A Framework for Monitoring and Adapting Business Processes Using Aspect-Oriented URN

Alireza Pourshahid

Thesis submitted to the
Faculty of Graduate and Postdoctoral Studies
In partial fulfillment of the requirements for the degree of

Ph.D. in Computer Science

Under the auspices of the Ottawa-Carleton Institute for Computer Science

uOttawa
University of Ottawa
Ottawa, Ontario, Canada
April 2014

Abstract

Context: Organizations strive to improve their business processes, and adaptive business processes have recently attracted much attention in that context. However, much research in that area has a narrow focus and does not consider a comprehensive view of the organization and its goals. In addition, Business Intelligence-based monitoring methods are useful for business process improvement but they often present information in a format that is not entirely suited for decision making.

Objectives: The main objectives of this thesis are to provide:

- A framework to model goals, processes, performance, situations, and improvement patterns using one modeling notation, in an iterative and incremental manner;
- A method for the modeling and analysis of cause-effect relationships between indicators used to measure goal satisfaction; and
- A technique allowing the detection of undesirable, sub-optimal conditions and the application of improvement patterns to the context.

Method: We develop an iterative framework based on the *User Requirements Notation* (URN) for modeling, monitoring and improving business organizations and their business processes. In addition, we introduce a formula-based evaluation algorithm allowing better analysis of the relationships between the business performance model elements (namely indicators). Furthermore, we use a profiled version of the *Aspect-oriented URN* (AoURN) with extensions (*Business Process Pattern* profile), for detecting undesirable conditions and for business process adaptation. We validate the novelty and feasibility of our approach by performing a systematic literature review, by assessing it against Zellner' *mandatory elements of a method*, by developing tool support, by performing a pilot experiment and by using real-life examples from different sectors (healthcare and retail).

Results: The two examples show that through the framework's iterative approach, organizations at different levels of maturity in their business improvement journey can benefit from the framework. Furthermore, our systematic literature review shows that although there are existing works that enable our vision, most of them have a narrow focus and do not cover the three organization views that are of interest in this research. AoURN allows analysts to find repeated patterns in a context and bundle goal, performance and process models as a self-contained unit. AoURN hence enables the modeling of complex circumstances together with analysis techniques for what-if analysis and process adaptation, all using a unified and integrated modeling language. Finally, the pilot experiment suggests that, with some level of documentation and training, users who are already familiar with URN can use the profiled AoURN provided in this thesis as well as the discussed improvement patterns.

Acknowledgment

I would like to gratefully thank Dr. Daniel Amyot and Dr. Michael Weiss, whose valuable guidance and supervision helped me very much throughout this research. I was very lucky to have them both as my supervisors. Furthermore, I would like thank Dr. Gunter Mussbacher for his comments and for a stimulating research direction I was able to build on.

Many thanks to NSERC (Discovery grant program) for supporting and funding this research.

I would also like to thank my family for their constant support, especially my parents and my wife, Dr. Azalia Shamsaei, for their understanding throughout my studies, as well as all the people involved in our research team, including Dr. Greg Richards for his help with my second business example.

Table of Contents

List of Figures

List of Tables

List of Acronyms

Acronym	Definition
AoGRL	Aspect-Oriented and Goal-oriented Requirement Language
AOP	Aspect-Oriented Programming
AoUCM	Aspect-Oriented Use Case Maps
AoURN	Aspect-Oriented User Requirements Notation
BSS	Balanced Scorecards
BI	Business Intelligence
BIM	Business Intelligence Model
BPEL	Business Process Execution Language
BPEL4WS	Business Process Execution Language for Web Services
BPA	Business Process Analysis
BPI	Business Process Improvement
BPM	Business Process Management
BPMN	Business Process Model and Notation
BPMS	Business Process Management System
BPP	Business Process Pattern
DWAP	Data Warehouse Approval Process
EBRC	Enhanced Business Reporting Consortium
GRL	Goal-oriented Requirement Language
ITU	International Telecommunication Union
KPI	Key Performance Indicator
MEM	Mandatory Elements of Method
NFR	Non-Functional Requirements
SM	Strategy Maps
UCM	Use Case Map
UML	Unified Modeling Language
URN	User Requirements Notation
XBRL	Extensible Business Reporting Language
WICI	World Intellectual Capital Initiative

Chapter 1. Introduction

This thesis describes an iterative and incremental Business Process Improvement (BPI) framework that allows organizations to use a unified modeling language in order to model their goal, performance, process, and decision models, describe the relationship between the model elements, and monitor and improve their processes using a set of predefined redesign patterns.

This introductory chapter summarizes some of the concepts used in this thesis, the problems it addresses, the motivation for this research, and the research method. It also lists the publications based on this research and outlines the content and major/minor contributions of this thesis.

1.1. Concepts

This section defines the basic thesis terminology related to business process improvement, business intelligence, and modeling.

- **Business process:** Although, as suggested by Siegel [132], it is not practical to come up with a single definition of business process, most of the existing definitions consist of two main elements: 1) performed activities in one form or another, 2) the goals or objectives that the business strives toward. The Workflow Management Coalition [150] defines a business process as "a set of one or more linked procedures or activities which collectively realize a business objective or policy goal, normally within the context of an organizational structure defining functional roles and relationships". We use this definition in this thesis as we focus on the impact of business processes on business objectives.
- **Business process model:** A depicted sequence of activities and responsibilities that identifies the roles and responsibilities of users and organization units using a specific graphical modeling notation.

- **Business goal model:** The depicted goals of an organization that illustrate the business objectives and the objectives of different stakeholders, which allow one to analyse the current state of the business with respect to the defined goals.
- **Key Performance Indicators (KPI):** KPIs are the important quantifiable aspects of a business, calculated based on some defined measurement points and a specified rule for calculation [80]. According to Falck and Karlsson [44], KPIs can be both financial (e.g., financial solidity) and non-financial (e.g., customer satisfaction). This thesis provides means to monitor such KPIs, to use them for business process improvement, and to calculate them using other KPIs or measurement points as inputs and formulas.
- **Business performance model:** The depicted Key Performance Indicators of an organization as a graphical model that shows the relationships between the KPIs, and the relationships between the KPIs and the qualitative goals of the organizations that may not be as measurable as the KPIs. This model is at a lower level of abstraction than the goal model and can allow one to measure the progress of the business towards its goals using actual numbers.
- **Decision support system (DSS):** DSS is a computerized system that helps decision makers make informed decision by providing various up-to-date and comprehensive information regarding the context where the decision is being made.
- **Business Intelligence (BI):** According to Kimball and Ross [75], Business Intelligence is “a generic term to describe and leverage the organizations’ internal and external information assets for making better business decisions”. In the context of this thesis, the definition is more specific and focuses on BI as a tool that uses internal and external information assets to provide the KPI values used for monitoring the business in a goal-oriented monitoring system.
- **Process redesign patterns:** Process redesign patterns are best practices that have proven to be effective in improving business processes. The main process redesign patterns that we focus on in this thesis are those Reijers introduced in [123] and [124], which are both well cited (644 and 264 citations, respectively, according to Google Scholar, October 2013). Reijers’ work was the main reference on the topic when this research was started, and Reijers remains a well-cited expert

in this area of research nowadays. These patterns each could have potential positive or negative impact on time, cost, quality, and flexibility measures of business processes.

- **Aspect-Oriented Modeling (AOM):** Aspect-Oriented Modeling addresses problems in software engineering that occur because the units of interest to a requirements engineer cannot readily be encapsulated with units in downstream life cycle activities [27]. This mismatch results in *scattering* (parts of a requirements unit are distributed over many downstream units) and *tangling* (one downstream unit contains parts of many different requirements units) causing significant maintenance problems. Tarr *et al.* [140] refer to this as the tyranny of the dominant decomposition, since a chosen modularization technique inevitably will cause unwanted side effects such as scattering and tangling. While aspects were first introduced at the programming level with Aspect-Oriented Programming (AOP) [74], research emphasis has recently shifted towards Aspect-Oriented Modeling and application of Aspect-Oriented methods in various domains.

1.2. Problem Statement

In this thesis, we focus on helping businesses improve their performance by addressing several shortcomings in current business performance improvement frameworks and their supporting technologies, including:

- Lack of a business process redesign approach that allows one to model goals, performance and processes of organization in the form of reusable patterns to support the detection of common undesirable conditions in organizations.
- Lack of (semi-)automated approaches for detecting when and where business process redesign patterns can potentially be applied to business processes, applying them and investigating the potential outcome.
- Disconnection between Decision Support / Business Intelligence Systems outputs and decision-making processes used by decision makers.

- Dependency of the existing methods on a high maturity of the organization, on the availability of much information, and on infrastructures to provide tangible value for the business.

Furthermore, since the User Requirement Notation (URN) [62] is used as the main graphical modeling notation in this thesis, we also need to address the following shortcomings of this modeling notation:

- Lack of an approach that allows one to define accurate relationships between elements of a performance model (KPIs).
- Lack of an algorithm that allows the analysis and evaluation of the relationships between the KPIs and the propagation of the aggregated results all the way up to the organization goals.
- Further shortcomings that are found throughout this research and are required for the intended application such as lack of repetition containers, absent containers, and raw data input elements (to be explored in section 5.5).

1.3. Motivation

The goal of this research is to allow businesses to better monitor their business and processes and eventually to improve them. Although existing technologies and methods provide many facilities for businesses to achieve this goal, we believe there is still potential for contributions on this front.

Regarding the monitoring part, Ko and Abdullaev [76] estimate that more than 50% of BI implementations fail to achieve their goals. Reasons for this include cultural resistance, lack of relevance, lack of alignment with business strategy, and lack of actionable and "institutionalized" decision support technologies [51]. Many of these problems could be attributed to approaches used for defining the data to be delivered by the BI tool. In addition, the inability of existing tools to properly define the relationships between objectives and KPIs of the organization is yet another reason for this failure. According to Popova and Sharpanskykh [112], even when relationships can be defined such as in the ARIS model [35], which allows users to define cause-effect relationships using Balanced

Scorecards and connect KPIs to strategic goals, the analysis options are inadequate due to a lack of formal modeling foundations and proper representations.

Regarding the process improvement part, there has been some work done by Reijers and Liman-Mansar [123][124] to capture common improvement approaches in the form of business redesign patterns that contribute to the improvement of processes from four main perspectives, namely time, quality, cost, and flexibility. Although these redesign patterns could be used as a guideline for the improvement step, most of the improvement methods do not utilize these patterns. On the contrary, existing methods rely heavily on human innovation and creativity rather than on rationality. As suggested by Forster [45], using patterns can help to further rationalize and formalize process improvement methods.

Although there have been many achievements in the industry in this regard, with the exception of some contributions [13][51], most tools and methodologies do not allow stakeholders to go through all steps of a BPM project from modeling to performance improvement as an iterative and incremental approach with proper tool support. In this thesis, we propose a cohesive framework to address this issue.

1.4. Research Method

In this thesis, an iterative approach of research and validation has been followed. At every step, some level of validation was performed to evaluate the direction of research. This approach is adapted from the "Design science in information systems research" research method, proposed by Hevner [55] and used in other theses like [129], and it relies on iterative improvements to the research. To be more specific, we follow the steps highlighted in Figure 1, which are grouped under three cycles used in Hevner's proposed approach but adapted to the needs of this research, especially in terms of evaluations.

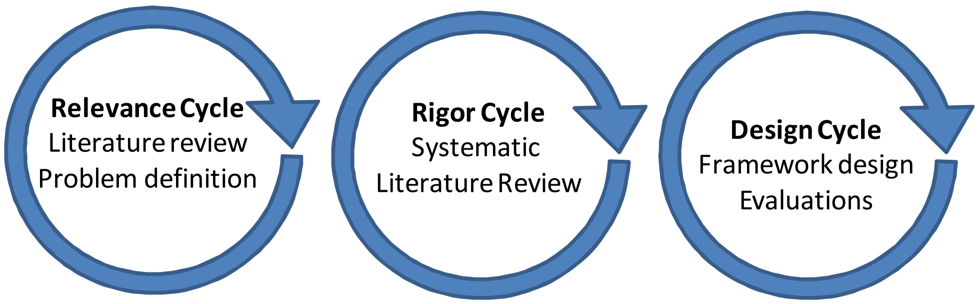


Figure 1: Research method [129]

1. **Relevance Cycle**
   - Background review: In this step, we conducted some initial background review to make ourselves more familiar with the area of research, pinpoint the problem areas, and perform some brainstorming on opportunities in our area of interest.
   - Problem definition: Once a better understanding of the area of research was gained in the background review step, in this step the topic and general scope of the thesis were defined.
   - Research design: In this step, we better defined the steps of the Rigor and Design cycles, including examples to be used for defining and validating our approach, the required systematic review to better understand the state of the art, and validation assessments to be performed at the same time.
2. **Rigor Cycle**
   - Systematic review and validation: We designed a two-step systematic review to first make sure there are no other reviews on our topic of interest, and second to find existing research results on the application of aspect-oriented approaches to business process adaptation and improvement. This step helped us to better understand the state of the art, related research, and to validate our research topic.
3. **Design Cycle**
   - Initial definition of the frameworks: Since the focus of this research was twofold (i.e., enhancement to monitoring and decision making in businesses as well as process improvement using an aspect-oriented approach, namely the Aspect-oriented User Requirements Notation – AoURN), we

first performed some research on each front independently to gain better understanding of the scope and identify the potential contributions.

- Initial validation using real-world examples: We also used two distinct real-life examples to illustrate and validate the main aspects of our research.
- Integration and detailed definition of the framework: We merged the two overlapping frameworks that we had proposed (for decision making and for aspect-oriented process improvement) into one integrated framework, introduced in this thesis.
- Evaluation using the *Mandatory Elements of Method* (MEM): In [155], Zellner suggests that for a Business Process Improvement (BPI) methodology to be considered complete, it has to have five main elements: 1) Procedure model, 2) Technique, 3) Results, 4) Role, and 5) Information Model. He also evaluates some of the existing BPI methodologies using this structured approach. We evaluate our proposed framework using the same approach.
- Further validation using real world examples: In this step, we further validate the aspect-oriented approach by defining more generic aspect-oriented based redesign patterns and by using them in real-life examples. Furthermore, we validate the integrated framework by extending our retail store real-life example to use aspect-oriented based redesign patterns for process improvement.
- The above validation steps were also used as feedback loops for previous cycles. They helped us i) iterate on the framework and problem definition and ii) improve both along the way. For instance:
  - Modeling the generic patterns and real-life examples helped find the required extensions to AoURN and rationalize the problem definition from a fully-automated approach to a more practical semi-automated approach, and
  - The MEM evaluation helped identify and add missing elements to the framework.

- Finally, a pilot experiment was conducted to validate the usability of the notation and patterns. The experiment uses a set of demographical questions to assess the level of participants' expertise in URN and AoURN and a set of technical questions that is the core of the experiment. Participants are also asked to express their opinions about the notation.

1.5. Publications

The following is the list of peer-reviewed publications that are direct or indirect results of the research done for this thesis. The first author represents the main author.

Refereed Journal Publications and Book Chapters

- **Pourshahid, A.,** Amyot, D., Mussbacher, G., Weiss, M., and Shamsaei, A.: "A Systematic Review and Assessment of Aspect-oriented Methods Applied to Business Process Adaptation". *Journal of Software (JSW)*, Vol. 7, No. 8, Academy Publisher, pp. 1816–1826 (August 2012)
- **Pourshahid**, A., Peyton, L., Ghanavati, S., Amyot, D., Chen, P., and Weiss, M.: "Model-Based Validation of Business Processes". Book chapter in: V. Shankararaman, J.L. Zhao and J.K. Lee (Eds) *Business Enterprise, Process, and Technology Management: Models and Applications*, IGI Global, USA, pp. 165–183 (2012)
- **Pourshahid**, A. Mussbacher, G. Amyot, D. Weiss, M.: "Toward an Aspect-Oriented Framework for Business Process Improvement". *International Journal of Electronic Business,* Inderscience Publishers, Vol. 8, No. 3, pp. 233–259 (2010)
- **Pourshahid, A.**, Chen, P., Amyot, A., Peyton, L., Weiss, M., Ghanavati, S., and Forster, A.J.: "Business Process Management with the User Requirements Notation". *Electronic Commerce Research,* Vol. 9(4), Springer, pp. 269–316 (December 2009) (61 references as of April 2014)

Refereed Conferences and Workshops

- **Pourshahid, A.**, Mussbacher, G., Amyot D., Weiss, M.: "Requirements for a Modeling Language to Specify and Match Business Process Improvement Patterns". In: *3rd Workshop on Model-Driven Requirements Engineering (MoDRE 2013)*, IEEE CS, pp. 10–19 (2013)
- Shamsaei, A., Amyot, D., **Pourshahid, A.**, Mussbacher, G., Tawhid, R., Yu, E., Braun, E., and Cartwright, N.: "An Approach to Specify and Analyze Goal Model Families". In: *7th System Analysis and Modelling (SAM 2012)*, LNCS 7744, Springer, pp. 34–52 (2012)
- Shamsaei, A., **Pourshahid, A.**, and Amyot, D.: "A Systematic Review of Compliance Measurement Based on Goals and Indicators". In: *Third Int. Workshop on Governance, Risk and Compliance – Applications in Information Systems (GRCIS 2011)*, CAiSE 2011 Workshops. LNBIP 83, Springer, pp. 228–237 (2011)
- **Pourshahid,** A., Richards, G., and Amyot, D.: "Toward a Goal-Oriented, Business Intelligence Decision-Making Framework". *MCeTECh 2011, E-Technologies: Transformation in a Connected World*, LNBIP 78, Springer, pp. 100–115 (2011)
- Shamsaei, A., **Pourshahid, A.**, and Amyot, D.: "Business Process Compliance Tracking Using Key Performance Indicators". *6th International Workshop on Business Process Design (BPD 2010)*, BPM Workshops, LNBIP 66, Springer, pp. 73–84 (2010)
- **Pourshahid**, A., Mussbacher, G., Amyot, D., and Weiss, M.: "An Aspect-Oriented Framework for Business Process Improvement". *4th International MCeTech Conference on e-Technologies*, LNBIP 26, Springer, pp. 290–305 (2009)

1.6. Contributions and Thesis Outline

This section gives the highlights of the thesis chapters, together with their *major* and *minor* contributions towards solving the problems discussed in section 1.2. The value and impact of these contributions, together with threats to validity and future work, will be further discussed in the conclusion.

Table 1: Thesis contributions and outline

Chapters	**Contributions**
Chapter 1. Introduction	
Chapter 2. Background	• **Minor** o Comparison of several modeling languages using a set of relevant criteria.
Chapter 3. Systematic Literature Review	• **Minor** o Repeatable systematic review on aspect-oriented methods applied to business process adaptation.
Chapter 4. A Framework for Modeling, Monitoring and Adapting Business Processes Using AoURN	• **Major** o An **aspect-oriented approach** that allows organizations to detect the potential problems in their business processes, business goals, and performance models using predefined generic or custom **redesign patterns** and to apply the predefined solutions to those problems accordingly. o An iterative and incremental **framework** for bringing organizations to a state that allows them to monitor and improve their business processes.
Chapter 5. Detection and Improvement Patterns	• **Major** o Modeling of several generic **process redesign patterns** using AoURN. o Identification of **requirements for a process modeling language** to capture the models to support the proposed approach. o A set of **proposed changes to AoURN** as a profile called *Business Process Pattern (BPP) Profile*, to realize the discussed requirements.
Chapter 6. Enhanced Performance View	• **Major** o A method for defining **relationships between the Key Performance Indicators** of organizations on performance models using mathematical formulas. o A new **evaluation algorithm** for URN's Goal-oriented Requirement Language, with implementation, which allows the evaluation of the models defined using the proposed approach.

Chapters	Contributions
Chapter 7. Healthcare Example	• **Minor** o Models of processes, goals and indicators for a real-life healthcare example.
Chapter 8. Retail Business Example	• **Minor** o Models of processes, goals and indicators for a real-life retail example.
Chapter 9. Validation and Evaluation	• **Minor** o Evaluation of the framework based on Mandatory Elements of a Method.
Chapter 10. Conclusions	

Chapter 2. Background

In this chapter, we review some of the background knowledge required by the reader to better understand the core contributions of this thesis. We first discuss business process management and adaptation. Then, we elaborate on the shortcomings of current Business Intelligence-based decision-making / monitoring solutions. Finally, we introduce the modeling language used in this thesis and review the related contributions made using this language as well as available tool support.

2.1. Business Process Management

Business Process Management (BPM) has recently gained momentum among e-business technologies. BPM can be realized through methodologies, techniques, or software, in a way that helps organizations bring together processes and their context including people, documents, information sources, organizational structures, and applications [141]. As a methodology, BPM helps organizations gain control over their business processes by modeling, validating, analyzing, and monitoring them. BPM provides process visibility for the organizations, and hence makes both human-centric and electronic-centric processes more manageable [35].

A BPM methodology is typically an iterative lifecycle composed of several steps and usually starts with the modeling and validation of the business processes. The next steps in the lifecycle are the automation and execution of these processes. Then, the processes are monitored and, based on the monitoring results, they may be redesigned and improved to better achieve the expected goals [114]. In addition to such methodologies, *Business Process Analysis* (BPA) ontologies have been suggested to make the analysis effort more effective and to reduce the gap between information technology (IT) and the business world [110]. Furthermore, there has been some work done by Reijers [123] to capture common improvement approaches in the form of *business redesign patterns* that contribute to the improvement of processes from four main perspectives, namely time, quality, cost, and flexibility. Although these redesign patterns could be used as guidelines for the improvement step, most of the improvement methods do not utilize these patterns. On the contrary, existing methods rely heavily on human innovation and creativity rather than on rationality. As suggested by Feng *et al.* [45], using patterns can help to further rationalize and formalize process improvement methods.

Software applications supporting BPM are often called *Business Process Management Systems* (BPMSs). Existing BPMSs, such as Appian enterprise BPM suite, G360, Tibco iProcess Suite, EMC BPM Suite, and Fujitsu Interstage BPM Suite [20], provide various methods for process monitoring. Kronz [80] explains that monitoring in these systems is usually done by defining calculation rules for specific measurement points, which quantify important business concerns called Key Performance Indicators (KPIs). However, these systems usually do not provide the means for process improvement. Moreover, they do not support the application of process redesign patterns. Therefore, improvements can be done solely based on human knowledge and experience.

Furthermore, available process modeling notations such as the Business Process Model and Notation (BPMN), the Unified Modeling Language (UML), Event-driven Process Chain (EPC), Yet Another Workflow Language (YAWL), and Integrated DEFinition for Process Description Capture Method (IDEF3) do not provide means to observe or simulate the impact of redesign patterns on KPIs and business goals before the patterns are implemented in the system [114]. They also do not allow the comparison of candidate redesign patterns with respect to their impact on high-level business goals. Therefore, support for effectively selecting the most appropriate pattern for the current business condition is very limited. Essentially, these features are not supported because there is no method to automatically reflect changes to the process model in the KPIs and business goal models – possibly even based on historical expectations. Hence, a change to the process model requires all the other related models to be tracked and modified manually, which can be a tedious and error-prone task.

2.2. Business Process Adaptation and Improvement

Adaptive business processes and service-based systems are gaining a lot of attention in both academia and industry because businesses have to react quickly to changes in the market place. Customers now expect custom and personalized services while competitors offer new services every day that businesses have to keep up with. Furthermore, when considering suppliers of different services, especially electronic ones, there are many options available to choose from. Meanwhile, businesses have to constantly monitor, evolve and improve their processes to reduce their costs and increase customer satisfaction. Finally, businesses have to deal with new constraints including standards, *Service Level Agreements* (SLAs), and legal requirements. According to a Gartner survey in 2009, business process improvement was the number one priority of 1,526 CIOs who participated in the survey [155]. Unless a reliable infrastructure is provided to allow businesses to constantly monitor their processes using best practices and to constantly react to changing conditions while considering their ultimate business objectives, businesses will find it difficult to keep up with competition and react promptly to market demands.

According to existing studies [26][30], most of the current approaches only target adaptation partially with a narrow focus and the existing frameworks are not comprehensive. As suggested by Zellner in [155], although there is much discussion surrounding process improvement, there is still insufficient support and methodological explanations related to the "act of improving the process, and improving itself seems to be a black box". Furthermore, current adaptive systems suffer from several issues such as focusing on process *instances* as opposed to process classes, being reactive as opposed to proactive, using rigid and inflexible specifications, requiring human intervention, and not considering the business context [72]. For instance, much of the existing research focuses on fault handling, service replanning or replacement, and requirements [106][121][127]. Process adaptation should be considered at all levels of the organization and not just in the aforementioned areas. Using existing technologies, the migration of running process instances to a newly improved process model can be challenging and needs special attention. Although some aspect-oriented approaches (and non-aspect-oriented ones [122]) have addressed such technology limitations, there are some concerns around the complexity of the infrastructure [121] and the learning curve [42] involved in aspect-oriented approaches.

2.3. Business Intelligence-Based Decision Making

The fundamental goal of BI is to enable informed decision making. Ko and Abdullaev [76] estimated, however, that more than 50% of BI implementations fail to influence the decision-making process in any meaningful way. Reasons for this include cultural resistance, lack of relevance, lack of alignment with business strategy, and lack of actionable and "institutionalized" decision support technologies [51]. Many of these problems could be attributed to approaches used for defining the data to be delivered by BI tools.

The role of BI tools can be categorized as data support and decision support [63]. *Data support* is related to the delivery of accurate, up-to-date data. *Decision support* is related to the assistance provided to the user in actually making a decision based on the available data. Most BI systems do a good job of data support, but data support does not necessarily translate into decision support since the data can be delivered in a variety of formats that might or might not fit with the task the user needs to accomplish. Korhonen *et al.* [78] point out that one of the key challenges faced in institutionalizing decision aids is the validation of decision models used by decision makers. The point here is that delivery of data that does not match the way in which the user plans to use it requires manipulation, thus slowing down the decision-making process.

This issue of fitting data presentation to decision-making tasks was a topic of research in the Information Systems (IS) literature in the 1980s and 1990s, as discussed by Benbasat *et al.* [14][15][16][17] as well as Bruggen *et al.* [21]. One of the most important arguments emerging out of this literature is the notion of cognitive fit, a condition described by Vessey [145] that results when a good match exists between the problem representation (i.e., in this case the way data is presented by the BI tool) and the cognitive task (the way data is used) involved in making decisions.

The concept of cognitive fit is supported by research in the field of behavioral decision making, which demonstrates that decision makers tend to make better use of information that is explicitly displayed [135]. The less data manipulation required by the user, the more explicit the display [67]. Huang *et al.* [60] suggest that, based on Cognitive Load Theory, explicitness enhances cognitive fit resulting in lower "cognitive load" (i.e., reduced mental effort by the user), which facilitates the decision-making process.

Cognitive fit is enhanced when data is presented in a form that fits well with the process the decision maker uses to make decisions.

In terms of this decision-making process, the key output in goal-directed systems is improving the likelihood of goal accomplishment. Goals are influenced by activities in the organization that often call for resource allocation. For example, should a manager invest more in advertising in order to improve revenues or would an investment in training have more of an impact? These types of decisions call for an understanding of associations between variables (i.e., impact of advertising and training on revenue growth). The problem is that in most BI tools, such associations are not defined. Decision makers need to process the data by first identifying the cause-effect model then estimating the strength of the associations between variables. According to Popova and Sharpanskykh [112], even when relationships can be defined such as in the ARIS model [35], which allows users to define cause-effect relationships using Balanced Scorecards and connect KPIs to strategic goals, the analysis options are inadequate due to a lack of formal modeling foundations and proper representations. There were some attempts in the industry (e.g. [61]) to address this issue using *strategy maps* [69]. However, such attempts do not reflect the process models of the organization and rely on predefined rigid models that get in the way of process adaptation.

