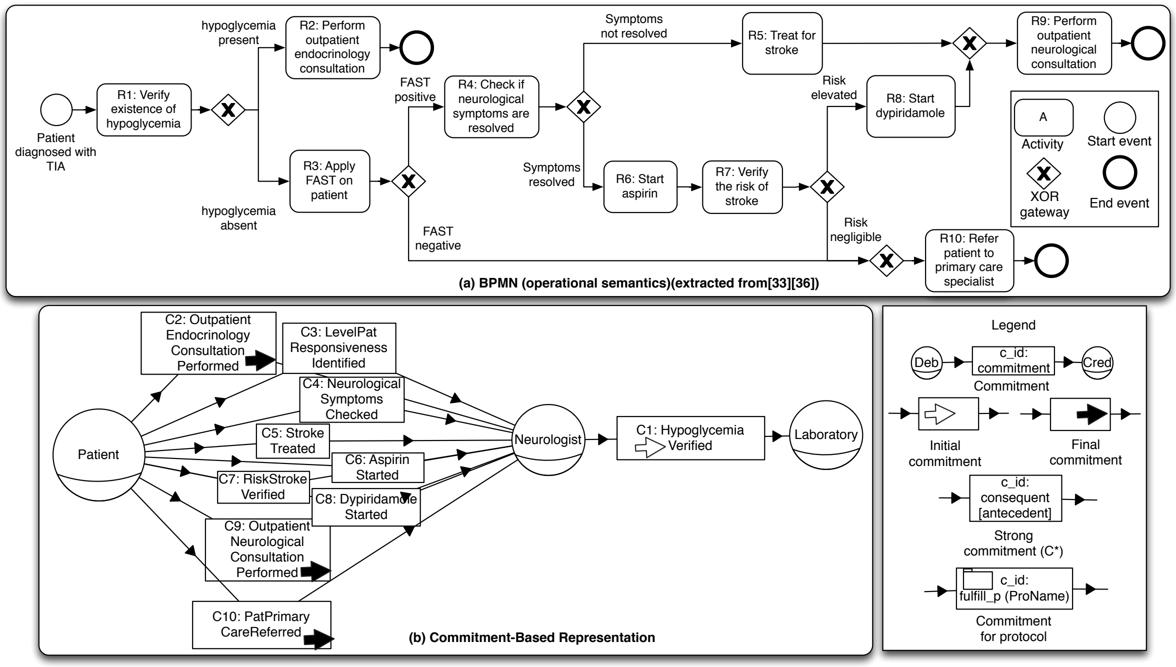


Figure 9.2: Transient Ischemic Attack (TIA) Clinical Guideline using (a) an operational workflow language (BPMN) and (b) a commitment-based representation

Conclusions. The TIA guideline imperative representation has been instrumental in enabling us to learn about the TIA CG domain, to discover the domain characteristics of clinical guidelines (e.g. its decision-intensive nature) and to understand how current approaches address CG representation. In this context, we realized that most of the languages for representing CGs follow a task-based paradigm [151] in which recommendations are represented as actions and decisions in a rigid flowchart-like (imperative) structure [44], like the BPMN representation in Figure 9.2(a). However, adopting this approach indeed introduces a number of shortcomings in the CG representation from a domain point of view, that are required by imperative process languages, like inexistent ordering constraints between multiple recommendations. Although it is out of the scope of this thesis to provide a more extensive discussion about the topic of CG representation, our intention here is to demonstrate how a commitment-based approach could help to tackle some of the problems with the imperative representation.

For that, we introduce in Figure 9.2(b) the respective commitment-based representation of Figure 9.2(a). While activities in CGs represent recommendations for healthcare providers on how to address particular clinical circumstances, commitments instead capture these recommendations as compromises of the healthcare provider who is executing the guideline towards the patient (and also the compromises of other healthcare providers in the scope of the guideline). In the remainder, we point out some of the shortcomings introduced by imperative languages and contrast the corresponding representation with the Azzurra model:

- *Negative recommendations:* Imperative process models (Figure 9.2(a)) describe recommendations like “Treat for stroke” and “Apply FAST on patient” as activities. This approach works well for positive recommendations, i.e., actions that have to be performed. Differently, for negative recommendations as “Do not provide aspirin” (which is admissible from a business perspective [214]), the activity-based representation fails. Indeed, the existence of negative recommendations suggests that recommendations are not actions themselves, but rather positive and negative restrictions on the behavior (actions). By centering the representation on commitments, Azzurra specifies restrictions on behavior, defining correctness criteria that should not be violated. In this case, the issue with negative recommendations can be solved by specifying a commitment whose consequent is a negative correctness criterion (for example, $\neg AspirinProvided$);

- *Ordering constraints:* As the imperative representation represents recommendations as actions that have to be performed (and actions are represented in sequence within the imperative paradigm), the paradigm imposes a natural sequence among these recommendations. From a domain perspective, however, ordering constraints among recommendations are not necessary or even desirable [214] (this lack of sequence can be indeed evidenced by the existence of negative recommendations). Differently, Azzurra does not impose any order among commitments, but when necessary, they can be specified by matching commitment's consequent and antecedent;

- *Conflicting recommendations:* in the imperative representation, recommendations are modeled as labeled activities (textual information) and no mechanisms are specified to correlate related actions (for instance, "Provide aspirin" and "Do not provide aspirin" are modeled as unrelated actions in the specification). As a consequence of that, external rules must be defined to capture conflicting actions, whereas automatic detection could be performed by reasoning over the meaning of the actions [214]. In a commitment-based approach, as commitment's consequent capture recommendations (for instance, for a recommendation "Provide aspirin", the commitment consequent is $AspirinProvided$), conflicting recommendations could be automatically detected. For instance, in a hypothetical situation in which aspirin conflicts with clopidogrel, the knowledge base could capture this conflict as a rule and design-time model-checking techniques could be applied to reason about conflicting commitments (for example, two commitments whose consequent are $AspirinProvided$ and $ClopidogrelProvided$ cannot exist in the same Azzurra specification). Alternatively, other conflicting recommendations could also be detected, like "Provide aspirin" ($AspirinProvided$) and "Do not provide aspirin" ($\neg AspirinProvided$).

- *Compliance checking:* As a guideline specification is intended to provide recommendations for healthcare providers to execute actions, from a practical point of view, they have the freedom to either change the suggested care actions (i.e., change the actions that satisfy a given recommendation/commitment) or even to completely skip certain recommendations when necessary. However, compliance with guidelines is assessed in a strict manner by only matching recommended actions with executed actions [195]. This means that, although they are free to select the best care actions at runtime, substitutions in the recommendations will accuse false cases of non-compliance. Azzurra leverages compliance to the business level, by not specifying concrete actions to be executed, but rather correctness criteria. This opens the possibility of using alternative actions to fulfill the commitment (depending on how suitable they are in relation to the executing context), as long as they satisfy the commitments. Furthermore, by capturing actors and their commitments, accountability can be easily checked in an Azzurra specification. This is also fundamental in a medical context, once responsibility for care actions need to be strictly tracked along the treatment process.

9.1.3 Scenarios Discussion

The fracture treatment scenario shows how Azzura natively supports modeling business processes in the healthcare domain; this style of modeling has advantages in other domains too. Unlike current languages, that center their representation either in *activities* or *data objects*, Azzurra captures the *social nature* of the interactions between process participants by expressing these interactions in terms of commitments (correctness criteria based on social expectations).

Centering the representation in terms of activities/data objects leads to an operational business process representation, once the behavior is specified in terms of specific operationalizations to achieve the desired outcomes, rather than *what* is supposed to be achieved. As a general consequence of the shift in the representation, specifications in Azzurra allow one to capture business processes in more strategic terms. In particular, the benefits of such approach in the first scenario can be manifested as (i) the ability to focus on the social perspective of the business processes, (ii) it enables a more flexible representation of the process than its respective counterparts in other process languages. This flexibility is manifested through the ability to specify different sequences of commitments that can be satisfied by different concrete activities.

While in the first scenario Azzurra provides increased flexibility for business process specification, the second scenario demonstrates that a shift in the modeling paradigm is rather fundamental to address the representational needs (knowledge structure) of clinical guidelines. To enumerate the CG flexibility needs more concretely, guidelines are inherently *decision-intensive* and act as abstract templates/blueprints that provide evidence-based decision support for healthcare providers. They do not prescribe the actual behavior within the business process, but rather *constraints on the behavior* and require subsequent adaptation and personalization to obtain a concrete medical treatment (actions) for a given patient [44]. As a result, it is not possible to define a priori all the variants in the execution of a business process (imperative modeling would require doing so). Azzurra, on the other hand, represents these guidelines through correctness criteria in terms of commitments.

In summary, Azzurra better supports not only the representation of the knowledge structure of the domain (by being able of representing negative recommendations as well as the essential ordering constraints), but also presents an advantage for the reasoning techniques that must be executed on the basis of CG models, as reasoning about conflicting recommendations and checking compliance. More specifically, considering compliance checking, Azzurra expands the notion of compliance to the business level, once correctness criteria allow one to consider different actions that satisfy commitments and not to necessarily stick to one particular activity as it is done in the current practice. Moreover, relying on commitments between agents, Azzurra natively supports accountability, i.e., enables determining at all times which agents are compliant, and which ones have violated a commitment they are responsible for.

9.2 Empirical Evaluation with Master Students

The evaluation conducted with the two previous modeling scenarios provided interesting insights regarding Azzurra's suitability for the specification of business processes. Within the first scenario, we demonstrated that a shift in the representation focus from activities to commitments enabled Azzurra to provide a more flexible solution for the representation of business processes than its process modeling counterparts. Within the second scenario, the language does not only provide a more flexible solution, but can also better capture the intrinsic characteristics of the clinical guidelines domain. The overall conclusion of the evaluation with scenarios enabled us to realize that a focus on commitments allows Azzurra to provide a more flexible representation for processes which is beneficial for the representation of clinical guidelines due to their intensive flexibility nature.

On the basis of the conclusion that Azzurra can better cope with the representational flexibility needs from clinical guidelines and as a clinical guideline consists of one concrete example of unstructured process, our intuition rests on the fact that Azzurra can better capture the features of unstructured business processes. In order to characterize unstructured processes and their flexible nature, we introduce the spectrum of work. In BPM literature, several classifications exist for business processes according to their characteristics [53, 196]. A common classification scheme considers the level of structuring or predictability, thus dividing business processes into a spectrum of work of four types (see Figure 9.3) [53, 196, 201]. The level of structuring and predictability basically considers the extent to which the behavior of a given business process is predictable at modeling time.

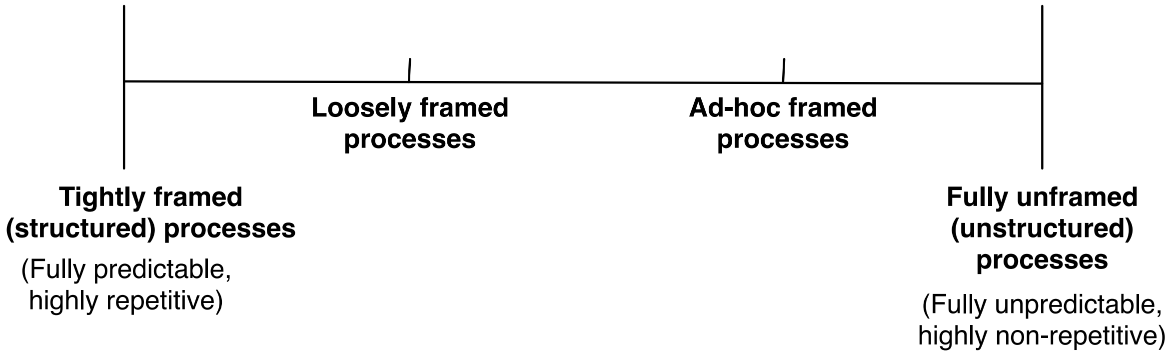


Figure 9.3: The Spectrum of Work in BPM adapted from [162]

In the leftmost extreme of the spectrum, a *tightly framed (or structured) process* comprehends those processes whose execution of activities consistently follows a predefined process model [53, 196]. Since a formal representation of these processes can be easily described prior to their execution, tightly framed processes are characterized as fully predictable and repetitive and after their design-time description, they can be repeatedly instantiated at runtime. Examples of this category are production and administrative processes [45] and as well as bank transactions that are executed in an exact sequence to comply with legal norms.

Even though tightly framed processes usually have a predictable behavior, a certain degree of unpredictability is expected due to the occurrence of exceptions and evolutions within the domain. Therefore, a *loosely framed process* corresponds to a process in which it is possible to represent the process behavior and a set of constraints a priori [196], such that the process model describes the "standard way of doing things" while requiring additions, removals or generation of alternative sequence of activities during runtime [45].

Contrasting with tightly and loosely framed processes that can be described a priori by an explicit process model, the behavior of *ad-hoc framed process* cannot be determined in terms of an explicit process logic during design time due to a lack of domain knowledge or the complexity of task combinations. Instead, only structured fragments can be identified a priori and properly composed on a per-case basis, while process parts that are undefined or uncertain can only be specified and incorporated as the process evolves [45].

Finally, within the rightmost category of the spectrum, *fully unframed (or unstructured) processes* have sufficient variability in such way that no process description can be pre-defined at all [45, 196]. As a result, process participants need to make decisions using their knowledge to create activities on demand. The creation of such activities is based on situation-specific parameters whose values are determined as the process execution proceeds. Besides choosing activities on demand, they also dynamically decide the execution order of such activities.

With these insights provided by the evaluation phase 1 with scenarios, in this second evaluation phase, we perform an experiment with students to check the validity of our insights regarding the suitability of Azzurra and BPMN for structured and unstructured processes. BPMN has been chosen for the comparison under consideration due to its wide acceptance and popularity as a standard for business processes representation [161, 84]. More specifically, with this experiment, we want to acquire objective and statistically significant evidence regarding the suitability of Azzurra for unstructured processes. In order to perform the experiment, we elaborated the following propositions:

P1. Azzurra produces models of better quality than BPMN in the representation of unframed (unstructured) business processes;

P2. BPMN produces models of better quality than Azzurra in the representation of tightly framed (structured) business processes.

9.2.1 The Experiment Process

The design of our experiment has been conducted on the basis of guidelines for experimentation in software engineering [204, 100]. According to such guidelines, the experiment process can be divided into five main activities depicted in Figure 9.4.

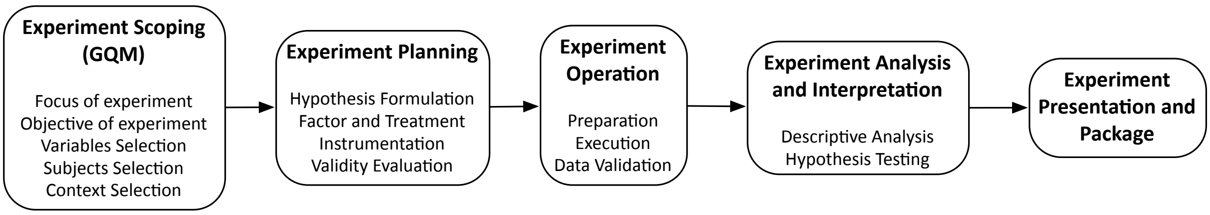


Figure 9.4: The Experimentation Process According to [204]

Within the **Scoping** activity, the experiment is defined in terms of problem statement and goals, defining *why* the experiment is needed. According to the Wohlin‘s guidelines [204], the Goal, Question, Metric (GQM) template [12] comprehends a suitable instrument for defining the scope of a given experiment. Our GQM template is described in Section **Experiment Scoping and Planning**.

The **Planning** activity is the phase in which the foundation of the experiment is laid, defining *how* it is conducted. The steps conducted in the scope of our planning activity are described in Section **Experiment Scoping and Planning**.

The **Operation** activity encompasses the preparation of subjects and required material on which the experiment is executed (i.e., *objects*), the actual execution of the experiment as well as the collection of measurements (see Section **Experiment Operation**). The **Analysis and Interpretation** activity focuses on qualitatively and quantitatively processing the outcomes of the experiment (Sections 9.2.2 and Section 9.2.3). Finally, the results are presented in the course of the **Presentation and Package** (leading to the present paper).

Experiment Scoping and Planning

Our experiment starts by scoping its objectives using the GQM template depicted in Table 9.1:

Table 9.1: GQM for our experiment

Focus of the experiment: Analyze Azzurra specification language and compare it with the BPMN modeling language.
Objective of the experiment: Checking the adequacy of the Azzurra and BPMN languages for the representation of structured and unstructured business processes.
Variables selection: We compare Azzurra and BPMN modeling languages in terms of model quality.
Subject: From the point of view of M.Sc. students enrolled in classes of Organizational Information Systems.
Context of the experiment: M.Sc. students creating Azzurra and BPMN models.

In the following, the planning phase of our experiment required us to elaborate the *hypotheses* (together with the *independent* and *dependent variables*), *factors* and *treatments* applied to our experiment.

Hypothesis Formulation. As we intend to compare Azzurra and BPMN for structured and unstructured processes, we construct three null hypotheses, one for each factor and a third one for the interaction between the factors [204].

- **Null Hypothesis $\mathbf{H_{0\text{-}1}}$:** There is no significant difference in model quality of Azzurra and BPMN modeling languages.
- **$\mathbf{H_{a\text{-}1}}$:** There is a significant difference in the model quality of Azzurra and BPMN modeling languages.
- **Null Hypothesis $\mathbf{H_{0\text{-}2}}$:** There is no significant difference in model quality of structured and unstructured scenarios.
- **$\mathbf{H_{a\text{-}2}}$:** There is a significant difference in model quality of structured and unstructured scenarios.
- **Null Hypothesis $\mathbf{H_{3}}$:** There are no significant interactions between the type of modeling language and types of business processes in terms of model quality.
- **$\mathbf{H_{a\text{-}3}}$:** There are significant interactions between the type of modeling language and types of business processes in terms of model quality.

Note that our hypotheses are elaborated in terms of model quality (*dependent variable*). In order to select the metrics for measuring model quality in our evaluation, we get inspiration from the field of Ontology Engineering; more precisely, we use a formal evaluation framework [189] that defines the dimensions of *precision* and *coverage* to define the quality of a given ontology (model).

In [189], a conceptualization comprehends a set of conceptual relations about a certain portion of reality perceived by an agent, defining a set of intended models I_K. In this context, the role of an ontology is to provide a specification of such conceptualization, precisely capturing the intended models according to such conceptualization and excluding the non-intended ones. Considering that it is not always easy to find the right set of entities so that an ontology admits only the intended models [80], ontologies are considered only approximations of conceptualizations. Consequently, the formal framework of Staab et al. [189] proposes a schema for evaluating ontologies with respect to the degree of approximation they can provide to their respective conceptualizations. To evaluate such degree of approximation, the *precision* and *coverage* metrics are introduced and can be mathematically defined as:

$$P = \frac{|I_K \cap O_K|}{|O_K|} \textit{(precision)} \qquad C = \frac{|I_K \cap O_K|}{|I_K|} \textit{(coverage)}$$

In Ontology Engineering, *precision* measures how much the represented models O_K are relevant according to the set of intended models I_K, while *coverage* measures how much of the intended models I_K are represented by the ontology O_K. We use analog reasoning for our evaluation of Azzurra and BPMN modeling languages. In our case, business processes are considered the target conceptualization that can be represented by two distinct ontologies, i.e., the Azzurra and BPMN modeling languages. Every business process has a natural language description that admits a number of execution paths (in our case, the set of intended models I_K corresponds to the set of intended execution paths $I_{execPath}$) and specifications in BPMN and Azzurra provide representations of such execution paths ($R_{execPath}$). Therefore, *precision* measures how many paths which are represented in the model are correct in relation to the intended paths prescribed by the natural language description, while *coverage* measures how many paths provided in the natural language description are indeed captured in the model representation. In our case, *precision* and *coverage* are mathematically defined as follows:

$$P = \frac{|I_{execPath} \cap R_{execPath}|}{|R_{execPath}|} \textit{(precision)} \; C = \frac{|I_{execPath} \cap R_{execPath}|}{|I_{execPath}|} \textit{(coverage)}$$

Factor and Treatment. As the aim of our experiment is to investigate whether the Azzurra modeling language has a more faithful representation of unstructured business process than the BPMN modeling language, we have two factors: factor A is the type business process modeling language (whose treatments are Azzurra and BPMN modeling languages) and factor B is the type of business process under consideration (whose treatments are unstructured and structured business processes). Factors and treatments are depicted in Table 9.2:

Language Type (Factor A) / BP Type (Factor B)	Azzurra	BPMN
Structured		
Unstructured		

Table 9.2: Factors and Treatments applied in our experiment

Instrumentation. Participants used a free online modeling tool[1] for the elaboration of BPMN 2.0 models and a plug-in[2] developed at the University of Trento for the elaboration of Azzurra models. At the end of the experiment, they provided the source of Azzurra and BPMN models for later evaluation of the results.