The lack of formal modeling foundations leads to higher mental effort on the part of the decision maker, increasing the cognitive load and thus reducing decision efficiency. The graphs versus tables literature (as studied by De Sanctis [36] and Vessey [145]), for example, argues that decisions can be improved (i.e., faster and more accurate decisions can be made) when values for each of the variables in the data set are displayed in their proper context. We argue therefore that if the decision model to be used by the decision maker (i.e., the cause-effect relationships among variables relevant to the decision) is displayed by the BI tool, and if the model is linked to the decision's context (in this case, the desired strategic outcomes), BI-based decision-making can be greatly enhanced. This argument is also made by Yu *et al*. in the context of the development of the Business Intelligence Model (BIM) language [153].

2.4. User Requirements Notation

The *User Requirements Notation* (URN) is an International Telecommunication Union standard for capturing early requirements in the form of scenarios and goals [6][9][62]. URN consists of two complementary sub-languages called *Goal-oriented Requirement Language* (GRL) and *Use Case Maps* (UCM) for goal modeling and scenario modeling, respectively. URN facilitates the elicitation, specification, analysis, and validation of early requirements expressed in the form of goals and scenarios. URN's unique capabilities for modeling both processes with UCM and goals with GRL in a unified way are a significant advantage over other process modeling notations. The integrated view of UCM and GRL not only answers the *where*, *what*, *who*, and *when* questions of process models, but also answers *why* a particular part of a process exists. Furthermore, URN's analysis capabilities can be used to evaluate the goal model for trade-off analysis among stakeholder goals and to establish a test suite for the scenario model.

The latest version of the URN standard (released in 2012) also includes supports for indicators in GRL, based on our previous work [114][119]. This further adds to the suitability of this notation for business process modeling, monitoring, and management.

2.4.1 Comparing Potential Modeling Notations

We performed a comparison, illustrated in Table 2, between URN (including its aspect-oriented extensions – AoURN) and other potential candidate modeling languages based on some of the existing research [22][59][119] and our own investigation. The following are the basic requirements a modeling language must fulfill to allow organizations to model their business processes and organizational goals as well as to define, detect, and apply Business Process Improvement patterns.

- *Goal modeling / intentional facets of business process*: One of our main requirements is the support for modeling the goals of the organization as well as the impact of business processes on these goals, in order to be able to see how well a process meets the overall stakeholder goals and to support trade-off analysis. Although there are many languages that support process modeling and that are widely used in research and industry (e.g., BPMN [92]), there are however very few languages that also support goal modeling.

- *Process modeling*: One of the main objectives of this thesis is to allow one to discover, model and improve business processes in an organization. Although there are popular languages that enable modeling goals well (e.g., *i** [152] and KAOS [34]), they do not support process modeling as part of the main language. There may be non-standard extensions or integration between these languages and other process modeling languages (e.g., BPMN) that provide both goal and process modeling. However researching on these extensions and finding proper tool support, as required for the example-based evaluation of our method, is out of scope of this thesis.
- *Performance / KPI models*: A suitable language for our research has to allow connection between organization goals, process models, and performance models. This can be done best through indicators/KPIs [153].
- *Evaluation*: the evaluation of the goal and performance models and an ability to perform analysis on them are other important factors in the selection of our modeling language [57]. One should be able to observe the effect of the changes that are done on the processes for improvement or the effect of the decisions made in the organization goals and performance models. The evaluation should also allow for business monitoring in order to assess how well a process meets the overall stakeholder goals and KPI targets.
- *Support for coordinated, heterogeneous patterns matching* (aspect-oriented approach) against behavioral and intentional models plays a significant role in matching and deploying process improvements. Therefore, the candidate language has to support some aspect-oriented-like process and goal modeling and allow one to integrate these concepts. Unless such support is available, structural as well as context-related properties cannot be detected in a business process model, and the specified improvements cannot be applied to that model.

Furthermore, the current state of tool support is also vital for the success of this research. Major tool development is not within the scope of this thesis. Therefore, it is vital to for us to choose a language with tool support at the level that allows us to experiment and validate our framework. The tool has to support the above modeling features and provide integration with external data sources providing KPI values.

Table 2: Comparing potential modeling notations

Languages	Goals	Processes	Performance	Evaluation	CPM
AoURN [93]	Y	Y	Y	Y	Y
URN [62]	Y	Y	Y	Y	N
EEML [79]	Y	Y	N	N	N
*i** [152]	Y	N	N	Y	Y
BIM [58]	Y	N	Y	Y	N
NFR [91]	Y	N	N	Y	N
KAOS [34]	Y	N	N	Y	N
Tropos [19]	Y	N	N	Y	N
BPMN [92]	N	Y	N	N	Y
EPC [84][131]	N	Y	N	N	N
UML-AD [84]	N	Y	N	N	Y
YAWL [1]	N	Y	N	N	N
IDEF3 [84]	N	Y	N	N	N
Petri Nets [111]	N	Y	N	N	N

Y: Supports N: Does not support CPM: Coordinated pattern matching

Note that in Table 2, *i**, BPMN and UML Activity Diagrams (UML-AD) do not support aspects and coordinated pattern matching per se, but tentative aspect-oriented extensions have been reported in the literature, for instance by Alancar *et al.* for *i** [3], Patiniotakis *et al.* for BPMN 2.0 [109], and Charfi *et al.* [24].

Hence, although several modeling languages other than AoURN exist, the support for KPIs and performance modeling, the ability to combine process and goal models and perform analysis on both, the support for coordinated pattern matching and the fact that URN is a standard modeling language, all together are unique and have convinced us that AoURN is the best language to use in the context of our research. Furthermore, existing tool support for using BI systems as a source of data is an additional reason justifying the selection of AoURN. URN and AoURN are hence further detailed in this chapter.

2.4.2 Goal-oriented Requirement Language

The Goal-oriented Requirement Language (GRL) is a graphical notation used to model and analyze goals. GRL enables the modeling of stakeholders, business goals, qualities, alternatives, and rationales. Modeling goals of stakeholders with GRL makes it possible to understand stakeholder intentions as well as problems that ought to be solved. GRL enables business analysts to model strategic goals and concerns using various types of intentional elements and relationships, as well as their stakeholders called actors. Core *intentional elements* include goals, softgoals for qualities, and tasks for activities and alternative solutions. Intentional elements can also be linked by AND/OR *decomposition*. Elements of a goal model can influence each other through *contributions*, displayed as arrows. Qualitative positive (make, help, some positive) and negative (break, hurt, some negative) contribution levels exist, as well as quantitative contribution levels on a scale going from -100 to +100. GRL captures business goals of many stakeholders, alternative solutions that are to be considered for a system and how they impact stakeholder goals, decisions that were made, and rationales that helped make these decisions. The notation elements are summarized in Figure 2.

Figure 5 shows an example of GRL model. Three types of intentional elements have been used in this model. Softgoals (⬭, e.g., Reduce Costs) describe the goals that are strived towards but for which there is no clear-cut criteria or state that indicates full achievement, e.g., Reducing costs is a continuous goal. Tasks (⬡, e.g., Approval Process) model potential solutions for achieving higher-level goals. Finally, Key Performance Indicators (KPIs) (⬡, e.g., Number of Mistakes) indicate metrics of the system normalized to a scale of -100 to 100 [116]. Furthermore, KPIs are assigned to time (🅣), cost (◉), and quality (◉) categories in Figure 5. In addition, URN traceability links (▶), which can link any pair of elements in a URN model, are used here to relate tasks in this goal model to their representation in the corresponding UCM process model (Figure 6).

GRL evaluation strategies enable modelers to assign initial satisfaction values to some of the intentional elements (usually alternatives at the bottom of a goal graph) and propagate this information to the other elements through the decomposition, dependency and contribution links. Strategies act as *what-if* scenarios that can help assess the impact of alternative solutions on high-level goals of the involved stakeholders, evaluate trade-offs during conflicts, and document decision rationales. Amyot *et al.* [7] discuss different goal evaluation algorithms (using qualitative values, quantitative satisfaction values between -100 and +100, or mix of both types) for GRL.

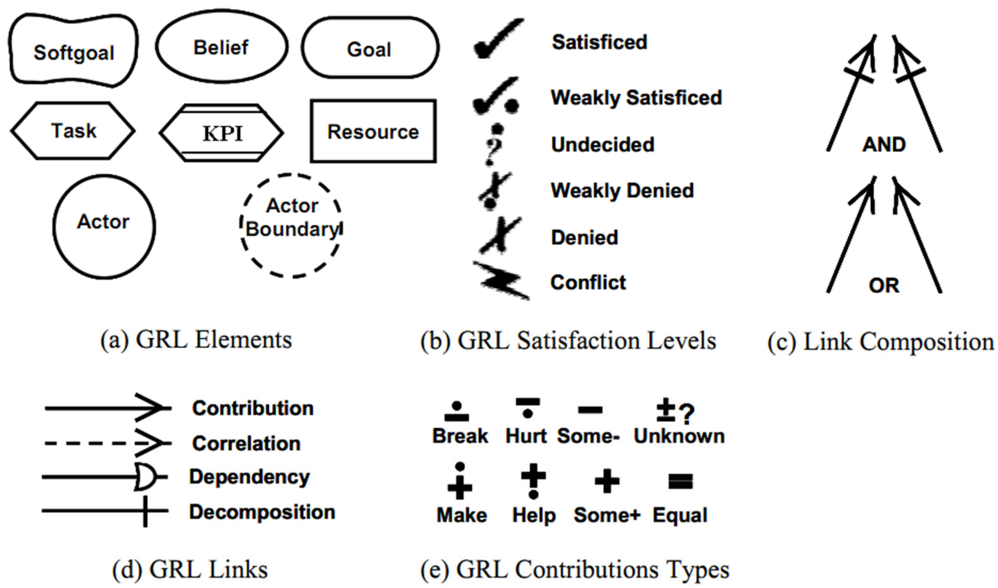


Figure 2: Basic elements of the GRL notation

2.4.3 Use Case Maps

The Use Case Map (UCM) notation is a visual process modeling language for specifying causal scenarios and optionally allocating their activities to an underlying structure of components and actors. UCMs model scenarios and processes with causal relationships, where *responsibilities* (✕) are sequenced and may be assigned to components. A UCM path begins at a *start point* (●) and ends with an *end point* (▌). *Stubs* (◇, e.g., Ethical Review in Figure 6) are containers for sub-models and denote here the three major steps in the process (i.e., perform an ethical review, privacy review, and technical review). Expanding a stub leads to a sub-map (the stub's plug-in), which provides more details about the step. For instance, as illustrated in Figure 6, the Privacy Review stub has a sub-map with four steps. These steps are defined using responsibilities, which are used to specify atomic tasks (i.e., tasks for which we do not have more detailed information) or tasks that we do not want to represent in more detail. Note that in this example, each stub has two possible exits: one if a review approved the data warehouse access request and one if it was rejected. This is also reflected in the sub-map of the Privacy Review stub as it has two end points. UCMs can be used to model as-is and to-be business processes. A summary of the UCM notation is illustrated in Figure 3. Some of the more advanced notational elements include timers, failure paths, and various kinds of dynamic stubs and of components.

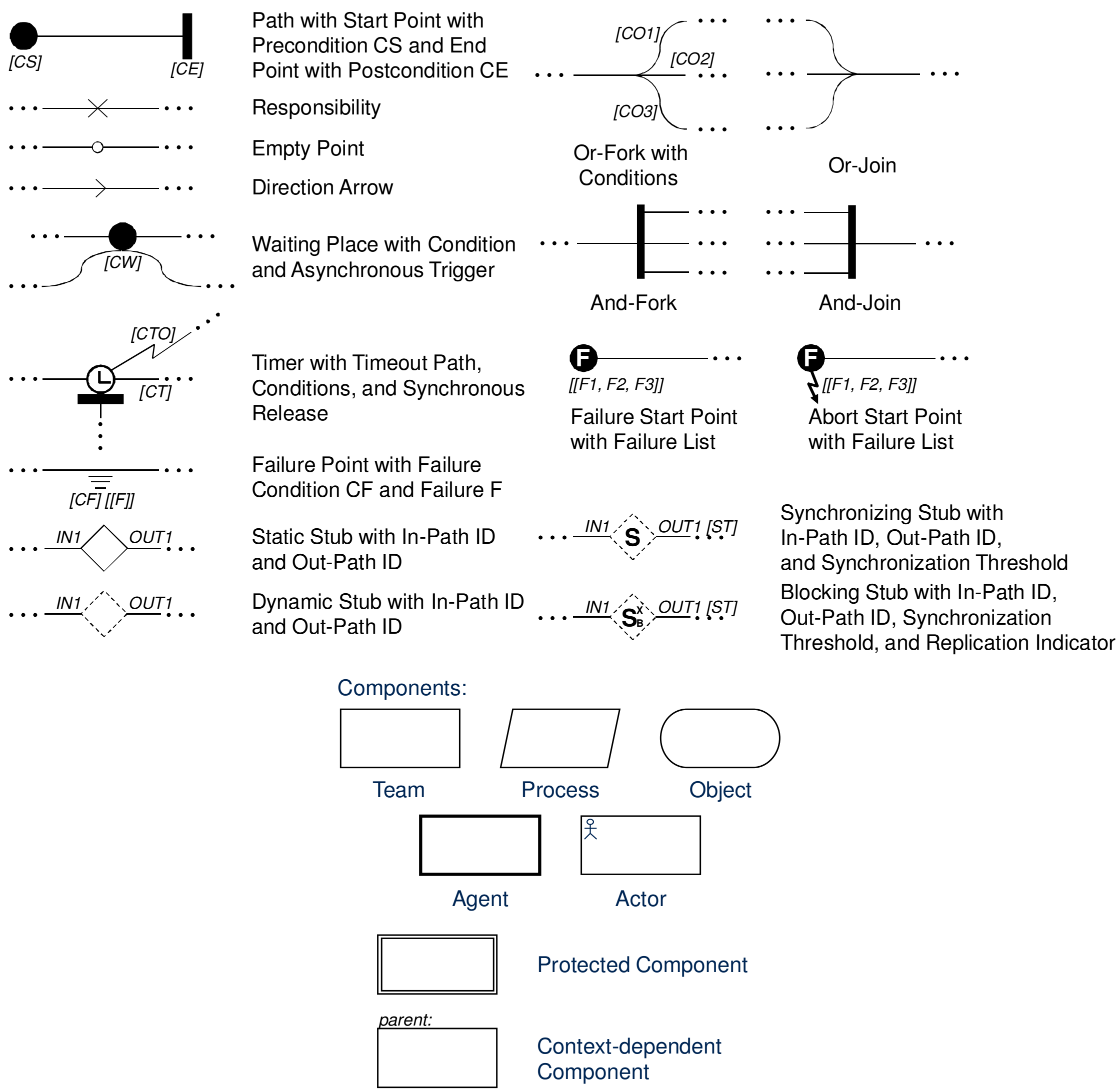


Figure 3: Basic elements of the UCM notation

2.5. URN and BPM

Although the primary application domains for URN target reactive systems and telecommunications services, this language has also been applied successfully to the modeling and analysis of business goals and processes [147]. Goal models combined to process models have been used elsewhere to assess the risk and viability of business solutions [12] and model different concerns of interest to different stakeholders [29].

URN's unique capabilities for modeling both processes with UCMs and goals with GRL in a unified way is a significant advantage over other process modeling notations. The integrated view of UCM and GRL not only answers the where, what, who, and when questions of process models, but also answers why a particular part of a process exists. Using URN, people with sufficient business knowledge and experience can align business goals and processes [84]. While the first version of the URN standard (2008) required better support for exception and cancellation handling in process and workflow models [94], the latest version (2012) solves this issue with the addition of failure points and failure paths.

In order to better support business process monitoring and performance management, Pourshahid *et al.* [114] have extended the first version of GRL with the concept of *Key Performance Indicators* (⬭), now standardized in URN as of 2012. In the extension of Pourshahid *et al.*, KPIs can also be analyzed from various angles called dimensions (⬚, not yet part of the URN standard), in a way similar to what is found in common BI systems. Dimensional data allows one to look at the data from different points of view and filter or aggregate the data based on the defined dimensions. For instance, in Figure 4, staffing cost can be aggregated in all locations in all years of store operations or can be analyzed for Store1, 2, 3 and the online store individually and in a specific month or year.

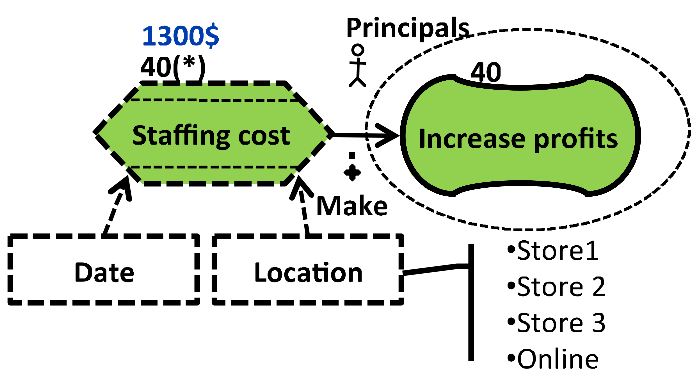


Figure 4: Example of a KPI with dimensions and an evaluation

From a static semantics perspective, KPIs and intentional elements share a common super-class in the URN meta-model, which allows them both to be associated to other GRL intentional elements (such as goals and tasks) and even other indicators through decomposition and contributions links. However, quantitative KPIs additionally include specifications of *worst*, *threshold*, and *target* values in a particular *unit*. For example, a Staffing cost KPI (see Figure 4) could have a target value of $1000, a threshold value of $1,500, and a worst value of $2,500. KPIs also contain a *current* value, which is either defined in a GRL evaluation strategy or provided by an external source of information such as a database, an Excel sheet, a BI tool, external sensors, or web services. Chen [25] and Johari [66] have explored and implemented different solutions for feeding external data to KPIs in their theses.

The KPI is a metric of the system that normalizes the current value to a [-100, 100] scale, which enables it to be used like any other intentional element in a GRL model. For instance, if the current Staffing Cost is $1,300, then the normalization function, which takes here |threshold-current| / |threshold-target|*100, will result in a satisfaction level of 40. Furthermore, when the current value is between the threshold value and the worst value (e.g., $2,500), then the normalization function becomes |threshold-current| / |worst-threshold| * (-100), which results in a negative value (e.g., -100 here). If the result is higher than 100, then it becomes 100 (symmetrically, if it is lower than -100, then it becomes -100). Such an evaluation strategy was used in Figure 4. After the satisfaction level (i.e., 40) is calculated, the usual GRL quantitative algorithm is used to evaluate the satisfaction level of the goals connected to the KPI (i.e., Increase profits). Note also in this model that Staffing cost could be drilled down (e.g., explored) according to the Date and Location dimensions.

Figure 5 and Figure 6 illustrate a hospital's Data Warehouse Approval Process (DWAP). Figure 5 shows the goals and performance indicators of DWAP and consists of GRL intentional elements linked together using contribution links. The *goal view* part at the top of Figure 5 shows that the highest-level goals of the hospital's DWAP are to increase health care quality, to reduce costs, and to comply with regulations.

The *performance view* part at the bottom of Figure 5 shows several tasks that may be performed for the hospital's DWAP such as Ethical Review, Go over Check List, and Privacy Review. Furthermore, several system indicators are modeled with the help of KPIs that can be measured in real life and are related to the tasks and higher-level goals as indicated by the contribution links. Examples of KPIs are Average (Avg.) Approval Turnaround time, Avg. Cost per Application, etc.

The Approval Process task in Figure 5 is linked to the whole diagram in Figure 6. This link is illustrated using the URN traceability symbol as well as the letter **a** to distinguish this link from the other two links in these two diagrams. The other two tasks (i.e., Ethical Review and Privacy Review) are linked to individual UCM model elements. The UCM model provides further behavioral details about the linked tasks. Given that the modeling of goals and processes is often done iteratively, some elements may not be linked at some point in time (e.g., Technical Review in Figure 6). Completeness and consistency analysis based on the traceability links between the goal and process views helps ensure that missing model elements are detected and that model elements that are not justified are removed.

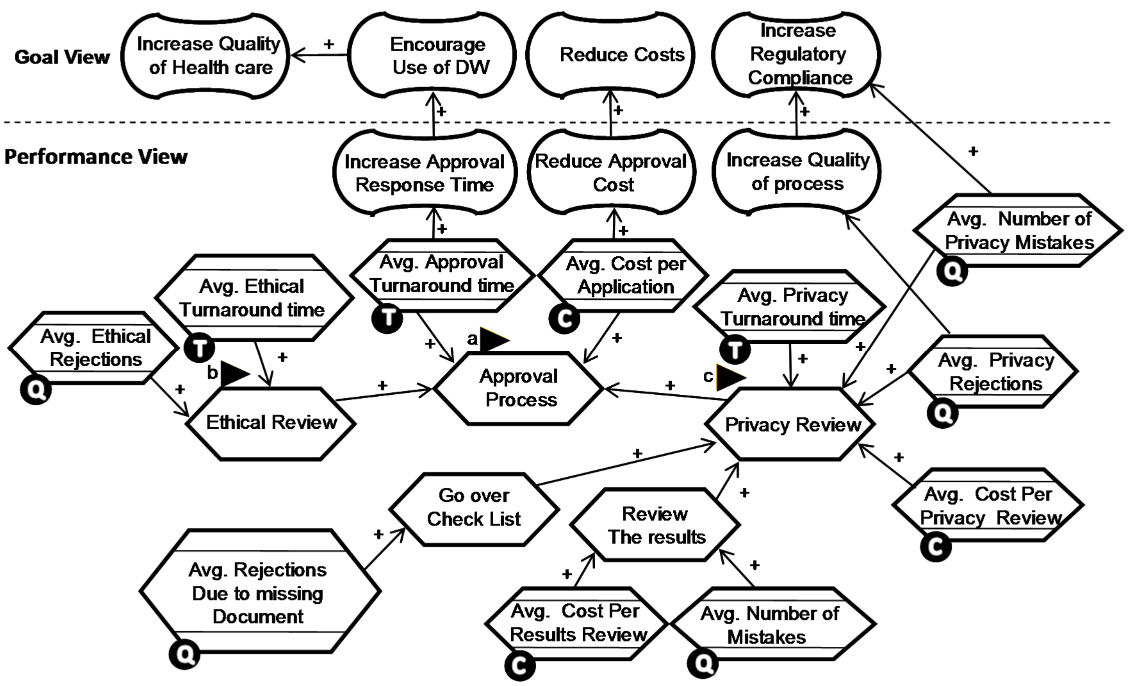


Figure 5: Data warehouse approval process – goal and performance views

GRL does not provide means similar to UCM for modeling hierarchical layers. Therefore, several model elements that appear at different layers of a UCM model may appear in one flat diagram in the GRL model. For instance, Go over Check List in Figure 5 is at the same level as Ethical Review, while in Figure 6, it is captured as a sub task under Privacy Review. However, GRL allows one to have many diagrams in a model, where each diagram provide a partial view of the model (and they could overlap in content).

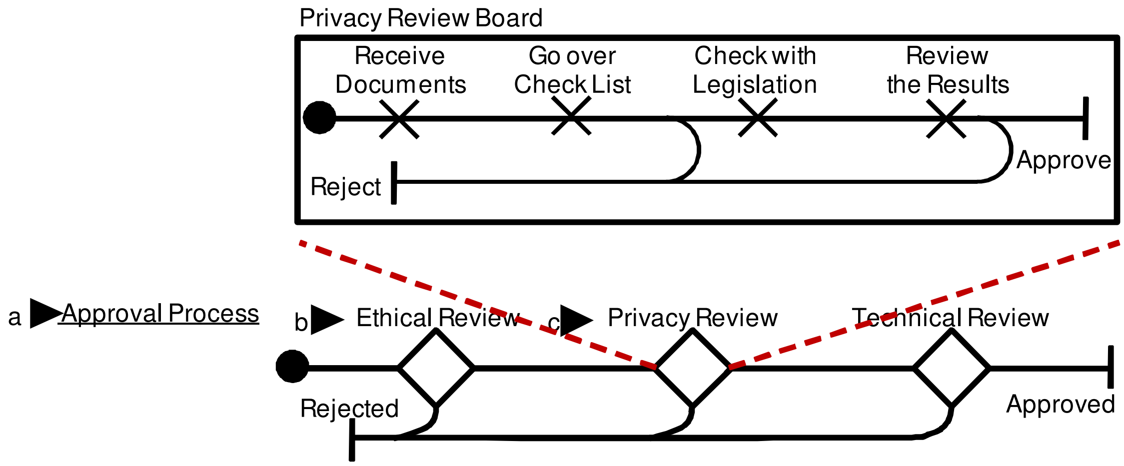


Figure 6: Data warehouse approval process – process view

The BPM framework developed in Pourshahid *et al.* [114] consists of four main views and several components (Figure 7). The process view captures the workflow in the business process with the help of UCM, from very high levels of abstraction down to the task and responsibility level describing the atomic parts of the process. The goal view captures the business goals related to the process with the help of GRL. Goal modeling can focus initially on high-level strategic goals of the business, which are later decomposed into low-level operational goals and even tasks. It is common to see the same tasks shared in both the low-level goal view and the low-level process view. Therefore, the process and goal views can be associated together, defining which part of a process impacts which business goals. In addition, the performance view introduced in [115] is associated with both the process and goal views. It illustrates how processes perform with respect to the business goals using Key Performance Indicators in GRL. Finally, the *validation view* introduced by Pourshahid *et al.* [113] defines the requirements and restrictions against which the process view should be validated. For instance, corporate policies, laws, or service-level agreements can be considered as some validation criteria that need to be satisfied by the process view.

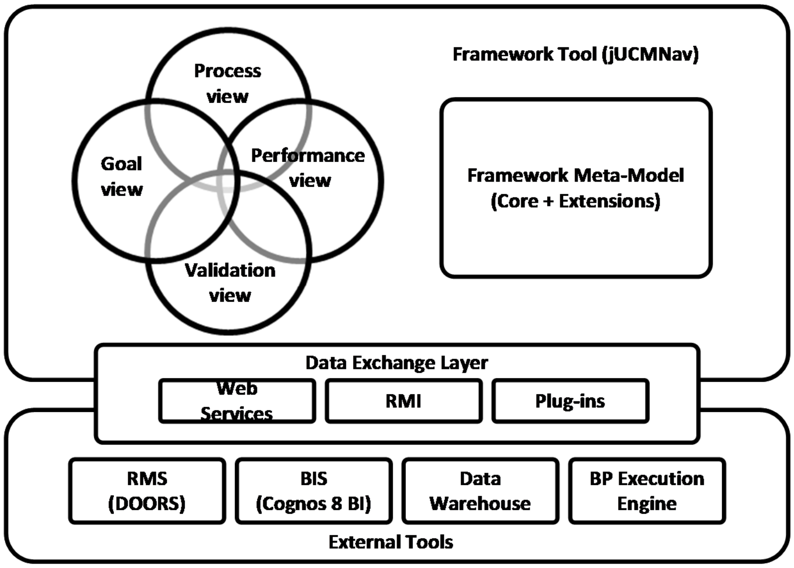


Figure 7: Framework core views and components

The BPM framework uses the built-in evaluation mechanisms of URN to evaluate high-level business goals and other validation criteria based on the satisfaction levels of low-level goals and the values of KPIs. For example, monitoring the Data Warehouse Approval Process (Figure 5) yields process measures that in turn result in initial satisfaction values for the KPIs. These values are then propagated to higher-level nodes in the goal graph until the highest-level goals have been evaluated. Therefore, by measuring KPIs in the real system and utilizing the GRL model, an assessment of the highest-level goals such as increasing quality and reducing cost can be made and subsequently used for supporting decisions about the state of the hospital's DWAP.

2.6. Redesign Patterns

Patterns are reusable solutions to repeatedly observed problems in a specific context [4]. Patterns are often presented in a structured format [47]. The context of a pattern is the precondition that makes the pattern applicable and the solution resolves the described design trade-offs (i.e., the forces involved) [95][96]. There have been several efforts to formalize design patterns [139] and to define and automatically apply patterns [39].

Process redesign patterns are best practices that have proven to be effective in improving business processes. The generalized BPI patterns discussed by Reijers [123] are redesign patterns that can help with process improvement in four categories: cost, time, quality, and flexibility. Table 3 shows the list of redesign patterns in six groups and their impact on the four major improvement categories. Furthermore, in [124], Reijers and Liman discuss the impact of process redesign patterns on various entities involved in business processes, which is also reflected in Table 3.

Table 3: Redesign patterns and their impact on the four improvement categories

Redesign Pattern		Impacts	Time	Cost	Quality	Flexibility
Task Pattern	**Task Elimination**	**Operations**	↑	↑	⇣	N/A
	Task Composition	**Operations**	↑	N/A	⇡	⇣
	Task Automation	**IT**	↑	↑	↑	↓
Routing Patterns	**Resequencing**	**Behavior**	↑	↑	N/A	N/A
	Knockout	**Behavior**	⇡	↑	N/A	⇣
	Control Relocation	**Customers**	N/A	N/A	↑	N/A
	Contact reduction	**Customers**	⇡	⇣	⇡	N/A
	Parallelism	**Behavior**	↑	⇣	⇣	⇣
	Triage	**Operations**	↑	↑	↑	↓
Allocation Patterns	**Case Manager**	**Structure**	N/A	⇣	↑	N/A
	Case Assignment	**Structure**	↑	N/A	↑	↓
	Customer Teams	**Structure**	↑	N/A	↑	↓
	Flexible Assignment	**Structure**	↑	N/A	⇡	N/A
	Resource Centralization	**Structure**	⇡	↓	N/A	↑
	Split Responsibilities	**Structure**	⇣	N/A	↑	N/A
Resource Patterns	**Numerical Involvement**	**Structure**	↑	⇡	⇣	N/A
	Extra Resource	**Population**	↑	↓	N/A	⇡
	Specialist-Generalist	**Population**	↑	⇣	↑	↓
	Generalist-Specialist	**Population**	N/A	⇡	N/A	↑
	Empower	**Population**	⇡	⇡	⇣	N/A
External Party Patterns	**Integration**	**Customers**	↑	↑	N/A	⇣
	Outsourcing	**External**	N/A	↑	⇣	N/A
	Interfacing	**External**	⇡	⇡	⇡	⇣
	Buffering	**Structure**	↑	⇣	N/A	N/A
	Trusted Party	**External**	↑	↑	N/A	N/A
Integral Business Process Patterns	**Case Types**	**Structure**	⇡	⇡	⇣	⇣
	Exception	**Behavior**	↑	⇣	N/A	⇣
	Case-Based Work	**Structure**	↑	⇡	N/A	N/A

↑: Positive Impact ↓: Negative Impact ⇡: Maybe Positive Impact ⇣: Maybe Negative Impact

As seen from this table, each pattern has a different impact on the processes. Therefore, the applicability of these patterns is dependent on the condition and target improvements. For instance, if we want to improve the turnaround time of a process, redesign patterns from the task pattern group are usually good candidates. The same patterns however, may have a negative impact on the quality of the process. Therefore, it is very important to select the right redesign process and observe and validate the results after applying them to the process.

In the context of business process improvement, business process redesign patterns are the *units of interest* to business modelers. These patterns are crosscutting in two ways. First, each pattern may be applied multiple times to a business process. Furthermore, the business process patterns may impact not only the process view but also the goal and performance views. Due to the crosscutting of the patterns and since aspects are often viewed as transformations, and since the purpose of business process redesign patterns is also to transform the current business process, we view business process redesign patterns as *crosscutting concerns*. With AoURN, covered next, these patterns can be encapsulated as concerns in their own modules and selectively applied to an existing process.

2.7. Aspect-oriented URN

The *Aspect-oriented User Requirements Notation* (AoURN) is a modeling framework proposed by Mussbacher that extends URN with aspect-oriented concepts [94][99], allowing modelers to better encapsulate crosscutting concerns that are hard or impossible to encapsulate with URN models alone.

AoURN treats concerns as first-class modeling elements. AoURN groups all relevant properties of a concern such as goals, behavior, and structure, as well as pointcut expressions needed to apply new goal and scenario elements to a base model or to modify existing elements. A *pointcut expression* is a pattern that must be matched in the base model if the aspect is to be applied, thus determining the base model locations to which the aspect is applied (i.e., in the context of this thesis, a pointcut expression determines where a business process redesign pattern is applied).

AoURN adds aspect concepts to URN's sub-notations, leading to Aspect-oriented GRL (AoGRL) [100][103] and Aspect-oriented UCMs (AoUCM) [100][101]. AoURN uses standard URN diagrams to describe *pointcut* expressions; it is hence only limited by the expressive power of URN itself as opposed to a particular composition language.

GRL pointcut expressions are shown on a *pointcut graph* and make use of *pointcut markers* and *pointcut deletion markers* to indicate the pattern to be matched (see Figure 8.a, and also the AoURN notation summary in Figure 11). All elements without pointcut markers are added to the matched location in the GRL model, while elements with a pointcut deletion marker are removed. AoGRL does not distinguish between pointcut graphs and *aspect graph* and combines goals and tasks of the aspects and pointcut expressions in the same graph.

For example, the pointcut graph in Figure 8(a) matches against an actor named Stakeholder, a softgoal with any name, and a goal named Stakeholder Goal C that is linked to the softgoal with a correlation. Note that all these elements are tagged by *pointcut markers* or *pointcut deletion markers*.

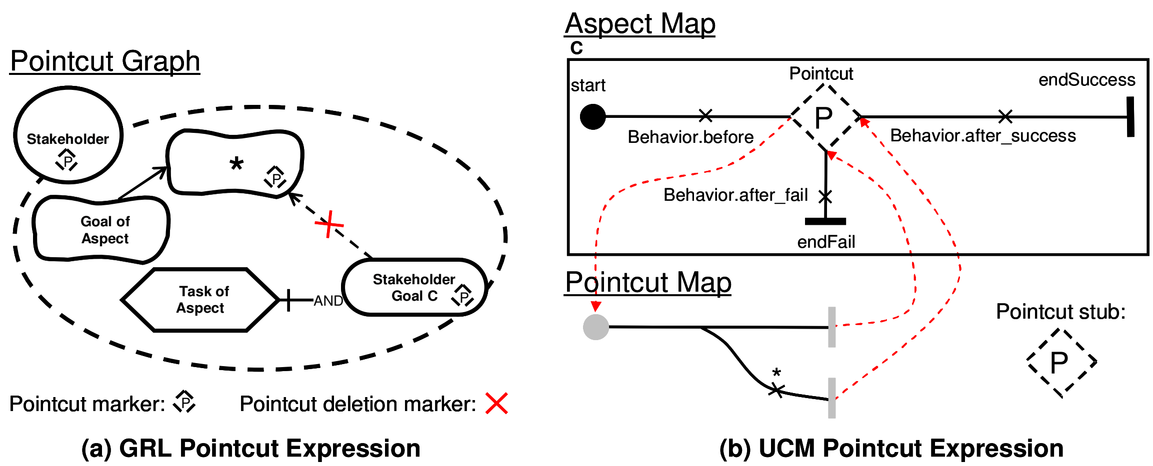


Figure 8**:** GRL and UCM pointcut expressions

UCM pointcut expressions define the pattern to be matched with a *pointcut map*. The aspectual properties are shown on a separate *aspect map*. The aspect map is linked to the pointcut expression with the help of a *pointcut stub*. The causal relationship of the pointcut stub and the aspectual properties visually defines the composition rule for the aspect, indicating how the aspect is inserted in the base model (e.g., before, after, instead of, in parallel, interleaved, or anything else that can be expressed with the UCM notation). For example, the pointcut map in Figure 8(b) matches against an OR-fork followed by a responsibility with any name on one branch.

The base models in Figure 9 match against the patterns described by the pointcut graph and the pointcut map. The pointcut graph in Figure 8(a) is matched because the base model in Figure 9 contains an actor named Stakeholder and a goal named Stakeholder Goal C that is linked by a correlation to a softgoal. The pointcut map in Figure 8(b) is matched because the base model in Figure 9 contains an OR-fork followed by responsibility R1 on one of its branches.

AoURN employs an aspect composition technique that can fully transform URN models. *Aspect markers* (small red diamonds) indicate locations affected by an aspect (Figure 9). When an aspect marker is selected, the modeler is taken to the *AoView* where only those aspectual properties are highlighted that are relevant to the selected aspect marker. AoViews are based on pointcut graphs and pointcut maps which are defined by the modeler and therefore do not require complex layout issues to be automatically resolved for the complete model composed of base and aspectual elements.

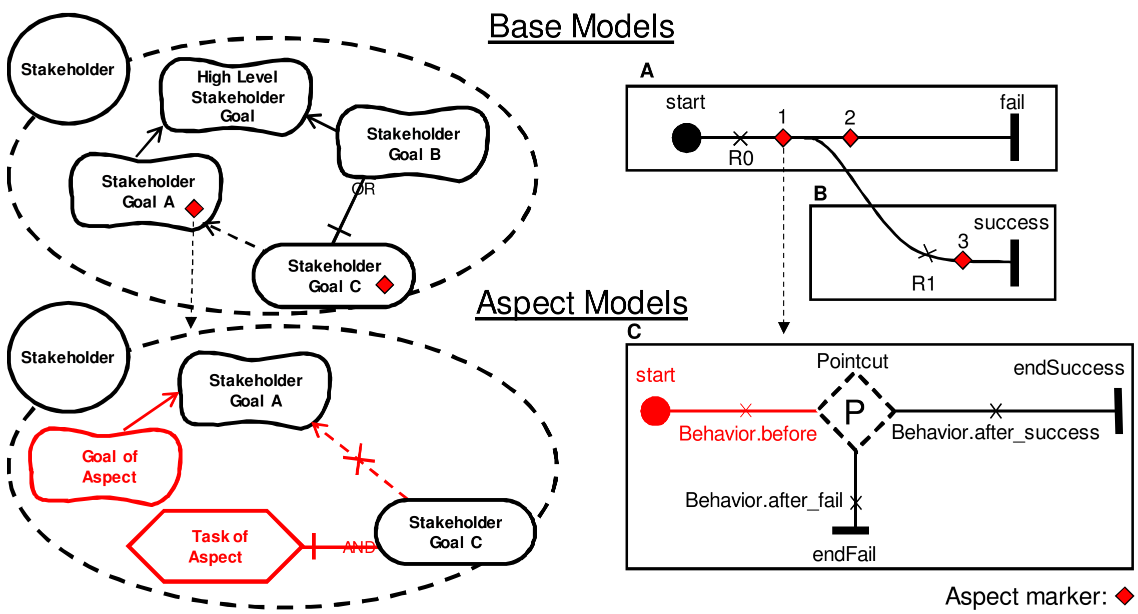


Figure 9: Composed base models with aspect markers and AoViews

For example, the left aspect marker in the GRL base model in Figure 9 indicates that Stakeholder Goal A is affected by the GRL aspect described on the pointcut graph in Figure 8(a). As described by the pointcut graph, the new softgoal Goal of Aspect is linked to Stakeholder Goal A and the correlation with Stakeholder Goal C is removed (shown in the GRL AoView in Figure 9). The second aspect marker in the GRL base model indicates that Stakeholder Goal C has been affected. The AoView of this aspect marker would highlight that the correlation between Stakeholder Goal A and Stakeholder

Goal C is removed and that a decomposition link is added from Stakeholder Goal C to the Task of Aspect (i.e., the AoView highlights everything that changed for the Stakeholder Goal C modeling element).

Similarly, the UCM base model shows three aspect markers and the AoView of the aspect marker 1 is shown in Figure 9. The aspect map in Figure 8(b) defines that Behavior.before is to be added before any match of the pointcut expression because all matches are represented by the pointcut stub in the aspect map. Hence, Behavior.before is highlighted in the UCM AoView. Aspect markers 2 and 3 link to Behavior.after_fail and Behavior.after_success, respectively.

In this thesis we also use a novel *simultaneous pattern matching* concept. In this new type of matching, the pointcut expression is a combination of both GRL and UCM models connected together using URN links. This type of pointcut expression only matches if both parts of the defined pattern, GRL and UCM, are found in the base model. Figure 10 illustrates an example of such pattern. In this example, although the UCM pointcut expression is very simple and can match any responsibility, it only matches to a responsibility that is linked to a task, in the GRL side of the base model, with incoming contributions from KPI elements of specific types (i.e., Quality and Cost KPIs). In addition, this match only happens if the evaluation value of the KPI is less than a significantValue variable defined by the modeler).

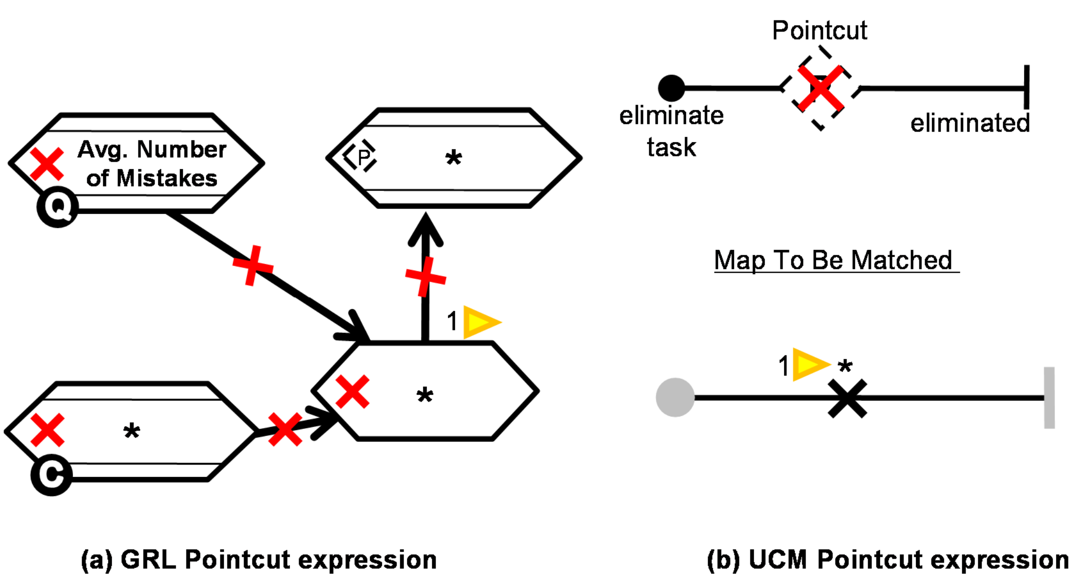


Figure 10: Example of AoURN concurrent GRL and UCM pattern matching

Figure 11 summarizes a subset of the AoURN notation. The *replacement pointcut stub* is used to indicate composition rules that replace the matched base elements with aspectual elements. If such a composition rule is used, *tunnel entrance* and *tunnel exit* aspect markers indicate the start and end of the replacement in the UCM base model, respectively. Finally, the *anything pointcut element* is used on a pointcut map and specifies that any sequence of path elements in the base model may be matched against the anything pointcut element.

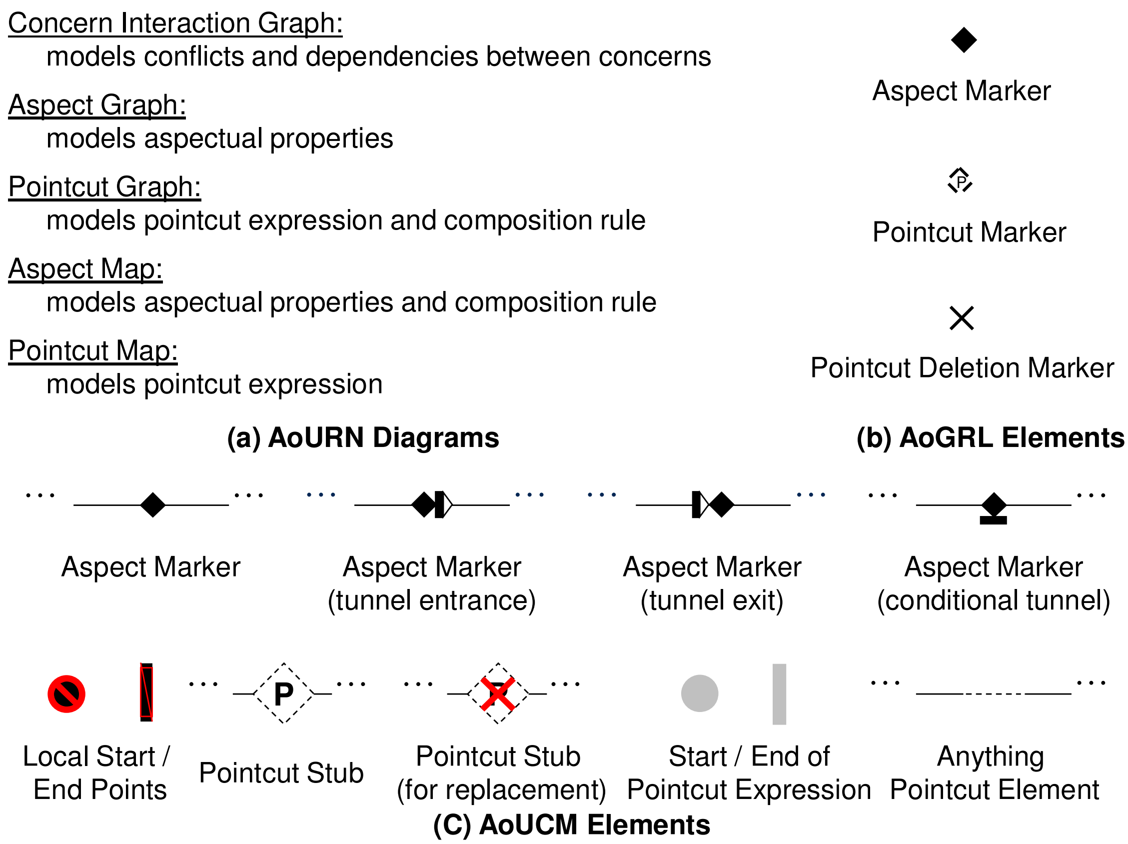


Figure 11: Subset of the AoURN notation

2.8. Current State of Tool Support

jUCMNav [104][124] is the most comprehensive URN modeling tool available today. This Eclipse plugin supports the visual modeling of URN models composed of many GRL and UCM diagrams (Figure 12). On the analysis side, jUCMNav also supports the execution and testing of UCM scenarios as well as quantitative, qualitative, and hybrid propagation algorithms for evaluating GRL models [7].

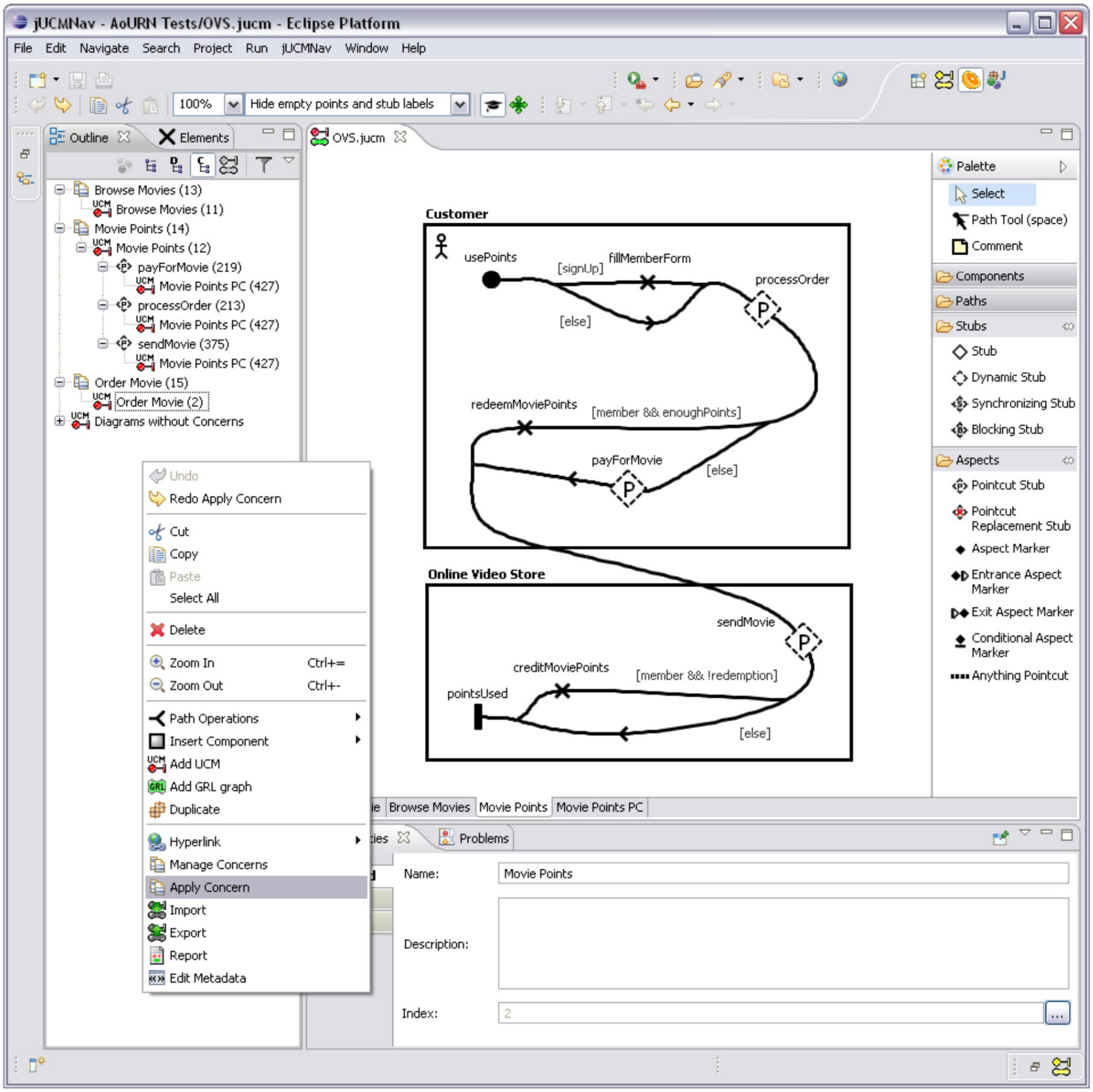


Figure 12: jUCMNav tool, with partial support for AoURN

By formalizing the data exchange layer, external information systems such as data warehouses can be connected to the jUCMNav tool. Integration with external tools including the IBM DOORS Requirement Management System [124] and the Cognos Business Intelligence System [66][114], as well as extensions to URN [115] helped with the development of an integrated BPM framework for process validation and process performance monitoring [113]. In the past, we have documented with jUCMNav processes and their goals in URN models comprised of 50+ interlinked GRL and UCM diagrams.

Chen in [23] and Pourshahid *et al.* in [114] have introduced and implemented a service-oriented architecture enabling the use of underlying data and BI reports by the jUCMNav tool. jUCMNav is connected to BI systems via a web service. All the information generated by the BI system, from raw data to very complex data warehouses, can hence be used as qualitative data to initialize the KPIs used in the GRL model, and against which goal satisfaction is evaluated. KPIs can also be fed directly in GRL strategies imported from Comma Separated Value files.

In addition, Mussbacher [92] added proof-of-concept support for AoURN in jUCMNav (Figure 12). Although this prototype does not provide complete support for AoURN modeling, it supports the matching and composition algorithms for AoUCM. However the matching algorithm for AoGRL is yet to be implemented. Furthermore, AoUCM specification is fully supported while complete support for AoGRL specification requires the manual update of jUCMNav metadata. A proposed list of feature for complete AoURN modeling environment is available in Table 14 of [92].

2.9. Chapter Summary

Business process management is one of the technologies that enterprises and research has been paying attention to over the past several years. Many organizations work towards automating their process using Business Process Management Systems.

Furthermore, businesses look for opportunities to improve their processes to save money and enhance their customers' satisfaction. Various technologies, including Business Intelligence, are used to monitor Key Performance Indicators in organizations to help with such process improvement efforts. In addition, there exist best practices and predefined patterns that support process improvement.

Many process modeling notation exist for modeling business context, including processes and objectives. AoURN is one of these languages. AoURN adds aspect-oriented concepts on top of URN, a standard modeling notation that allows both processes and goals of an organization, as well as their Key Performances Indicators, to be modeled and analyzed using an integrated notation. Our comparison of many languages shows that AoURN offers many of the capabilities required for this research, although even this language needs enhancements to cover requirements of business process improvements in practice, as will be discussed in coming chapters.

The next chapter further explores, through a systematic literature review, how aspect-oriented methods are currently applied to business process improvements and adaptation.

Chapter 3. Systematic Literature Review

In this chapter, we describe a systematic literature review of aspect-oriented methods applied to business process adaptation, conducted in order to investigate related work.

3.1. Introduction

Adaptive business processes and service-based systems are gaining much attention in both academia and industry because businesses have to react quickly to changes in the market place. Customers now expect custom and personalized services while competitors offer new services every day that businesses have to keep up with. Furthermore, when considering suppliers of different services, especially electronic ones, there are many options available to choose from. Meanwhile, businesses have to constantly monitor and evolve their processes to reduce their costs and increase customer satisfaction. Finally, businesses have to deal with new constraints including standards, Service Level Agreements (SLAs), and legal requirements. Unless a reliable infrastructure is provided to allow businesses to constantly monitor their processes using best practices and to constantly react to changing conditions, businesses will find it difficult to keep up with competition and react promptly to market demands.

Research has proposed many different solutions for process adaptation. Among others, aspect-oriented methods have recently attracted much attention. Aspect-Oriented Programming (AOP) [74] has been around for 15 years and has proven to be a useful modularization approach for improving separation of concerns and for composing features and services dynamically. However, only in recent years has research started applying the same ideas to the business process adaptation area.

Although many surveys and literature reviews on business processes were published, to our knowledge no review that specifically targets the application of aspect-oriented methods to business process adaptation currently exists (except for this chapter's review, recently published in [118]). To validate this claim and to evaluate the extent to which aspect-oriented methods are applied to business process adaptation, we have performed a two-step systematic review. The first step of our study shows that among the 40 survey/review papers on business processes we examined, none focuses on aspect-oriented process adaptation. Yet, the second step of our research illustrates that aspect-oriented methods have been significantly applied to business process adaptation and service-based systems adaptation. This motivates the need for a new systematic review of the literature in this area.

Finally, another motivation for this literature review is the validation of our own aspect-oriented process modeling adaptation framework [117], to be introduced in Chapter 4. This review suggests that our framework is not only novel but also realizable. Indeed, much of the research we found in our review focuses only on specific elements of adaptation while our approach offers a more comprehensive view and takes goals, performance, processes, and constraints of the organizations into consideration. In addition, the availability of much prior research in terms of aspect-oriented implementations of business process execution and service composition infrastructures supports the realizability of our framework.

3.2. Research Method

In our research, we followed the approach defined by Kitchenham [81] for systematic literature reviews. We designed a two-phase systematic review. In both phases, we first selected the related work using search engines and cited references. Afterwards, we performed an analysis on the related work. In the second phase, we also conducted a detailed review of a selected subset of initial results.

A well-established body of knowledge exists for the business process arena. Therefore, to assure there is not already a literature review answering our research questions, in the first phase, we looked at existing survey and literature review papers. In the second phase, we focused on studying the existing work on applying aspect-orientation to business process adaptation. This second phase is also used to validate our research agenda from the novelty and feasibility points of view.

3.2.1 Research Questions

We define three main research questions for this study:

1) Is there any existing survey or systematic review related to adaptive business processes?
2) Have aspect-oriented methods been used for business processes adaptation? If yes,
    a. What are the main applications of aspect-oriented methods in the area of adaptive business processes?
3) Is our proposal for an AoURN-based framework novel and realizable given prior research and existing infrastructures?