Validity evaluation. We enumerate the main threats to the validity of our experiment using the Wohlin‘s categorization [204]:

Threats to construct validity. The threats in this category are: (i) a major threat to construct validity is that the chosen business processes may not be representative samples for the structured and unstructured types of business processes. To mitigate this issue, we have chosen already consolidated scenarios within the BPM literature as representatives from structured and unstructured processes; (ii) furthermore, the domain knowledge involved in the description of the scenarios may entail some difficulty during the modeling process; (iii) the fact that BPMN is an imperative language,

[1] `www.lucidchart.com`

[2] `https://trinity.disi.unitn.it/azura/azura/`

CHAPTER 9. EVALUATION WITH MASTER STUDENTS

while Azzurra is declarative may also entail additional difficulties as there is some evidence that imperative languages are more understandable than declarative ones [152]; (iv) hypothesis guessing may also represent a threat as subjects can be conditioned by the results they are providing. We mitigated this threat by carefully formulating questions on the basis of correct usage and preference of modeling languages.

Threats to external validity. Here, our largest threat is the use of students as subjects in our experiment. Further, they had prior training in BPMN and UML activity diagrams during the course lectures. To mitigate these issues and make their background more uniform, we have provided preliminary training in both Azzurra and BPMN languages by means of one example. In order to encourage subjects to participate, they could earn at most one point in the overall course grade on the basis of the correct usage of languages constructs.

Threats to conclusion validity. The two threats to conclusion validity are the low number and homogeneity of the samples (students) that may impact our ability to reveal patterns in the data. Besides that, the first author of this paper evaluated the number of admissible execution paths for each scenario, together with their respective representations in Azzurra and BPMN.

Threats to internal validity. This type of validity is threatened by the effect of order in which the subjects apply the treatments (structured and unstructured) as students may learn the content of natural languages descriptions, and the second models are easier to produce. To mitigate the effect of the order, the order is assigned randomly to each subject. By having the same number of subjects starting with the first treatment as with the second, the design is balanced [204].

Experiment Operation

Preparation. We continue following the same rationale of evaluation through modeling scenarios. In particular, we have used same business process from Scenario 2 used in [36] (i.e., the TIA clinical guideline) as a representative of an unstructured business process and the X-Ray Medical Order (extracted from [162]) as the representative of a structured business process. The selection of both scenarios as representatives of unstructured and structured business process has been supported by BPM literature that positions clinical guidelines as unstructured processes [45] and the X-Ray Medical Order as a structured process [162].

Next, a natural language description[3] has been extracted from literature in order to be applied to the subjects. Further, the corresponding Azzurra and BPMN models have been built in advance for each scenario by the first author with the purpose of ensuring that those process models to be built in each scenario indeed covered the core concepts of both modeling languages.

Experiment execution. The experiment has been conducted in July 2015 with master's students in Computer Science in the scope of the Organizational Information Systems Course at University of Trento. In total, 17 subjects participated in this empirical test. The experiment has been structured in different parts:

- **Introduction Phase (15 min)**: General instructions about the experiment and introduction to Azzurra modeling language and modeling tool together with a presentation about BPMN. It is also important to note that students had prior contact with BPMN along the course lectures;
- **Experiment phase (40 min, i.e., 20 min for each language)**:

[3] Scenario descriptions, experimental results and data analysis are available at `https://www.dropbox.com/s/8qlwd5svqbt3hmw/Empirical%20evaluation.zip?dl=0`

Group 1 models the structured scenario using Azzurra and BPMN, whereas group 2 models the unstructured scenario using Azzurra and BPMN;

- **Questionnaire phase (15 min)**: General questions concerning the background of the subject and questions regarding the elaboration of models relative to scenario 1 and 2.

Data validation. The obtained data were checked for consistency and plausibility. We discarded the inputs from two students due to incompleteness; thus, we could employ data from 15 students in the data analysis.

9.2.2 Experiment Analysis and Interpretation

To report experimental results, Table 9.3 shows mean, median and standard deviation values for precision and coverage by language and process type:

Table 9.3: Precision and Coverage by Language and Process Type

		Azzurra			**BPMN**		
		Mean	Median	Std. dev.	Mean	Median	Std. dev.
Unstructured	Precision	1.00	1.00	0.00	1.00	1.00	0.00
	Coverage	0.89	1	0.18	0.34	0.36	0.07
Structured	Precision	1.00	1.00	0.00	0.95	1.00	0.13
	Coverage	0.82	0.75	0.19	0.82	0.75	0.19
Overall	Precision	1.00	1.00	0.00	0.97	1.00	0.09
	Coverage	0.85	1.00	0.18	0.60	0.50	0.28

We conducted statistical analysis to test whether the null hypothesis H_0 can be rejected, thereby allowing us to draw conclusions about our studied phenomenon: the modeling of structured and unstructured business processes.

For the selection of the statistical tests, we followed the guidelines prescribed by Harvey [85, Chap. 37]. As the participants of our experiment applied both methods, to test $H_{0\text{-}1}$, we can use the paired t-test or its non-parametric analog, Wilcoxon test. However, the participants did not switch scenario type and, therefore, to test $H_{0\text{-}2}$ we use the unpaired t-test or its non-parametric analog, Mann-Whitney (MW) test. Finally, to test $H_{0\text{-}3}$ we need to investigate the difference between the combination of two factors (type of language and type of process), which requires ANOVA test or its non-parametric analog, Kruskal-Wallis (KW) test [204]. We checked the normality of data by Shapiro-Wilk test which returned *p-value* = 0.0013 for coverage and *p-value* = $6.8 \cdot 10^{-11}$ for precision. Thus, we used non-parametric tests for all three hypothesis. Further, for all statistical tests, we use a threshold of 5% for α, the probability of committing Type-I error [204].

Null Hypothesis $H_{0\text{-}1}$ (Azzurra vs. BPMN): The results of the Wilcoxon test revealed a statistically significant difference between two modeling languages with respect to coverage (test results: $W = 7$, $Z = 2.09$, *p-value* = 0.04, Cohen's d = 1.06) and no significant difference in precision (*p-value* = 0.32). The power of the Wilcoxon test for coverage is 0.72. Therefore, we cannot reject the null hypothesis both for coverage and precision. However, to achieve 80% power for coverage we would need a sample size of 16 participants, while we had 13 participants. For Azzurra, the overall mean coverage is 0.85, whereas for BPMN the overall mean coverage is 0.6. As coverage describes the percentage of the intended interpretations (according to the natural language description) that are indeed captured by the model, a mean coverage of 0.85 means that 85% of all intended paths are captured in the model, whereas 15% of them are not. In fact, this is a reasonable advantage from Azzurra, once the language specifies process paths in terms of correctness criteria, whereas BPMN requires a more verbose style of specification, demanding exhaustive specification of all potential process paths. It is natural that some intended process paths are not captured in the BPMN representation. Observe also the significant difference in terms of coverage between Azzurra (0.893) and BPMN (0.345) for unstructured processes. As unstructured processes potentially have a large number of process paths, this difference in terms of coverage between both languages becomes even more evident for such kind of processes.

Null Hypothesis $H_{0\text{-}2}$ (Structured vs. Unstructured): To test this hypothesis, we should use MW test which assumes the equality of variance. However, the Levene's test for homogeneity of variance returned *p-value* = 0.37 for precision and *p-value* = 0.04 for coverage. Therefore, we cannot rely on the results of the MW test for coverage. To mitigate this issue, we cross-validate the results of MW test with KW test which does not require an equal variance. The MW test results did not reveal significant difference between two process types both for precision (*p-value* = 0.35) and coverage (*p-value* = 0.11). The KW test returned *p-value* = 0.11 for coverage, which supported the results of MW test. In order to achieve statistically significant results for coverage with 80% power, we would need a sample size of 54 participants. The results show that the process type did not affect the performance of the participants. The null hypothesis $H_{0\text{-}2}$ cannot be rejected for any of the variables.

Null Hypothesis $H_{0\text{-}3}$ (Language & Process Type): The results of KW test revealed a statistically significant effect of the combination of language and process type on coverage ($\chi^2(3) = 15$, *p-value* = 0.002) and no effect on precision (*p-value* = 0.44). Therefore, the null hypothesis $H_{0\text{-}3}$ can be rejected only for coverage. A post-hoc test using MW test with Holm correction showed the significant differences between coverage of the results produced by participants who used BPMN on unstructured process and other participants who used BPMN on structured process (MW test results: *p-value* = 0.002, Cohen's d = 3.23) or Azzurra on unstructured (*p-value* = 0.003, Cohen's d = 4.02) and structured process (*p-value* = 0.002, Cohen's d = 3.23). It means that there is a significant difference in terms of coverage between Azzurra and BPMN for unstructured processes, as described above, whereas for structured processes both Azzurra and BPMN have equal performance in terms of coverage.

9.2.3 Experiment Discussion

Our aim is to investigate the suitability of the Azzurra language for representing unstructured processes and its superiority in terms of model quality in relation to BPMN. In our approach, model quality is measured in terms of *precision* and *coverage*, two metrics extracted from the field of Ontology Engineering for the evaluation of ontology quality. Regarding our propositions introduced in Section 9.2, our findings suggest that:

P1. The Azzurra modeling language is significantly better than BPMN in terms of coverage for the representation of unstructured processes, but the power of the test is not enough to completely reject null hypothesis H_{0-1} (see the discussion of null hypothesis $H_{0\text{-}1}$).

P2. No definite conclusion can be drawn, due to the absence of statistically significant difference between the two modeling languages with respect to precision (see the discussion of null hypothesis $H_{0\text{-}1}$).

The superiority of Azzurra over BPMN in terms of coverage for unstructured processes can be explained by the representational style of Azzurra and BPMN: Azzurra requires correctness criteria to be specified as commitment's consequents, whereas BPMN imposes the need of exhaustive specification of all activities and paths. First, if we consider the advantage of Azzurra over BPMN in terms of coverage (by measuring how many intended paths are captured by its corresponding representation), an Azzurra representation "covers" more paths than its counterpart in BPMN, as Azzurra's correctness criteria captures all possible paths in an implicit way as opposed to explicitly capturing all paths. Therefore, there is a higher chance that some paths are indeed forgotten during the modeling process in a BPMN representation.

Second, considering Azzurra's suitability for unstructured processes, these processes are characterized by an "on-the-fly" creation of activities, lacking also a pre-defined execution order among activities. Therefore, their textual description allows several interpretations regarding the potential paths to be captured (e.g., for three activities A, B and C, it is possible to capture 3! paths). Azzurra's features can cope better than BPMN with both aspects of unstructured processes: via commitments, modelers can specify obligations to be fulfilled and participants can dynamically select which activities to perform to fulfill such obligations at runtime. Further, a commitment-based representation also allows one to specify lack of structure necessary for unstructured processes, refraining from capturing a specific order to fulfill them. Differently, as we have noticed during the evaluation of experiment's results, students commonly captured only the most trivial sequence of activities in BPMN, missing all the other possible interpretations according to the natural language description.

Our experimental evaluation considered the metrics of *precision* and *coverage* to determine the quality of models representations in terms of domain faithfulness and language expressiveness, rather than the focusing on the modelers' perception. To overcome this issue, we distributed a questionnaire among participants. In this survey, there is a significant preference of BPMN in relation to Azzurra. This answer should be interpreted with care for two reasons. First, the questionnaire revealed prior process modeling experience of subjects in BPMN both in academia and industry. Second, imperative process modeling has its roots in imperative and declarative computer programming languages which have been used in computer science since the 50s and 60s. Third, there is evidence that imperative languages are more understandable than the declaratives ones [152]. As familiarity is a very important aspect for the usability of modeling languages, preference of BPMN seems to natural in this case.

Although we effectively conducted the experiment with a homogeneous group of master's students, some limitations must be considered. In particular, the relatively low number of experimental subjects constitutes a limitation in terms of statistical significance of our conclusions. Moreover, while BPMN models have been produced on the basis of a professional tool, the usage of a prototypical implementation of the Azzurra modeling tool may be also considered a disadvantage in relation to its respective counterpart in BPMN models.

9.3 In-Depth Evaluation with Novices

Within the third evaluation phase, the first author of this thesis together with supervisors have supervised three master students (Paul Ssekamatte [188], Melkamu Emiru [193] and Shumet Nigatu [190] in a quality comparison between Azzurra and BPMN modeling frameworks with respect to expressiveness, usability and comprehensiveness criteria. The main goal of this phase is to evaluate the overall quality of the Azzurra and BPMN modeling frameworks in terms of the achievement of these requirements. The evaluation process can be summarized into three main steps (depicted in Figure 9.5):

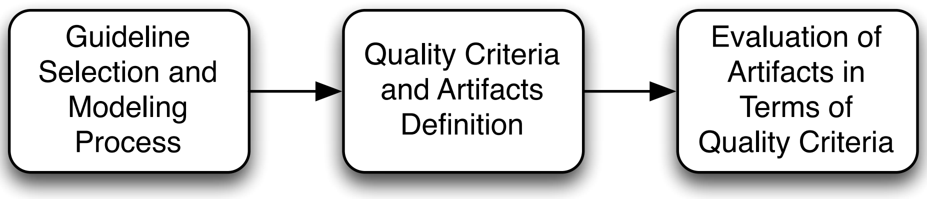


Figure 9.5: Steps of Third Evaluation Phase with Master Students (Novices)

In the remainder of this section, we detail the execution of each of these steps.

9.3.1 Guideline Selection and Modeling Process

Contrasting with evaluation phase 1 in which the domain requirements from the TIA clinical guideline (scenario 2, Section 9.1.2) have been acquired from an imperative style CG representation, each master student here has received a natural language description of three different clinical guidelines (namely, Australian malaria, lung cancer and asthma guidelines).

In order to exemplify the nature of the textual guidelines, Figure 9.6 depicts natural language descriptions stemmed from the lung cancer clinical guideline [150, 193]. In this case, the clinical guideline for diagnosis process contains a semi-structured representation of key phases of the process (colorful figure), whereas the clinical guideline for treatment (black-and-white figure) consists of purely natural language descriptions. Both types of descriptions have been used by master students and subsequently by the first author of this thesis to discover the process logic from the guidelines. Such natural languages descriptions of clinical guidelines are usually called clinical practice guidelines (CPGs) in the medical literature, while their formalization into some computer interpretable format are called computer-interpretable guidelines (CIGs) [151].

On the basis of these textual guidelines, each master student elaborated a corresponding Azzurra and BPMN process representations. For the elaboration of Azzurra models, students used the plug-in[4] developed at University of Trento, while commercial tools[5] have been used for the elaboration of BPMN models.

The modeling process has been carried out in three iterations of regular meetings involving the students and (co)-supervisors in order to discuss modeling decisions and how to capture certain domain features using both

[4] `https://trinity.disi.unitn.it/azura/azura/`

[5] Signavio: `https://www.signavio.com` [188, 193] and Bizagi: `https://www.bizagi.com` [190]

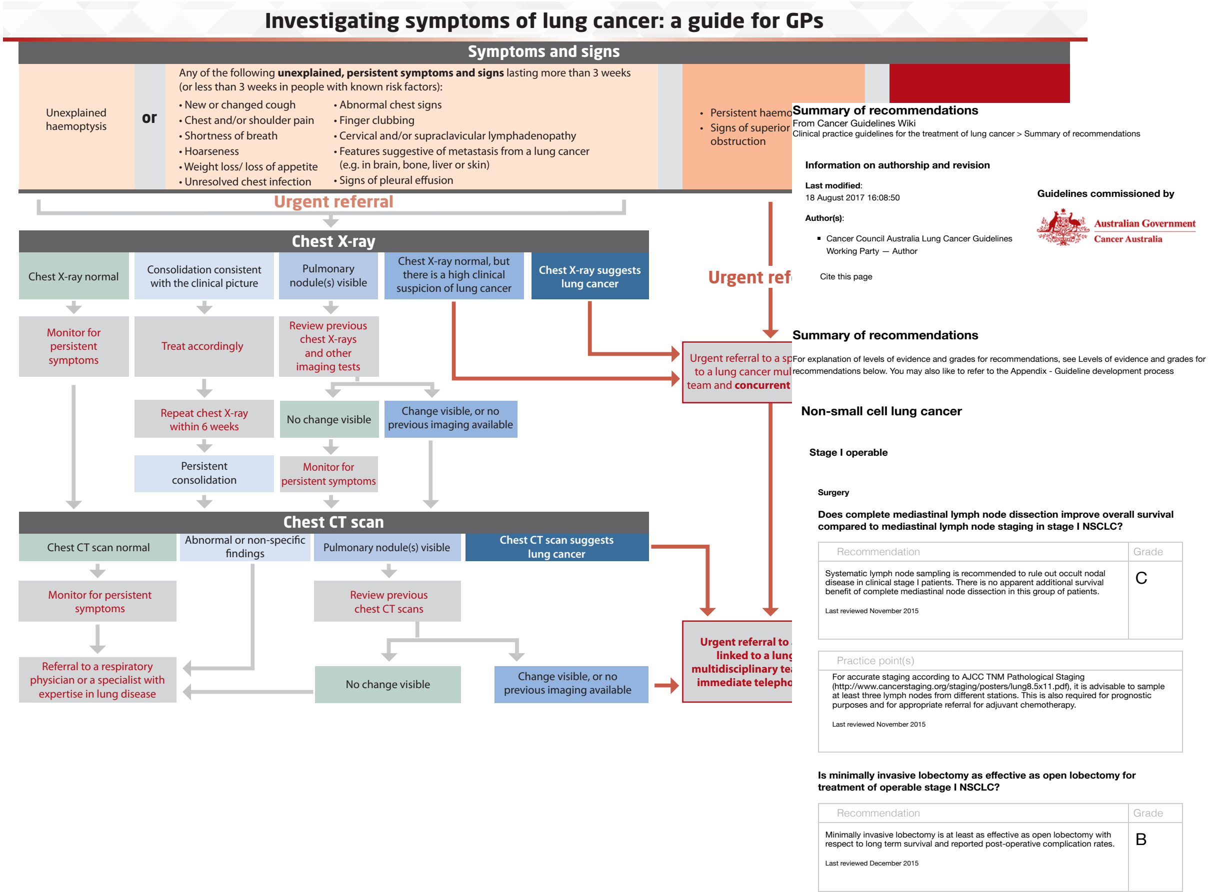


Figure 9.6: An Excerpt of Natural Language Descriptions of Lung Cancer Guidelines [150]

process languages. Once the refinement of models has been completed, students carried out peer evaluation on the models with the aim of validating their correctness and consistency with the guidelines and their correctness in terms of the usage of the modeling constructs of both languages.

In total, we summarize the statistics concerning the results of this modeling effort into Tables 9.4, 9.5 and 9.6:

The resulting models are depicted in Figure 9.7, 9.8 and 9.9. Figure 9.7 presents a BPMN representation of the small cell lung cancer guideline [193], whereas Figures 9.8 and 9.9 depicts its Azzurra counterparts, with Figure 9.8 depicting the Azzurra Social View and Figure 9.9 depicting the Azzurra Protocol View.