3.2.2 First Phase Queries

In the first phase of the review and to answer the first question, we used three queries (i.e., *("Business Process" AND Survey)*, *("Business Process" AND "systematic review")*, and *("Business Process" AND "literature review")*) and performed searches on Google Scholar for each query individually. The initial result just for the first query was 54,300 papers. As studying all these papers would be impractical, we limited the search to titles of articles. This reduced the search results significantly to 72 papers.

3.2.3 First Phase Method Details

We started the review process by selecting a subset of the search results using two main criteria. First, we defined six keywords (i.e., adaptation, adaptive, redesign, improve, enhance, and reengineering). We searched for these keywords in the papers and reviewed the papers with any of those keywords. The intention was to make sure we would only review the papers relevant to the research questions we were interested in. Furthermore, we also excluded any paper that was published before the year 2000. Since there has been much research and industry evolution in this area and since using automated methods to evolve business processes is a relatively new concept, we believe any earlier survey is no longer relevant to our topic.

3.2.4 First Phase Result

In first phase, we found 72 papers in total, of which only 40 were written after year 2000. We did not find any survey/review on the application of aspect-oriented methods to business process adaptation. Therefore, we decided to continue with the second phase.

3.2.5 Second Phase Queries

In the second phase of the review process, we used two queries (i.e., *("aspect oriented" AND "business process adaptation")* and *("aspect oriented" AND "adaptive business process")*) and performed the search on the Google Scholar, SpringerLink, IEEE Explore, and ACM Digital Library search engines. 49 papers were returned in total, including 11 duplicates (i.e., 38 papers were unique). Furthermore, when reviewing the papers and their references, we also added 22 other papers from the references to our result set. These were papers commonly cited among the 38 papers in the initial list. Therefore, the total number of papers we analyzed in the second phase was 60. Note that the earlier version of this survey published in 2012 in [118] had only 54 papers and was refreshed in 2013 with six additional papers, covered in this chapter.

3.2.6 Second Phase Method Details

In this phase, we used refined criteria (Table 4) to select a subset of the papers for a deep review that can help us answer the second and third research questions but also to compare the selected papers with our proposed framework, which exploits aspect-oriented adaptation, process measures (Key Performance Indicators) and redesign patterns for process improvement.

Table 4: Second phase study criteria

Criteria
Is the focus on adaptive processes/systems?
Is an aspect-oriented technique used?
Is the focus on process improvement?
Are process redesign patterns used?
Are business goals and KPIs used?

3.2.7 Second Phase Results

We did not find any match that uses all the concepts we are using in our research on process improvement. We observed that the papers using aspect-orientation mainly focus on the techniques summarized in Table 5. In section 3.3, we review the papers that actually help us answer our research questions.

3.3. Summary of Second Phase Papers

In this section, we review the final set of papers selected using the criteria listed in Table 4. We first focus on papers that use aspect-oriented methods for business process adaptation. Then, we turn our attention to papers that use other methods (these papers mainly come from the additional references collected manually). Finally, we review papers that have a more generic look at service-based adaptive systems.

3.3.1 Aspect-oriented Methods

A review of the first group of papers shows that current research using aspect-oriented methods for business process and service adaptation is mainly focused on using the (non-mutually exclusive) techniques summarized in Table 5.

Table 5: Aspect-oriented methods summary

Papers	**Techniques**
[5][43][53][56][107]	Composing and swapping services based on QoS, cost, rules, and in the event of failure.
[107][120][128][136]	Extracting roles and crosscutting concerns from composite services.
[148]	Customizing process instances based on user profiles or SLAs.
[43]	Adaptive service composition and collaboration policies.
[24][144][109]	Using monitoring aspects to detect undesired conditions.

In [53], Hermosillo *et al.* discuss challenges of process adaptation using BPEL and process execution engines including 1) the lack of specification in BPEL to force execution engines to implement a consistent monitoring API, 2) the addition of unnecessary code to the core business process definition, and 3) downtime caused by process redeployment.

The authors propose a framework called CEVICHE (Complex Event processing for Context-adaptive processes in pervasive and Heterogeneous Environments) to address the mentioned problems. CEVICHE uses a Complex Event Processing (CEP) engine to trigger the adaptation aspects after detecting a pre-defined adaptation situation. The events are in fact context information that is monitored in real-time. Furthermore, they use an AO4BPEL engine to perform the adaptation. This approach supports *before*, *after*, and *around* advice types that allows one to execute a task before, after, and both before and after a process step. In the example used in the paper, the adaptation is used in an online car rental store to eliminate optional steps of the renting process when store traffic is high. The CEVICHE framework is further completed in [54] and covers *undoing* the adaptations as well.

In [56], the same authors discuss the same framework, but this time they use a healthcare process as an example and argue that in different situations, different levels of information may need to be gathered about the patient that cannot always be anticipated. Therefore, their framework could help with the adaptation of such processes according to the detected situations. This approach shows a significant enhancement in the process execution infrastructure in terms of adaptation of business processes. However, the adaptation points and the adaptation rules should be predefined in the business process model and can only handle known specific cases. In other words, this is not a generic framework that can be used to improve the design of the process models, but rather one that executes already-known alternative processes after detecting pre-defined situations.

Furthermore in [5], Algahtani and Zedan aim to solve several problems with service-based technologies using a combination of an event-driven architecture and aspect-oriented methods. The targeted problems are the lack of design-time adaptability, lack of testability for composition correctness, lack of behavioral features, and lack of runtime adaptability. Similar to CEVICHE, the proposed system intercepts the events that trigger behaviors based on a set of pre-defined rules. Then, the appropriate behaviors are weaved into the system to address the situation raised by the events.

Similar to the two previous research contributions, Rahman *et al.* [120] propose an Event-Condition-Action (ECA) based architecture using aspect-oriented methods to adapt rule-based service-oriented systems. The goal of this architecture is to increase the adaptability of rules and use of rules in the composition of web services. Rule-based operations are extracted as aspects and applied to join points at run-time. Therefore, when rules are changed, workflows are easily adapted to the new rules.

Furthermore, Patiniotakis *et al.* follow a similar track in [109]. They extend AO4BPMN and integrate it with Situation-Action-Networks (SANs) to enable the detection of situations that require process adaptation. The focus is on monitoring quality thresholds, which then trigger process adaptation using a reasoning mechanism. This research is close to our proposed approach in the sense that it uses a behavioral notation (BPMN) and a goal modeling like notation (SAN) to monitor pre-defined situations and react on them. In this case, the goal model plays a role in monitoring the process model and triggers the adaptation when defined situations are detected, while in our research the two parts of the model work together and a set of predefined patterns is used to detect potential improvements in the business processes.

Narendra *et al.* [107] focus on run-time adaptation of non-functional features (e.g., security and scalability) of composite web services. They propose an aspect-oriented approach and a language for specifying the non-functional properties of composite services. This approach allows for the adaptation of web services without impacting user experience or QoS.

In [128], Sánchez, and Villalobos use AOP to define dynamic and flexible executable workflow definitions. The main argument the authors make for using an aspect-oriented technique in process modeling is the presence of process parts that are not necessarily domain related but that play a more supportive role in the process model. They argue that process definitions are often tangled with extra tasks that are meant to support the main activity. Some examples of such tasks are recurring tasks like data storage, resource allocation or restrictions (e.g., maximum time allotted to complete a task). Although the paper focuses on the dynamicity of process models, the intention is the reuse of common tasks and services and easier extension of the process model, not the monitoring and improvement of the models.

Stearns and Piccinelli [136] describe the crosscutting nature of business transactions in terms of organizational boundaries. A business process (e.g., order process), when executed in an e-business environment, usually involves several functional units of the organization as well as other external organizations (e.g., suppliers) to fulfill the requirements of the process. Furthermore, the paper discusses increased challenges in terms of process adaptability due to business environment changes or customer circumstances. According to the authors, processes within the boundaries of the organization are easier to adapt than processes that go beyond the organization's boundaries. To address this problem, the authors suggest capturing the processes in the form of aspects that can be automatically projected depending on requirements. They suggest the separation of roles from the actual function (e.g., transfer of merchandise). They discuss an example showing that a transport function can be assigned to three different roles (buyer, seller, or a third-party supplier). Depending on the business model or customer requirements, the business process structure and flow remain the same, but the role that executes the delivery part of the process changes.

In addition, Wen *et al.* [148] discuss an approach for using aspect-oriented technology to provide personalization and customization in IP Multimedia Subsystem Networks. The authors argue that existing solutions are not flexible enough for today's agile and dynamic environments. Service providers need to be able to react to changes more rapidly and be able to customize services more dynamically using a service control layer proposed in this paper. The approach suggested by this paper achieves this control using the AOP paradigm to model service control requirements (e.g., authorization or event-based charging) and apply them to the appropriate points in the services without changing and redeploying the core composite service.

In [43], Erradi *et al.* discuss an ongoing effort regarding the development of an aspect-oriented service composition method with the goal of increasing configurability and dynamicity of web services. The framework will help with the adaptation of business rules, collaboration policies between web services and the addition of functional and non-functional extensions to core services. The motivation for this work is the shortcoming of existing approaches in terms of adding optional extensions as well as applying crosscutting concerns. Furthermore, there is a lack of proper approaches for responding to new requirements and changes in business rules. To address all these challenges, the authors augment existing frameworks with aspect-oriented features so that core functionalities can be easily extended. The new proposed framework is called AdaptiveBPEL. Likewise, Charfi *et al.* [24] provide a good infrastructure, complementary to ours, in terms of the adaptation of business processes execution environments. However, it lacks the holistic view of the organization and looks at the building blocks of the composite services as opposed to the overall performance with respect to organization goals and performance views.

In [23], Charfi and Mezini introduce a container for AO4BPEL, which is a well-known work in the research community and is considered the basis for several subsequent publications by these authors. In [24], Charfi *et al.* build on their previous work, using aspects, to achieve web service composition and propose a flexible plug-in based architecture allowing self-adaptation logic to be deployed on the running process instances. The authors argue that manual fault management in the unstable web services environment is not the right approach, because the process instances executing important business transactions using composite services could be interrupted. In addition, the cost of manual intervention is high. Furthermore, the authors believe other suggested approaches based on extensions to orchestration engines as well as BPEL processes consisting of self-healing logic are not satisfactory in terms of extensibility, flexibility, and scope. The authors suggest two types of aspects to address the shortcoming of other approaches: monitoring aspects and adaptation aspects. While the former is used to detect faulty situations, the latter addresses the detected problems. The authors have developed three generic plug-ins including one for replacing faulty services, one for detecting and reacting to service policy updates, and a third one for monitoring of services SLA and changing services when they do not satisfy the requirements. Although the flexibility is impressive in terms of providing modifications to composite services, the work is limited to monitoring the service SLAs as opposed to the process as a whole with respect to the organization goals. The objective of this work and of any other work at this level is immediate reaction to the run-time problems whereas our work mainly focuses on monitoring and improving the business process in the long term. However, this work and all the related work at this level demonstrate that the implementation of our suggested approach is feasible at the process execution level.

Finally, Verheecke *et al.* [144] propose an aspect-oriented method for the dynamic selection and swapping of web services using several criteria including their QoS, cost, and availability. The services selection can be done based on new incoming application requirements or based on the monitoring results and performance of the current web services as measured by dynamically added measure points. Similar to the other research in this category, this work is also focused on an atomic level of service composition. In this case, the focus is even narrower and is on the selection and swapping of specific services as required. An interesting part of this research, though, is the dynamic monitoring aspects, which allows monitoring of services based on the defined selection policy. For instance, if the selection policy requires the fastest service, the performance of the service will be measured using the measurement points added to steps in the service on the fly. Although, much of the research in this area uses aspects for service swapping and making modification to composite services, this work as well as Charfi's [24] are among the few that discuss the use of monitoring aspects to monitor and detect undesirable conditions.

3.3.2 Other Methods

The next group consists of papers returned in our systematic search results that attempt to address the process adaptation issue but without using aspect-oriented technology. We learned the lessons summarized in Table 6 by reviewing these papers. Although these papers do not directly help us answer the research questions, reviewing them allowed us to look at this research area from different angles and to realize some of the downsides of aspect-oriented approaches.

In [50], Graml *et al.* propose a method to extract *business rules* from process definitions in order to make processes more agile and adaptive. They believe in the separation of rules from the actual process definitions. Placing rules in a rule engine enables the adaptation of business processes at run-time just by changing the rules. Their suggested approach relies on a web-service based integration of a process execution engine and rule engine. Although they use aspect-oriented concepts to integrate constraints with business processes, they do not use any AOP technology for the implementation.

Ramakrishnan [121] suggests an approach to provide self-adaptive, process-based web service composition. The main goal of the paper is proper handling of *fault situations*, especially failures of partner services. The authors achieve this goal by instrumenting existing BPEL processes so that when the satisfactory service is not provided, an alternative service provider is used. The authors review some of the aspect-oriented approaches in their related work and argue that aspect-oriented based approaches require extensions to standard BPEL engines.

Table 6: Other methods summary

Papers	**Highlights**
All papers in this part	System adaptation is an important problem and many researchers are trying to address it using a variety of approaches. While policy-based approaches are gaining the attention of several researchers, much research has already been done on this topic from various angles and using different approaches.
[71][127][41]	Process adaptation should be considered at all levels, not just at the process design level, and should consider various information related to user context (e.g., environment and devices)
[126]	The migration of running process instances to a newly improved process model can be challenging and needs special attention.
[42][73][83]	Some authors believe aspect-oriented based methods are complicated and could have a longer learning curve for the practitioners. In addition, the higher level of abstraction of policy-based approaches may be more suitable for capturing management information.
[121]	While aspect-oriented based approaches always require infrastructure supporting aspect-oriented technology, other approaches can achieve some level of adaptation (e.g., fault handling) without extending the process execution engines.
[106][121][127][32]	Many of the suggested approaches only use QoS and focus on a very narrow view of adaptation (e.g., risk mitigation, fault handling, service replanning or replacement, and requirements) without much consideration for business value.

In addition, Na *et al.* [106] suggest a method for *adaptive replanning* (changing the service bindings) of service-based systems. Their main goal is to improve the replanning process from three points of view: 1) trigger, 2) service selection, and 3) cost and effect. According to the authors, most of the approaches that have been suggested for replanning of service-based systems use an exclusive strategy for replanning and do not consider the effectiveness of the replanning based on the current state of the system. Therefore, in the proposed system, they focus on a closer relationship between replanning and the system situation using a quantitative evaluation method to estimate the impact of the change on the system and a solution space management model to identify the search scope in which the best solutions can be found. Most of the related work as well as the approach suggested in this paper rely on QoS to trigger the adaptation process while, in a business environment, using only QoS is not the perfect measure for changing the composition. We believe the business context, including the outcome of the process and KPIs defined to measure the overall performance of the business as well as the satisfaction level of business goals, needs to be considered in order to achieve a good results that contributes to the bottom line of the business.

Ruy *et al.* [126] present a framework that allows businesses to evolve executing *process instances* to newly defined protocols if possible. In this paper, business protocol definitions are equivalent to business processes. The framework also provides tool support for detecting the process instances that can be migrated to the new protocols. The main goal of the paper is to provide the supporting environment for ever-changing services in a composite services environment. The authors use finite state machines to model the business processes and illustrate the Australian citizenship application process as an example in the paper. This framework does not use aspect-oriented approaches to perform the modification on the process models, but instead use protocol change operators including AddTransition, RemoveTransition, AddState, and RemoveState to perform modifications on the service models.

From a different angle, Erradi *et al.* [42] take a *policy-based* approach on business process adaptation and propose a middleware architecture called MASC to implement that approach. The main goal of the authors is the separation of concerns between process definition and the monitoring and control of the process. Furthermore, a new language, WS-Policy4MASC, is used to define the policies and monitoring rules. The authors suggest that adaptation can be studied from three different dimensions: 1) adaptation target (i.e., at a class or instance level), 2) adaptation approach (i.e., dynamically / at run-time or statically / at design time), and 3) adaptation goal (i.e., customization, correction, optimization, and prevention). This paper mainly focuses on dynamic customization and correction processes, which are mainly composed of adding, removing, and replacing tasks and fixing the faults reported during execution. Furthermore, the authors argue that the policy-based approach to adaption is easier to understand compared to alternatives like aspect-oriented programming.

Likewise, Lu takes a *policy-based* approach in [86]. He proposes an autonomic business-driven method to maximize the business value while considering several policies and constraints. This method also uses WS-Policy4MASC to define the business metrics and policies that need to be taken into account during the adaptation process. Lu compares this work with aspect-oriented approaches and claims that the proposed approach, unlike existing aspect-oriented ones, not only considers QoS but also business value and goals into account. In addition, it provides better abstraction for specifying management information compared to aspect-oriented approaches.

Similarly, Xiao *et al.* discuss a *policy-based* approach for process adaptation in [151]. Their method mainly relies on predefined process fragments that can be used to compose a process defined at a higher level of abstraction. The high-level process is defined using a process template and process fragments are selected using defined policies and constraints. Although this approach seems promising as it offers a flexible predefined process infrastructure, it is not ready yet to be used for adapting live processes on the fly.

In [83], Lian *et al.* propose an *agent-based, context-aware* framework that helps with the adaptation of business processes. The framework mainly focuses on handling fault and error situations and takes a proactive approach in addressing exceptions. The designed architecture sets an agent layer on top of the process execution layer to monitor and adapt the processes, as well as react to unexpected events. The authors do not use any aspect-oriented technique, but reference many aspect-oriented research contributions as related work. They argue that while aspect-oriented methods can be used to insert rules into processes, these methods could also increase complexity in terms of managing conflicts between different rules.

Eleonu and Oruh pay special attention to mobile systems in [41], and work towards an architecture that allows for *context-aware* process adaptation. The main differentiator of their architecture is separation of context information from application and business logic with the goal of making the information usable in various applications. The context in this case is the environment surrounding the user and the device being used by the user, e.g., time of the day, temperature, battery, activity, and location. The goal is to adapt the applications and process using such context information to create business opportunities and improve the user experience. The authors propose to investigate use of aspect-oriented programming in the development process.

Kalavathy *et al.* [73] propose a *policy-based* architecture for self-adaptation of service-oriented media services. The architecture is not generic but is meant to be used for media services only. The authors suggest that in any service adaptation framework, separation of the adaptation policies from the base process is important, and their architecture supports this approach. Similar to [83] and [42], the authors also believe their approach is better than the aspect-oriented programming alternative because it deals with the problem at a higher level of abstraction and is easier and more understandable by the users of the system.

In [71], Kazhamiakin *et al.* break down service-based applications into several layers and argue that monitoring and adaptation in each layer independently is not the right approach. For instance, if a business process is improved but the underlying services used to execute the process still uses low-quality services, the improvement will not be maximized. Furthermore, they illustrate the conceptual model of an architecture that helps address this problem.

In [127], Sawyer *et al.* use an *i*-based* modeling approach to handle dynamic changes of a system at the requirements level. In their method, a visual model of dynamically adaptive system requirements is created. The requirements for the system when it works in a stable environment are separated from the adaptive requirements. Although the suggested method uses a goal modeling language, it covers only the system requirements level and does not address the need for monitoring and improving on the process models.

In [32], Conforti *et al.* use a YAWL-based environment to detect risk situations in business processes at runtime and mitigate risks by recommending alternative actions to process administrators. Although the proposed approach evaluation shows it can be effective, it is a very prescriptive and targeted approach limited to risk situations.

3.3.3 Generic Studies

The last group consists of papers that have performed more generic studies on adaptive systems and processes. While analyzing these papers, we learned several lessons, summarized in Table 7.

Table 7: Generic studies summary

Papers	Highlights
[70]	Current process modeling notations are not good enough for defining flexible business processes.
[26][30]	Adaptation could be performed at different levels, have different goals, and use different approaches. In addition, most of the existing approaches only target this space partially and it is hard to come up with a single framework that covers everything.
[72]	Current adaptive systems have several issues including having a focus on process instances as opposed to classes, being reactive as opposed to proactive, the use of rigid and inflexible specifications, requiring human intervention, and not considering the business context.

Kapuruge *et al.* performed a survey [70] on current approaches for providing *flexibility* in business modeling and service composition. The focus of the paper is mainly on defining flexibility requirements and assessing current work considering those requirements. The flexibility requirements specified by this paper are divided in three groups: 1) process definition flexibility including: configurable design, built-in context awareness, ease of understanding, late specification, automated change verification, and merging of business process definitions; 2) process instance flexibility including: process instance deviation, process instance handling as a case, process instance migration to a new process class design, and ease of human understanding and intervention; and 3) services relationship flexibility including: ability to change service interfaces, ability to change service bindings, and ability to change inter-service relationships. According to this paper, one of the problems of current graphical modeling languages is the use of strict sequences of atomic-level tasks to form a process. This approach, while helpful with understandability of the process for humans, reduces the flexibility of the process model dramatically.

In [26], Cheng *et al.* discuss a software engineering research roadmap for self-adaptive systems from four points of view: modeling dimensions, requirements, engineering, and assurance. The paper suggests that an adaptive system has to support methods for expressing goals and monitoring changes, mechanisms to perform change, and also evaluations of effects on the system.

In [30], Courbis and Finkelstein argue that adaptation can be 1) static or dynamic, 2) manual or automatic, and 3) proactive or retroactive. The paper suggests that aspect-oriented approaches must be used in all layers of process-based systems to achieve maximum flexibility and adaptability. The three layers suggested by this paper are semantic analyzers, BPEL engines, and BPEL processes.

Finally, Kazhamiakin *et al.* [72] discuss the adaptation of services from several points of view, including personalization based on user preferences, changes to address QoS requirement, and changes in functionality. A taxonomy is also presented that defines the adaptation arena by answering why, what, and how questions in this context. In addition, the authors suggest AOP in related work as a good approach for the adaptation of software systems due to the flexibility and dynamicity that AOP brings to the table. The paper also presents a summary of the adaptation approaches proposed by other researchers in this body of knowledge. This summary looks at adaptation from five different angles:

- Usage of adaptation: the authors suggest that adaptation is mostly used for recovery, optimization, and customization. They believe the use of contextual factors (e.g., business context) as well as proactive approaches can be enhanced in adaptation frameworks.
- Subject of adaptation: according to the authors, there are few researchers focusing on adaptation at the class level. The majority of research focuses on adaptation at the composition or instance level. Our systematic review also confirms this result. The authors regard more holistic approaches at the class level, as proposed in our framework, to provide better long-term impact on businesses.
- Adaptation strategy: according to the authors, the existing approaches have several issues. As these approaches are usually not proactive, future issues cannot be easily prevented. Furthermore, most of the frameworks do not have a distribution and co-ordination module for the adaptation activities.

- Adaptation specification: most of the existing adaptation specifications lack flexibility and are defined at design-time for specific situations. Therefore, it is hard to use them in more generic situations.
- Decision and autonomy: researchers address the adaptation decision problem differently, at design-time in a predefined way or dynamically. Furthermore, in some cases, human intervention is required for the final decisions on the modifications.

3.4. Threats to Validity

There are several main threats to the validity of our systematic literature review. One may argue that the scope of our research in both phases one and two was not wide enough. In addition, one may suggest that we could have biased the selection of publications.

In the first phase, because of a very high number of search results, we have limited our publication selection scope significantly (i.e., we only searched on article titles and we only used one meta-search engine). Therefore, there is a risk that our answer to the first research question is not correct, which means there could be existing literature reviews focusing on the application of aspect-oriented methods to business process adaptation. This was mitigated to some extent by the use of the Google Scholar engine, which is fairly global and up to date.

In the second phase of our study, we could have used several more queries, for example ("*aspect oriented*" AND "*dynamic business processes*"), to increase the scope of our research. However, to make the research feasible, we chose a targeted set of queries. We believe we have mitigated this risk in two ways: first, by manually looking at commonly cited articles by the papers we found, and second, by comparing our conclusions with some more generic surveys and studies (i.e., [26], [30], [70], and [72]).

Finally, considering that this review is also used as a validation of our own research agenda from feasibility and novelty points of view, one may argue that we could have biased the selection of articles used in the second phase of the research. We have addressed this threat in three ways. First, we have systematically selected the papers and documented all steps, allowing them to be replicated by any independent party. Second, some of the conclusions we reached are confirmed by other higher-level studies on the topic of adaptive service-based systems (i.e., [26], [30], [70], and [72]). Third, we took a very neutral position by applying the lessons learned from the study to our own research and criticizing the downsides of our own proposed framework in the summary section and throughout the chapter.

3.5. Chapter Summary

In this chapter, we conducted a systematic review on the application of aspect-oriented techniques to business process adaptation. The review was done in two phases. In the first phase, we answered the first question of this research by looking at 72 existing surveys/reviews on business processes to assure ourselves no one had already performed a systematic review on our topic of interest. In the second phase we focused on finding papers that apply aspect-oriented methods to process adaptation to answer three remaining questions: 2) have aspect-oriented methods been used for business processes adaptation? If yes, 2.a) what are the main applications of aspect-oriented methods in the area of adaptive business processes? 3) Is our proposal for an AoURN-based framework novel and realizable given prior research and existing infrastructures?

After analyzing 60 (54 in the first version and six additional papers in 2013) papers and performing a deep review of a subset extracted using the criteria listed in Table 4, we can answer the aforementioned research questions. For several years, mainly after 2005, significant attention was devoted to the application of aspect-oriented techniques to business process and service-based system adaptation. Much of the research is focused on adaptation at the process execution level and in the following areas: i) composing and swapping services based on QoS, cost, and rules, and in the event of failure, ii) extracting roles and crosscutting concerns from composite services, iii) customizing process instances based on user profiles or Service Level Agreements, iv) adapting service composition and collaboration policies, and v) using monitoring aspects to detect undesired situations.

Although the results may convey that this area of research is mature enough, we believe there is still much work that can be done. However, the underlying research in this area makes us confident that our proposed framework is realizable in practice.

As also indicated in [72], much current work focuses on instance-level adaptation as opposed to considering the overall performance of the process and improving the process model. Most approaches do not take the context of the business into consideration.

As illustrated in Table 8, this is where our framework exceeds other research by considering processes, business goals, performance models (KPIs), constraints, and even redesign patterns (for process improvement), to provide a more comprehensive framework.

Furthermore, current techniques are more reactive as opposed to proactive. They mainly use pre-defined rules, policies, or hard-coded situations that are hard to use in a more generic way. Moreover, most of the existing frameworks require some level of human intervention for the final decision-making. The other aspect that could be improved in the existing research is the lack of focus on all layers of service-based systems. While adaptation could be done at many different levels of a system, many proposed approaches only focus on one specific layer. This is true for our proposed framework as well, which only focuses on process adaptation at the business process model level. Although this is still beneficial, a process that is improved to perfection but still uses poor services during the execution will not perform as expected. Therefore, to have a better picture and feedback loop, even when we are focused on improving business process models, we have to monitor the service composition and process execution layers as well.

We also learned that there are concerns in the research community regarding the complexity of aspect-oriented approaches. Some authors [42][73][83] claim that the learning curve in aspect-oriented approaches from the point of view of the end user could be (too) high. In addition, managing the interaction between aspects when multiple aspects are applied to a process could increase the complexity of the system. Lu [86] argues that policy-based approaches are easier since they deal with the adaptation / rule definition at a higher level of abstraction that is closer to the language users often understand. We also had a similar experience with our aspect-based redesign patterns. They can become very complex and hard to explain to an audience without prior experience in Aspect-oriented URN.

Table 8: Reviewed papers compared with our framework [117]

		Ref.	Author - Year	Adaptive systems	Aspects used?	Process improve prove-ment?	Redesign patterns?	Business goals & KPIs?
A	CS	[5]	Algahtani and Zedan - 2009	Y	Y	Y	N	N
		[56]	Hermosillo *et al.* - 2010	Y	Y	Y	N	N
		[53]	Hermosillo *et al.* - 2010	Y	Y	Y	N	N
		[54]	Hermosillo - 2012	Y	Y	Y	N	N
		[107]	Narendra *et al.* - 2007	Y	Y	N	N	N
		[43]	Erradi *et al.* - 2005	Y	Y	N	N	N
	E	[120]	Rahman *et al.* - 2008	Y	Y	N	N	N
		[107]	Narendra *et al.* - 2007	Y	Y	N	N	N
		[128]	Sánchez and Villalobos - 2008	Y	Y	N	N	N
		[136]	Stearns - 2002	Y	Y	N	N	N
	CP	[148]	Wan, *et al.* - 2008	Y	Y	N	N	N
	MA	[24]	Charfi *et al.* - 2009	Y	Y	Y	N	N
		[144]	B. Verheecke *et al.* 2003	Y	Y	N	N	N
		[109]	Patiniotakis *et al.*- 2013	Y	Y	Y	N	N
O		[50]	Graml *et al.* - 2007	Y	N	N	N	N
		[127]	Sawyer *et al.* - 2007	Y	N	N	N	N
		[126]	Ruy *et al.* - 2007	Y	N	N	N	N
		[71]	Kazhamiakin *et al.* - 2009	Y	N	Y	N	N
		[42]	Erradi *et al.* - 2006	Y	N	N	N	N
		[73]	Kalavathy *et al.* 2010	Y	N	N	N	N
		[83]	Lian *et al.* - 2010	Y	N	N	N	N
		[106]	Na *et al.* - 2010	Y	N	N	N	N
		[121]	Ramakrishnan - 2009	Y	N	N	N	N
		[86]	Lu - 2011	Y	N	Y	N	N
		[151]	Xiao *et al.* – 2011	Y	N	N	N	N
		[32]	Conforti *et al.* – 2012	Y	N	N	N	N
		[41]	Eleonu and Oruh - 2013	Y	N	N	N	N
G		[70]	Kapuruge *et al.* - 2010	Y	N	N	N	N
		[26]	Cheng *et al.* - 2009	Y	N	N	N	N
		[30]	Courbis and Finkelstein - 2005	Y	Y	N	N	N
		[72]	Kazhamiakin *et al.* - 2010	Y	N	N	N	N
P		[117]	Pourshahid *et al.* - 2011	Y	Y	Y	Y	Y

A: Aspect-oriented-based methods, **G:** Generic studies, **O:** Other methods, **P:** Proposed method
CS: Composing and swapping services based on QoS, cost, rules, and failures
E: Extracting roles and crosscutting concerns from composite services
CP: Customizing process instances based on user profiles or SLAs
MA: Using monitoring aspects to detect undesired situations

The next chapter will provide the details of our framework for modeling, monitoring and adapting business processes with the help of the Aspect-oriented User Requirements Notation.

Chapter 4. A Framework for Modeling, Monitoring and Adapting Business Processes Using AoURN

This chapter introduces the new framework for modeling, monitoring, and adapting business processes, which is at the core of this thesis. After a brief overview in section 4.1 and the introduction of the elements of the framework in section 4.2, the steps of the two framework's lifecycles, analysis and adaptation, are presented in sections 4.3 and 4.4 respectively. Subsequently, we discuss in section 4.5 the roles and responsibilities required in an organization to use the framework, and then summarize the chapter in section 4.6.

4.1. Framework Overview

The goal of the framework is to allow an organization to evaluate business goals and processes, to discover, model, monitor, and detect undesirable conditions, and to adapt (improve) accordingly. The framework, whose high-level view is shown in Figure 13, consists of two main intertwined *lifecycles*.

The first lifecycle (*analysis*) targets modeling relevant artefacts, finding problem areas, and making decisions to improve the business while monitoring the impact of these decisions. The inputs to this lifecycle are business objectives and process definitions in their existing forms (*as is*). In some businesses, when the framework starts being used, process definitions may not exist yet, whereas in other businesses with a higher level of maturity, such definitions may be available as (graphical) models. The outputs from the analysis lifecycle are the models needed for the second lifecycle (with performance and decision information), which allow businesses to start monitoring processes and goals in the next steps, together with potential problem areas.

The second lifecycle (*adaptation*) supports the improvement and adaptation of the business processes that are pinpointed as problem areas by the analysis lifecycle. The inputs to the adaptation lifecycle are the models from the analysis lifecycle together with redesign patterns, while the outputs are *adapted* versions of the models. The framework is iterative and incremental and meant to help organizations at any level of maturity in terms of models and data definition availability.

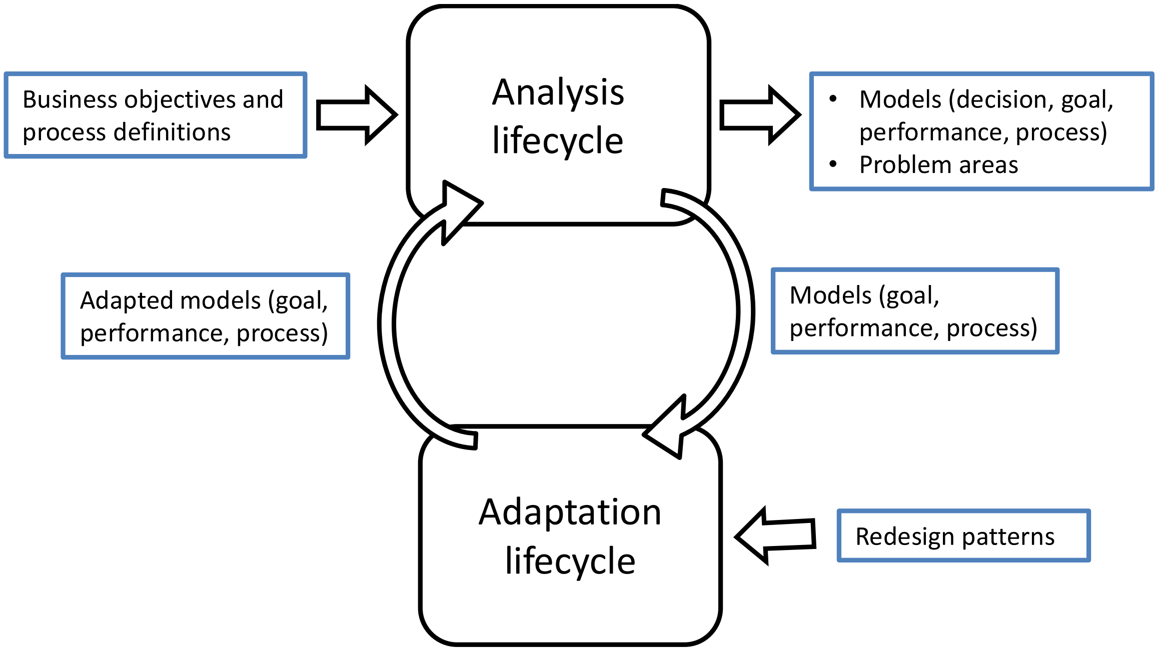
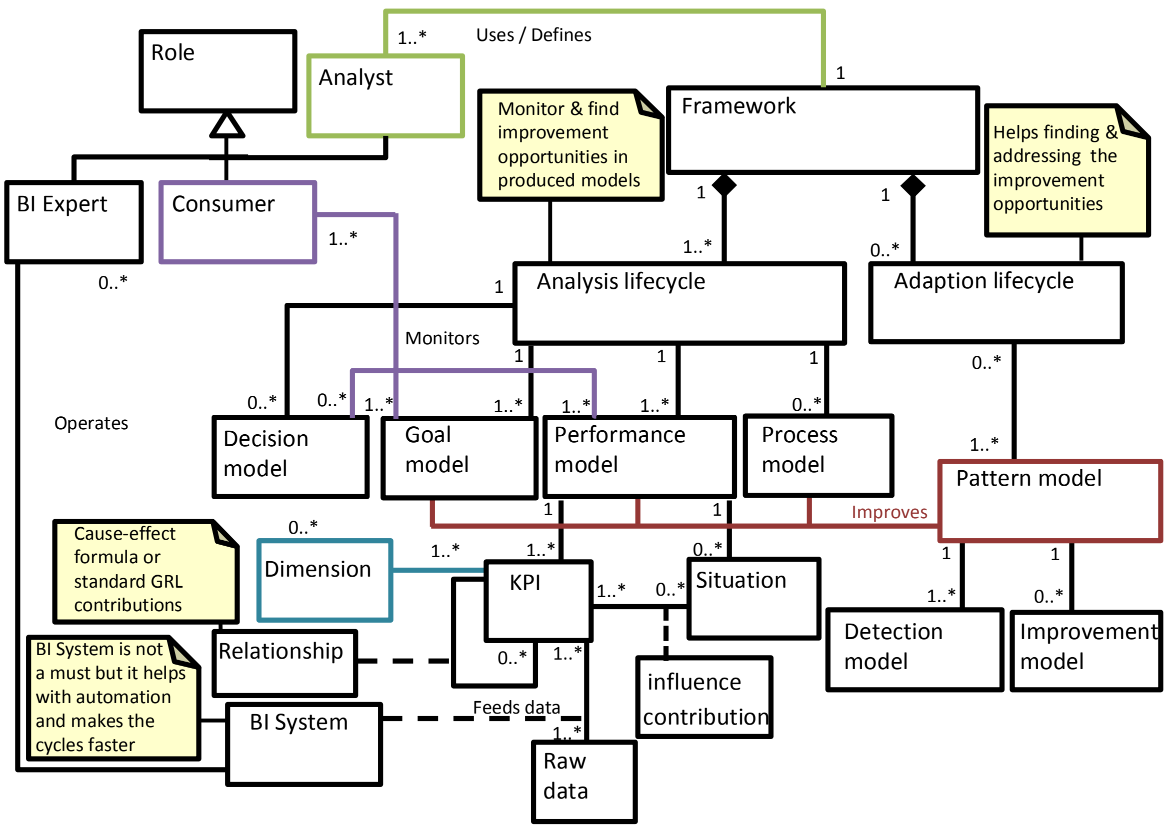


Figure 13: Framework high-level view, with lifecycles' inputs and outputs

4.2. Framework Meta-Model

In Figure 14, we formalize the framework meta-model with its high-level components and the roles involved in making the framework operational. The framework's main two subsets cover the analysis lifecycle and adaptation lifecycle. The analysis lifecycle produces four types of models that help with providing a comprehensive view of the organization goals, KPIs, processes, and business decisions. The adaptation lifecycle uses pattern models to detect undesired conditions based on predefined best practices and to provide improvement options when included in the patterns. Although BI systems are not a mandatory part of the framework, they can be used to feed data into the framework and help with real-time monitoring of KPIs.

Three types of roles are involved in the implementation and usage of the framework: *analysts*, *BI experts,* and *consumer*s of the framework (i.e., those who monitor reports and make decisions). These roles, to be further discussed in section 4.5, interact with the system at different levels. In a nutshell, analysts interact the most with the system, its artefacts, and the other roles in the framework. They implement the framework and make it available to other roles. BI experts help the analysts by providing the underlying data sources and BI systems to feed the required values to the framework models automatically. Finally, consumers mainly use the models for decision-making and monitor the satisfaction level of business goals and KPIs. Consumers are usually the either tactical who monitor the performance models or strategy managers who monitor the high-level business goals.



*Note: The colors in the above figure have been used to distinguish the overlapping lines and have no other significance.

Figure 14: Framework meta-model

The following is a description of the artefacts produced by the analysis lifecycle:

- *Decision model*: shows the various business decision options to the analysts and consumers and allows them to visually see the available options. This model is captured using GRL diagrams (Figure 15). This enables analysts and consumers to use GRL evaluation algorithms to perform impact and trade-off analysis.

- *Goal model*: captures the high-level goals of the organization and allows both analysts and consumers to monitor the satisfaction level of organization goals. This model is captured using GRL diagrams.
- *Performance model*: shows business KPIs, together with their relationships to raw data inputs, organization goals, and other KPIs. Furthermore, this model also captures the KPI dimensions. The performance model is created by analysts and used by consumers and analysts to monitor KPI values of interest. In addition, BI experts can use this model to create the required BI data sources for automated monitoring. This model is captured using GRL diagrams plus *raw data* elements, which are a new type of model element proposed in this thesis.
- *Process model*: captures the business processes of the organization using UCM. This model is mainly used by the analysts in the adaptation phase to make changes to the processes.

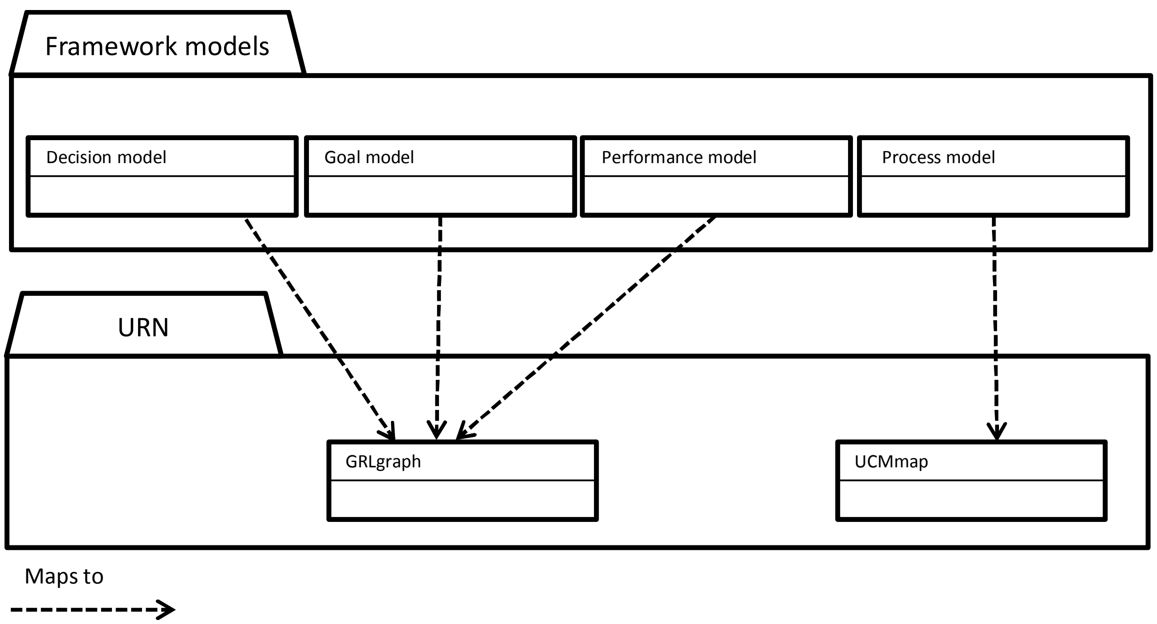


Figure 15: Mapping of the framework models to URN

The framework's patterns consist of *detection models* and possibly of *improvement models* (Figure 14). As shown in Figure 16, the patterns' models are further divided into intentional and behavioral models, which respectively capture the goals/KPIs and processes of the organizations. A pattern must at least have one intentional detection model to detect an undesirable condition and it can be further completed by having intentional improvement, behavioral detection and behavioral improvement models. Figure 16 also illustrated the mapping of patterns to AoURN models. This topic will be further discussed in section 5.2.

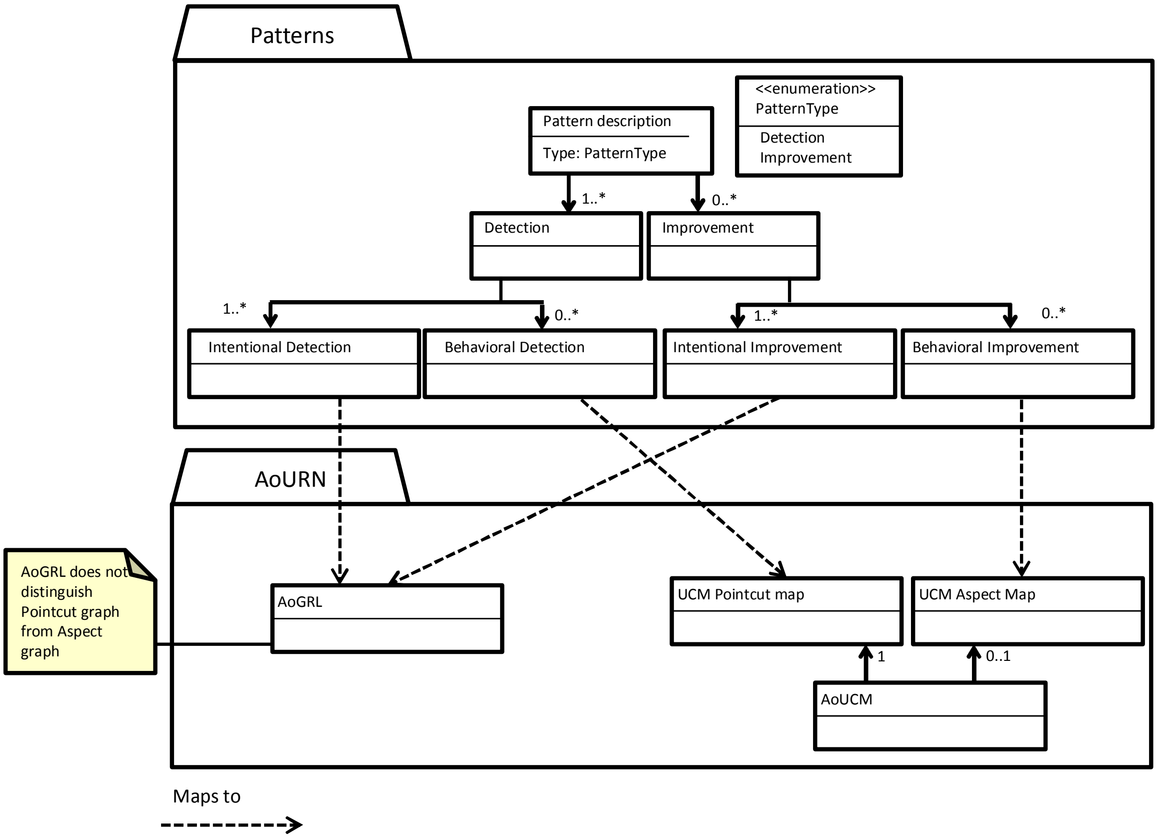


Figure 16: Mapping of the framework patterns to AoURN

Note that all these models are actually views of one single AoURN model. This enables elements to be referenced in multiple models (e.g., a goal common to the goal view and the performance view). URN links can also be used to relate any pair of elements from different models/views, and type this relationship (e.g., a “refinement” URN link from a GRL task to a UCM map/process). This information enables the goal, performance, decision and process models to be considered all at once using existing (and extended) analysis algorithms and tools.

4.3. Analysis Lifecycle

Based on the reasoning behind the notion of cognitive fit discussed in section 2.3, the framework defines the organizational goals and links these goals explicitly to performance models including the relevant Key Performance Indicators. The framework can be used by organizations at any level of maturity and readiness in terms of gathering and monitoring data for BI-based monitoring of the business. In particular, unlike many simulation approaches, it does not necessitate up front large quantities of data to be useful. We believe different organizations however have different needs and may be in different states when they decide to incorporate such a framework. Consequently, we are suggesting an iterative method consisting of three basic steps involving iterations that build upon each other (Table 9).

Table 9: Analysis lifecycle

Steps	Activity	Input	Output	Involved roles
Step 1	Create goal models	Interview data	Goal models	Analysts: Interviewers & Modelers Consumers: Interviewees
	Define the KPIs	Interview data & Existing KPI monitoring systems	Performance model	Analysts: Interviewers & Modelers Consumers: Interviewees
	Identify analysis types & Specify new KPIs	Goal and Performance models	Analysis types	Consumers: Identify the required analysis
Step 2	Add new KPIs	Performance model	Refined performance model	Analysts
	Refine cause-effect relationships	Performance model	Refined performance model	Analysts
	Connect KPIs to BI data sources	Performance model	Performance model connected to BI data sources	BI Experts: BI related activities Analysts: Provide the required information for BI experts
	Create decision options diagrams	Interview data, business plans and other related documents	Decision model	Analysts: Interviewers & Modelers Consumer: Interviewees
	Make a decision	Decision model, Goal model and Performance model	Changes in the business	Consumers: Make decisions Analysts: Provide input to consumers as required
	Model required processes	Interview data & current process models	Process model	Analysts: Interviewers & Modelers Consumers: Interviewees
	Adaptation lifecycle (Optional)	Models (Goals, Performance, Processes)	Adapted models	Analysts
Step 3	Model situations	Performance model	Performance & Goal models + Situations	Analysts
	Add required monitoring KPIs	Performance model	Performance model + New KPIs	Analysts
	Monitor and refine the model	Goal and Performance model	Refined models and a new iteration	Analysts: Monitor and refine Consumers: Monitor

In the **first step**, an initial model of the organization's goals is created [45]. This model can be built by analysts based on interviews with executives and operational managers (i.e., consumers), as we experimented with in our examples. This goal model can consist of long term, short term, strategic and operational goals of the organization as well as contribution and decomposition relationships between them. Furthermore, in this step, analysts define *the KPIs that support the goals* (e.g., financial KPIs) and add them to the model. This can be a challenging task and be very dependent on the level of maturity of the organization. For instance, in the two organizations we have studied as part of this research, one small organization had a very limited set of data and was using a spreadsheet to monitor the business while the other (and larger) one had many indicators available and was using a sophisticated Business Intelligence system. Our discussions with both organizations however demonstrate that organizations at any point within this wide range of information management capabilities can benefit from applying such goal-based model. After defining the *performance and goal models*, consumers will have a better picture of their business, will *identify the type of analysis* they want to perform on the model and *specify the new KPIs required* to do so if any is missing.

In the **second step**, analysts explore the existing models, discover and *add the new KPIs* and the new dimensions to the model required to perform a better job in monitoring various aspects of the business. Note that not all the KPIs need to be dimensional and if the available data is not as granular as is required for a dimensional model, or if all the data is not available, a step-by-step approach can be used leading to a number of model iterations as additional data becomes available.

In addition, during this process analysts *refine the relationships* between KPIs in the goal model (hence improving cognitive fit). These relationships create a graphical model, which can be used to analyze what-if scenarios. The relationships are defined using mathematical formulas that are discussed comprehensively in Chapter 6. For example, a relationship between the "New customer count" KPI and the "Online marketing campaign budget" and "Flyer budget" KPIs could look like the following:

New customer count = Online marketing campaign budget in $ / (0.5 $/customer)
+ Flyer budget in $ / (0.8 $/customer)

Such formula capturing relationships between KPIs can be refined based on historical data, for example through data mining. In cases where organizations do not have historical data and are in their early iterations of creating their performance model, the initial formula used to define the KPI relationships can be based on industry standards (e.g., samples in Table 15); analysts have to start somewhere, even if the relationships are not right the first time. As organization gathers more information, this historical data can be integrated to the model. As will be seen later in the retail example (Chapter 8), the models can be used to illustrate the expected impact of actions taken by involved consumers and analysts.

Once a performance model is ready, as an optional step, BI experts can get involved and use that model as a reference to understand the monitoring needs of the organization and create (or reuse) the underlying BI infrastructure, including data warehouses and BI reports, to feed the performance model. This is a collaborative effort between analysts and BI experts and may result in refining the performance model by adding raw data elements to read the data directly from BI systems.

Analysts can also add a *decision model* and connect the model's options to the goals and KPIs of the organization. A decision model outlines the different options available to organizations to achieve their goals. This diagram helps consumers to visualize the options and define GRL strategies that reflect the result of selecting one of the options. Note that this is not a new type of graphical diagram and it is only a terminology choice in our framework and a suggestion to isolate the decision options from goal and performance models to help improve the analysis.

Moreover, organizations can continually adapt the models by saving the initial iteration as a "snapshot" and comparing it to actual results achieved by decisions. Gathering these snapshots will eventually create a "decision trail" that displays context, decisions taken and results of these decisions allowing managers to make better decisions in the future. In addition, decision trails allow organizations to refer back to the rationale they used for making successful or unsuccessful decisions. At this time, the snapshots are not currently supported by tools at the model level, and analysts have to save various model versions and rely on BI tools and databases for snapshot capabilities. This functionality needs to be further developed in the supporting tool in the future but is out of the scope of this thesis.

Furthermore, the notation used for the framework and the modeling tool allow the organization to model *business processes* (with UCM) in an integrated environment with the KPI models. This often can be used when the analysts believe some of the processes in the organization need closer attention to enhance the KPIs of interest. Analysts can model the as-is and proposed to-be processes and use the KPI model to monitor the expected results. When process adaptation is considered to be an option for improving certain aspect of a business, analysts can use the process *adaptation lifecycle*, which is an integrated part of this thesis' framework.

In the **third step**, analysts add the expected impact, if any, of the decision made in the second step to the model that illustrates either *threats* or *opportunities* involved. Such expected impact is modeled with *situations*, a new concept in our framework that is elaborated on in section 6.3.1. This framework's situations are akin to situations in the Business Intelligence Model (BIM) [58][65], which is focused on SWOT (Strengths, Weaknesses, Opportunities, and Threats) analysis. Situations, in this thesis, capture the opportunities or threats that affect the way a business operates. For instance, investment in an online store has the risk of reducing the profile of the business for a period, which is a situation. Using *stereotyped GRL softgoals* to capture situations, we are able to the show the qualitative impact of situations on the rest of the model. The most challenging aspect of this step is modeling a qualitative situation factor that influences a quantitative KPI. In this case, we model the impact by increasing the range of acceptable values for a KPI. In other words, once we have estimated the impact of the situation, we allow the acceptable range of the measured KPI to deviate accordingly from its target value. However, this approach is only an example and analysts can define any mathematical formula for how situations influence the rest of the model. The formula-based evaluation algorithm that is used to analyze the impact of the situations on the model is discussed extensively in Chapter 6.

In this step, analysts also add the *required KPIs* and dimensions to the model that allows analysts and consumers to better observe the impact of decisions. If analysts expect a decision to change anything in the organization, they will examine that hypothesis using the appropriate KPIs and GRL strategies. When the *adaptation lifecycle* is used, the new KPIs can be based on the intentional improvement model of the redesign patters that are used. In this case, the changes to the model can influence more than just the KPIs including goals and processes.

Finally, analysts *monitor* the impact of the changes made to the business or processes and compare expected results against actual results. Based on this comparison, analysts *refine* the models as required and record the data.

Monitoring of the business could be done in two ways. In the first approach, analysts and consumers observe the results of business goals and KPIs one by one and look for signs of unsatisfied business goals and performance indicators in the models. Although this approach is effective in simple scenarios, it needs a lot of attention and constant monitoring. When complicated, undesirable conditions involving more than just one goal and KPI need to be detected, additional tool support can be helpful. The approach discussed in section 4.4.1 helps with complicated situations by allowing analysts to model and detect undesirable conditions.

In summary, the analysis lifecycle is based on creating an initial model, which is then refined by expanding data sources, capturing decisions made and the results of those decisions, and building historical decision trails that inform future models.

4.4. Adaptation Lifecycle

The intentions of the adaptation lifecycle, whose steps are defined in Table 10, are to allow analysts to, in the **first step,** detect the undesired situations, in the **second step**, find the most appropriate redesign patterns and finally, in the **third step**, apply the patterns and improve processes that do not satisfy their goals.

The analysts use this lifecycle on its own or after modeling the artefacts required for the *analysis lifecycle* (e.g., the process, goal, and performance models). If the business intention is to perform process improvement, business process models are required, otherwise, if the intention is to only monitor the business goals and KPIs, analysts could only model goal and performance models of the organizations. Furthermore, the partial pattern matching discussed in section 5.4 allows organizations to add the required missing elements to their models, which further allows them to monitor their business for undesired situations more effectively in future monitoring iterations.