Paul Ssekamatte [188]				
Clinical Guidelines	**Azzurra**		**BPMN**	
	# Roles	**# Commitments**	**# Tasks**	**# Flows**
Asthma diagnosis in adults and children	5	18	18	19
Management of acute asthma in children	7	45	60	88
Management of acute asthma in adults	6	30	39	50
Initial management of life threatening, asthma in children	6	27	20	28
Initial management of life threatening, asthma in adults	6	24	16	21

Table 9.4: Paul Ssekamatte's Statistics [188] (CGs with 44 Pages in Natural Language)

Melkamu Emiru [193]				
Clinical Guidelines	**Azzurra**		**BPMN**	
	# Roles	**# Commitments**	**# Tasks**	**# Flows**
Diagnosis	16	15	15	30
NSCLC treatment	5	46	60	50
SCLC treatment	4	15	15	5
Follow Up	2	18	20	10
Preventive Care	2	20	20	5

Table 9.5: Melkamu Emiru's Statistics [193] (CGs with 24 Pages in Natural Language)

9.3.2 Quality Criteria, Artifacts Definition and Evaluation of Artifacts in Terms of Quality Criteria

In order to compare both modeling frameworks (Azzurra and BPMN), expressiveness, usability and comprehensiveness criteria have been selected from literature [62, 59] and the artifacts (e.g. languages, tools, etc.) in which such quality criteria are measured have been also defined. In this

Shumet Nigatu [190]				
Clinical Guidelines	Azzurra		BPMN	
	# Roles	# Commitments	# Tasks	# Flows
Initial Management of Malaria	3	24	38	22
Ward Monitoring and Discharge Plan	3	20	27	10
Public Health Response	6	18	38	21

Table 9.6: Shumet Nigatu's [190] (CGs with 22 Pages in Natural Language)

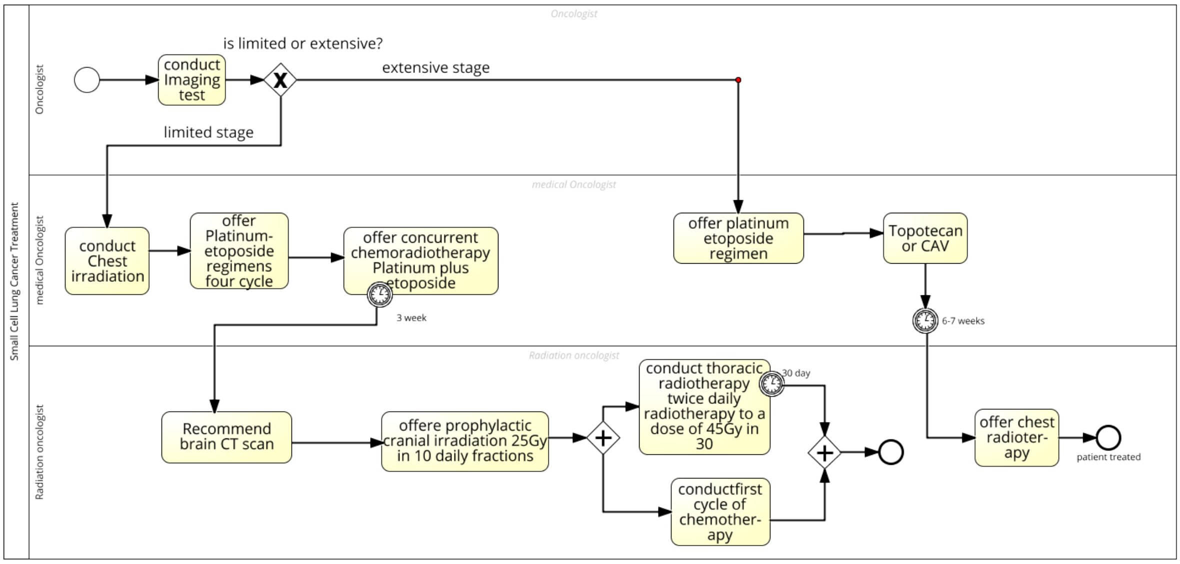


Figure 9.7: The BPMN Representation of Small Cell Lung Cancer Guideline [193]

context, expressiveness refers to Azzurra and BPMN modeling languages, usability refers to both modeling languages and modeling tools and comprehensibility refers to overall Azzurra and BPMN modeling frameworks (including modeling languages, tool support, documentation and modeling guidelines). In the following, we provide a definition of expressiveness, usability and comprehensiveness criteria and the metrics used to measure in which extent both approaches meet the desired criteria:

1. **Expressiveness [62]** is defined as the ability of a language to capture

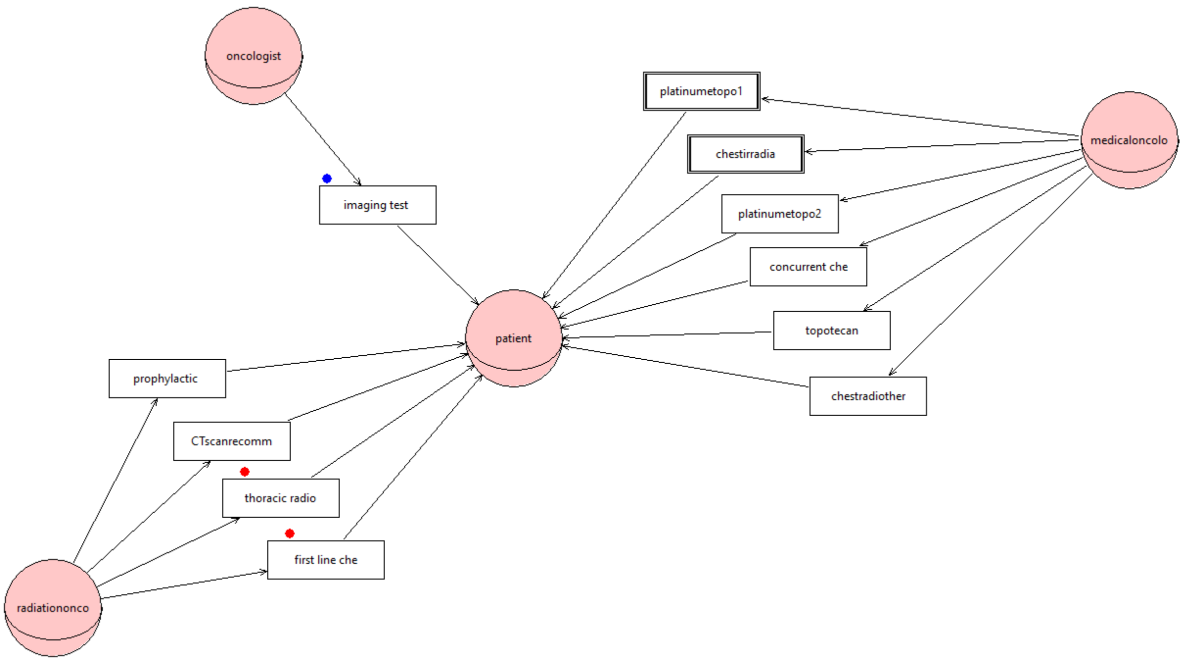


Figure 9.8: The Azzurra Representation of Small Cell Lung Cancer Guideline (Social View) [193]

```
protocol SCLCtreatment(key keyName, FirstAgent : firstAgent, SecondAgent : secondAgent){
agent-variable :
            radiationonco : varID, patient : varID, medicaloncolo : varID, oncologist : varID;
commitments :
         init -> C1: c(oncologist,patient, true, stageidentified)
         imagingtestconducted -> C2: c*(medicaloncolo,patient, limitedstagecancerfound, chestirradiation)
         chestirradiation -> C3: c(medicaloncolo,patient, true, platinumetopo2)
         platinumetopo -> C4: c(medicaloncolo,patient, true, chemoradiooffered)
         imagingtestconducted -> C5: c*(medicaloncolo,patient, extensivestagecancerfound, platinumetoposideoffered)
         platinumetoposideoffered -> C6: c(medicaloncolo,patient, true, topotecanoffered)
         concurrentche -> C7: c(radiationonco,patient, true, CTscanrecommended)
         CTscanrecomm -> C8: c(radiationonco,patient, true, prophylacticoffered)
         prophylactic  -> C9: c(radiationonco,patient, true, thoracicradiooffered)final
         prophylactic  -> C10: c(radiationonco,patient, true, firstlinecheoffered)final
         topotecanoffered -> C11: c(medicaloncolo,patient, true, chestradiotherapyoffered)
         deadline (C4, 3 weeks)
         deadline (C9, 30 days)
refinements:
kb:
}
```

Figure 9.9: The Azzurra Representation of Small Cell Lung Cancer Guideline (Protocol View) [193]

information about a certain domain of discourse. The expressiveness of a language can be measured by taking a benchmark of relevant domain concepts and mapping this benchmark to the concepts of the language defined by its meta-model. Misalignments in the mapping may reveal absence or excess of modeling constructs [62]. In our evaluation, the domain of discourse is the clinical guidelines and the benchmark of relevant domain concepts are their representational requirements imposed on both process modeling languages. Table 9.7 depicts such representational requirements (benchmark of relevant concepts) and the concepts provided by both process languages for their representation.

Expressiveness			
	Representational Requirements from Clinical Guidelines	**Concepts of Azzurra**	**Concepts of BPMN**
1	Recommendation	Commitment's consequent	Activity
2	Patient's state (context or pre-conditions)	Conjunctions and disjunctions in commitment's antecedent	Gateway and labeled control-flow links
3	Triggering events (context or pre-conditions)	Commitment's triggering event	Events (labeled elements)
4	Organizational context (context or pre-conditions)	Conjunctions and disjunctions in commitment's antecedent	Gateway and labeled control-flow links
5	Involved roles (context or pre-conditions)	Roles, agents, delegations (as commitment's refinements)	Pools and lanes
6	Ordering constraints between recommendations	Match between commitments' consequent and antecedent	Control-flow links between activities
7	Negative recommendations	Negative commitment's consequent	-
8	Recommendation's intention	Operational Goal (from SIENA)	-

Table 9.7: Evaluation of Achievement of Expressiveness Requirement

Evaluation (Expressiveness). The leftmost column of Table 9.7 depicts the CGs representational requirements. Such benchmark of relevant concepts have been discovered by the three master students in the context of their work and subsequently identified by the first author of this thesis in a subsequent evaluation. In the remainder of this discussion, we follow the ordering of lines from Table 9.7 to discuss the representation of such requirements.

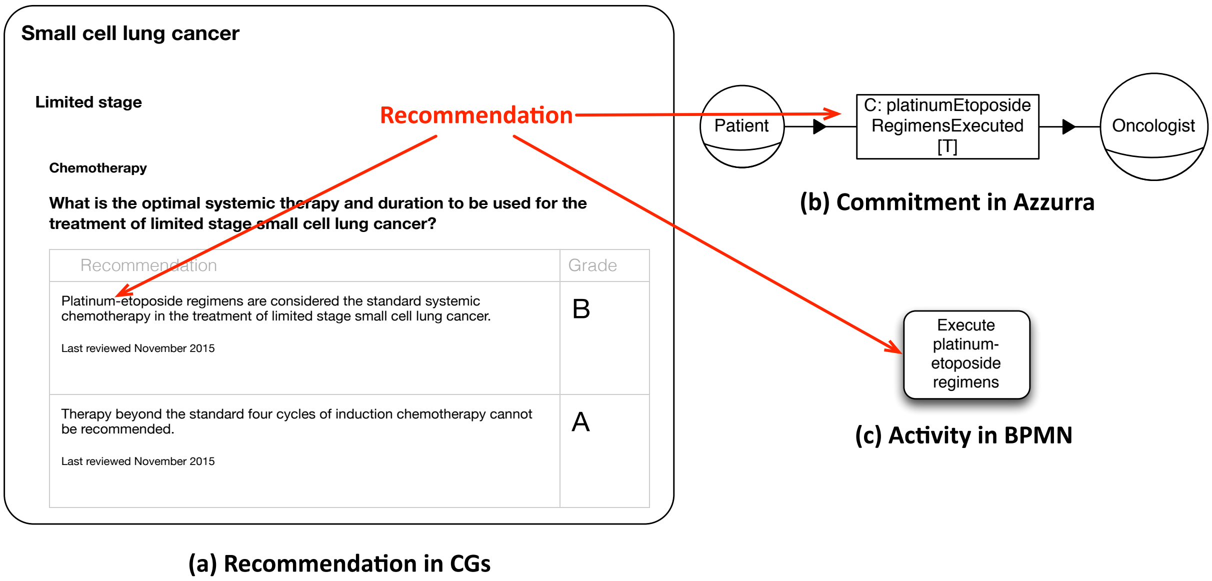


Figure 9.10: An Excerpt of Natural Language Descriptions of Lung Cancer Guidelines [150] (a), Representations of Recommendations in Azzurra (b) and BPMN (c)

The central CG concept consists of *recommendations* to be executed by healthcare practitioners (line 1) (as also discovered in the second scenario of evaluation phase 1 (TIA CG, Section 9.1.2)). Figure 9.10(a) depicts an excerpt of the natural language small cell lung cancer guideline which shows a recommendation for the treatment of cancer lung. In the context, such *recommendations* have been represented by commitments' consequent in Azzurra (Figure 9.10(b)) and activities in BPMN (Figure 9.10(c)).

In CGs, healthcare practitioners select the most appropriated recommendations to execute based on a given *context* (or pre-conditions). Such context is characterized by [151]: (i) the patient state (e.g. patient has fever or not) (line 2), (ii) triggering events (e.g., an abnormal test result obtained for a patient or the event of completion of a deadline) (line 3), (iii) organizational setting context (e.g. availability or absence of particular diagnosis/treatment technique) (line 4) and (iv) involved organizational roles (e.g. either a rheumatologist or a nurse may execute the recommendation) (line 5).

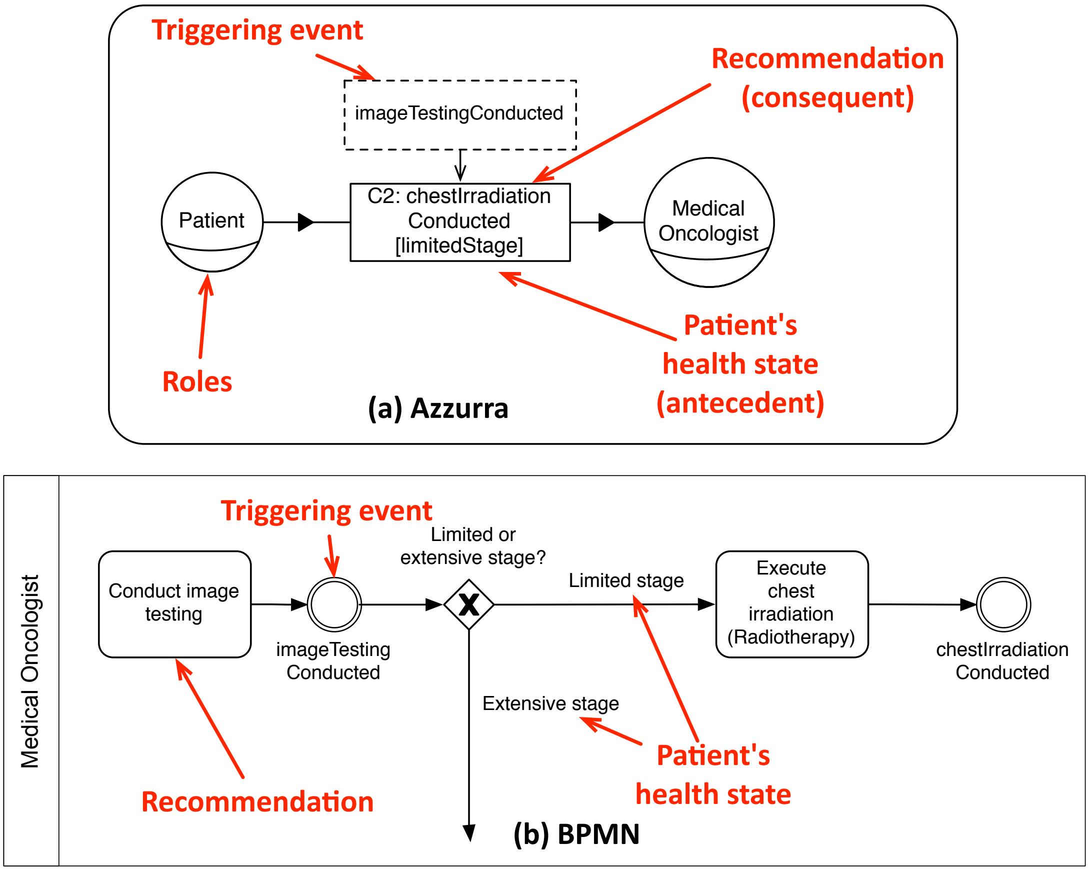


Figure 9.11: Representation of Commitment's Antecedent (patient's health state) in Azzurra (a) and BPMN (b) [150]

For the representation of CG context, Azzurra provides native support for the representation of many its elements. First, *patient's health state* and *organizational context* can be represented as a set of conjunctions and disjunctions in commitment's antecedents. For example, within the CG natural language description (Figure 9.10(a)) the recommendation under consideration assumes the patient to have a small cell lung cancer in a "limited stage" state to be applicable (for "extensive stage" state cancer, there is another recommendation). In Azzurra (Figure 9.11(a)), the patient's health state ("limited stage") is represented as a commitment's antecedent. This decision can be accounted by the fact that commitment's antecedent models that if some proposition is brought about (in this case, some patient's health state), then the commitment's consequent should be brought about. As the commitment's consequent consists of the recommendation under consideration ("conduct chest irradiation"), the semantics of this model state that if the patient has a "limited stage lung cancer", then the CG recommends the medical oncologist to "conduct a chest irradiation". The same modeling situation is represented in BPMN (Figure 9.11(b)) by including a gateway ("Limited or extensive stage?") to represent a decision performed by the medical oncologist to check whether the cancer is in "limited stage" or "extensive stage" (represented as labelled control-flow links).

Second, *triggering events* that might happen in the real world can also be captured by Azzurra as commitment's triggering events (e.g. the event of completion of an abnormal test result obtained for a patient or the event of completion of a deadline). In Figure 9.11(a), the completion of an exam of image testing (*imageTestingConducted*) to check the stage of the cancer is represented as a triggering event in Azzurra[6]. For the representation of events in BPMN, the language offers a number of different types of events (e.g. timed, message, handling or triggering compensation events, among others), but they are also labeled elements in the language. Figure 9.11(b) shows the corresponding representation of the completion of the "conduct image testing" recommendation as a labeled event "image

[6]Although it is not possible to represent triggering events in current version of Azzurra's graphical syntax (only in the Protocol View), Figure 9.11 depicts the *imageTestingConducted* event for illustrative purposes. In future work, we intend to make improvements in Azzurra's graphical notation.

testing conducted".

In terms of the involved *roles* within the CG execution, Azzurra supports the representation of roles, agents and commitment's delegations (represented as commitment's refinements). Figure 9.11(a) shows the patient and medical oncologists as CG involved roles, with "patient" as the creditor (the agent who receives the commitment) and "medical oncologist" as the debtor (role who is committed) in the Azzurra specification. BPMN also supports the representation of the CG involved roles in terms of pools and lanes, but there are no mechanisms in the language for specifying delegations among activities. Figure 9.11(b) shows the medical oncologist represented in a BPMN pool as the responsible for the execution of activities (recommendations).

Another important aspect of the CG representation regards the expression of *ordering constraints between recommendations* (Table 9.7, line 6). For example, in the natural language version of lung cancer guideline is written: *"It is appropriate to obtain a brain CT scan before embarking on prophylactic cranial irradiation, to exclude pre-existing brain metastases. If brain metastases are detected then a palliative rather than a prophylactic dose of whole brain radiotherapy may be delivered." [150].* In Azzurra, the representation of ordering constraints among recommendations is represented by matching the commitment's consequent $C_A(cred, deb, P, \mathbf{Q})$ with the commitment's antecedent $C_B(cred, deb, \mathbf{Q}, R)$ in order to denote that commitment C_A activates commitment C_B. In BPMN, such ordering constraints are represented by connecting control-flow links among activities (e.g. A $\twoheadrightarrow$ B to denote that activity A activates activity B). Figure 9.11(b) depicts the activation of the "Execute chest irradiation (radiotherapy)" activity by the link stemming from the gateway.