Table 10: Adaptation lifecycle

Step	Activity	Input	Output	Involved roles
Step 1	Monitor & detect undesirable conditions	Models (Goals, Performance, Processes)	Flagged undesirable conditions List of matched patterns	Analysts
Step 2	Find the applicable redesign patterns	List of matched patterns	Selected list of patterns that: - Are not false positive - Improve the intended areas	Analysts
Step 3	Apply the best redesign patterns	Selected patterns & base models	Refined models	Analysts
	Analysis lifecycle' monitoring	Refined model	New observations	Analysts & Consumers

4.4.1 Detecting Undesirable Conditions

Undesirable conditions are conditions (or obstacles) in the organization that work against the goals of the business. These conditions can be detected in the goal and performance models of the organization by monitoring the satisfaction values of goals and KPIs. However, conditions can be more complex than just monitoring one KPI. For instance, a condition can involve a combination of KPIs and process structure.

As part of the performance modeling step, each KPI is normalized. The worst possible value, the threshold value, and the target value are defined for real world values of each GRL KPI and then mapped to -100, 0, and 100 on the GRL scale, respectively. The normalization step therefore maps a real-life evaluation value of a KPI into its GRL satisfaction value. Now, the business processes may be monitored. Any KPI value that is not satisfactory (i.e., that is far off the target value) indicates a possible area of improvement.

Monitoring each KPI on its own is easy but monitoring several interconnected KPIs for a predefined undesired condition could be complex and error-prone if done by a human. Our pattern matching approach helps analysts detect these complex conditions by defining the undesired patterns.

An undesired condition is detected when a match of its pattern can be found in the models of the organization. The pattern could look for a combination of unsatisfied KPIs and goals. In addition, it could also take the process structure into consideration. Note that there might be undesirable conditions that cannot be detected with this approach. For instance, if a process cannot be modeled using the modeling language of choice, obviously the undesired conditions in this process cannot be detected.

Figure 17 presents an example of detection of undesired condition where three goals, one process, and two KPIs are being monitored. Even in this simple model, there are different combinations of conditions that could have a negative impact on the business goals. For example, let us assume that Approval Process has a low satisfaction value. This could either be due to a low satisfaction value in the *Average (Ave.) Turnaround time* KPI or in the *Average* (*Ave.) Cost per Application* KPI. If an analyst wants to be alerted only when the cost of the approval process goes up, then we need to define a pattern model that detects the condition when the satisfaction values of the cost KPI, the Approval process, and the Reduce Cost model elements are low. This combination is completely different form the condition where a decline in turnaround time might be of interest to the analysts.

The undesired conditions could be anywhere in the range going from very custom to very generic. For instance, they could describe a very specific condition like in aforementioned example and only detect one problem while not addressing the problem. They could also be part of a generic redesign pattern that comes with potential solutions to improve the processes connected to the KPIs and goals being monitored.

Businesses can add to their repository of undesired conditions over time and eventually turn some of the very custom conditions into patterns that can be applied to the other contexts of the business.

The next step for a business after detecting these undesired conditions is to make changes to the business to address the root cause of the problem. This can be done partly by human intervention and partly by using predefined redesign patterns as discussed in the next section.

4.4.2 Finding and Applying the Applicable Patterns

Process redesign patterns are modeled as aspects (as will be seen in section 5.1), which are later applied to the existing process. The users of the framework, in addition to standard generic redesign patterns, may model customized redesign patterns or alternative versions of the processes. To aid the selection of the appropriate redesign pattern, KPIs are categorized according to four redesign pattern groups. The filled circles with **T**, **Q**, **F**, and **C** indicate the *time*, *quality*, *flexibility* and *cost* groups for the KPIs, respectively. As shown in Figure 17, these icons are used to annotate the KPI model elements.

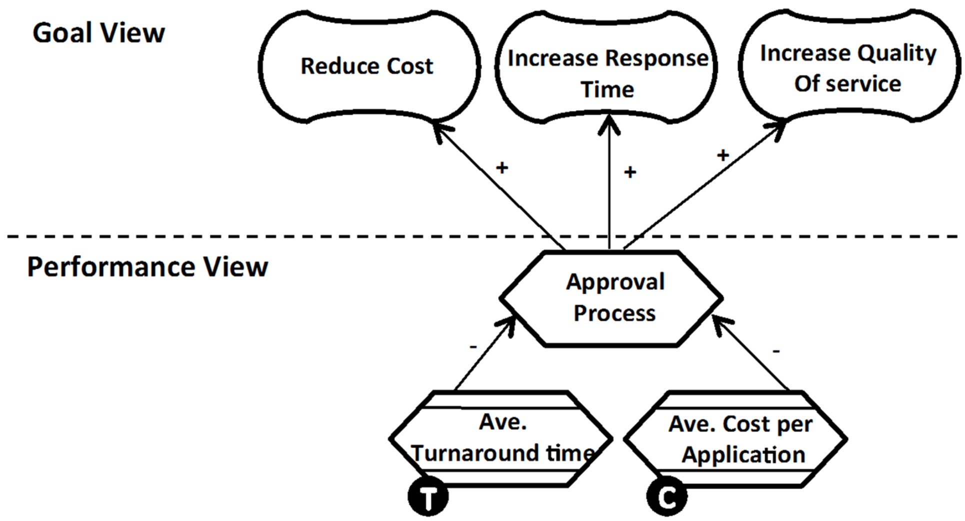


Figure 17: An example of KPIs annotated with KPI groups

The presence of KPI groups helps determine candidate redesign patterns for undesirable conditions. For instance, if the KPI with the unsatisfactory value is categorized as a time KPI, all redesign patterns with a positive impact on time are potential candidates for improving the observed values of the KPI and consequently the process. However, all of the possible candidate redesign patterns may not be applicable to the target process for which the KPI has been defined. Therefore, in the next step, we use additional characteristics of redesign patterns to reduce the number of candidates. These characteristics are identifying features of the redesign pattern that are expressed by the modeling language of choice, e.g., the redesign pattern may require a certain sequence of process model elements or may require that certain KPIs perform worse than other KPIs. A redesign pattern, therefore, is only applicable to the process model, if a match of its expected model elements can be found in the base model. After identifying the applicable patterns and if more than one possible choice exists, users can apply all the possible options one by one and decide which one is the most appropriate. As an applied pattern changes not only the process model but also the goal and performance models, it is possible to observe the impact of the applied pattern on all models.

The advantage of modeling redesign patterns with aspects is that the whole pattern, i.e., the characteristics of the pattern but also its impact on the process, goal, and performance models, can be composed in one properly encapsulated unit. This facilitates the reuse of the pattern in different applications and enables reasoning about the use of the pattern and its composition with other patterns.

Before this thesis, all AoURN pointcut expressions explored in the literature [99] were defined either just in the process view or just in the goal view of a URN model. However, for this thesis, both goal and process models need to be covered at the same time, a) because redesign patterns require both to be matched properly in order to describe the identifying characteristics of the pattern and b) to ensure that all models remain aligned with each other. This provides us with the ability to apply the required changes to goal and performance models after applying redesign patterns to the process models. Such changes in the goal or performance models are required when the redesign pattern eliminates, adds, or updates tasks in the process model. Therefore, the same tasks and the corresponding KPIs should be added, eliminated, or updated in the goal and performance models, respectively. The modified model of a redesign pattern may even introduce up-to-now unidentified goals and KPIs to the goal and performance models accordingly, contributing further to a more comprehensive set of models.

4.5. Organizational Roles and Responsibilities

Although the iterative and incremental nature of the framework allows organizations at any size/maturity to start using it, depending on the size of the organization and level of investment, not all aspects of the framework may be readily achievable. The complexity of some aspects of the framework requires specialized skills that not all users in an organization have, and the intention of the framework is also not to expose everyone to the same level of detail and complexity.

Figure 18 illustrates the required roles and responsibilities in an organization to use the framework. In a large organization, most of the users, primarily managers and business users, will only deal with the least complex aspect of the methodology (for consumers) and mainly use a subset of goals and KPIs to monitor their targets. A smaller number of analysts and BI experts will be involved in the more complicated aspects of the methodology. Initially, large organizations with access to such expertise will likely be the key adopters. In larger organizations people with different functional roles in the organization may take one of these defined roles in the framework. For instance, both executives and line-of-business users can take the consumer role in the framework. In smaller organizations, one may take more than one of defined roles in the framework. For instance, the owner of a small retail store may become both the consumer and the analyst.

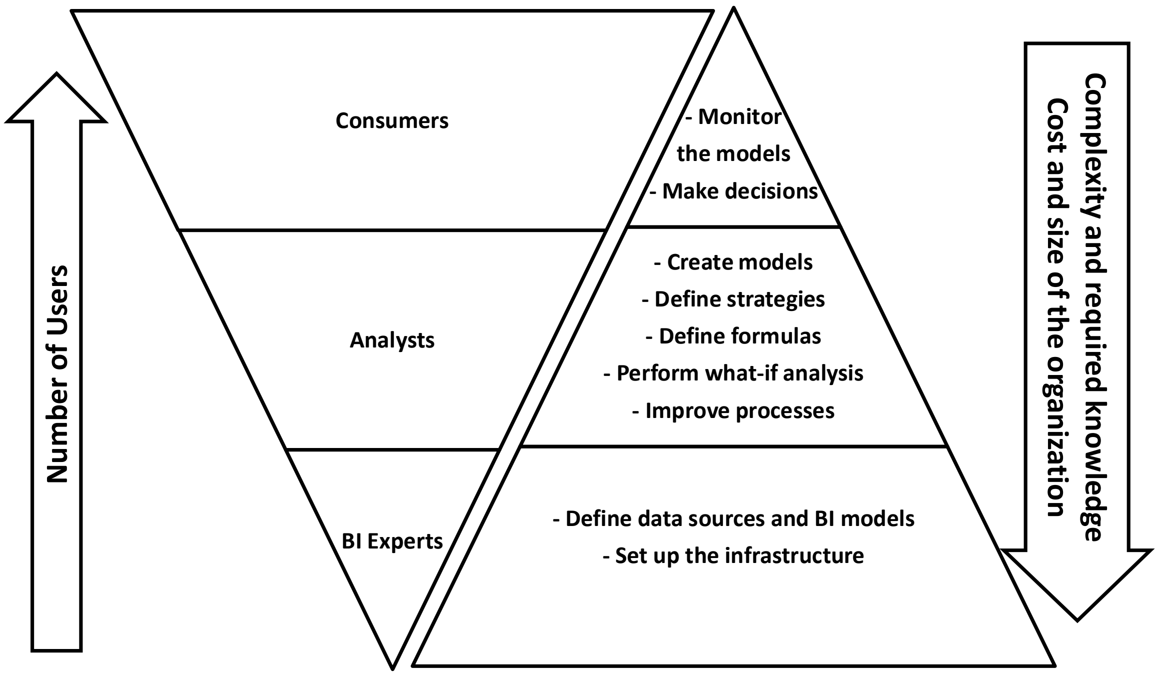


Figure 18: Roles and responsibilities involved in the methodology

4.6. Chapter Summary

In this chapter, the new framework was described at a high level of abstraction. First, an overview and the framework meta-model were presented. Then, the analysis and adaptation lifecycles and their relationships were elaborated. Finally, the roles and responsibilities involved in the framework were discussed.

In summary, the framework helps organizations to first model the context required to monitor their key performance indicators and business objectives as well as their business processes and the decisions that they need to make. Furthermore, it allows the business to define targets for their key performance indicators and detect and improve the undesired conditions in the business. The framework can start with very simple models at a very high level of abstraction and, if required, iteratively and incrementally lead to mode complex and concrete models that involve more roles in the organization.

The next chapter focuses on AoURN-based detection and improvement patterns supporting the adaptation lifecycle. This is a major contribution required to make this framework practical beyond the more common URN-based modeling needed for the analysis lifecycle.

Chapter 5. Detection and Improvement Patterns

In this chapter, we use the list of redesign patterns introduced in section 2.6 and identify which of these patterns can be modeled with AoURN, either only as detection patterns or both for detection and improvement. *Detection patterns* only detect an undesirable condition or find potential application of a redesign pattern but, due to various reasons, human involvement is required for taking further action. *Improvement patterns* not only detect an undesirable condition but also improve the process model by applying a change to the process without as much human involvement as required by detection patterns.

Furthermore, the patterns will be defined as generically as possible to allow them to be applied to various business contexts. In Chapter 7, some of these patterns will be applied to a business process in the healthcare domain and in Chapter 8 some of the patterns will be applied to the retail domain.

In addition, the shortcomings of AoURN for capturing the patterns and the generic requirements for a modeling language "X" (a language independent from URN) to do so are discussed. These requirements can be used to improve AoURN or add the same enhancements to other languages. Finally, potential realization of the requirements in AoURN are demonstrated and discussed in context of the various improvement patterns.

5.1. Business Process Improvement Patterns

Patterns, introduced in section 2.6, are reusable solutions to repeatedly observed problems in a specific context [4]. The context of a pattern is the precondition that makes the pattern applicable and the solution resolves the described design trade-offs (i.e., the forces involved) [95][96]. Although there exist many ways to formalize design patterns [139], to our knowledge and based on our systematic literature review, there has not been any effort to formalize BPI patterns and apply them to adapt business processes while taking *concrete performance measurements* of a business process into account.

The User Requirements Notation (URN) [62] has been used previously to model architectural patterns [95], more formally defining them while focusing on highlighting their impact on forces and enabling pattern users to perform trade-off analysis [96]. URN was also evaluated and improved to better support business process and workflow patterns [94].

The Business Process Improvement (BPI) patterns introduced by Reijers [123], expressed in plain English, touch on various improvement possibilities and show examples of use at different levels of abstraction. Therefore, these patterns represent an excellent data set to derive the requirements for a comprehensive BPI language (called language X from now on). Since these patterns cover many different best practices accepted in the process improvement community [124], it is likely that any proposed requirements covering these patterns will also enable us to model other potential patterns. Table 11 illustrates the patterns and describes their purpose.

As discussed in Table 3, redesign patterns can improve business processes from four different aspects (i.e., forces): cost, time, quality, and flexibility. Usually, not all aspects are impacted positively or equally. For instance when the “Extra Resource” pattern is used, time is impacted positively while cost is impacted negatively. Therefore, it is important to decide which aspect of the business requires improvement, what the priorities are, and what slack is available (e.g., money is available) before the right pattern can be selected and applied to a business process.

The patterns also affect various entities involved in the business process context including operations, IT, customers, structure, and population as elaborated in [124]. For example while the “Task Elimination” pattern influences operations, the “Case manager” pattern affects the structure.

Although these patterns are helpful for business process analysts, it is not easy to decide when to apply the patterns, especially when there are many existing process models in an organization. This motivates our approach to help analysts detect where and when the patterns can be applied and their potential impact [117].

Table 11: Business process improvement patterns

BPI Pattern	Description	G	R
Knockout	Order knock-outs in an increasing order of effort and decreasing order of termination probability	I	HS
Task Elimination	Removes the tasks that do not provide significant value from the business processes	I	HS
Control Relocation	Move control towards customer	I	HS
Case-Based Work	Remove batch-processing and periodic activities from business process	I	HS
Split Responsibilities	Do not assign responsibility to resources from different functional units	I	HS
Resequencing	Move the tasks to a better place in the processs to reduce the setup time	I	HS
Parallelism	Consider executing the sequential tasks in parallel	I	HS
Case Manager	Make one person responsible for handling of each type of process and point of contact with customer	I	HS
Task Automation	Use technology to improve processes	D	BD
Outsourcing	Outsource part of the entire business process	D	BD
Trusted Party	Use information from a trusted party instead of determining the information	D	BD
Extra Resource	Increase the resources	D	IL
Task Composition	Combine small tasks and divide large tasks	D	HS
Contact Reduction	Reduce the number of contacts with customers and third parties	D	HS
Numerical Involvement	Reduce the number of involved parties	D	HS
Interfacing	Standardize the interface with other parties to reduce the error	D	HS
Integration	Integrate process with supplier or customer to increase efficiency and reduce overhead	N	BD
Buffering	Buffer information instead of requesting it	N	BD
Resource Centralization	Use technology to treat distributed resources as centralized resources	N	BD
Case Types	Detect similar tasks for case types and create a new business process	N	IL
Exception	Isolate exception cases from the normal flow	N	IL
Flexible Assignment	Assign resources for maximum flexbility, e.g., assign specialists first to leave the generalists free	N	IL
Case Assignment	Perform as many steps as possible related to one order/case with least number of resources	N	IL
Customer Teams	Consider assigning teams out of different departments to complete a specific customer case / order	N	HS
Specialist-Generalist	Increase number of specialists/generalists depending on the case	N	HS
Empower	Give decision making power to workers to reduce overhead	N	HS
Triage	Divide general task into alternative	N	HS

G: Group, R: Reason for human intervention I: Improvement, D: Detection, N: Not-Modeled
IL: Instance Level information required, HS: Human Smarts required, BD: Business Decision

In Table 11, we divide the patterns into three groups (column **G**), namely Improvement (I), Detection (D), and Not-Modeled (N). The patterns in Group I can be modeled and used to improve business processes, if the language that is chosen for modeling the patterns supports the requirements identified in this thesis. A model of the BPI patterns in this group consists of a detection model (i.e., a pattern expression that matches the As-Is process and goals) and an improvement model (i.e., a description of the To-Be process and goals illustrating the suggested changes to be applied to the As-Is process and expected effects on the goal models). The generic suggested changes often require human intervention unless customized for a specific context.

Group D (Detection) consists of patterns that only have detection models, but do not have improvement models. Therefore, the patterns in this group only find potential deficiencies in the process and leave their resolution to the user.

Finally, patterns in Group N (Not-Modeled) cannot be modeled even with languages that support all requirements identified in this research; this means that neither a detection model nor an improvement model can be defined for patterns in this group. Eleven of the patterns introduced by Reijers [123] fall in this group.

We performed this grouping after trying to model these patterns and realizing that only patterns with the following characteristics can be modeled with language X:

- The improvement model operates at the process model level, i.e., the process is changed for all process instances;
- The improvement model changes the process flow / steps and structure;
- The improvement model adds and removes model elements to and from the business process being improved; and
- The pattern is specific enough to be detectable by the detection model, i.e., it is not at a level of abstraction higher than operational business processes.

Patterns requiring modeling capabilities outside the aforementioned characteristics were placed in Group N (e.g., the "Integration" and "Exception" patterns).

In addition, we observed that patterns with the following characteristics are harder to model. In some cases, without knowing enough about the context it is not possible to formulate a generic improvement model. Consequently, those patterns fall into Group D:

- Require human smarts (HS): e.g., resource elimination, merging/splitting tasks, change responsibilities, and major change to process flow;
- Changes are at business model level and have significant impact on how the business operates (BD): e.g., outsourcing a part of business, investment in technology automation, and changing the organization structure;
- Process instance level information is required (IL): e.g., involves resource skills, data used in the process, and context (e.g., type of order/customer).

Although it is disappointing that all patterns cannot be modeled, we are not the first researchers who come across this issue while working on formalizing patterns and modeling them using graphical notations. El Boussaidi et al. [40] faced similar issues and were unable to model more than 40% of a common set of design patterns, and this 40% corresponded to the patterns that defined a poor (and detectable) solution with a fix.

Column **R** in Table 11 specifies why we were not able to capture the improvement models in a generic enough way for application of the patterns without human intervention.

The first and second issues (HS and BD) are both related to the decision-making part of the improvement process, which due to its impact on the business has to be made by a human. Although there has been much progress in the field of Artificial Intelligence, crucial decisions still require human involvement. For instance, the "Case Manager" pattern requires changes in responsibility and level of authority as well as the flow of the process. Although the pattern can be modeled to detect and suggest an improvement, the final decision and change in the process needs to be made by a person (who also takes responsibility for the actions taken).

The third issue (IL), though, is mainly caused by lack of data and by limitations in capturing information at the level of process instances. For example, the "Exception" pattern is related to the discovery of process instances that do not fit into the majority of the cases, so they can be isolated. We do not address this issue in this research and leave the description of IL requirements for future work. Note, however, that process modeling languages meant for process execution automation as well as logs and data captured in relation to process instances in other enterprise systems (e.g., Data Warehouses) could be utilized for this kind of patterns.

Before taking a closer look at the detailed language requirements imposed by the BPI patterns, there are five basic requirements any modeling language must fulfill to define and detect the required business context, scenarios, and stakeholder goals for BPI patterns at the requirements level:

- support for modeling the behavioral facets of business processes (i.e., the language must support workflow/scenario modeling in some way);
- support for modeling the intentional facets of business processes (i.e., the language must support the modeling of stakeholder goals as well as the impact of business processes on these goals to support reasoning about trade-offs);
- support for modeling of key performance indicators (KPIs) for improved intentional modeling with business process monitoring capabilities;
- support for evaluation and analysis of intentional (goal) models and KPI models, using data provided to the model to allow for business monitoring to quantitatively assess how well a process meets the overall stakeholder goals and KPI targets; and
- support for coordinated, heterogeneous pattern matching against behavioral and intentional models to ensure that structural as well as context-related properties can be detected in the business process model and the specified improvements can be applied to the business process model.

As discussed in section 2.4.1 and illustrated in Table 2, AoURN fulfills these five basic requirements. Therefore, we have used AoURN in this thesis.

5.2. Mapping Patterns to AoURN

The patterns that can be modeled consist of the entities illustrated at the top of Figure 19. The minimum required entity for a pattern that can be modeled is the intentional detection description. For instance, a pattern that only describes one or a set of related KPIs with a defined condition would fit into this category. The most complete pattern, on the other hand, is a pattern that has both intentional and behavioral detection descriptions, as well as the description for intentional and behavior improvement. As an example, a pattern describing i) the goals, KPIs, and process structure required in the base model for the pattern to match, as well as ii) the improvement that can be made to the process and the impact on the performance aspects and goals of the process, fits in this category.

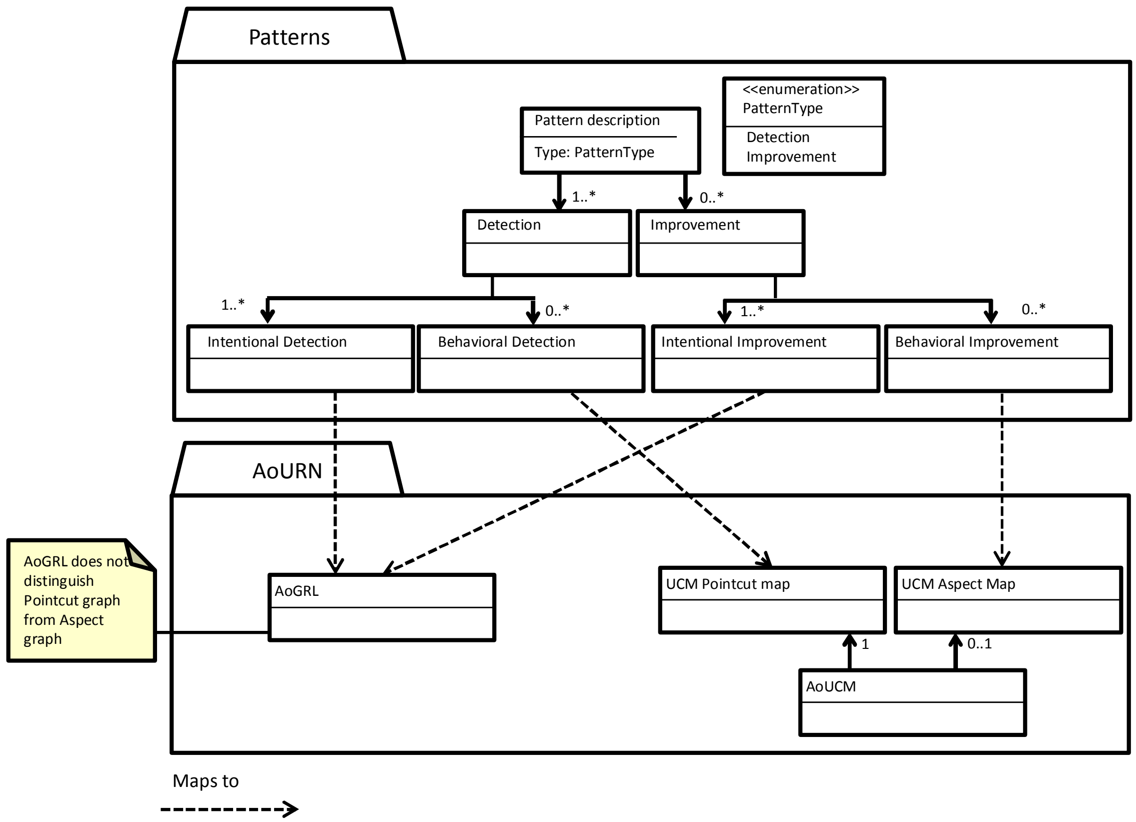


Figure 19: Mapping of the framework patterns to AoURN

Unfortunately, the description of the patterns and best practices that are used as example in this thesis and discussed in [123] do not always follow a concise structure. Patterns can be written in abstract descriptions that may not map directly to the elements illustrated in Figure 16. Therefore, a complete formalization for going from a pattern description to an AoURN model is not possible unless the pattern is very well defined and follows the prescribed structure. Nevertheless, the mapping provided in Figure 16 helps in cases where the pattern description is precise. In addition, this mapping helps with recognising the incomplete patterns, which allows one to fill up the gaps using information from other sources or from the business context. Furthermore, the mapping can also help guiding the formalization of new patterns.

As shown in Figure 19, the intentional parts of the patterns are mapped to AoGRL models and the behavioral parts of the patterns are mapped to AoUCM.

5.3. Example Patterns

In this section, we go over examples of the formalization of detection and improvement patterns. These examples were selected from Table 11 to cover a variety from both groups of patterns. Furthermore, the requirements for language X and the proposed enhancements to AoURN to address these requirements are discussed in the context of the patterns as required. The selected patterns provide enough coverage to explain the approach; they were chosen to highlight the expressiveness problems of AoURN and are applicable to the examples used in future chapters. The patterns from Table 11 that have not been illustrated do not introduce any new language construct or challenge.

The patterns that are discussed in this section are generic patterns and some require *customization* before being applied to a specific context. The customization can be done in two, non-mutually exclusive ways: 1) change a detection pattern to make it applicable more precisely to a given context, and 2) change the expected results (values and functions) according to the businesses' historical data or specific calculations. Pattern customization will be revisited in section 5.4.1.

As a reminder and for convenience, URN and AoURN language constructs are repeated here in Figure 20, Figure 21, and Figure 22. The notations have been introduced in sections 2.4 and 2.7.

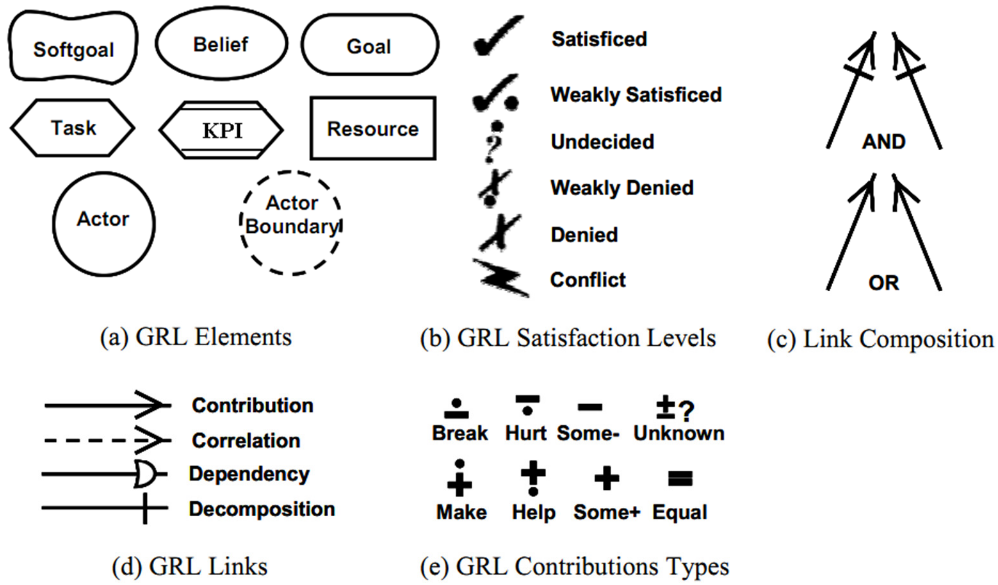


Figure 20: Basic elements of the GRL notation

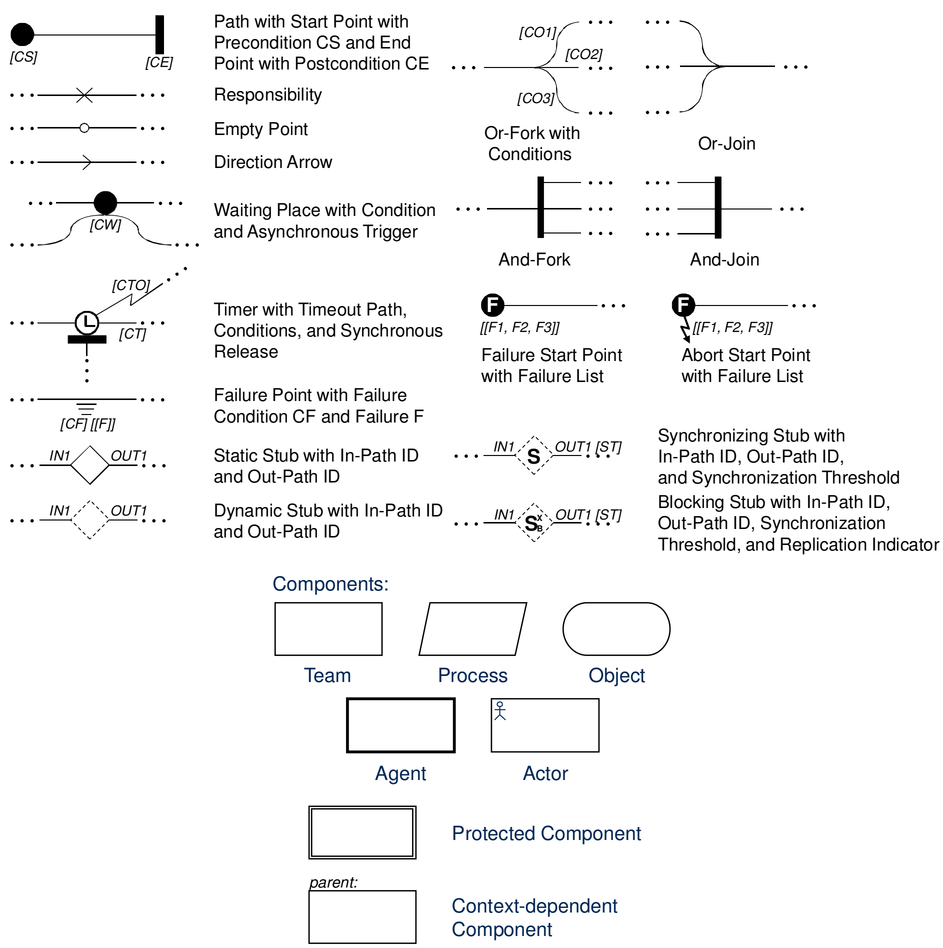


Figure 21: Basic elements of the UCM notation

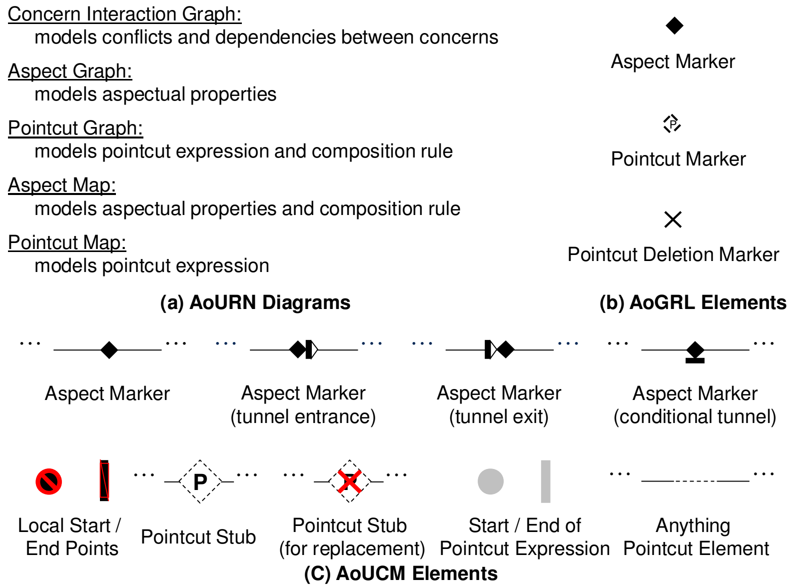


Figure 22: AoURN model elements reference

5.3.1 The Knockout Pattern

The Knockout redesign pattern reorders a sequence of tasks based on their failure rate and effort. Figure 23 shows the generic AoURN model for the Knockout redesign pattern. The aspectual model for the Knockout redesign pattern captures in a generic way the constraints of the pattern.

First, the AoGRL pointcut expression (Intentional Detection model in Figure 23) stipulates that there is an evaluation value of a KPI in the time category that is not satisfactory (i.e., the top KPI in the AoGRL pointcut expression as indicated by the T in the circle and > #TurnaroundTime, respectively). *#TurnaroundTime* is a constant value defined by the modeler, capturing the main premise of the Knockout redesign pattern, and "C:" indicates that there is a *Condition* that needs to be satisfied for the pattern to match. This condition by default is checked against the KPI EvaluationValue unless specified otherwise using a dot notation (for example KPI.SatisfactionValue). As the pattern positively impacts the time category, it should be applied in a condition where a KPI from the time group is not performing as desired.

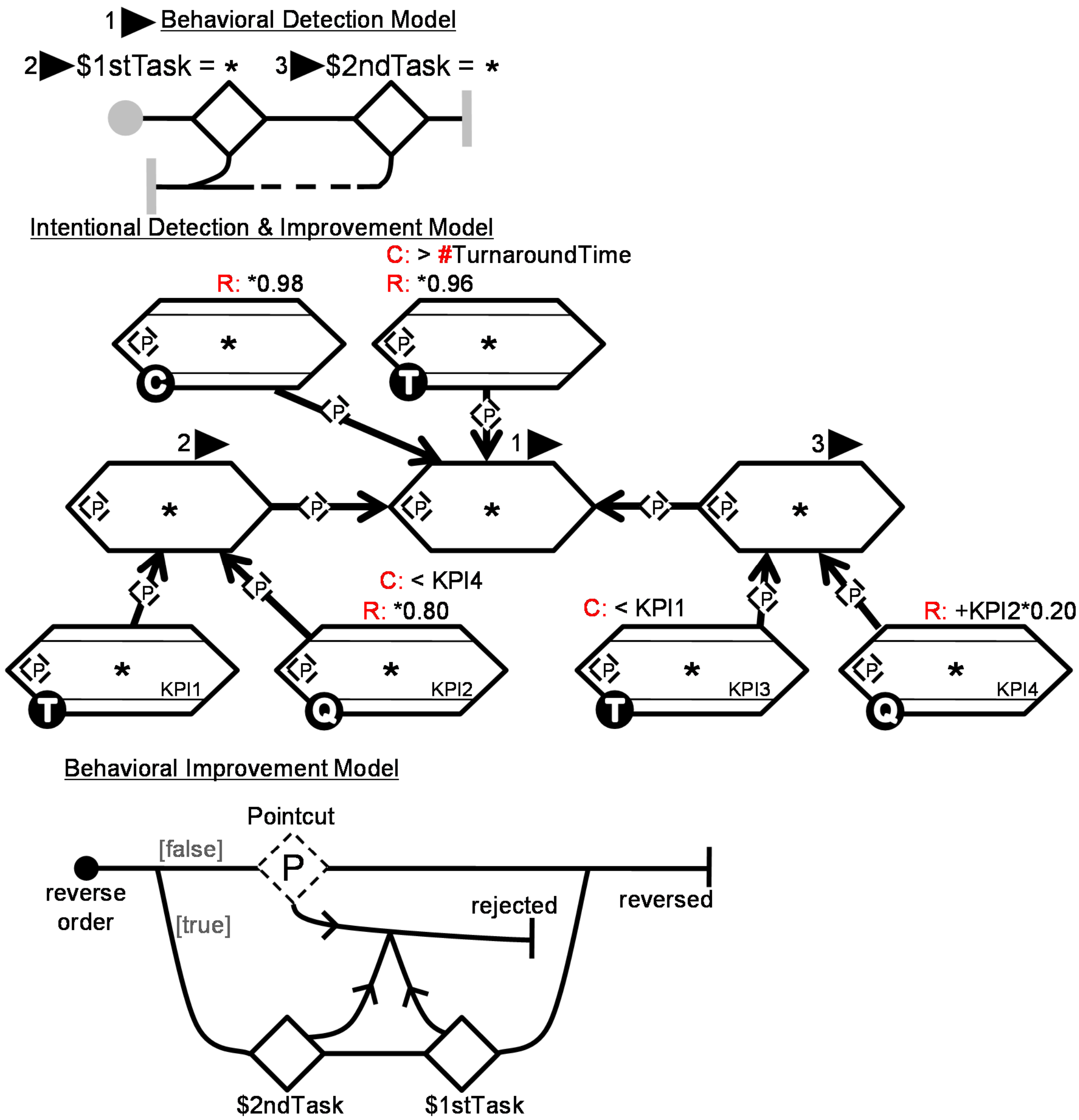


Figure 23: Generic aspectual model for the Knockout redesign pattern

Furthermore, the unsatisfactory KPI is connected to a task (the one with URN link 1) which is further connected with two other tasks (the ones with URN links 2 and 3). All three tasks are traced with the help of the URN links to the two stubs and the map in the AoUCM pointcut expression (Behavioral Detection model in Figure 23). Therefore, additional properties from the AoUCM model must be satisfied for a successful match of the pointcut expression and for the Knockout pattern to be applied. The URN links indicate that the top level task is described by a map and that the other two tasks appear as stubs on the map (i.e., the two other tasks are a refinement of the higher-level task). More specifically, the behavioral detection model indicates that the redesign pattern applies to a series of two stubs (i.e., two process steps) that either succeed or fail. Note that the dashed portion of the pointcut expression matches against any sequence of UCM modeling elements.

At this point, the behavioral detection model would match against a large number of consecutive stubs with two out-paths as long as the overall task associated to the map of the two stubs has a time KPI with an unsatisfactory value. To further improve the accuracy of the Knockout pattern description, further matching criteria are defined for the two other tasks. Note that the task with URN link 2 is the first task and the task with URN link 3 is the second task in the sequence as defined by the AoUCM pointcut expression and the URN links. Each of the two tasks has two more KPIs connected, one from the time category and one from the quality category. Constraints for these four KPIs state that:

- Constraint A: the evaluation value of KPI1 (the time KPI of the first task) must be higher than the evaluation value of KPI3 (the time KPI of the second task);
- Constraint B: the evaluation value of KPI2 (the quality KPI of the first task) must be lower than the evaluation value of KPI4 (the quality KPI of the second task).

This reflects the characteristics of the Knockout redesign pattern as it applies only to conditions where the second task takes less time to do (time KPI is lower) but results in rejections more often (quality KPI is higher) than the first task. If this is the case, then it is advantageous to move the second task ahead of the first task. This change to the process is described by the Knockout Pattern map (Behavioral Improvement model in Figure 23). This map describes the aspectual behavior to be applied if the pointcut expression can be matched for the existing process. As an alternative to the replacement pointcut stub, AoURN describes more complicated replacements of matched model elements with an OR-fork with [false] and [true] branches. The [false] branch describes what is being replaced (i.e., the matched model elements represented by the pointcut stub), while the [true] branch describes the new behavior. On the [true] branch the order of the two stubs from the pointcut expression are switched. The variables $1stTask and $2ndTask defined by the pointcut expression allow matched elements to be reused in the Knockout Pattern map (i.e., the aspectual scenario).

Finally, the AoGRL pointcut expression in Figure 23 also defines the anticipated impact of applying the Knockout pattern on the performance model by describing the changes to the evaluation values for matched KPIs, i.e., the real-life metrics. The annotation R: *0.96 in Figure 23 indicates that as a *Result* of applying the pattern, the *evaluation value* of the unsatisfactory KPI is expected to improve by 4%, i.e., it takes 4% less time than before to complete the process. The expected results are heuristic values that an organization would specify to restrict the application of the pattern to situations where the cost of implementing the pattern is justified. An example of this particular pattern's application is illustrated in section 7.2.1.

5.3.2 The Task Elimination Pattern

This pattern is often used to remove process steps without significant value [123]. Therefore, the aspectual model for the Task Elimination pattern (see Figure 24) indicates with the pointcut replacement stub that the responsibility matched by the AoUCM pointcut expression (i.e., behavioral detection mode) is removed from the model. Since the aspect does not add any responsibility of its own to the base model, this is an actual removal and not just a replacement.

The responsibility that is being removed, however, also needs to match characteristics defined by the AoGRL pointcut expression (i.e. intentional detection model). The task linked to the responsibility must have a contribution link with a quality KPI. This quality KPI is used to measure the value that this tasks adds to the process. Furthermore, the evaluation value of this KPI must be below *#significantValue* – a constant defined by the user. If these conditions hold, the task and its two KPIs are also removed from the model as indicated by the pointcut deletion markers. This pattern is an example of a generic pattern that needs to be tailored to a more specific condition to avoid false positives. There is an example of a customized version of this pattern in section 7.2.2.

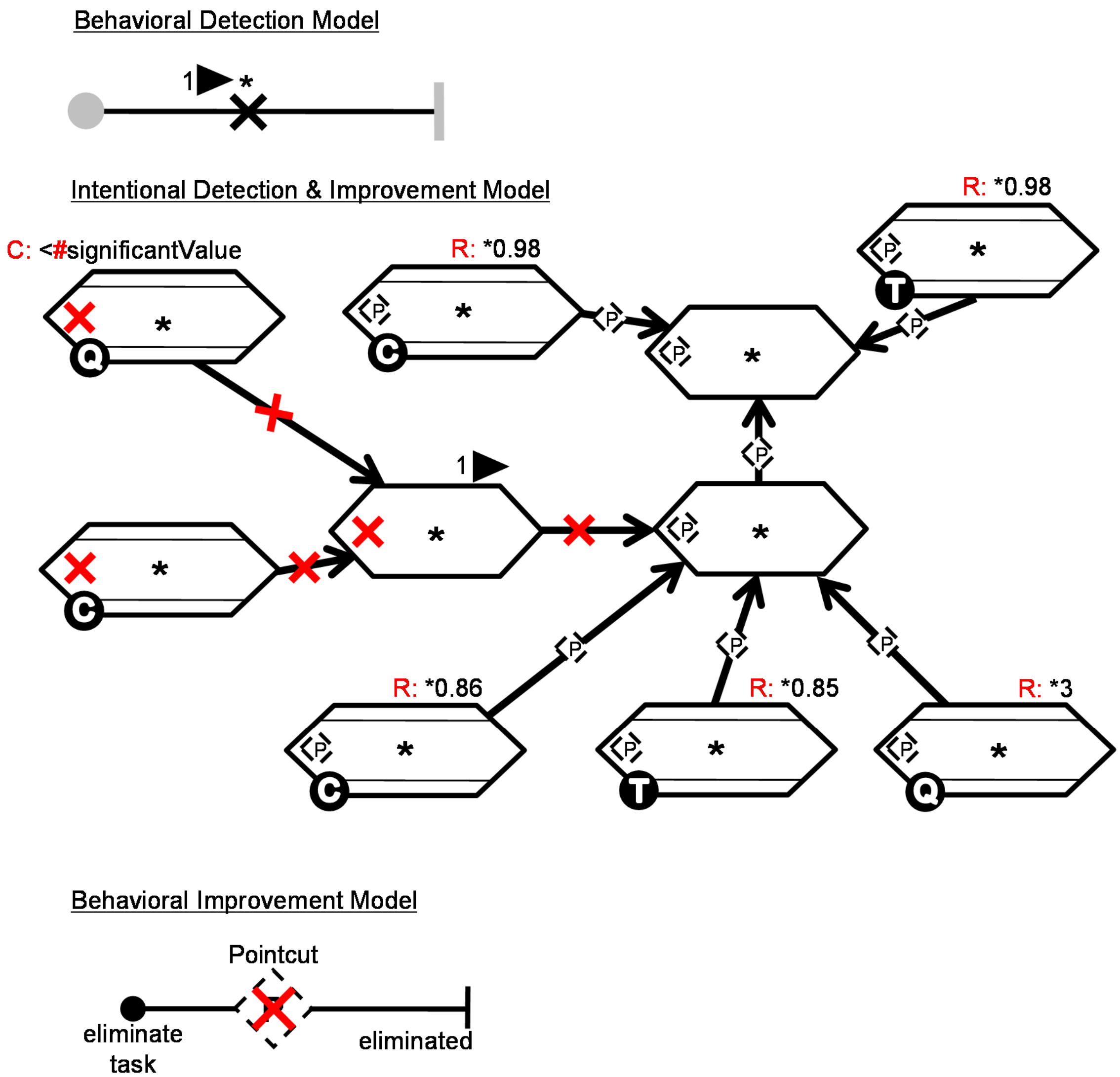


Figure 24: Generic aspectual model for the Task Elimination pattern

In addition, the impact of the removal is shown for the KPIs of two other tasks linked directly and indirectly with the removed task. As the pattern removes a task from the overall process, reductions in cost and turnaround time are anticipated (i.e., the evaluation value of KPIs in the cost and time categories decrease: *0.98, *0.86, *0.98, and *0.85) and hence satisfaction will increase. On the other hand, the quality aspect of the process that was controlled by the removed task will decline albeit only by a small number.

5.3.3 The Control Relocation Pattern

This pattern is relevant to application submission processes and is often used to move input validation checks to the client side [123]. Therefore, the aspectual model for the Control Relocation pattern (see Figure 25) indicates that the responsibility Go over Check List is removed from its current location and reinserted before the Submit end point in the submission process. This, however, is only done if the intentional detection model is satisfied, i.e., the evaluation value of the quality KPI Avg. Rejection due to Missing Docs, which is linked to the task of the responsibility is above a certain *#qualityTarget*. Note how the behavioral detection mode (i.e., AoUCM pointcut expression) is an example for interleaved matching and allows two UCM paths representing separate sub-processes to be matched. This is possible because the two sub-processes are linked via their parent maps. This connection can be matched by the anything pointcut element in the AoUCM pointcut expression of this pattern.

Applying the Control Relocation pattern impacts the evaluation values of the quality KPI Avg. Rejection due to Miss* as well as other cost and time KPIs. Using Miss* in this case allows the pattern to match against a broader set of base models and makes the pattern more generic (e.g., Missing Docs, Missing Documents, and Missing Material would all match). The evaluation value of the quality KPI is estimated to be reduced by 80% (=*0.20), thus increasing the satisfaction value of this KPI. The evaluation values of the cost and time KPIs are also decreasing (R: *0.965, R: *0.93, R: *0.975, and R: *0.94), therefore also increasing their satisfaction values.

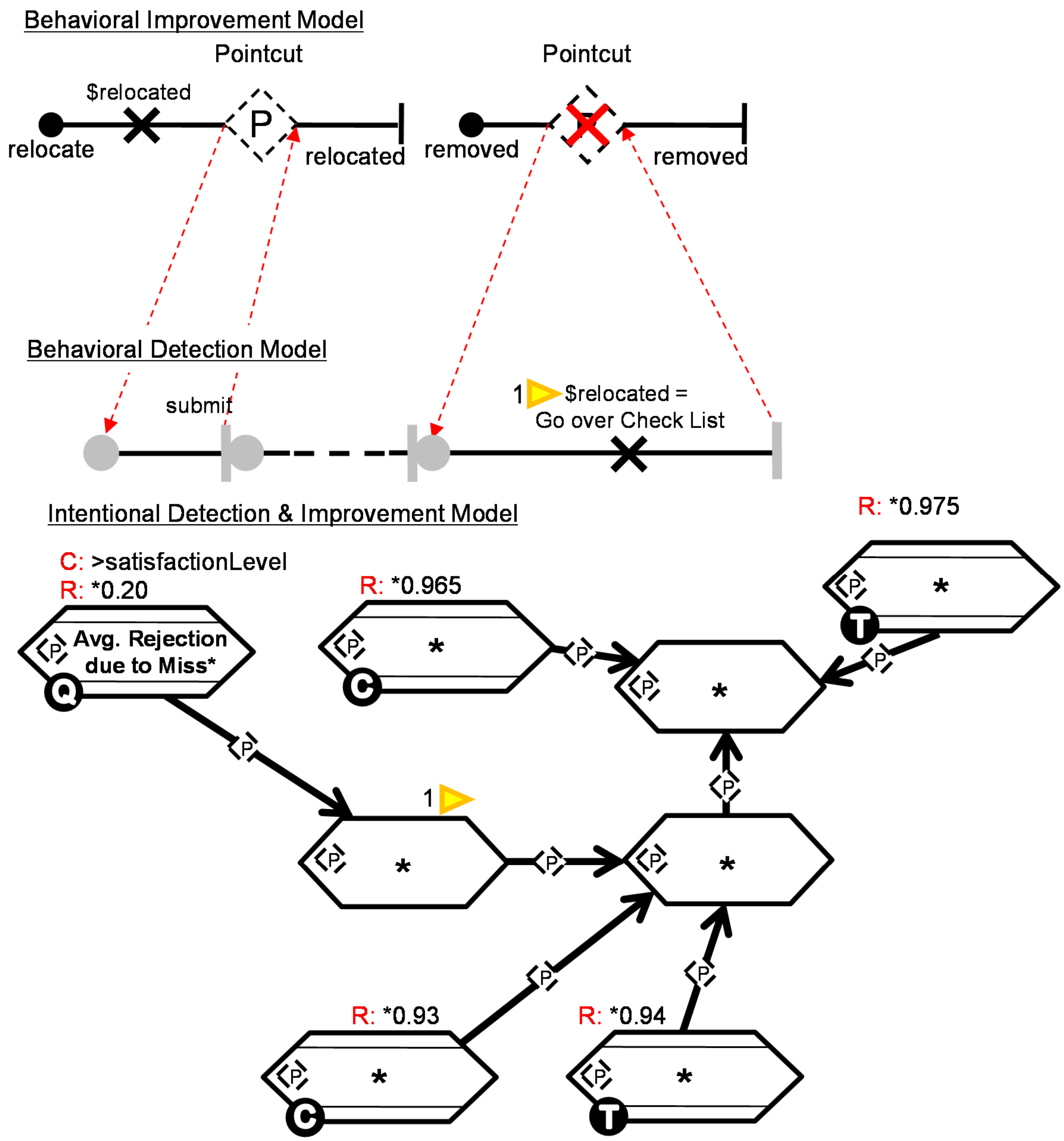


Figure 25: Generic aspectual model for the Control Relocation pattern

5.3.4 Task Composition Pattern

The "Task Composition" pattern combines small tasks to reduce setup time and improve turnaround time of a process. Although combining tasks on the surface may seem simple, deciding on which tasks to combine and the roles that will perform the combined task as well as optimizing the combined task requires human smarts. Therefore, we consider this pattern a "Detection" only pattern. Figure 26 and Figure 27 respectively show the behavioral and intentional detection models for this pattern, which helps identify some of the proposed requirements.

The behavioral detection model in Figure 26 matches when there is repetition of either two or more UCM responsibilities (×) or stubs (◇) (i.e., Option A and Option B, respectively).

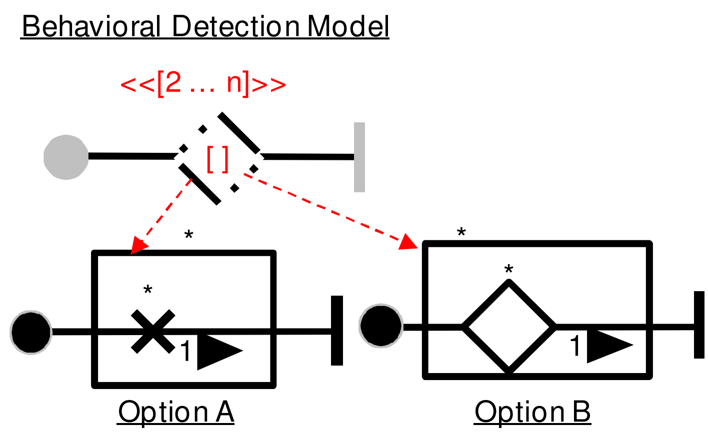


Figure 26: Task Composition – Behavioral detection model

Modeling repetition is necessary for this pattern and some of the other patterns (e.g., "Contact Reduction", "Parallelism", and "Case Manager"). Therefore, we require a *Repetition Container* model element allowing us to capture repetition. In AoURN, we address this requirement by adding a new stub type called "repetition stub" (◇). We use the array and stereotype syntax to show the expected number of repetitions. For instance, in this example pattern, we have used <<[2 … n]>> to show the behavioral detection model only matches, if at least two repetitions exist. Furthermore, the array syntax in combination with URN links (▶) allows each instance of the repetition to be referenced in the intentional detection model. For example, in Figure 27, [1] and [n] are used to reference the first and n^{th} instance of the repetition and [1 … n] is used to indicate that the *Time KPI is defined for all instances of the repetition. We believe an array of model elements or a construct consisting of multiple model elements is required for flexibility and scalability to allow us to model cases like the "Task Composition" intentional detection model. Otherwise, for model elements like the *Time KPI in Figure 27, one KPI per instance must be added to the detection model.

In addition, since the repetitions may describe elements at a lower level of abstraction (i.e., responsibilities) or higher level of abstraction (i.e., stubs) or a mix of both, we need an approach to show alternative possibilities (i.e., Option A and Option B in Figure 26). This requirement is not limited to this example pattern. Any other pattern that has alternative workflow structures to match against has the same requirement. For instance, the “Extra Resource” pattern can be applied either to a process with one resource performing a task repeatedly or to a process with multiple resources processing a queue, which would be modeled in a different way, hence, requiring a different detection approach. Consequently, we are proposing another type of model element to behave as an *Options Container*. In AoURN, we translate this requirement to the “options stub” ( ). The stub illustrated in Figure 26 is a *Composite Container* supporting the semantics of both the repetition stub and the options stub, which is yet another requirement for language X. In AoURN, we translate this requirement to the new “composite stub”. Note that these proposed model elements not only help with defining BPI patterns, but also improve the modularity of requirements models that describe more complex behavioral requirements of a system.

The intentional detection model in Figure 27 matches against a KPI ( ) of type time (Ⓣ), with a name ending in Time (*Time), and associated with the matched responsibility or stub as indicated by the URN link (▶). A *Condition* (C: <#eachTaskTimeValue) constraints the match to those with a time value below a certain *Constant* (#). In general, an element that needs to be matched is identified by the conventional AoURN icon. Therefore, two more model elements need to be matched, i.e., the ones labeled StartTime and EndTime.

While modeling this pattern, we observed the lack of a model element allowing us to use *Raw Data Input* to calculate more complex KPI values in Figure 24. Therefore, we are proposing a new model element and visualize this element in AoURN as a “data model element" ( ). The difference between a KPI and a data model element is that, for a KPI, a conversion from a real-world value to a GRL satisfaction value takes place, while this does not happen for a data model element and its real-world value is used as is in the GRL model. A data model element can only contribute to KPIs.