As argued in Section 9.1.2, guidelines' recommendations are produced in an empirical, systematic and evidence-based process, most of the times resulting in care actions to be performed in face of certain patient's health state. However, in the course of producing such statements (recommendations), guidelines developers also commonly identify care actions to be avoided (*negative recommendations*) (Table 9.7, line 7). For example, for a lung cancer in stage III inoperable, the lung cancer guideline [150] says: *"What are the principles of radiation therapy in the definitive management of stage III inoperable non small-cell lung cancer (NSCLC)?" Elective nodal irradiation is not recommended."*. Azzura represents such negative recommendations by means of negative commitment's consequent, such as $\neg ElectiveNodal\ Irradiation$ (in the same spirit of the TIA CG, Section 9.1.2, with the recommendations "Provide aspirin" ($AspirinProvided$) and "Do not provide aspirin" ($\neg AspirinProvided$)). Figure 9.12 depicts such negative recommendations modeled in Azzurra. Unfortunately, BPMN does not provide support for the representation of negative recommendations (i.e., activities that should *not* be executed).

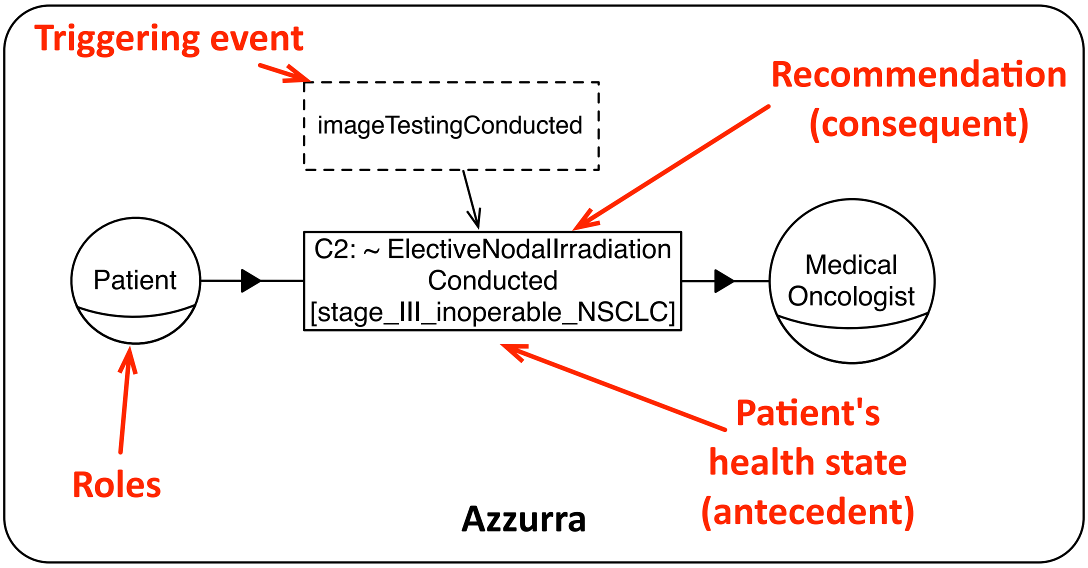


Figure 9.12: An Excerpt of Natural Language Descriptions of Lung Cancer Guidelines [150]

Each CG recommendation to be executed or avoided is provided with a certain intention (i.e., goal) in mind (Table 9.7, line 8). An example of recommendations' goals can be found in the excerpt of CG lung cancer previously described in this section: *"It is appropriate to obtain a brain CT scan before embarking on prophylactic cranial irradiation, to exclude pre-existing brain metastases. If brain metastases are detected then a palliative rather than a prophylactic dose of whole brain radiotherapy may be delivered." [150].* In other words, the excerpt states that the medical oncologist has to "provide brain CT scan exam" in order to "exclude pre-existent brain metastases". Figure 9.13 depicts this recommendation with its associated goal in Azzurra. Although the language does not provide direct support for the representation of operational goals, its association with the SIENA language and the methodological support for the derivation of commitments from operational goals (Chapter 7, Section 7.4) can be considered a beneficial aspect of the Azzurra language towards solving the problem of CG representation. Unfortunately, BPMN also does neither provide support for the representation of goals nor methodological support for the derivation of activities from goals.

Discussion (Expressiveness). In order to determine whether Azzurra and BPMN languages support the representation of clinical guidelines requirements from Table 9.7, we had to investigate the modeling constructs of both languages in order to select the most appropriate construct for each representational requirement. On the basis of such investigation, we have proposed the mapping proposed in Table 9.7. The mapping for BPMN language is already proposed in the current literature (e.g., see [203]), while the Azzurra mapping has been established in the context of the TIA guideline effort (Section 9.1.2). Although other CG representational requirements have been also identified in the context of the current modeling effort with students (e.g. cyclical, periodical recommendations and potential effects of drugs on patient's health state), we leave their discussion and represen-

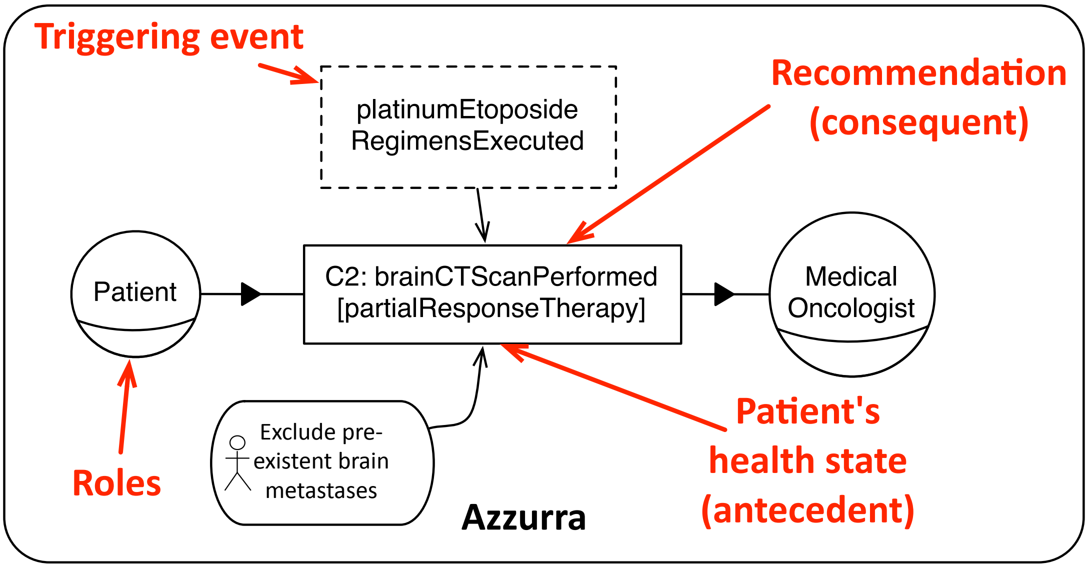


Figure 9.13: An Excerpt of Natural Language Descriptions of Lung Cancer Guidelines [150]

tation as future work.

In order to objectively measure expressiveness in our evaluation, we have graded Azzurra and BPMN with respect to the achievement of each expressiveness requirement from Table 9.7 using the following conventions:

0. No support for the requirement
1. Partial support for the requirement
2. Satisfactory support for the requirement
3. Very well support for the requirement

Table 9.8 depicts the grades obtained by each language for each expressiveness requirement. The grades achieved by each language in this table have been assigned on the basis of the aforementioned expressiveness evaluation discussion. Below, we enumerate the reasons for each grade received by both languages:

1. Azzurra and BPMN provide full support for the representation of recommendations (line 1) and ordering constraints (line 6) and therefore,

Expressiveness			
	Representational Requirements from Clinical Guidelines	**Concepts of Azzurra**	**Concepts of BPMN**
1	Recommendation	3	3
2	Patient's state (context or pre-conditions)	3	1
3	Triggering events (context or pre-conditions)	3	1
4	Organizational context (context or pre-conditions)	3	1
5	Involved roles (context or pre-conditions)	3	2
6	Ordering constraints between recommendations	3	3
7	Negative recommendations	3	0
8	Recommendation's intention	3	0

Table 9.8: Grades Obtained for Azzurra and BPMN Languages Regarding the Achievement of Expressiveness Requirement

both languages have received maximum score in the three expressiveness requirements;

2. For the representation of context in guidelines, Azzurra captures patient's health state (line 2), events (line 3) and organizational context (line 4) in terms of propositions in a knowledge base. For this reason, we considered the language to fully provide support for their representation (scoring 3), since they are natively represented in the language's knowledge base. In contrast, CG context is modeled as gateways and labeled control-flow links (events are also labeled elements) in BPMN. As labels, these elements are not natively represented in the language and it is not possible to reason with them. Therefore, BPMN has been considered to provide only partial support for the requirement (scoring 1);

3. For the representation of CG roles (line 5), Azzurra has been graded

with 3 as it fully provides support for their representation. In contrast, BPMN allows the representation of roles, but not delegations and therefore, it has been considered to provide satisfactory support for the requirement (scoring 2), but not full support;

4. As BPMN does not allow the representation of negative recommendations (line 7) and recommendation's intention (line 8), the language has been considered to provide no support for the representation of the requirement (scoring 0), while Azzurra has been considered to provide full support (scoring 3) as it allows their representation.

As can be seen from Table 9.8, Azzurra achieved the highest grades for all expressiveness requirements, while BPMN does not fully support some of them (lines 2, 3, 4, 5, 7 and 8). Consequently, we conclude that Azzurra is more expressive than BPMN for the representation of representational requirements from clinical guidelines.

2. Usability [59] measures the degree to which an artifact (system, product or service) can be used to achieve some pre-defined user's goals with effectiveness, efficiency and satisfaction within a specified context. In our case, the artifacts are Azzurra and BPMN process languages and their modeling tools. Effectiveness, efficiency and user's satisfaction (usability) measure the extent to which both languages and the used tools support modelers to develop models within both languages. In the context of our evaluation, usability is measured in terms of the metrics summarized in the leftmost column of Table 9.9:

Discussion (Usability). In Table 9.9, Azzurra and BPMN have been graded by students using the following conventions:

0. No support for the requirement
1. Partial support for the requirement
2. Satisfactory support for the requirement

Usability		
Usability Metrics	**Azzurra**	**BPMN**
Easiness of Language Learning and Use	1	2
Easiness of Differentiation Between Different Concepts	2.5	2.5
Tool Support	1	3

Table 9.9: Evaluation of Achievement of Usability Requirement

3. Very well support for the requirement

The grades achieved by each language in this table have been calculated by an average of the grades assigned by each student. As can be seen from this table, students considered BPMN easier to grasp than Azzurra (denoted by higher grade assigned to BPMN in relation to Azzurra in the *easiness of language learning and use* criteria). In terms of *easiness of differentiation of concepts* in both languages, Azzurra and BPMN have received the same score. Below, we enumerate some of the reasons why they considered Azzurra more difficult to learn and use than BPMN:

1. Ssekamatte [188] mentions the syntax and semantics of Protocol View were somehow challenging for a newcomer, requiring him some practice until fully grasping the concepts. For Nigatu [190], Azzurra models are not easy to understand and interpret, (unlike BPMN) due to the syntax and semantics in which the Protocol View is defined, especially for large models. Despite the noticed challenges, he (Nigatu) thinks that Azzurra could be usable in small modeling efforts;

2. Ssekamatte [188] and Nigatu [190] mention that lack of structure in Azzurra specifications as one of the factors that harm usability with the language. Unlike BPMN in which the sequence of process is naturally determined by the linkage of activities within the process control-

flow, Azzurra does not impose the need for explicitly capturing a sequence among commitments. As a result of that, the modeler has to explicitly search in the natural language specification for events that trigger commitments antecedents, consequents, deadlines, among others. This implies more detail to be captured per commitment, thus increasing the time required for a modeling effort using Azzurra compared to BPMN.

3. Another consequence stemmed from the lack of Azzurra structure is the difficulty in reading and navigating in Azzurra models [188, 190]. For example, since roles and agents are linked through commitments, it becomes difficult to connect two roles that are located in extreme positions in a large model. Although the tool provides the functionality of zooming in/out to make visualization of the diagram easier, the intrinsic lack of sequence in Azzurra specification makes the readability and navigation difficult;

4. As commitments in Azzurra have to be a social interaction between two roles, the language semantics requires the prior identification of two roles in which the commitments can be established. This language feature makes the representation of commitments slight difficult, especially in those situations in which the two involved roles are not clear from the natural language specification [188];

5. In relation to BPMN, Nigatu [190] cites the easiness in understanding and using BPMN concepts to positively contribute to shortening the time for a given modeling effort with the language. He also mentions an easiness in interpreting visual BPMN models.

6. Due to the complexity of clinical guidelines, the size of specifications may become huge both in BPMN and Azzurra. In BPMN, this prob-

lem can be somehow tackled by the introduction of sub-processes to split large representations. However, as Azzurra does not contain the mechanisms of sub-processes, Ssekamatte [188] suggested that Azzurra could be enhanced with mechanisms for capturing the concept of sub-process, both by creating notational elements and modeling guidelines that determine when a process should be split into sub-processes.

3. Comprehensiveness measures in which extent the overall framework encompasses artifacts that facilitate the use of the modeling approach. In the context of our evaluation, comprehensibility is measured in terms of the metrics summarized in Table 9.10:

Comprehensiveness		
Comprehensiveness Metrics	**Azzurra**	**BPMN**
Availability of Modeling Tool	1.5	3
Availability of Language Documentation (includes guidelines for modeling and for the tool)	1	3

Table 9.10: Evaluation of Achievement of Comprehensiveness Requirement

Discussion (Comprehensiveness). In Table 9.10, Azzurra and BPMN have been graded by students using the following conventions:

0. No support for the requirement
1. Partial support for the requirement
2. Satisfactory support for the requirement
3. Very well support for the requirement

The grades achieved by each language in this table have been calculated by an average of the grades assigned by each student. As can be seen from this table, students considered the BPMN approach more comprehensive than Azzurra, which is denoted by higher grades assigned to BPMN in relation to Azzurra in both criteria (*availability of modeling tool* and *availability of language documentation*). Below, we enumerate some of the reasons why they considered BPMN more comprehensive than Azzurra:

- Ssekamatte [188] and Nigatu [190] mention the availability of many BPMN tools (e.g. Signavio[7] and Bizagi[8], whereas Azzurra contains just a prototypical implementation developed at University of Trento[9]. Furthermore, BPMN tools usually provide documentation for BPMN modeling efforts, in terms of modeling patterns and tool documentation [188, 190];

- Nigatu [190] and Emiru [193] mention a lack of usability in the Azzurra tool concerning a limitation in the number of characters for naming commitments. Ssekamatte [188] argues that Azzurra tool is still under development and as a consequence of that, the tool still presents some bugs and some missing functionalities.

9.3.3 In-Depth Evaluation Discussion

This third evaluation phase intended to compare Azzurra and BPMN modeling frameworks with respect to expressiveness, usability and comprehensiveness. For that, three master students have used natural language descriptions of Australian malaria, lung cancer and asthma clinical guidelines to model them using both languages and subsequently performed an evaluation of each language.

On the basis of the evaluation discussed in this section, Azzurra is considered better than BPMN regarding expressiveness for the representation of clinical guidelines, while BPMN is considered better than Azzurra for us-

[7] `https://www.signavio.com`

[8] `https://www.bizagi.com`

[9] `https://trinity.disi.unitn.it/azura/azura/`

ability and comprehensiveness. Such conclusions have been acquired based on the grades received by both languages in Tables 9.8, 9.9 and 9.10.

In order to complement the evaluation performed by the three master students, the first author of this thesis also modeled the lung cancer guideline to understand the difficulty of performing the task. Figures 9.14 and 9.15 depict the BPMN model from the small cell lung cancer guideline resulted from this modeling effort. Differences regarding the BPMN model captured by the master student (Figure 9.7) and the first author of this thesis (Figures 9.14 and 9.15) stem from the lack of clarity regarding sequence in most of the recommendations in the clinical guideline. As a consequence of that, the master student captured only the “happy path” within the guideline which could be naturally inferred by reading the guideline in sequence, whereas the first author attempt to capture all possible sequencing combinations for recommendations. As BPMN imposes the need of explicitly capturing ordering constraints among recommendations, the model of the first author of this thesis became much larger to represent all possible sequences combinations among recommendations.

In contrast, as Azzurra does not impose any sequencing among recommendations, the Azzurra specification from the first author of this thesis does not differ from the master student, as the only point of divergence would be the sequencing among recommendations.

Besides the challenges associated with the representations in BPMN and Azzurra languages, the discovery of process logic from natural language clinical guidelines represented an additional challenge in our experience. This can be accounted by the complexity of domain knowledge associated with the natural language descriptions in the medical domain. In this context, in comparison with the evaluation conducted in the first evaluation phase (Section 9.1.2) that discovered the process logic from the BPMN representation, this third evaluation phase was more difficult.

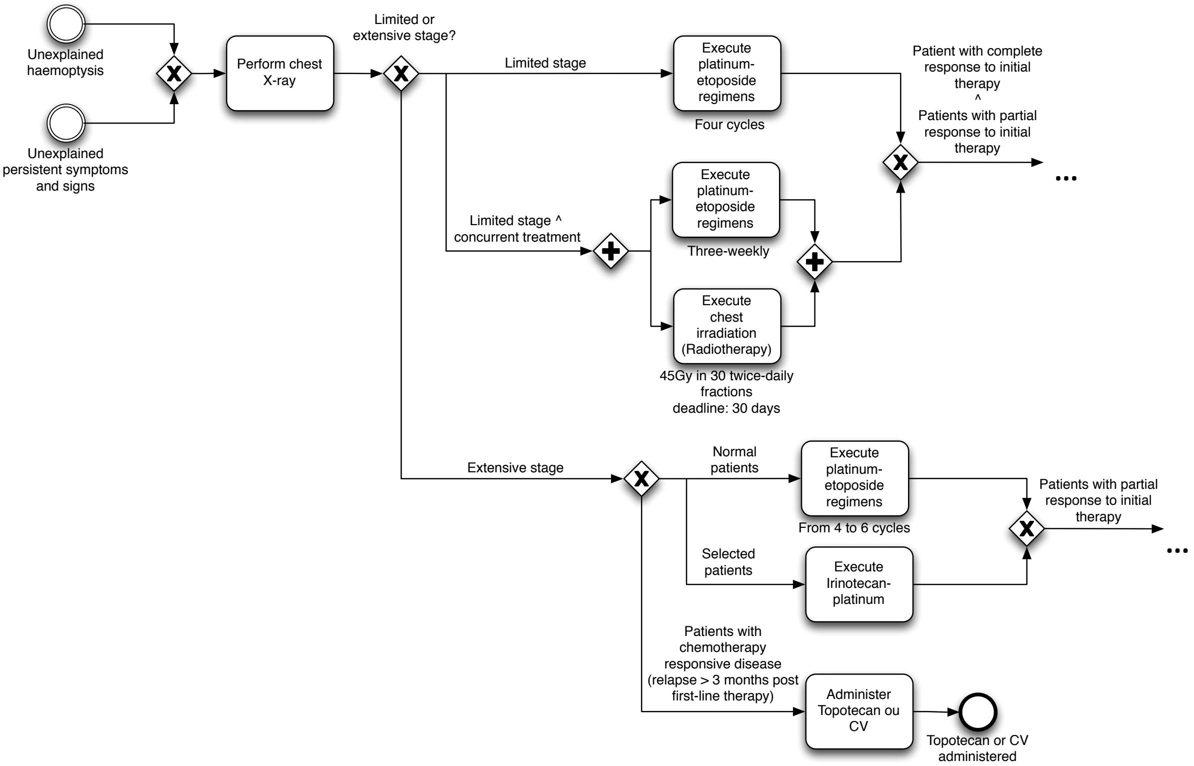


Figure 9.14: BPMN Representation of Small Cell Lung Cancer Guideline (Modeled by First Author of This Thesis)

9.4 Summary

This chapter reports three evaluation phases of the Azzurra modeling language. More specifically, the first phase reports an evaluation of the Azzurra language using two real-world scenarios from the medical domain with the purpose of illustrating Azzura features. Within the fracture treatment scenario, our purpose is to compare Azzurra's representational features with the current state of the art process modeling languages (BPMN, Declare and Artifact-centered). The second scenario uses the TIA clinical guideline to show certain domain features from medical processes that could be better supported by a commitment-based representation.

The second phase empirically evaluates the Azzurra and BPMN modeling languages for the representation of structured and unstructured pro-

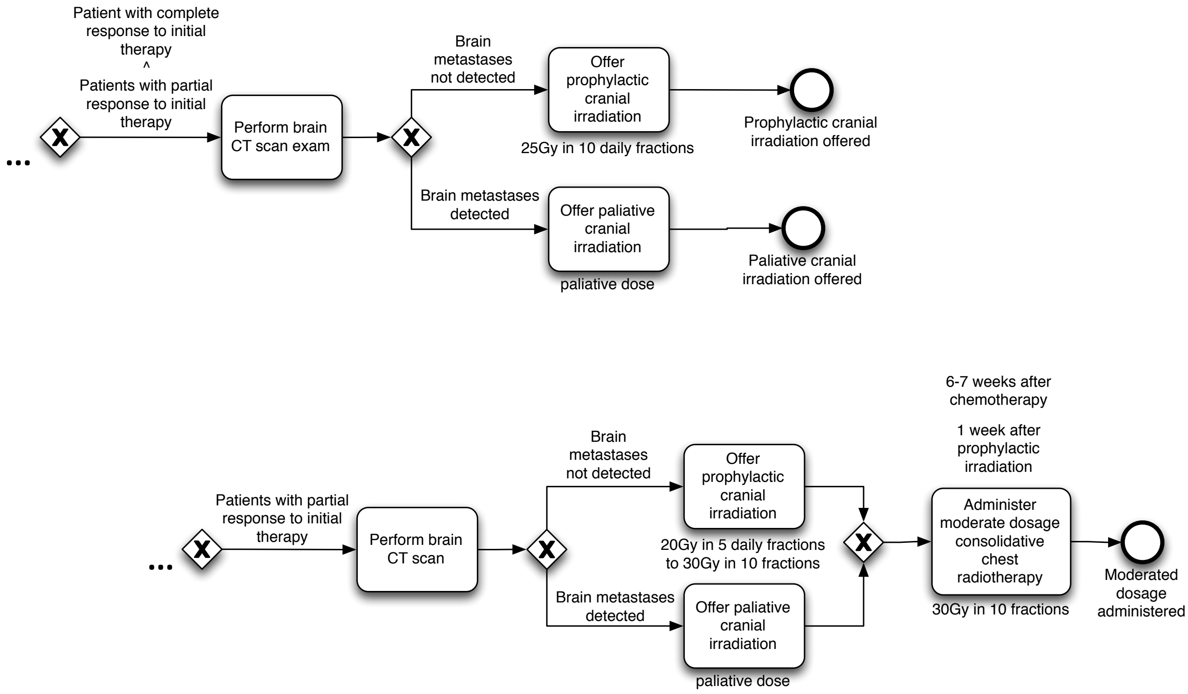


Figure 9.15: BPMN Representation of Small Cell Lung Cancer Guideline (Modeled by First Author of This Thesis)

cesses in terms *precision* and *coverage*, two metrics used in the evaluation of ontology quality in the field of Ontology Engineering. Our empirical results indicate that Azzurra can be considered superior to BPMN for the representation of unstructured processes. However, no further claims can be stated concerning the superiority of BPMN over Azzurra for the representation of structured processes.

The third phase reports on a modeling effort in which the first author of this thesis has supervised three master students [188, 193, 190] in conjunction with her supervisors in a quality comparison between Azzurra and BPMN modeling frameworks in terms of expressiveness, usability and comprehensiveness criteria. This modeling effort reveals that students consider Azzurra better than BPMN regarding expressiveness for the representation of clinical guidelines, while BPMN is considered better than Azzurra for usability and comprehensiveness.

Chapter 10

Contributions, Limitations and Future Work

In this chapter, we review the requirements for the development of strategic enterprise architectures and the challenges for their development. Subsequently, we enumerate the contributions of this thesis, including a discussion about the limitations and future work of our strategic enterprise architecture approach.

10.1 Motivations Summary

In Chapter 2, we have introduced the requirements for strategic enterprise architectures that describe the desired characteristics and components to be addressed by a strategic enterprise architecture approach. Such desired characteristics have been derived from a careful analysis of Management literature that describes the needs of supporting the overall process of management of organizations. Therefore, such characteristics are the desired properties to be embedded in an enterprise architecture approach that supports the management process of organizations.

In basic terms, organization's management process is structured into five management functions (planning, organizing, staffing, leading and controlling) in Management literature. This thesis has been particularly focused on providing support for the exercise of the planning function. In practice, the planning function consists of setting the goals the company has to achieve followed by an allocation of actions and resources intended to achieve such goals. Figure 2.1 has been introduced in Chapter 2 to depict the steps of the enterprise planning process and repeated here in Figure 10.1.

In order to support the enterprise planning process, the enterprise modeling language should be able to capture the conceptualization inherent to this phenomena. As can be observed from Figure 10.1, the enterprise planning process must deal with: (i) motivational aspects (goals of different shades that drive the overall scope of purpose of the enterprise architecture) (step 1) (ii) mechanisms for capturing uncertainties and embedding flexibility as the enterprise planning process plans today what will be realized in the future (step 2) and (iii) behavioral perspective (business processes which consists of the means by which goals are realized) (step 6).

From a methodological point of view, a strategic enterprise architecture approach must support the elaboration of goals of different shades and their characteristics accordingly. Further, the elaboration of business processes and their internal control-flow must be also supported. Full traceability between motivational and behavioral perspectives must be also ensured by means of methodological guidelines. In terms of reasoning of strategic enterprise architectures, automated reasoning techniques should support the several steps of the enterprise planning process.

In order to provide support for the representation and analysis of strategic enterprise architectures, many approaches exist in a number of areas of Computer Science. More specifically, motivational modeling is mainly addressed by Goal-Oriented Requirements Engineering (GORE) and Enterprise Modeling (EM), whereas behavioral modeling is mainly addressed

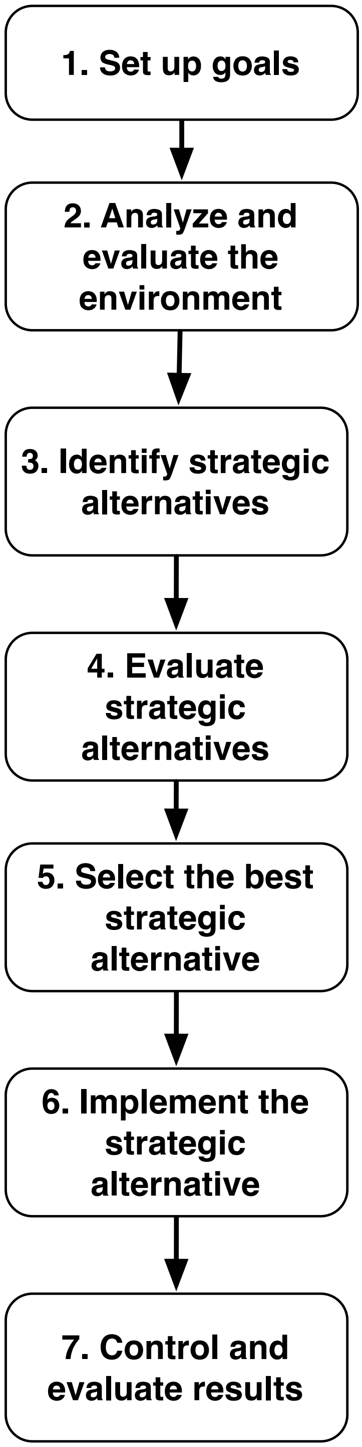


Figure 10.1: Steps in a Basic Planning Process [153] (Initially introduced in Chapter 2)

in Business Process Management (BPM). Hybrid approaches that acknowledge the benefits of integrating motivational and behavioral concepts also exist in EM and BPM. We have provided a comprehensive summary of such work in Chapter 4.

In this context, in the scope of motivational modeling, existing approaches do not recognize the existence of goals of different shades. GORE approaches recognize the importance of goals to capture stakeholders' requirements for a target software system, but the concept of goal is not further refined into different goal categories. Enterprise modeling approaches borrow GORE goals and its relations to represent enterprise's strategic concepts such as "Increase company's sales", but do not distinguish among different shades of goals. Hybrid approaches create their own goal ontologies to represent either strategic goals (e.g. "Increase sales") or operational goals (e.g. "create order") interchangeably. In Chapter 4, we have analyzed how GORE, EM and hybrid approaches meet the requirements of strategic enterprise architectures. We concluded that research in goal modeling is fragment along multiple veins in which each approach represents a type of goal, but a unique approach that encompasses the full goal hierarchy is still missing. We have also realized that none of the approaches acknowledge the existence of environmental factors that may impact the achievement of goals during organization planning.

Regarding the interconnection between motivational and behavioral perspectives, we have concluded in Chapter 4 that GORE approaches establish a simplistic relationship between both perspectives in which usually a "plan" achieves a goal, but does not refine this plan into its detailed control-flow. In contrast, EM, hybrid and BPM approaches enable the representation of business processes and their control-flows in terms of activities, but the interactions and expectations of process participants are not addressed in such approaches. Consequently, as the social perspective of processes is not captured, a seamless integration between motivational and behavioral perspectives are not provided by such approaches.

A direct consequence of a deficient support in the representation of strategic enterprise architectures is reflected in its automated reasoning as the enterprise planning process (Figure 10.1) is not fully supported by any approach. For example, reasoning techniques cannot perform analysis with goals of different shades and environmental factors. Only the BIM forward and backward reasoning techniques are able to reason with environmental factors, but cannot address reasoning with multiple levels of abstraction in goal hierarchies. Most of the approaches cannot support the generation of strategic alternatives. Only three approaches (Tropos, i* and BIM backward reasoning) can generate strategic alternatives but no reasoning with different shades of goals. Furthermore, only Lapouchnian et al. [116] and Morrison et al. [136] can support the implementation of strategic alternatives by means of processes.

On the basis of the conclusions acquired with the current state of the art, we have chosen the frameworks that meet most of the requirements for strategic enterprise architectures. A description and analysis of such approaches are reported in Chapter 3. They are: the *BIM* and *BMM frameworks*, *Management Literature*, *CGM formalism*, *Commitments and Protocols* and the *KAOS approach*.

The BIM framework presents a great match between the represented concepts and the requirements stipulated for strategic enterprise architectures. Therefore, the BIM concept of "goal" has been chosen as the starting point for creating goals of different shades together with the concept of situation and domain assumptions for capturing environmental factors that impact goal achievement. In order to refine BIM goals, the BMM specification has been used for the differentiation among the several goal types due to its rich vocabulary of goal-related concepts. However, practical efforts of refining BIM goals in terms of finer-grained goal distinctions from BMM are hindered by the absence of well-defined, concrete criteria in BMM.

In order to overcome the lack of concrete criteria in BMM and to acquire a common semantic foundation for integrating motivational and behavioral perspectives, we referred to conceptualization provided by Management Sciences in this thesis. In Management literature, our initial objective was to find clear differentiation among goals and conceptual integration between motivational and behavioral perspectives. However, we faced similar issues of unclear distinctions as in BMM. Therefore, even with models that lack precise semantics, Management Theories have been chosen as the conceptual basis in this thesis, as it makes available a rich set of conceptual tools like assumptions, scenario analysis, goal and operations planning.

Given our final interest in automatically reasoning with our conceptual models, they needed to be specified using formal rigor. In this context, we need to either assign our own formal semantics for strategic enterprise architecture models or find existing languages with already well-established semantics for reasoning. We have opted for the second solution by choosing the CGM formalism as the backend tool for our strategic planning approach. Finally, commitments and protocols have been used to specify processes in social terms, while KAOS refinement patterns and operationalization process have provided methodological support for deriving of business process's control-flow specified in Azzurra from SIENA operational goals.

10.2 Thesis Contributions

In order to provide adequate support for the requirements for strategic enterprise architectures, this thesis strived to improve the support for strategic enterprise architectures by means of languages, methodology and automated reasoning support. Below, we review the contribution of the SIENA framework.

SIENA Language (Chapter 5). In order to provide support for the representation of strategic enterprise architectures, we have presented the SIENA language which is structured in terms of Goal and Operations View in order to respectively represent motivational and behavioral perspectives.

In the Goal View, the SIENA language contributes to the current state of the art by proposing goals of different shades (mission, vision, strategic, tactical and operational goals) and *dimensional refinement operators.* Dimensional refinement operators enable the refinement of strategic goals in terms of time, location and products/services dimensions. In comparison with traditional OR-refinements that capture alternatives for achieving goals, they go beyond by enabling the specification of different alternatives to achieve strategic goals in different points of a given dimension. SIENA also borrow situations and domain assumptions from BIM in order to support the representation of environmental factors that impact the achievement of goals. In the Operations View, SIENA introduces the distinctions of operations and business processes. Operations are used to support the planning of the achievement of strategic and tactical goals, whereas business processes consist of behavior exhibited for delivering company's services and products. With such conceptual model, we have proposed a hierarchical architecture for strategic enterprise models.

In Chapter 8, we have assessed the SIENA language in terms of the achievement of the requirements for strategic enterprise architectures from Chapter 2 (Section 8.1). In this assessment, we concluded that the language meets all requirements for strategic enterprise architectures (with the exception of support for control evaluation of implemented strategic alternatives (R3.2.4 requirement)). We have also demonstrated the real-world applicability of the SIENA and Azzurra languages by modeling the real hospital scenario of our previous effort (Section 8.2). Finally, we concluded our evaluation of the SIENA framework with a comparison of our framework and the ArchiMate framework. In this comparison, we have concluded that SIENA/Azzurra languages are more expressive than Archimate and therefore, it advances the state of the art in the representation, methodology and reasoning of strategic enterprise architectures.

Azzurra Language (Chapter 7). In order to provide support for the representation of social perspective of business processes, the Azzurra language introduces the specification of business processes founded on the concepts of social commitments and protocols. Centering the representation in terms of both concepts enables the specification of correctness criteria to be achieved as commitments' consequents, rather than specific operationalizations (activities) that must be carried out. Besides both modeling constructs, Azzurra contains other business primitives that focus on the obligations that process participants have towards each other, including delegations, deadlines and role adoption constraints. By centering the representation on commitments, Azzurra advances the state of the art in process modeling with the specification of the social perspective of business processes, contrasting with the current state of the art that only captures the operational perspective. In order to unambiguously specify the language, Azzurra has been presented in terms of its syntax, semantics, graphical notation and prototypical implementation of the language.

In Chapter 9, we evaluated the Azzurra language using two real-world scenarios from the medical domain (Section 9.1). In the first scenario, we compare Azzurra's representational features with the current state of the art of process modeling languages, whereas the second self-evaluation highlights certain domain features of the scenario that could be better supported by a commitment-based representation. The second phase (Section 9.2) reports on an experiment conducted with master students to investigate the suitability of Azzurra and BPMN for the representation of structured and unstructured processes in terms *precision* and *coverage*, two metrics used in the evaluation of ontology quality in the field of Ontology Engineering. Our empirical results indicate that Azzurra can be considered superior to BPMN for the representation of unstructured processes. However, no further claims can be stated concerning the superiority of BPMN over Azzurra for the representation of structured processes. Finally, the third phase (Section 9.3) reports on a modeling effort in which the first author of this thesis has supervised three master students [188, 193, 190] in conjunction with her supervisors in a quality comparison between Azzurra and BPMN modeling frameworks in terms of expressiveness, usability and comprehensiveness criteria. The results of this modeling effort revealed that students consider Azzurra better than BPMN regarding expressiveness for the representation of clinical guidelines, while BPMN is considered better than Azzurra for usability and comprehensiveness.

Methodology for Strategic Enterprise Architectures (Sections 5.2 and 7.4). Methodological guidelines for strategic enterprise architectures have been proposed in Section 5.2 and 7.4. In Section 5.2, these methodological guidelines specify how to elaborate goals of different shades and how to achieve traceability between motivational and behavioral perspectives by deriving operations from tactical goals and business processes from operational goals. In Section 7.4, guidelines specify how to derive process's control-flow specified in Azzurra from SIENA operational goals.

Strategic Planning Approach (Chapter 6.) In order to provide support for automated reasoning in strategic enterprise architectures, Chapter 6 initially proposes a formal representation of strategic goals, their AND/OR refinements and dimensional refinement operators. Subsequently, this formalization is used as input in a strategic planning technique that supports the enterprise planning process from Chapter 2. Such strategic planning technique allows the generation of optimum strategic plans to achieve strategic goals on the basis of objective functions, constraints and scenarios.

Chapter 6 also evaluates this strategic planning technique by performing tests with the CGM tool using different types of SIENA models. Regarding the achievement of the requirements for strategic enterprise architectures (Chapter 2), our strategic planning approach fully addresses support for automated reasoning in strategic enterprise architectures (with an exception for control and evaluation of implemented strategic alternatives). In this context, the initial formalization step provides formal rigor for SIENA's specifications and enables the reasoning with goals in multiple levels of abstraction. With the usage of the CGM tool, our strategic planning approach allows the identification of strategic alternatives and generation of optimum strategic alternatives (strategic plans) that explore enterprise variability and constraints. The mapping of situations and domain assumptions to CGM modeling constructs also enables the realization of scenario analysis which corresponds to the assessment of environment and their impacts on the achievement of strategic goals.

Overall, the language, methodology and reasoning proposed in this thesis can support business managers in formalizing strategic planning activities. In the current state of the art of Management Sciences, such strategic planning activities have informal methodological support (mainly textual) and no automated support for their development. In particular, on the basis of the contributions here proposed, we believe that the conceptualization proposed by the SIENA Language together with its methodological guidelines can contribute to make enterprise modeling and reasoning more practical in a number of ways. First, by providing key motivational and behavioral distinctions together with concrete guidelines for model elaboration, SIENA can help business managers to initiate discussions regarding which outcomes (goals) to achieve, which strategies to adopt in order to succeed in business and in which extent the elaborated business goals are achievable. The use of dimensional refinement operators can support distinct business analysis according to different dimensions (e.g. products/services, time and location) of the enterprise. Second, our strategic enterprise models can be used as an overall guidance not only with respect to which outcomes to achieve (goals), but how to realize them in terms of operations/business processes. Third, our strategic planning technique can provide concrete strategic plans to achieve strategic goals. As our automated technique requires numeric values (costs, time, etc.) to be attached to operations, this drives managers to numerically estimate the overall attributes of implementing certain strategies under the likelihood of certain business scenarios. With this estimation in hands, they can use the automated reasoning technique in order to adopt the strategic options with most advantages (e.g. cheapest strategic plan or the fastest strategic plan) in their corresponding business scenario.

10.3 Limitations

With the introduction of the SIENA framework described along this work, we have significantly advanced the current state of the art in representation and reasoning in strategic enterprise architectures. However, our approach presents a number of limitations that we enumerate in the remainder of this section.

Completeness of Requirements for Strategic Enterprise Architecture. In Chapter 2, we have used the research questions introduced in Chapter 1 to refine them in terms of the requirements for strategic enterprise architectures. Although we believe that such requirements present the prominent features of a given strategic enterprise architecture approach, they may be not necessarily complete in all settings. More specifically, existing requirements may become more or less necessary depending on the characteristics of certain domains or even new requirements may be introduced. For example, in the metal manufacturing company used throughout this thesis, an emphasis was put on the elaboration of strategic goals with a focus on competitive requirements. In contrast, competitive requirements were not the focus for the hospital setting modeled in Chapter 8 due to the non-profit (public) characteristic of the company. Despite the specificity of requirements required for specific domains, we still believe that the requirements enumerated in this thesis reach an effective level of generalization of the needs of strategic enterprise architectures because they have been derived from Management literature.

Absence of Bottom-Up Methodological Guidelines. In Chapter 5 (Section 5.2), we have provided methodological guidelines on how to elaborate goals of different shades and how to derive behavioral elements from motivational elements in a top-down fashion. Although these guidelines in principle are useful for keeping traceability between motivational and behavioral perspectives, they ignore the strategy formation that might emerge without a prior planning in a bottom-up fashion [135]. Furthermore, different strategy formation schools in Management literature defend distinct ways in which strategies are formed [135]. Therefore, new methodological guidelines should be developed taking such strategy formation processes into account.

Goal Modeling Limitations. SIENA Models (Chapter 5) share many shortcomings with goal models, such as complexity and lack of scalability [94]. Although our methodological guidelines strive to improve model elaboration and understanding, they can only partially alleviate their complexity and lack of scalability. if the SIENA model becomes too complex, the methodological guidelines can guide the final users to read the model, but cannot support in tackling the inherent model complexity, especially if the final user is not familiar with the domain. In this context, the development of modules [94] in the area of goal model scalability could support in tackling goal modeling complexity in SIENA.

Limitations in Strategic Planning Technique. In Chapter 6, although our approach has advantages over traditional goal analysis, it still presents some limitations. First, the numeric treatment for the distribution of weights (targets) of sub-goals is still very simplistic and requires an extensive work in the definition of rules for such distribution based on the type of operator (AND or OR) for each type of refinement dimension (time, location, product/service). Second, as strategic goals refinements are manually performed, the approach lacks scalability in modeling if the size of the model grows. In order to cope of both shortcomings, in particular, we envision that the semantics of drill-down/roll-up operations from data warehouses [192] can be further explored for the automatic generation of goal refinements and for checking the consistency of such refinements in future work. Third, the CGM formalism used for providing the formal semantics for SIENA language limits the use of constraints to only linear constraints (e.g. $x*a + y*b$) and therefore, more complex constraints during strategic planning cannot be used. Fourth, as CGM only shows only one optimum solution per time, this is also a limiting factor for our approach.

Usage of Formal Models in Business Context. In Chapter 6, we have mapped SIENA strategic, tactical goals and situations to the CGM formalism. This step has been performed with the purpose of providing formal rigor to SIENA models and subsequently enabling automated reasoning with them. In Chapter 7 (Section 7.4), we had the same approach by mapping operational goals and their refinements to KAOS semantics (LTL formalization and goal patterns) in order to enable the checking of correctness and completeness of goal refinements. Although the ultimate goal of our SIENA framework is to enable business representation and analysis for business managers, the formal nature of goal models in both approaches hinder their understanding, consisting in a very complex artifact for such type of users. However, in order to alleviate this problem, our approach initially builds informal goal models in Chapter 5 and subsequently maps such informal models to CGM. We believe this initial step may support final users to get an initial contact with goals models and their refinements, thus facilitating their understanding. The same considerations may be also applied to Azzurra models.

Lack of Tool Support for SIENA Models. Although our SIENA framework (Chapter 5) provides a strategic enterprise language suited for business analysis, this language is not tool-supported. In this context, the mapping to the CGM formalism intended to assign formal semantics for reasoning for our strategic enterprise models. In order to tackle this problem by developing a tool for SIENA models, two approaches may be considered. The first one may consider the development of a tool suite composed by SIENA and Azzurra modeling tools, whereas the second approach considers the development of two distinct modeling tools. In order to decide which solution to adopt, we have to consider whether both languages integrate. In this context, we believe that the second approach should be taken into account since both languages do not integrate as they do not have overlapping concepts. Instead, the connection of both languages is established via the methodological guidelines of Chapter 7 (Section 7.4).

10.4 Future Work

10.4.1 Further Validation

Validation Studies for SIENA Modeling Framework. In this thesis, we have applied the SIENA and Azzurra modeling languages (Chapter 5 and 7) in two company scenarios, i.e., the metal manufacturing in Chapter

5 and the hospital example in Chapter 8 (Section 8.2). In this context, although we believe the framework proposed in this thesis has considerably advanced the representation and analysis of strategic enterprise architectures, it still needs to undergo further validation beyond the application in the two aforementioned scenarios. We believe that the application in other examples of enterprises might reveal other concepts, methodological guidelines and reasoning needs which are currently not covered. Furthermore, different types of enterprises may also require evolutions and adaptations in the current version of concepts, methodological guidelines and reasoning technique. For example, the hospital scenario has a knowledge-intensive characteristic that leads to business processes with uncertainties and adaptations at execution time, whereas the manufacturing company is based on standardized production requirements, that leads to little variations in business processes at runtime. Hence, the particular features of each domain may impact in the future (re)design of the SIENA framework. In this context, in order to strengthen the validation of the framework, we need real-world companies with different characteristics (e.g., public vs. private, small vs. medium vs. large enterprises) from different domains (product vs. service industries, private vs. non-governmental organizations).

Validation Studies for Azzurra Modeling Language. A very natural direction for our future work regards the replication of the empirical experiment of Azzurra language and structured vs. unstructured processes (Section 9.2) performed with master students. In that respect, we first envision an experimental design that encompasses a higher number of students in order to be able to validate some of our hypothesis (e.g., the difference of structured and unstructured processes). Alternatively, we would be also interested in repeating the similar experiment with BPM experts within an industrial setting. The adoption of industrial experts would allow us to not only gain more statistical power in our analysis but could be also instrumental in acquiring insights regarding the acceptance of Azzurra within the industry. A second future work direction for our work concerns the elaboration of modeling patterns and guidelines for process representation using Azzurra, similarly as the existent ones for BPMN [131]. Finally, the usage of the same dataset with different metrics for the evaluation of process models (as the one proposed in [168]) could yield us different conclusions regarding the suitability of both process languages.

10.4.2 Additional Frameworks Features

Refinements or Extensions in Representational Support

Representation of Core and Support Operations. In SIENA language (Chapter 5), tactical goals that implement strategic goals are divided into *initiatives* (single projects that establish mechanisms for implementing strategic goals) or *established responsibilities* (responsibilities of every functional area to accomplish its part of the organization's strategy). During the evaluation with the SIENA and Azzurra languages in the scope of the hospital scenario (Section 8.2.4), we have noticed that established routines might be operationalized by *core operations* (related to the core business of the company), *support operations* (executed to support the realization of core operations) and *management operations* (for managing the organization) in line with the conceptualization proposed by Michael Porter [157] in Management literature. Therefore, such distinctions could be introduced in the language.

Representation of Indicators. Although the representation of indicators is an important feature for measuring in which extent goals are being achieved, our approach does not provide a solution for their representation. In fact, a careful analysis of the metal manufacturing goal hierarchy (Figure 3.1) in Management literature reveals the representation of indicators as a natural direction for our work. For example, for an "Increase sales by 12% over 2 years" strategic goal, the target value is an increase of 12%. In order to be able to measure the sales increase to determine the extent to which the goal is achieved, we need to create a "sales volume" indicator and measure it after 2 years. In this context, we can use the representational features of BIM framework [89] in order to propose such extension.

Representation of Contingency Goals. SIENA language and modeling guidelines (Chapter 5) are based on the core distinctions from Management Literature, i.e., strategic, tactical and operational goals. However, the framework refrains from addressing the representation of contingency goals which could be incorporated in the framework in the future. Contingency goals are defined as "an alternative goal or courses of action to reach that goal if and when circumstances and assumptions change so drastically as to make an original plan unusable" [153, p. 164]. They are useful for addressing evolutions and adaptations within the business context.

Representation of Runtime and Awareness Goals. In Chapters 5 and 6, the SIENA language proposes motivational distinctions that represent requirements (goals) and their realizations (strategic plans) at design time. An important future direction for research consists in using such design-time goal models for monitoring the achievement of SIENA operational goals and propagating such monitoring to higher layers up. In order to do that, we need to enrich our goal models at design-time, thus transforming them into runtime goals [35] that specify additional behavioral details about how goals should be achieved at runtime. Besides monitoring the achievement of SIENA operational goals, it would be also interesting to consider adaptations within the goal hierarchy by pruning or adding new goals in SIENA. The decision of which goals are pruned or added depends on how frequent they succeed or fail. In this context, in order to measure the success/failures of goals in SIENA, awareness goals [186] should be incorporated into the language. The incorporation of runtime and awareness goals in SIENA would support the achievement of the requirement R3.2.4 that requires support for control and evaluation of implemented strategic alternatives.

Representation of Capabilities and Resources. In the comparison between SIENA and ArchiMate frameworks (Section 8.3), we have noticed that ArchiMate represents capabilities and resources which are used for strategic planning. Therefore, as SIENA final goal is to support enterprise planning process, such concepts should be also embedded in our framework.

Improvements in the Azzurra Language. In Chapter 7, we have used the concepts of commitments and protocols to capture the social perspective of business processes. In this context, other types of contractual elements (e.g., obligations, duties, permissions, prohibitions, power/liability, immunity/disability and permission/no-right and protected liberty) should be also used to represent behavioral specifications. The usage of recommendations of clinical guidelines (Sections 9.1.2 and 9.3) could be used as a good motivation for studying the different nature of contractual elements.

Second, the in-depth evaluation of Azzurra language with novices (Chapter 9.3) revealed the need of embedding the representation of sub-processes in Azzurra. Third, in the context of the Azzurra language, future research could be performed in order to: (i) develop an enactment engine that supports remedies for noncompliance with Azzurra models; (iii) improve Azzurra's graphical notation and (iv) investigate the joint usage of Azzurra specifications and operational business process models represented using artifact-centered languages.

Different Types of Methodological Guidelines. In order to perform synchronized movements between the Goal and Operations Views within the SIENA language (Chapter 5), our framework needs the incorporation of different types (e.g. bottom-up) of methodological guidelines. Such methodological guidelines are useful for both starting and established companies. For starting companies, the company can derive its realizing business process architecture from organization's goals. For already established companies, the idea is to synchronize both structures (goal hierarchy and business process architecture), enabling one to find misalignments between company's strategy and its corresponding architecture of processes, revealing "why" each company's process exists in the architecture. Furthermore, such methodological guidelines should also support the transition from an AS-IS business process architecture to a TO-BE business process architecture that realizes business goals, similarly to ArchiMate methodological guidelines.

Extensions regarding Value-Based Frameworks. The business process architecture introduced in the SIENA framework (Chapter 5) must work in a synchronized fashion in order to deliver services and products to the final company's customer. In this context, further work is required to extend the SIENA framework with concepts and methodological guidelines that regards value-based conceptualization [97] from Management Sciences. These value proposition conceptualization will also help to clarify the elaboration of mission statements at the strategic level of the SIENA framework.

Extensions in Strategic Planning Technique. In Chapter 6, we currently have the rules for dimensional refinements that define how to find targets for strategic or tactical sub-goals based on the target of the parent goal. These rules specify how to find targets for sub-goals based on mathematical refinements and properties inheritance from parent goals. However, new rules for finding sub-targets based on the type of operator (AND or OR) for each type of refinement dimension (time, location, product/service) should be also specified. More specifically, the semantics of drill-down/roll-up operations from data warehouses [192] can be further explored for the automatic generation of goal refinements. Second, the rules for refinement should be implemented in a tool, for supporting both the refinement process and for checking the consistency of such refinements. Third, scalability tests for our approach with larger goal models with a high number of constraints are also fundamental for the overall evaluation of our reasoning approach. Fourth, once we have a realization (strategic plan) generated with the execution of the CGM tool, we could apply stress testing analysis as proposed in [126] in order to evaluate how strong this strategic plan behaves in different business scenarios. Finally, we envision that similar ideas for strategic planning proposed in Chapter 6 can be applied in the generation of tactical plans.

Bibliography

[1] Faisal Aburub, Mohammed Odeh, and Ian Beeson. Modeling Non-Functional Requirements of Business Processes. *Information and Software Technology*, 49(11):1162 – 1171, 2007.

[2] João Paulo Almeida and Evellin Cardoso. On the Elements of an Enterprise: Towards an Ontology-based Account. In *Proceedings of the 26th ACM Symposium on Applied Computing (SAC'11)*, pages 323–330. ACM, 2011.

[3] Birger Andersson, Maria Bergholtz, Ananda Edirisuriya, Tharaka Ilayperuma, Paul Johannesson, Jaap Gordijn, Bertrand Grégoire, Michael Schmitt, Eric Dubois, Sven Abels, Axel Hahn, Benkt Wangler, and Hans Weigand. Towards a Reference Ontology for Business Models. In *Proceedings of the 25th International Conference on Conceptual Modeling (ER'06)*, pages 482–496. Springer Berlin Heidelberg, 2006.

[4] Birger Andersson, Nicola Guarino, Paul Johannesson, and Barbara Livieri. Towards an Ontology of Value Ascription. In *Proceedings of the 9th International Conference in Formal Ontology in Information Systems (FOIS'16)*, pages 331–344, 2016.

[5] Annie Antón, Michael McCracken, and Colin Potts. Goal Decomposition and Scenario Analysis in Business Process Reengineering.

In *Proceedings of the 6th International Conference on Advanced Information Systems Engineering (CAiSE'94)*, pages 94–104. Springer Berlin Heidelberg, 1994.

[6] Yudistira Asnar, Volha Bryl, and Paolo Giorgini. Using Risk Analysis to Evaluate Design Alternatives. In *Proceedings of the 7th International Conference on Agent-oriented Software Engineering (AOSE'06)*, pages 140–155. Springer-Verlag, 2007.

[7] Yudistira Asnar and Paolo Giorgini. Modeling Risk and Identifying Countermeasure in Organizations. In *Proceedings of the 1st International Workshop on Critical Information Infrastructures Security (CRITIS'06)*, pages 55–66. Springer Berlin Heidelberg, 2006.

[8] Carlos Azevedo, João Paulo Almeida Almeida, Marten van Sinderen, and Luis Pires. Towards Capturing Strategic Planning in EA. In *Proceedings of the IEEE 19th International Enterprise Distributed Object Computing Conference (EDOC'15)*, pages 159–168. IEEE Computer Society Press, 2015.

[9] Carlos Azevedo, Maria Eugenia Iacob, João Paulo Almeida, Marten van Sinderen, Luis Pires, and Giancarlo Guizzardi. Modeling Resources and Capabilities in Enterprise Architecture: A Well-Founded Ontology-Based Proposal for ArchiMate. *Information Systems*, 54:235–262, 2015.

[10] Benjamin Aziz, Alvaro Arenas, Juan Bicarregui, Christophe Ponsard, and Philippe Massonet. From Goal-Oriented Requirements to Event-B Specifications. In *Proceedings of the 1st NASA Formal Methods Symposium*, volume NASA/CP-2009-215407 of *NASA Conference Proceedings*, pages 96–105, 2009.

[11] Jay Barney. Firm Resources and Sustained Competitive Advantage. *Journal of Management*, 17(1):99–120, 1991.

[12] Victor Basili, Gianluigi Caldiera, and Hans Rombach. The Goal Question Metric Approach. In *Encyclopedia of Software Engineering*. Wiley, 1994.

[13] Howard Becker. Notes on the Concept of Commitment. *American Journal of Sociology*, 66:32–40, 1960.

[14] Boualem Benatallah, Fabio Casati, Farouk Toumani, and Rachid Hamadi. Conceptual Modeling of Web Service Conversations. In *Proceedings of the 15th International Conference on Advanced Information Systems Engineering (CAiSE'03)*, pages 449–467. Springer Berlin Heidelberg, 2003.

[15] Solvita Bērziša, George Bravos, Tania Cardona Gonzalez, Ulrich Czubayko, Sergio España, Jānis Grabis, Martin Henkel, Lauma Jokste, Jânis Kampars, Hasan Koç, Jan-Christian Kuhr, Carlos Llorca, Pericles Loucopoulos, Raul Juanes Pascual, Oscar Pastor, Kurt Sandkuhl, Hrvoje Simic, Janis Stirna, Francisco Giromé Valverde, and Jelena Zdravkovic. Capability Driven Development: An Approach to Designing Digital Enterprises. *Business & Information Systems Engineering*, 57(1):15–25, 2015.

[16] Kamal Bhattacharya, Cagdas Gerede, Richard Hull, Riu Liu, and Jianwen Su. Towards Formal Analysis of Artifact-Centric Business Process Models. In *Proceedings of the 5th International Conference on Business Process Management (BPM'07)*, pages 288–304. Springer Berlin Heidelberg, 2007.

[17] Steven Bleistein, Karl Cox, June Verner, and Keith Phalp. B-SCP: A Requirements Analysis Framework for Validating Strategic Align-

ment of Organizational IT Based on Strategy, Context, and Process. *Information and Software Technology*, 48(9):846 – 868, 2006.

[18] Marco Brambilla, Piero Fraternali, and Carmen Vaca. A Notation for Supporting Social Business Process Modeling. In *Proceedings of the 3rd International Workshop on Business Process Model and Notation (BPMN'11)*, pages 88–102. Springer Berlin Heidelberg, 2011.

[19] Paolo Bresciani, Anna Perini, Paolo Giorgini, Fausto Giunchiglia, and John Mylopoulos. Tropos: An Agent-Oriented Software Development Methodology. *Autonomous Agents and Multi-Agent Systems*, 8(3):203–236, 2004.

[20] Volha Bryl, Paolo Giorgini, and John Mylopoulos. Designing Cooperative IS: Exploring and Evaluating Alternatives. In *Proceedings of the 14th International Conference on Cooperative Information Systems (CoopIS'06)*, pages 533–550, 2006.

[21] Volha Bryl, Paolo Giorgini, and John Mylopoulos. Supporting requirements analysis in tropos: A planning-based approach. In *Proceedings of the 10th Pacific Rim International Conference on Multi-Agent Systems (PRIMA'07)*, pages 243–254. Springer Berlin Heidelberg, 2009.

[22] Janis Bubenko, Anne Persson, and Janis Stirna. EKD User Guide. Technical report, Department of Computer and Systems Sciences, Royal Institute of Technology (KTH) and Stockholm University, 2001.

[23] Cristina Cabanillas, M. Resinas, and Antonio Ruiz-Cortés. Designing Business Processes with History-Aware Resource Assignments. In *Proceedings of the 4th International Workshop on Business Pro-*

cess Model and Notation (BPMN'12), volume 132, pages 101–112. Springer-Verlag Berlin Heidelberg, 2012.

[24] Evellin Cardoso. On the Alignment Between Goal Models and Enterprise Models with an Ontological Account. Master's thesis, Federal University of Espírito Santo, 2009.

[25] Evellin Cardoso, João Paulo Almeida, and Renata Guizzardi. On The Support for the Goal Domain in Enterprise Modeling Approaches. In *Proceedings of the 14th IEEE International Enterprise Distributed Object Computing Conference Workshops (EDOCW'10)*, pages 335–344, 2010.

[26] Evellin Cardoso, João Paulo Almeida, and Renata Guizzardi. Analyzing the Relations Between Strategic and Operational Aspects of an Enterprise: Towards an Ontology-Based Approach. *International Journal of Organizational Design and Engineering (IJODE'12)*, 2(3):271–294, 2012.

[27] Evellin Cardoso, Renata Guizzardi, and João Paulo Almeida. Aligning Goal Analysis and Business Process Modeling: A Case Study in Healthcare. *International Journal of Business Process Integration and Management (IJBPIM'11)*, 5(2):144–158, 2011.

[28] Evellin Cardoso, Katsiaryna Labunets, Fabiano Dalpiaz, John Mylopoulos, and Paolo Giorgini. Modeling Structured and Unstructured Processes: An Empirical Evaluation. In *Proceedings of the 35th International Conference on Conceptual Modeling (ER'16)*, volume 9974 of *LNCS*, pages 347–361, 2016.

[29] Evellin Cardoso, John Mylopoulos, Alejandro Mate, and Juan Trujillo. Strategic Enterprise Architectures. In *Proceedings of the 9th*

IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM'16), pages 57–71. Springer International Publishing, 2016.

[30] Evellin Cardoso, Paulo Santos, João Paulo Almeida, Renata Guizzardi, and Giancarlo Guizzardi. Semantic Integration of Goal and Business Process Modeling. In *IFIP International Conference on Research and Practical Issues of Enterprise Information Systems (CONFENIS 2010)*, 2010.

[31] Amit Chopra, Fabiano Dalpiaz, Paolo Giorgini, and John Mylopoulos. Modeling and Reasoning about Service-Oriented Applications via Goals and Commitments. In *Proceedings of 22nd International Conference on Advanced Information Systems Engineering (CAiSE'10)*, volume 6051, pages 113–128. Springer, 2010.

[32] CIO Council. FEAF - Federal Enterprise Architecture Framework Version 1.1. Technical report, United States Federal Government, 1999.

[33] David Cohn and Richard Hull. Business Artifacts: A Data-Centric Approach to Modeling Business Operations and Processes. *IEEE Data Engineering*, 32(3):3–9, 2009.

[34] R.L. Daft and D. Samson. *Fundamentals of Management: Asia Pacific Edition.* Cengage Learning, 2014.

[35] Fabiano Dalpiaz, Alex Borgida, Jennifer Horkoff, and John Mylopoulos. Runtime Goal Models. In *Proceedings of the IEEE 7th International Conference on Research Challenges in Information Science*, pages 1–11. IEEE, 2013.

[36] Fabiano Dalpiaz, Evellin Cardoso, Giulia Canobbio, Paolo Giorgini, and John Mylopoulos. Social Specifications of Business Processes with Azzurra. In *Proceedings of the IEEE 9th International Conference on Research Challenges in Information Science (RCIS'15)*. IEEE, 2015. Best Paper Award, selected among 199 submissions.

[37] Anne Dardenne, Axel van Lamsweerde, and Stephen Fickas. Goal-Directed Requirements Acquisition. In *Proceedings of the 6th International Workshop on Software Specification and Design*, pages 3–50. Elsevier Science Publishers, 1993.

[38] Robert Darimont and Axel van Lamsweerde. Formal Refinement Patterns for Goal-driven Requirements Elaboration. In *Proceedings of the 4th ACM SIGSOFT Symposium on Foundations of Software Engineering (SIGSOFT'96)*, pages 179–190. ACM, 1996.

[39] Rob Davis. *ARIS Design Platform: Advanced Process Modeling and Administration*. Springer Publishing Company, Incorporated, 1st edition, 2008.

[40] Richard Dealtry. *Dynamic SWOT Analysis: Developer's Guide*. Dynamic SWOT Associates, 1992.

[41] Gero Decker, Oliver Kopp, Frank Leymann, and Mathias Weske. Interacting Services: From Specification to Execution. *Data and Knowledge Engineering*, 68:946–972, 2009.

[42] Nirmit Desai, Amit Chopra, and Munindar Singh. Amoeba: A Methodology for Modeling and Evolution of Cross-Organizational Business Processes. *ACM Transactions on Software Engineering and Methodology*, 19(2), 2009.

[43] Nirmit Desai, Ashok Mallya, Amit Chopra, and Munindar Singh. Interaction Protocols as Design Abstractions for Business Processes. *IEEE Transactions on Software Engineering*, 31(12):1015–1027, 2005.

[44] Claudio Di Ciccio, Andrea Marrella, and Alessandro Russo. Knowledge-Intensive Processes: An Overview of Contemporary Approaches. In *Proceedings of the 1st International Workshop on Knowledge-intensive Business Processes (KiBP'12)*, number 861, pages 33–47. CEUR Workshop Proceedings, 2012.

[45] Claudio Di Ciccio, Andrea Marrella, and Alessandro Russo. Knowledge-Intensive Processes: Characteristics, Requirements and Analysis of Contemporary Approaches. *Journal on Data Semantics*, 4(1):29–57, 2015.

[46] Jan Dietz. *Enterprise Ontology: Theory and Methodology*. Springer-Verlag New York, Inc., 2006.

[47] Jan Dietz. The Deep Structure of Business Processes. *Communication ACM*, 49(5):58–64, 2006.

[48] Virginia Dignum. *A Model for Organizational Interaction: Based on Agents, Founded in Logic*. PhD thesis, Utrecht University, Utrecht, The Netherlands, 2004.

[49] Remco Dijkman, Irene Vanderfeesten, and Hajo Reijers. Business Process Architectures: Overview, Comparison and Framework. *Enterprise Information Systems*, 10(2):129–158, 2014.

[50] George Doran. There's a SMART Way to Write Management's Goals and Objectives. *Management Review*, pages 35–36, 1981.

[51] Andrew DuBrin. *Essentials of Management*. Cengage Learning, 2011.

[52] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo Reijers. *Fundamentals of Business Process Management*. Springer Publishing Company, Incorporated, 2013.

[53] Marlon Dumas, Wil van der Aalst, and Arthur ter Hofstede. *Process-Aware Information Systems: Bridging People and Software Through Process Technology*. Wiley, 2005.

[54] Rami-Habib Eid-Sabbagh. *Business Process Architectures: Concepts, Formalism, and Analysis*. PhD thesis, University of Potsdam, 2015.

[55] Kathleen Eisenhardt and Jeffrey Martin. Dynamic Capabilities: What Are They? *Strategic Management Journal*, 21(10-11):1105–1121, 2000.

[56] Wilco Engelsman and Roel Wieringa. Goal-Oriented Requirements Engineering and Enterprise Architecture: Two Case Studies and Some Lessons Learned. In *Proceedings of the 18th International Working Conference on Requirements Engineering: Foundation for Software Quality (REFSQ'12)*, pages 306–320. Springer Berlin Heidelberg, 2012.

[57] Roberta Ferrario and Nicola Guarino. Commitment-Based Modeling of Service Systems. In *Proceedings of the 3rd International Conference on Exploring Services Science (IESS'12)*, pages 170–185. Springer Berlin Heidelberg, 2012.

[58] Fernando Flores, Michael Graves, Brad Hartfield, and Terry Winograd. Computer Systems and the Design of Organizational Interaction. *ACM Transactions on Information Systems*, 6:153–172, 1988.

[59] International Organization for Standardization (ISO). Ergonomics of Human-System Interaction. ISO DIS 9241-11.2(en), ISO, 2017.

[60] Mark Fox. The TOVE Project Towards a Common-Sense Model of the Enterprise. In *Proceedings of the 5th International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pages 25–34. Springer Berlin, 1992.

[61] Mark Fox, Mihai Barbuceanu, Michael Grüninger, and Jinxin Lin. Simulating Organizations: Computational Models of Institutions and Groups. chapter An Organization Ontology for Enterprise Modeling, pages 131–152. MIT Press, 1997.

[62] Andrew Gemino and Yair Wand. A Framework for Empirical Evaluation of Conceptual Modeling Techniques. *Requirements Engineering*, 9(4):248–260, 2004.

[63] Pankaj Ghemawat. Competition and Business Strategy in Historical Perspective. Technical report, Harvard Business Review, 2000.

[64] Aditya Ghose and George Koliadis. Auditing Business Process Compliance. In *Proceedings of the 5th International Conference on Service-Oriented Computing (ICSOC'07)*, pages 169–180. Springer Berlin Heidelberg, 2007.

[65] Paolo Giorgini, John Mylopoulos, Eleonora Nicchiarelli, and Roberto Sebastiani. Reasoning with Goal Models. In *Proceedings of the 21st International Conference on Conceptual Modeling (ER'03)*, pages 167–181. Springer Berlin Heidelberg, 2003.

[66] Jaap Gordijn. *Value-Based Requirements Engineering: Exploring Innovative E-Commerce Ideas*. PhD thesis, Vrije Universiteit Amsterdam, 2002.

[67] Jaap Gordijn and Hans Akkermans. Designing and Evaluating E-Business Models. *IEEE Intelligent Systems*, 16(4):11–17, 2001.

[68] Robert M. Grant. Toward a Knowledge-Based Theory of the Firm. *Strategic Management Journal*, 17(S2):109–122, 1996.

[69] Gemma Grau, Xavier Franch, and Neil Maiden. A Goal-Based Round-Trip Method for System Development. In *Proceedings of the 11th International Conference on Requirements Engineering: Foundations for Software Quality (REFSQ'05)*, pages 71–86. Essener Informatik Beitrge, 2005.

[70] Trisha Greenhalgh. *How to Read a Paper: The Basics of Evidence-based Medicine.* John Wiley & Sons, 2014.

[71] Greenpeace.org. Greenpeace Core Values. http://www.greenpeace.org/international/en/about/our-core-values/.

[72] Dominic Greenwood. Goal-Oriented Autonomic Business Process Modeling and Execution: Engineering Change Management Demonstration. In *Proceedings of the 6th International Conference on Business Process Management (BPM'08)*, pages 390–393, 2008.

[73] Dominic Greenwood and Giovanni Rimassa. Autonomic Goal-Oriented Business Process Management. In *Proceedings of the 3rd International Conference on Autonomic and Autonomous Systems (ICAS '07)*, pages 43–. IEEE Computer Society, 2007.

[74] IEEE Architecture Working Group. IEEE Std 1471-2000, Recommended Practice For Architectural Description of Software-Intensive Systems. Technical report, IEEE, 2000.

[75] Object Management Group. *Business Motivation Model - Version 1.0.* Object Management Group (OMG), 2015.

[76] Object Management Group. *Value Delivery Metamodel*. Object Management Group (OMG), 2015.

[77] The Open Group. TOGAF Version 9.1, an Open Group Standard. Technical report, The Open Group, 2014.

[78] The Open Group. ArchiMate 2.0 Specification. Technical Report C118, Van Haren Publishing, Zaltbommel, 2017.

[79] Michael Grüninger, Katy Atefi, and Mark Fox. Ontologies to Support Process Integration in Enterprise Engineering. *Computational & Mathematical Organization Theory*, 6(4):381–394, 2000.

[80] Nicola Guarino. *Formal Ontology and Information Systems*. IOS Press, 1st edition, 1998.

[81] Renata S. S. Guizzardi, Giancarlo Guizzardi, Anna Perini, and John Mylopoulos. Towards an Ontological Account of Agent-Oriented Goals. In *International Workshop on Software Engineering for Large-Scale Multi-agent Systems (SELMAS'07)*, pages 148–164. Springer Berlin Heidelberg, 2007.

[82] Renata S. S. Guizzardi and Ariane Reis. A Method to Align Goals and Business Processes. In *Proceedings of the 34th International Conference on Conceptual Modeling (ER'15)*, pages 79–93, 2015.

[83] David Harel and Bernhard. Rumpe. Modeling Languages: Syntax, Semantics and All That Stuff, Part I: The Basic Stuff. Technical report, Weizmann Science Press of Israel, Jerusalem, Israel, 2000.

[84] Paul Harmon. The State of Business Process Management 2016. Technical report, BPTrends, 2016.

[85] Motulsky Harvey. *Intuitive Biostatistics*. New York: Oxford University Press, 1995.

[86] William Heaven and Emmanuel Letier. Simulating and Optimizing Design Decisions in Quantitative Goal Models. In *Proceedings of the 19th IEEE International Requirements Engineering Conference (RE'11)*, pages 79–88, 2011.

[87] Jennifer Horkoff, Fatma Aydemir, Evellin Cardoso, Tong Li, Elda Paja, Mattia Salnitri, Luca Piras, Alejandro Mate, John Mylopoulos, and Paolo Giorgini. Goal-Oriented Requirements Engineering: An Extended Systematic Mapping Study. *Requirements Engineering Journal (REJ'17)*, 2017.

[88] Jennifer Horkoff, Fatma Başak Aydemir, Evellin Cardoso, Tong Li, Alejandro Maté, Elda Paja, Mattia Salnitri, John Mylopoulos, and Paolo Giorgini. Goal-Oriented Requirements Engineering: A Systematic Literature Map. In *IEEE 24th International Requirements Engineering Conference (RE16)*, pages 106–115. IEEE, 2016.

[89] Jennifer Horkoff, Daniele Barone, Lei Jiang, Eric Yu, Daniel Amyot, Alex Borgida, and John Mylopoulos. Strategic Business Modeling: Representation and Reasoning. *Software & Systems Modeling*, 13(3):1015–1041, 2014.

[90] Jennifer Horkoff, Tong Li, Feng-Lin Li, Mattia Salnitri, Evellin Cardoso, Paolo Giorgini, and John Mylopoulos. Using Goal Models Downstream: A Systematic Roadmap and Literature Review. *International Journal of Information System Modeling and Design (IJISMD'15)*, 6(2):1–42, 2015.

[91] Jennifer Horkoff, Tong Li, Feng-Lin Li, Mattia Salnitri, Evellin Cardoso, Paolo Giorgini, John Mylopoulos, and Joao Pimentel. Taking Goal Models Downstream: A Systematic Roadmap. In *Proceedings*

of the IEEE 8th International Conference on Research Challenges in Information Science (RCIS'14), pages 1–12. IEEE, 2014.

[92] Jennifer Horkoff and Eric Yu. Evaluating Goal Achievement in Enterprise Modeling – An Interactive Procedure and Experiences. In *Proceedings of the 2nd IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM'09)*, pages 145–160. Springer Berlin Heidelberg, 2009.

[93] Jennifer Horkoff and Eric Yu. Comparison and Evaluation of Goal-Oriented Satisfaction Analysis Techniques. *Requirements Engineering*, 18(3):199–222, 2013.

[94] Jennifer Marie Horkoff. *Iterative, Interactive Analysis of Agent-Goal Models for Early Requirements Engineering*. PhD thesis, University of Toronto, Toronto, Canada, 2012.

[95] Richard Hull, Elio Damaggio, Riccardo De Masellis, Fabiana Fournier, Manmohan Gupta, Fenno Terry Heath, III, Stacy Hobson, Mark Linehan, Sridhar Maradugu, Anil Nigam, Piwadee Noi Sukaviriya, and Roman Vaculin. Business Artifacts with Guard-Stage-Milestone Lifecycles: Managing Artifact Interactions with Conditions and Events. In *Proceedings of the 5th ACM International Conference on Distributed Event-based System (DEBS '11)*, pages 51–62. ACM, 2011.

[96] Paul Johannesson. Representation and Communication - A Speech Act Based Approach to Information Systems Design. *Information Systems*, 20(4):291 – 303, 1995.

[97] Gerry Johnson, Kevan. Scholes, and Richard Whittington. *Exploring Corporate Strategy*. Exploring Corporate Strategy. Financial Times Prentice Hall, 2008.

[98] Henk Jonkers, Marc Lankhors, Hugo ter Doest, Farhad Arbab, Hans Bosma, and Roel Wieringa. Enterprise Architecture: Management Tool and Blueprint for the Organization. *Information Systems Frontiers*, pages 63–66, 2006.

[99] Paul Joyce and Adrian Woods. *Strategic Management: A Fresh Approach to Developing Skills, Knowledge and Creativity*. Kogan Page, 2001.

[100] Natalia Juristo and Ana M. Moreno. *Basics of Software Engineering Experimentation*. Springer, 2010.

[101] Ajit Kambil, Ari Ginsberg, and Michael Bloch. Re-Inventing Value Propositions. Technical report, Information Systems Working Papers Series, 1996.

[102] Evangelia Kavakli and Pericles Loucopoulos. Goal-Driven Business Process Analysis Application in Electricity Deregulation. *Information Systems*, 24(3):187 – 207, 1998.

[103] Evangelia Kavakli and Pericles Loucopoulos. Goal Modeling in Requirements Engineering: Analysis and Critique of Current Methods. *Information Modeling Methods and Methodologies*, pages 102–124, 2005.

[104] Gerhard Keller, Markus Nüttgens, and August-Wilhelm Scheer. Semantische Prozessmodellierung auf der Grundlage "Ereignisgesteuerter Prozessketten (EPK)". *Veröffentlichungen des Instituts für Wirtschaftsinformatik*, 89, 1992.

[105] Rania Khalaf. From RosettaNet PIPs to BPEL Processes: A Three Level Approach for Business Protocols. *Data and Knowledge Engineering*, 61:23–38, 2007.

[106] Maxim Khomyakov and Ilia Bider. Achieving Workflow Flexibility through Taming the Chaos. In *Proceedings of the 6th International Conference on Object Oriented Information Systems (OOIS'00)*, pages 85–92. Springer, 2000.

[107] George Koliadis and Aditya Ghose. Relating Business Process Models to Goal-Oriented Requirements Models in KAOS. In *Proceedings of the Pacific Rim Knowledge Acquisition Workshop (PKAW'06)*, pages 25–39. Springer Berlin Heidelberg, 2006.

[108] George Koliadis, Aleksandar Vranesevic, Moshiur Bhuiyan, Aneesh Krishna, and Aditya Ghose. Combining i* and BPMN for Business Process Model Lifecycle Management. In *Proceedings of Business Process Management (BPM'06) Workshops*, pages 416–427. Springer Berlin Heidelberg, 2006.

[109] Birgit Korherr and Beate List. Extending the EPC and the BPMN with Business Process Goals and Performance Measures. In *Proceedings of the 9th International Conference on Enterprise Information Systems (ICEIS'07)*, pages 287–294, 2007.

[110] Phillip Kotler, Linden Brown, Stewart Adam, Gary Armstrong, and Suzan Burton. *Marketing.* Pearson Education Australia, 2007.

[111] Manolis Koubarakis and Dimitris Plexousakis. A Formal Model for Business Process Modeling and Design. In *Proceedings of the 12th International Conference on Advanced Information Systems Engineering (CAiSE'00)*, pages 142–156. Springer Berlin Heidelberg, 2000.

[112] Peter Kueng and Peter Kawalek. Goal-Based Business Process Models: Creation and Evaluation. *Business Process Management Journal*, 3:17–38, 1997.

[113] Marc Lankhorst. *Enterprise Architecture at Work: Modeling, Communication and Analysis.* The Enterprise Engineering Series. Springer Berlin Heidelberg, 2017.

[114] Michael Lanning and Edward Michaels. A Business Is A Value Delivery System. Technical report, McKinsey Staff Paper, 1988.

[115] Alexei Lapouchnian and John Mylopoulos. Modeling Domain Variability in Requirements Engineering with Contexts. In *Proceedings of the 28th International Conference on Conceptual Modeling (ER'09)*, pages 115–130. Springer Berlin Heidelberg, 2009.

[116] Alexei Lapouchnian, Eric Yu, and Arnon Sturm. Re-Designing Process Architectures Towards a Framework of Design Dimensions. In *Proceedings of the IEEE 9th International Conference on Research Challenges in Information Science (RCIS'15)*, pages 205–210, 2015.

[117] Alexei Lapouchnian, Yijun Yu, and John Mylopoulos. Requirements-Driven Design and Configuration Management of Business Processes. In *Proceedings of the 5th International Conference on Business Process Management (BPM'07)*, volume 4714, pages 246–261. Springer Berlin, 2007.

[118] Emmanuel Letier and Axel van Lamsweerde. Deriving Operational Software Specifications from System Goals. In *Proceedings of the 10th ACM SIGSOFT Symposium on Foundations of Software Engineering (SIGSOFT'02)*, pages 119–128. ACM, 2002.

[119] Emmanuel Letier and Axel van Lamsweerde. Reasoning About Partial Goal Satisfaction for Requirements and Design Engineering. In *Proceedings of the 12th ACM SIGSOFT Twelfth International Symposium on Foundations of Software Engineering (SIGSOFT'04)*, pages 53–62. ACM, 2004.

[120] Sotirios Liaskos, Sheila A. McIlraith, Shirin Sohrabi, and John Mylopoulos. Representing and Reasoning About Preferences in Requirements Engineering. *Requirements Engineering*, 16(3):227, 2011.

[121] Mikael Lind and Göran Goldkuhl. Reconstruction of Different Business Processes - A Theory and Method Driven Analysis. In *Proceedings of the 2nd International Workshop on Language/Action Perspective (LAP'97)*. Borås Studies in Information Systems, 1997.

[122] Mikael Lind and Gran Goldkuhl. The Constituents of Business Interaction - Generic Layered Patterns. *Data & Knowledge Engineering*, 47(3):327 – 348, 2003.

[123] B. Mahadevan. *Operations Management: Theory and Practice*. Pearson, 2010.

[124] Elisa Marengo, Matteo Baldoni, Cristina Baroglio, Amit Chopra, Viviana Patti, and Munindar Singh. Commitments with Regulations: Reasoning about Safety and Control in REGULA. In *Proceedings of the 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS'11)*, volume 2, pages 467–474, 2011.

[125] Ivan Markovic and Marek Kowalkiewicz. Linking Business Goals to Process Models in Semantic Business Process Modeling. In *Proceedings of the IEEE 12th International Enterprise Distributed Object Computing Conference (EDOC'08)*, pages 332–338, 2008.

[126] Alejandro Maté, Juan Trujillo, and John Mylopoulos. Stress Testing Strategic Goals with SWOT Analysis. In *Proceedings of the 34th International Conference on Conceptual Modeling (ER'15)*, pages 65–78. Springer International Publishing, 2015.

[127] Raul Medina-Mora, Terry Winograd, Rodrigo Flores, and Fernando Flores. The Action Workflow Approach to Workflow Management Technology. In *Proceedings of the 1992 ACM Conference on Computer-supported Cooperative Work (CSCW '92)*, pages 281–288. ACM, 1992.

[128] Lucas Meertens, Maria Eugenia Iacob, Lambertus Nieuwenhuis, Henk Jonkers, Marten van Sinderen, and Dick Quartel. Mapping the Business Model Canvas to ArchiMate. In *Proceedings of the 27th Annual ACM Symposium on Applied Computing*, pages 1694–1701. ACM, 3 2012.

[129] Dagmawi Mekuria, Chi Mai Nguyen, Roberto Sebastiani, Paolo Giorgini, and John Mylopoulos. CGM Tool, 2015.

[130] Ricardo Mendes, André Vasconcelos, Artur Caetano, João Neves, Pedro Sinogas, and José M. Tribolet. Representing Business Strategy through Goal Modeling. In *Proceedings of the 3rd International Conference on Enterprise Information Systems (ICEIS'01)*, pages 884–887, 2001.

[131] Jan Mendling, Hajo Reijers, and Wil van der Aalst. Seven Process Modeling Guidelines (7PMG). *Information and Software Technology*, 52(2):127–136, 2010.

[132] Paul Meyer. *Attitude Is Everything!* The Leading Edge Publishing Co, 2006.

[133] Henry Mintzberg. *The Structuring of Organizations: A Synthesis of The Research.* Theory of Management Policy Series. Prentice-Hall, 1979.

[134] Henry Mintzberg. *The Strategy Process: Concepts, Contexts, Cases.* Pearson Education Higher Education División. Pearson Education, 2003.

[135] Henry Mintzberg, Bruce Ahlstrand, and Joseph Lampel. *Strategy Safari: A Guided Tour Through The Wilds of Strategic Management.* Free Press, 2005.

[136] Evan Morrison, Aditya Ghose, Hoa Dam, Kerry Hinge, and Konstantin Hoesch-Klohe. Strategic Alignment of Business Processes. In *Proceedings of Service-Oriented Computing (ICSOC'12) Workshops*, pages 9–21. Springer Berlin Heidelberg, 2012.

[137] John Mylopoulos. Information Modeling in the Time of the Revolution. *Information Systems*, 23(3-4):127–155, 1998.

[138] Rajiv Nag, Donald Hambrick, and Ming-Jer Chen. What Is Strategic Management, Really? Inductive Derivation of A Consensus Definition of The Field. *Strategic Management Journal*, 28(9):935–955, 2007.

[139] Prabir Nandi and Jorge Sanz. Cross-Functional Operations Modeling as a Nexus of Commitments: A New Approach for Improving Business Performance and Value-Creation. In *Proceedings of the IEEE 15th Conference on Business Informatics (CBI'13)*, pages 234–241, 2013.

[140] Julio Nardi, Ricardo Falbo, Joao Almeida, Giancarlo Guizzardi, Luis Pires, Marten van Sinderen, and Nicola Guarino. Towards a Commitment-Based Reference Ontology for Services. In *In Proceedings of the IEEE 17th International Enterprise Distributed Object Computing Conference (EDOC'13)*, pages 175–184, 2013.

[141] Dina Neiger and Leonid Churilov. Goal-Oriented Business Process Modeling with EPCs and Value-Focused Thinking. In *Proceedings of the 2nd International Conference on Business Process Management (BPM'04)*, volume 3080, pages 98–115. Springer Berlin, 2004.

[142] Chi Mai Nguyen, Roberto Sebastiani, Paolo Giorgini, and John Mylopoulos. Multi-Objective Reasoning with Constrained Goal Models. *Requirements Engineering*, pages 1–37, 2016.

[143] Eetu Niemi. Enterprise Architecture Benefits: Perceptions from Literature and Practice. *Tietotekniikan Tutkimusinstituutin Julkaisuja; 18*, pages 1236–1615, 2008.

[144] Natalya Noy. Semantic Integration: A Survey of Ontology-Based Approaches. *ACM SIGMOD Record*, 33(4):65–70, 2004.

[145] Selmin Nurcan, Anne Etien, Rim Samia Kaabi, Iyad Zoukar, and Colette Rolland. A Strategy Driven Business Process Modeling Approach. *Business Process Management Journal*, 11(6):628–649, 2005.

[146] Department of Defense. Department of Defense Architecture Framework Version 2.2. Technical report, United States Department of Defense, 2010.

[147] Martin Op't Land, Erik Proper, Maarten Waage, Jeroen Cloo, and Claudia Steghuis. *Enterprise Architecture: Creating Value by Informed Governance*. Springer Science & Business Media, 2008.

[148] Alexander Osterwalder. *The Business Model Ontology - A Proposition In A Design Science Approach*. PhD thesis, University of Lausanne, Lausanne, Switzerland, 2004.

[149] Alexander Osterwalder and Yves Pigneur. *Business Model Generation: A Handbook for Visionaries, Game Changers, and Challengers.* John Wiley and Sons, 2010.

[150] Cancer Council Australia Lung Cancer Guidelines Working Party. Clinical Practice Guidelines for the Treatment of Lung Cancer. Technical report, Cancer Council Australia, 2017.

[151] Mor Peleg. Computer-Interpretable Clinical Guidelines: A Methodological Review. *Journal of Biomedical Informatics*, 46(4):744 – 763, 2013.

[152] Paul Pichler, Barbara Weber, Stefan Zugal, Jakob Pinggera, Jan Mendling, and Hajo Reijers. Imperative versus Declarative Process Modeling Languages: An Empirical Investigation. In *Proceedings of Business Process Management (BPM'11) Workshops*, pages 383–394. Springer Berlin Heidelberg, 2012.

[153] Warren Plunkett, Raymond Attner, and Gemmy Allen. *Management: Meeting and Exceeding Customer Expectations.* Cengage Learning, 2007.

[154] Viara Popova and Alexei Sharpanskykh. Formal Goal-based Modeling of Organizations. In *Proceedings of the 6th International Workshop on Modeling, Simulation, Verification and Validation of Enterprise Information Systems (MSVVEIS'08) in conjunction with (ICEIS'08)*, pages 19–28, 2008.

[155] Viara Popova and Alexei Sharpanskykh. Formal Analysis of Executions of Organizational Scenarios Based on Process-Oriented Specifications. *Applied Intelligence*, 34(2):226–244, 2011.

[156] M.E. Porter. *Competitive Advantage: Creating and Sustaining Superior Performance.* Free Press, 2008.

[157] Michael Porter. *From Competitive Advantage to Corporate Strategy*, pages 234–255. Macmillan Education UK, 1989.

[158] Alireza Pourshahid, Daniel Amyot, Liam Peyton, Sepideh Ghanavati, Pengfei Chen, Michael Weiss, and Alan Forster. Business Process Management WiththeUserRequirements Notation. *Electronic Commerce Research*, 9(4):269–316, 2009.

[159] Alireza Pourshahid, Iman Johari, Gregory Richards, Daniel Amyot, and Okhaide S. Akhigbe. A Goal-Oriented, Business Intelligence-Supported Decision-Making Methodology. *Decision Analytics*, 1(1):9, 2014.

[160] Dick Quartel, Wilco Engelsman, Henk Jonkers, and Marten van Sinderen. A Goal-Oriented Requirements Modeling Language for Enterprise Architecture. In *Proceedings of the IEEE 13th International Enterprise Distributed Object Computing Conference (EDOC'09)*, pages 3–13. IEEE Computer Society Press, 2009.

[161] Jan Recker. BPMN Research: What We Know and What We Don't Know. In *Proceedings of the 4th International Workshop on Business Process Model and Notation (BPMN'12)*, pages 1–7, 2012.

[162] Manfred Reichert and Barbara Weber. *Enabling Flexibility in Process-Aware Information Systems - Challenges, Methods, Technologies.* Springer, 2012.

[163] R. Reid and Nada Sanders. *Operations Management: An Integrated Approach.* Wiley, 2005.

[164] William Robinson and Sandeep Purao. Specifying and Monitoring Interactions and Commitments in Open Business Processes. *IEEE Software*, 26(2):72–79, 2009.

[165] Ben Roelens, Wout Steenacker, and Geert Poels. Realizing Strategic Fit Within the Business Architecture: The Design of A Process-Goal Alignment Modeling and Analysis Technique. *Software & Systems Modeling*, pages 1–32, 2017.

[166] Colette Rolland, Pericles Loucopoulos, Evagelia Kavakli, and Selmin Nurcan. Intention Based Modeling of Organizational Change: An Experience Report. In *Proceedings of the IFIP 8.1 International Workshop on Evaluation of Modeling Methods in Systems Analysis and Design (EMMSAD'09)*, pages 1055–1071. Springer Berlin Heidelberg, 1999.

[167] Jeanne Ross, Peter Weill, and David Robertson. *Enterprise Architecture As Strategy: Creating A Foundation for Business Execution.* Harvard Business Press, 2006.

[168] Anne Rozinat, Ana de Medeiros, Christian Günther, A. Weijters, and Wil van der Aalst. Towards an Evaluation Framework for Process Mining Algorithms. Technical report, 2007.

[169] Stuart Russell and Peter Norvig. *Artificial Intelligence: A Modern Approach (3rd edition).* Prentice Hall, 2009.

[170] Shazia Sadiq, Guido Governatori, and Kioumars Namiri. Modeling Control Objectives for Business Process Compliance. In *Proceedings of the 5th International Conference on Business Process Management (BPM'07)*, pages 149–164, 2007.

[171] Tiago Sales, Nicola Guarino, Giancarlo Guizzardi, and John Mylopoulos. An Ontological Analysis of Value Propositions. In *Proceedings of the IEEE 21st International Enterprise Distributed Object Computing Conference (EDOC'17)*, pages 184–193, 2017.

[172] Garth Saloner, Andrea Shepard, and Joel Podolny. *Strategic Management*. John Wiley, 2001.

[173] Emanuel Santos, Jaelson Castro, Juan Snchez, and Oscar Pastor. A Goal-Oriented Approach for Variability in BPMN. In *Workshop em Engenharia de Requisitos*, 2010.

[174] August-Wilhelm Scheer. *ARIS-Business Process Modeling*. Springer-Verlag New York, Inc., 3rd edition, 2000.

[175] Jaap Schekkerman. *How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework*. Trafford Publishing, 2004.

[176] Helen Schonenberg, Ronny Mans, Nick Russell, Nataliya Mulyar, and Wil van der Aalst. Towards a Taxonomy of Process Flexibility. In *Proceedings of CAiSE Forum*, volume 344 of *CEUR Workshop Proceedings*, pages 81–84. CEUR-WS.org, 2008.

[177] Russell Schutt. *Investigating the Social World: The Process and Practice of Research*. SAGE Publications, 2015.

[178] Clive Seale, Giampietro Gobo, Jaber Gubrium, and David Silverman. *Qualitative Research Practice*. SAGE Publications, 2007.

[179] Roberto Sebastiani, Paolo Giorgini, and John Mylopoulos. Simple and Minimum-Cost Satisfiability for Goal Models. In *Proceedings of the 16th International Conference on Advanced Information Systems*

Engineering (CAiSE'04), pages 20–35. Springer Berlin Heidelberg, 2004.

[180] Roberto Sebastiani and Silvia Tomasi. Optimization Modulo Theories with Linear Rational Costs. *ACM Transactions on Computational Logics*, 16(2), March 2015.

[181] Roberto Sebastiani and Patrick Trentin. OptiMathSAT: A Tool for Optimization Modulo Theories. In *Proceedings of the 27th International Conference Computer Aided Verification (CAV'15)*, pages 447–454. Springer International Publishing, 2015.

[182] Roger Sessions. Comparison of the Top Four Enterprise Architecture Methodologies. Technical report, ObjectWatch, Inc., 2007.

[183] Arash Shahin and M. Ali Mahbod. Prioritization of Key Performance Indicators: An Integration of Analytical Hierarchy Process and Goal Setting. *International Journal of Productivity and Performance Management*, 56(3):226–240, 2007.

[184] Munidar Singh. An Ontology for Commitments in Multiagent Systems: Toward a Unification of Normative Concepts. *Artificial Intelligence and Law*, 7:97–113, 1999.

[185] Pnina Soffer and Yair Wand. On the Notion of Soft-Goals in Business Process Modeling. *Business Process Management Journal*, 11:663–679, 2005.

[186] Vitor Souza, Alexei Lapouchnian, William Robinson, and John Mylopoulos. Awareness Requirements for Adaptive Systems. In *Proceedings of the 6th International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS '11)*, pages 60–69, 2011.

[187] R.M. Srivastava and Shubhra Verma. *Strategic Management: Concepts, Skills and Practices*. PHI Learning, 2012.

[188] Paul Ssekamatte. Modeling Clinical Guidelines as Business Processes: A Case Study for Asthma. Master's thesis, University of Trento, 2016.

[189] Steven Staab, Asunción Gomez-Perez, Walter Daelemana, Marie Reinberger, Nicola Guarino, and Natalya Noy. Why Evaluate Ontology Technologies? Because it Works! *IEEE Intelligent Systems*, 19(4):74–81, 2004.

[190] Shumet Tadesse Nigatu. Modeling Clinical Guidelines as Business Processes: A Case Study for Malaria. Master's thesis, University of Trento, 2016.

[191] David Teece and Gary Pisano. The Dynamic Capabilities of Firms: an Introduction. *Industrial and Corporate Change*, 3(3):537–556, 1994.

[192] Nectaria Tryfona, Frank Busborg, and Jens G Borch Christiansen. starER: A Conceptual Model for Data Warehouse Design. In *Proceedings of the 2nd ACM International Workshop on Data Warehousing and OLAP*, pages 3–8. ACM, 1999.

[193] Melkamu Tsegaye Emiru. Modeling Clinical Guideline As Business Processes: A Case Study For Cancer. Master's thesis, University of Trento, 2016.

[194] Mike Uschold, Martin King, Stuart Moralee, and Yannis Zorgios. The Enterprise Ontology. *The Knowledge Engineering Review*, 13(1):31–89, 1998.

[195] Joris van de Klundert, Pascal Gorissen, and Stef Zeemering. Measuring Clinical Pathway Adherence. *Journal of Biomedical Informatics*, 43:861–872, 2010.

[196] Wil van der Aalst. Business Process Management: A Comprehensive Survey. *ISRN Software Engineering*, pages 1–37, 2013.

[197] Wil van der Aalst, Marlon Dumas, Arthur ter Hofstede, Nick Russell, Eric Verbeek, and Petia Wohed. Life After BPEL? In *Proceedings of International Conference on European Performance Engineering Workshop (EPEW'05) and International Workshop on Web Services and Formal Methods (WS-FM'05), International Conference on Formal Techniques for Computer Systems and Business Processes*, pages 35–50. Springer-Verlag, 2005.

[198] Wil van der Aalst, Maja Pesic, and Helen Schonenberg. Declarative Workflows: Balancing between Flexibility and Support. *Computer Science-Research and Development*, 23:99–113, 2009.

[199] Wil van der Aalst and Mathias Weske. Case Handling: A New Paradigm for Business Process Support. *Data Knowledge Engineering*, 53(2):129–162, 2005.

[200] Axel van Lamsweerde. Requirements Engineering in the Year 00: A Research Perspective. In *Proceedings of the 22nd International Conference on Software Engineering (ICSE '00)*, pages 5–19. ACM, 2000.

[201] Jan vom Brocke and Michael Rosemann. *Handbook on Business Process Management 1: Introduction, Methods, and Information Systems*. Springer Publishing Company, Incorporated, 1st edition, 2010.

[202] Mathias Weske. *Business Process Management: Concepts, Languages, Architectures.* Springer, 2 edition, 2012.

[203] Szymon Wilk, Martin Michalowski, Wojtek Michalowski, Marisela Mainegra Hing, and Ken Farion. Reconciling Pairs of Concurrently Used Clinical Practice Guidelines Using Constraint Logic Programming. In *Proceedings of the AMIA Annual Symposium*, pages 944–953, 2011.

[204] Claes Wohlin, Per Runeson, Martin Höst, Magnus Ohlsson, Bjöorn Regnell, and Anders Wesslén. *Experimentation in Software Engineering: An Introduction.* Kluwer Academic Publishers, 2000.

[205] WS-CDL. Web Services Choreography Description Language Version 1.0. Technical report, W3C, 2005.

[206] Robert Yin. *Case Study Research: Design and Methods.* Applied Social Research Methods. SAGE Publications, 2003.

[207] Pinar Yolum and Munindar Singh. Flexible Protocol Specification and Execution: Applying Event Calculus Planning using Commitments. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS'02)*, pages 527–534, 2002.

[208] Eric Yu. *Modeling Strategic Relationships for Process Reengineering.* PhD thesis, University of Toronto, Toronto, Canada, 1996.

[209] Eric Yu and John Mylopoulos. An Actor Dependency Model of Organizational Work: With Application to Business Process Reengineering. In *Proceedings of the Conference on Organizational Computing Systems, (COOCS'93)*, pages 258–268, 1993.

[210] Eric Yu and John Mylopoulos. Using Goals, Rules and Methods to Support Reasoning in Business Process Reengineering. *International Systems in Accounting, Finance and Management*, 5(1):1–13, 1996.

[211] Eric Yu, John Mylopoulos, and Yves Lespérance. AI Models for Business Process Reengineering. *IEEE Expert*, 11(4):16–23, 1996.

[212] Eric Yu, Markus Strohmaier, and Xiaoxue Deng. Exploring Intentional Modeling and Analysis for Enterprise Architecture. In *Proceedings of the IEEE 10th International Enterprise Distributed Object Computing Conference Workshops (EDOCW'06)*, pages 32–32, 2006.

[213] John Zachman. A Framework for Information Systems Architecture. *IBM Systems Journal*, 26(3):276–292, 1987.

[214] Veruska Zamborlini, Marcos da Silveira, Cedric Pruski, Annette ten Teije, and Frank van Harmelen. Towards a Conceptual Model for Enhancing Reasoning About Clinical Guidelines: A Case-Study on Comorbidity. In *Proceedings of the 6th International Workshop on Knowledge Representation for Health Care (KR4HC'14)*, pages 29–44, 2014.

[215] Jelena Zdravkovic, Janis Stirna, Jan-Christian Kuhr, and Hasan Koç. Requirements Engineering for Capability Driven Development. In *Proceedings of the 7th IFIP WG 8.1 Working Conference on the Practice of Enterprise Modeling (PoEM'14)*, pages 193–207. Springer Berlin Heidelberg, 2014.