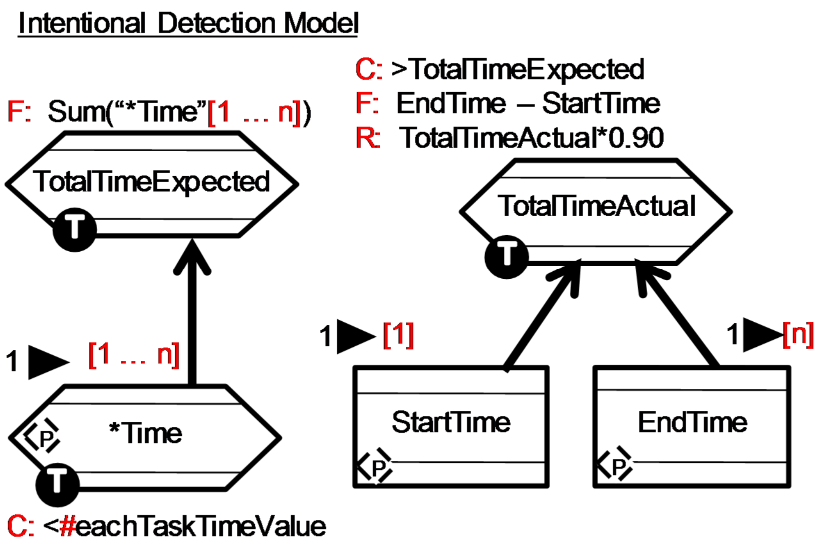


Figure 27: Task Composition – Intentional detection model

Both of the data model elements in Figure 24 need to be matched, i.e., the StartTime for the first responsibility or stub and the EndTime of the last (n^{th}) responsibility or stub. These two elements are then combined into the TotalTimeActual KPI as defined by a *Formula* (F: EndTime – StartTime), which calculates the actual duration of the process from the first to the last matched responsibility or stub. In addition, a second KPI (TotalTimeExpected) and formula is defined (F: Sum(“*Time”[1 … n])) that calculates the sum of all individual task durations. Another condition (C: >TotalTimeExpected) expresses the overall constraint that a combination of tasks takes longer than expected. Conditions, formulas, and constants are used very frequently for intentional detection models and are hence an integral part of the model.

Finally, an anticipated *Result* (R: TotalTimeActual*0.90) is specified for the TotalTimeActual KPI, meaning that the anticipated improvement is a 10% reduction of the duration of the process. This can be used to provide insight into relative advantages of several candidate/competing patterns that can be applied [117], e.g., preference is given to the pattern with the greater positive impact on the overall goals of the business. In summary, language X needs to support the ability to specify *Conditions*, *Formulas*, *Constants*, and expected *Results*.

5.3.5 Case Manager Pattern

"Case Manager" is an "Improvement" pattern that detects conditions requiring a managerial review or check at the end of a series of tasks or sub-processes performed by various roles in an organization. Although the pattern tends to improve the quality metrics of the processes, it may have negative effects on time and cost.

Figure 28 shows the detection and improvement models for this pattern. The intentional detection model of the pattern matches against quality KPIs (Ⓠ) with an evaluation value lower than a defined constant called #QualityTarget (note that quotes are used to reference the * KPI name to avoid confusion with the multiplication symbol). The anticipated result (R:) of the "Case Manager" pattern is an improvement of 20%. The KPI needs to be defined for a process consisting of a series of tasks as the URN link (▶) refers to the whole behavioral detection model and not an individual element.

The behavioral detection model uses the repetition stub (◇) proposed earlier. In this case, the repetition stub is used to show a series of at least #MinRoleCount sub-processes (i.e., stubs) executed sequentially in the process. This series of stubs, however, must not already be followed by a managerial check, necessitating an *Absent Container* model element allowing us to describe model elements that must not be matched. We address this requirement in AoURN with another new stub type called "absent stub" (◇).

The improvement model in Figure 28 uses conventional AoURN syntax to describe the effect of applying the "Case Manager" pattern. The pointcut stub (P) signifies the locations where the pattern can be applied in the process model because the detection model can be matched, i.e., it represents the detection model. Since the check stub is shown after the pointcut stub in the improvement model, it is applied after the matched locations in the process model. Note that the check stub in the improvement model actually reuses the same UCM sub-model used in the absent stub.

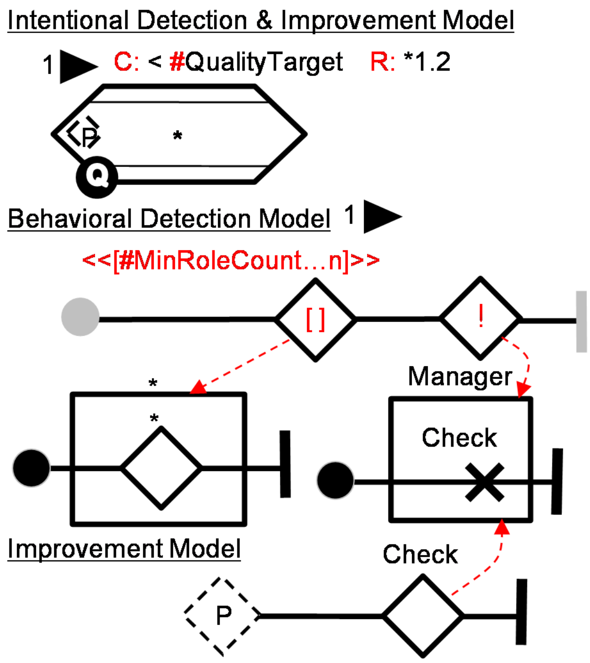


Figure 28: Case Manager – Intentional and behavioral detection models, as well as improvement model

5.3.6 Task Automation Pattern

Automation is a widespread practice for process improvement that can be as minor as automating an approval process in an organization or as major as changing a brick and mortar to an online business model. Therefore, coming up with a generic "Task Automation" pattern to improve all the potential automation opportunities in an organization is not feasible. However, providing a dictionary of opportunities based on best practices can be viable solution.

As this dictionary evolves over time, it is useful to separate it from the specification of the detection model. Hence, there is a need to specify a *Reference* to an external source as well as an *Action* to be performed with this external source. Figure 29 shows how this requirement is addressed in AoURN for the "Task Automation" pattern. In the behavioral detection model, a responsibility is given a new keyword (<<orList AutomationDictionary.#Term>>). orList indicates that the external source contains a list of items that can be matched. AutomationDictionary references the external source and #Term accesses a particular data attribute of this external source. The same approach can be used for the “Outsourcing” and “Trusted Party” patterns to find the potential sub-processes that can be outsourced or information that can be acquired from trusted parties.

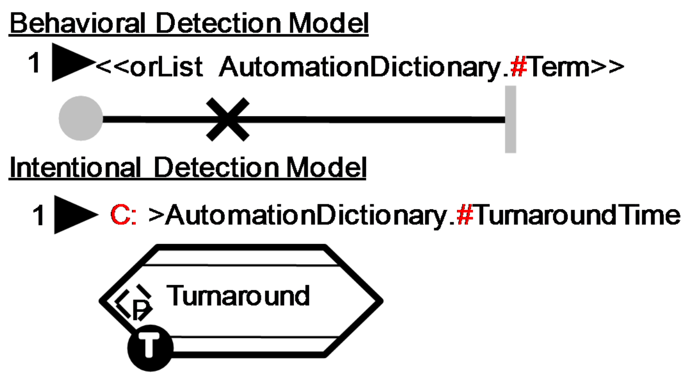


Figure 29: Task Automation – Behavioral and intentional detection models

As indicated by the URN link (▶), the responsibility must also satisfy a condition on its Turnaround KPI. In the condition, another data attribute of the external source is referenced. Note that several actions may be defined (e.g., orList, andList…). Furthermore, general issues of name-based matching such as how to match typos or synonyms are orthogonal to the discussed subject matter and have been discussed in a recent taxonomy [97].

5.3.7 The Resequencing Pattern

The Resequencing pattern moves a task or a sub-process to the best place in the process in order to reduce the setup time. Processes with multiple roles between which the flow goes back and forth, and the nature of a process that does not mandate a strict sequence, are usually good candidates for application of this pattern. Figure 29 shows the Resequencing pattern, which detects the cases where two distinct roles (i.e., $1stRole and $2ndRole) perform a sequence of tasks. In this case, the flow starts with $1stRole, goes to the 2ndRole, and finishes with the 1stRole. The intentional part of the detection model matches when the setup time for the second time that the task comes back to 1stRole is higher than a defined constant (i.e., #MaxSetuptime). The improvement model moves the $3rdTask performed by the 1stRole right after the $1stTask to reduce the setup time.

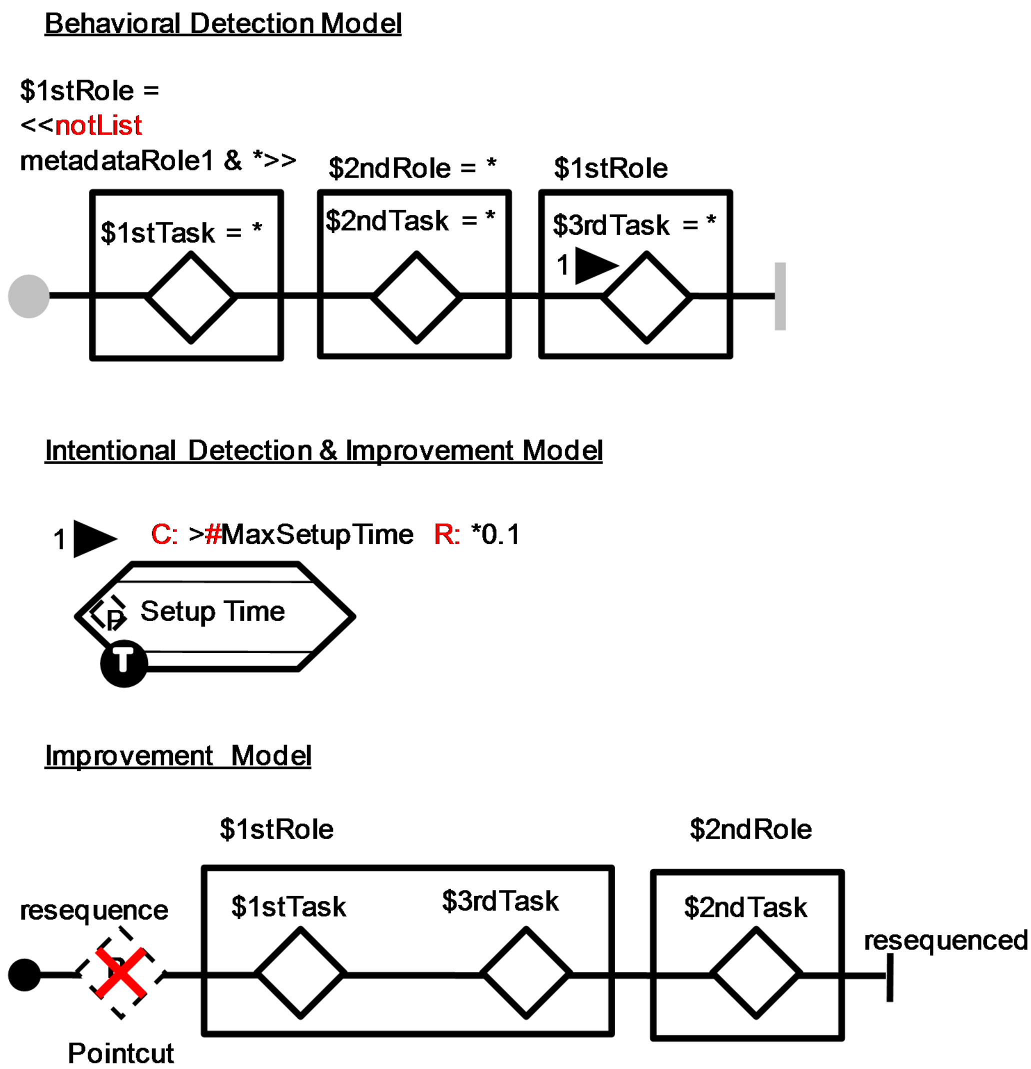


Figure 30: Resequencing Pattern – behavioral and intentional detection and improvement models

Note that the roles and tasks names in this pattern start with $, which indicates the use of variables. Using distinct variables instead of just the expected text pattern (e.g. *) tells the matching algorithm that even though $1stRole and $2ndRole can be anything (i.e., *) they are distinct. This will prevent the pattern to match against a process whose same role performs three different tasks sequentially.

Furthermore, the *notList* keyword with a metadataRole1 list is used to define the matching pattern for $1stRole (<<notList metadataRole1 & *>>). This tells the algorithm to match any role except for those that are on the list defined in metadataRole1. This syntax allows one to reduce the number of false positives in cases where the pattern matches but is not applicable. A good example is a process where a manager defines the work in $1stTask and reviews it in $3rdTask.

5.4. Application of Patterns

As can be seen in the examples of the previous section, the intentional detection model may be quite complex. Using typical pattern matching algorithms, the intentional detection model is matched in the base model, only if the exact same elements and conditions are present. Therefore, to enable matching, modelers are forced to have either:

a) very well defined base models using the same vocabulary/labels and structure as what is found in the patterns (might be too strict), or
b) very generic BPI patterns without strict labels or structures (might be too relaxed).

The first option could defeat the purpose of assisting modelers in the process improvement journey whereas the second option could cause many false positive matches. For example, the reason why the intentional detection model in Figure 27 is not matched could be that one of the conditions (e.g., C: <eachTaskTimeValue) is violated or because one of the data model elements (e.g., the StartTime raw data element) does not exist in the base model. However, these two reasons represent two completely different issues. The first means that the pattern should not apply, while the second one means that the pattern may potentially apply but a lack of data in the base model prevents a final determination of the applicability at this point in time.

In another example, the intentional part of Control relocation pattern, illustrated in Figure 25, may not find any match in a base model just because the base model uses a different label pattern than "Avg. Rejection due to Miss*", but not because other conditions in the pattern are not satisfied. On the other hand, the pattern could be made more generic by using * as the label instead of "Avg. Rejection due to Miss*", which in turn may match several cases. However, some of these cases may well be false positives.

Therefore, some customization may be required before patterns get used in an organization, either on the base model or on the generic patterns. Furthermore, the framework helps the analysts by providing support for *Partial Pattern Matching and Reporting*.

5.4.1 Pattern Customization

Pattern customization can be done on various attributes of the patterns. The most likely customizations are performed on:

- Model element labels: renamed to match the organization terminology.
- Conditions: Can be made looser or stricter.
- Expected results: Can be modified based on the organization's expectations or historical data.

However, in more extreme cases, one may also modify the structure of a pattern or use a pattern as a starting point to define a new pattern with a different set of expected model elements.

An example of pattern customization is illustrated in Figure 48. The Task elimination pattern has been customized with more specific model element labels to avoid false positives.

5.4.2 Support for Partial Pattern Matching and Reporting

To facilitate the application of the patterns and due to the aforementioned issues, language X is required to support a new matching algorithm with partial matching capabilities. The partial matching algorithm does not require the complete structure and conditions to be present in the base model to match. This algorithm matches in three stages. First, it matches against the behavioral detection model where an exact match is required. In the second stage, the algorithm looks for matches against the intentional detection model, but finds partial matches as well. For instance, if only one of three KPIs defined in an intentional detection model is found in the base model, the algorithm detects a partial match. In the third stage, the algorithm matches against the conditions defined for each matched KPI (note that not all conditions may hence be evaluated). Finally, the result of the partial matching algorithm is a report with the number of matches against the behavioral detection model further broken down into the following categories based on the match against the intentional detection model:

- the number of overall successful matches;
- the number of unsuccessful matches due to violated constraints; and
- the number of potential matches sorted using match probabilities calculated based on the number of matched model elements.

This three-stage partial matching approach is useful because not all model elements required by the intentional detection model to match need to exist in the base model right from the start. In other words, a modeler requires less preparation time for existing process models to use the BPI patterns. In addition, finding partial matches educates an organization to gather the data and add the KPIs and data model elements to their models as required for business processes measurement, monitoring, and improvement. Therefore, this approach reduces the complexity of the application of BPI patterns previously observed [117].

The three-stage partial matching approach, however, has implications on the visualization of matches as full and potential matches need to be differentiated. This highlights another requirement for language X, which is a method for identifying the different types of matches in the base model. Similarly, the patterns in Group D and I in Table 11 also pose different visualization requirements. Therefore, it must be possible to indicate the group to which a pattern belongs in the model. In AoURN, this requirement is realized by allowing the type of the pattern (i.e., one of the two groups D or I) to be defined for each of the patterns. An example of partial pattern matching algorithm is illustrated in section 8.4.

5.5. Requirements Summary

While the ability to match intentional and behavioral detection models is a basic requirement and is supported by some modeling languages, many other requirements related to the specification and matching of BPI patterns, as discussed with the help of examples in previous sections, are yet to be supported by modeling languages. These new requirements for language X are categorized in Table 13. They go beyond the basic requirements that have been already discussed in Table 2, which were used to select AoURN as a basis for this research.

We prioritize these requirements using five criteria (Table 12). Each requirement is tagged by these priority criteria in Table 13. The priority score of a requirement is the total number of points that the requirement acquires based on these tags. All of the criteria have fixed points except for Patterns (P), whose points are specified based on the number of patterns impacted by the evaluated requirement. The highest possible number of points for P is 16, which is the total number of patterns in Groups I and D. Due to space constraints, we only elaborate the tagging rationale for two of these requirements to convey the thought process we went through for this prioritization activity.

Table 12: Prioritization criteria

Criteria	**Description**	**Points**
Patterns (P)	must have for modeling *N* number of the BPI patterns (maximum for N is 15)	N
Framework (F)	must have for the success of the framework	5
Usability (U)	improves the usability of the framework and language	3
Expressivness (E)	increases the expressivness power of the modeling language	2
Scalable (S)	improves the scalability of the language	1

For example, if “Partial matching and reporting” is not supported by the language, either the defined patterns have to be very generic to match against incomplete base models, which causes many false positives and noise, or the base models have to be very comprehensive and developed having the patterns in mind, which defeats the purpose of the patterns. Therefore, we consider this requirement a must have for the success of the framework (F). If this requirement is supported, the patterns can be modeled including all the KPIs and be as specific as required without worrying about missing potential matches. Therefore, this requirement influences how all patterns are modeled (P = 16). Moreover, the reporting approaches discussed in section 5.4 increase the framework’s usability (U) by allowing modelers to complete their base model and find potential improvement opportunities. Finally, the reporting approach also helps with the scalability (S) of the framework because reporting a high number of matches in a concise manner allows the framework to be used against larger size models and does not rely on human eyes to detect full and partial matches. The total priority score for this requirement as shown in Table 16 is hence 26 (16 for P, 5 for F, 3 for U, and 1 for S).

Table 13: Prioritized requirements for language X

<u>**Requirement**</u> ***[AoURN Implementation]*** **Description of Requirement *(Priority Criteria → Priority Score)***
<u>R1. Pattern types identification</u> *[Aspect types]* Language X shall identify the type (i.e., "Detection" and "Improvement") of a BPI pattern to allow for differentiated treatment. *(P = 15, F, U, E → 26)*
<u>R2. Conditions, formulas, constants, and results</u> *[C, F, #, and R]* Language X shall support the specification and evaluation of conditions, formulas, and expected results for KPIs including the ability to use constants. *(P = 15, F, U, E → 26)*
<u>R3. Partial matching and reporting</u> *[Partial AoURN pattern matching]* Language X shall report the number of partial matches and clearly visualize model elements that could not be matched in a partial match to reduce the dependency of the pattern expression on existing KPIs in the base model. *(P = 15, F, U, S →25)*
<u>R4. Identify multiple matches</u> *[Annotated Aspect Markers]* Language X shall support the visualization of multiple matches (i) in a distinguishable way in the intentional and behavioral base models and (ii) ideally with only one model element for multiple, added instances in the intentional base model. *(P = 15, F, E → 23)*
<u>R5. Match KPI types and conditions</u> *[Improved AoURN pattern matching]* Language X shall support matching against the type of a KPI as well as the condition of a KPI. *(P = 15, F → 21)*
<u>R6. Detection highlighting</u> *[Improved AoURN pattern matching]* Language X shall support the visualization of the matches of a BPI pattern in the "Detection" group even though there is no implementation model to be applied to the base model. *(P = 9, F, U, S → 17)*
<u>R7. Repetition Container</u> *[Repetition Stub]* Language X shall support the specification and matching of a group of repeated model elements in behavioral models. *(P = 5, E, S → 8)*
<u>R8. Actions and References</u> *[AoURN Actions and References]* Language X shall support a set of advanced commands and the ability to reference external sources in the specification and matching of detection models. *(P = 5, E, S → 8)*
<u>R9. Array of model elements</u> *[<<[n…m]>>]* Language X shall support the specification and matching of similar intentional model elements that need to be repeated several times as an array of model elements for improved ease of modeling and scalability. *(P = 2, E, S → 5)*
<u>R10. Options Container</u> *[Option Stub]* Language X shall support the specification and matching of groups of alternative model elements in behavioral models. *(P = 3, E → 5)*
<u>R11. Absent Container</u> *[Absent Stub]* Language X shall support the specification and matching of a group of model elements that should not be in the base model for a successful match of behavioral models. *(P = 1, E → 3)*
<u>R12. Composite Container</u> *[Composite Stub]* Language X shall support the combination of various container semantics (e.g., Repetition Container with Options Container) for behavioral models. *(P = 1, E → 3)*
<u>R13. Raw data input</u> *[Data Model Element]* Language X shall support the specification and matching of raw data in intentional models to be used as input for the calculation of KPI values. *(P = 1, E → 3)*

P: Patterns, F: Framework, U: Usability, E: Expressiveness Power, S: Scalability

On the other hand, while lack of the "Options Container" prevents us from modeling some of the patterns (P = 3), it does not completely prevent us from using the framework (not tagged with F). It, however, is tagged with E because it increases the expressiveness power of language X with the ability to describe groups of alternative model elements. Therefore, the total priority score for this requirement, as shown in Table 13, is 5 (3 for P and 2 for E).

5.6. AoURN Profile for Language X

Language profiling is a technique that has been used in other research [129] with similar needs to take advantage of language extensibility features to tailor the language behavior while using existing tool support and object model to validate the research. This approach allows us to implement and further explore the ideas put forward in this thesis before proposing them as standard features of the language, which would imply modifications of the language's metamodel. URN, AoURN and the jUCMNav tool support profiling mainly through use of stereotypes (*metadata* whose name starts with "ST_") for tagging model elements, *concerns* for grouping model elements, *URN typed links* for associating model elements, and *OCL constraints* for specifying additional well-formedness rules on these stereotypes, concerns, and links [8][11]. These extension mechanisms do not require modifications to the metamodel of the language. Table 14 summarizes how each of the aforementioned requirements is defined in a new AoURN profile called *Business Process Pattern (BPP) Profile*. Note that the implementation of this profile and the related improved matching and evaluation algorithms are listed as future work.

Table 14: Requirements implementations in AoURN

Requirement ***[AoURN Implementation]*** **Implementation details**
R1. Pattern types identification *[Aspect types]* Aspect types are identified using a stereotype called ST_BPP_PATTERN. The possible values for this stereotype are BehavioralDetection, BehavioralImprovement, IntentionalDetection, IntentionalDetectionImprovement
R2. Conditions, formulas, constants, and results *[C, F, #, and R]* The specification of these patterns attributes is done using the following set of metadata. BPP (Business Process Pattern) is used to namespace these metadata elements: C → BPP_CONDITION, R → BPP_RESULT, # → BPP_#<*label*> Since formula is not specific to patterns and can be used in other context it uses the following metadata FormulaBasedGRLStrategyAlgorithm_evalFormula
R3. Partial matching and reporting *[Partial AoURN pattern matching]* The details for how partial matching will work is yet to be identified and is part of the future work.
R4. Identify multiple matches *[Annotated Aspect Markers]* After detection of multiple matches, an enumeration of alphabetical annotations (e.g., *a*, *b*, and *c* for three matches) is used to annotate the aspect markers. As a usability enhancement, color coding can be used between the annotations and aspect markers.

Requirement *[AoURN Implementation]* **Implementation details**
R5. Match KPI types and conditions *[Improved AoURN pattern matching]* They KPI types are specified using the ST_BPP_KPITYPE stereotype and the values of this stereotype are time, cost, quality, and flexibility. The BPP profile adds the appropriate KPI type annotation to the KPI model elements when this stereotype is present, and the matching algorithm uses the type to only match the KPIs that have identical types in the base model. The implementation of the improved matching algorithim is listed as future work.
R6. Detection highlighting *[Improved AoURN pattern matching]* The improved AoURN pattern matching algorithm works in two stages. In the first stage, it matches the pattern in the base model and marks the matches using aspect markers. In the second stage, the algorithm weaves in the improvement model. When the improvement model is not present, stage one still works independently and shows the aspect markers to indicate a match.
R7. Repetition Container *[Repetition Stub]* Repetition stub is specified using the following stereotype and value: ST_BPP_STUBTYPE = Repetition. The BPP profile renders the stubs identified as repetition stubs using the following visual representation
R8. Actions and References *[Improved AoURN pattern matching - Actions and References]* The actions and references are specified using the model element labels and a set of defined keywords. Therefore, no new stereotype or model element is required to capture them. The matching algorithm though, needs to be improved to detect these keywords as it parses the labels and take the appropriate action along the way.
R9. Array of model elements *[<<[n…m]>>]* The specification of array of the intentional model elements is done using the current model element labels and the <<*[n…m]*>> format, where *n* and *m* are integers or constants. The improved pattern matching algorithm detects this format and looks for multiple similar model elements in those cases.
R10. Options Container *[Option Stub]* Option stub is specified using the following stereotype and value: ST_BPP_STUBTYPE = Option. The BPP profile renders the stubs identified as option stubs using the following visual representation:
R11. Absent Container *[Absent Stub]* Absent stub is specified using the following stereotype and value: ST_BPP_STUBTYPE = Absent. The BPP profile renders the stubs identified as absent stubs using the following visual representation:
R12. Composite Container *[Composite Stub]* Composite stub is specified using ST_BPP_STUBTYPE stereotype and a comma separated list of values that specify the stub types that are used together. For example ST_BPP_STUBTYPE = Option, Repetition specifies a composite stub consist of Option and Repetition stub semantics. The BPP profile renders the stubs identified as composite by merging their visual representations.
R13. Raw data input *[Data Model Element]* Raw data element is specified using a GRL KPI model element marked by ST_BPP_RAWDATA. The KPI evaluation value is used to capture the raw data value and the other KPI values are ignored during the evaluation process. The BPP profile uses this visual representation to render raw data elements.

In addition to the above, the BPP profile uses a stereotype (ST_BPP_MODELTYPE) to distinguish among the four *model types* defined for the framework metamodel in section 4.2, namely goal, process, decision, and performance models. Finally, the concept of *dimension* defined in [115] and implemented in jUCMNav is also part of the profile.

Appendix B summarizes the stereotypes and metadata defined in this BPP profile, including those to be defined in the next chapter.

5.7. Chapter Summary

Business process improvement has been important for businesses for many years. Yet, the frameworks and approaches that currently exist are labor intensive, often remain at the level of guidelines, and fail to provide further assistance to analysts. Among other reasons, the lack of a process modeling language, referred to as *language X* in this chapter, which provides the required capabilities for such a framework, plays a key role in this deficiency.

In section 5.1, the minimum basic requirements for language X that can be used in a framework taking advantage of BPI patterns are discussed. We argue that, to the best of our knowledge, AoURN is the only language that addresses the minimum requirements among a set of well-known behavioral and intentional modeling languages. Hence, we use AoURN to demonstrate the specification and matching of business process improvement patterns, while still collecting requirements for language X independently of AoURN.

An examination of all improvement patterns illustrated by a representative set of example patterns as well as the preliminary requirements discussed in section 2.4.1 (i) shows clearly that currently popular business process and requirement modeling languages (including AoURN in its current state) are not yet capable of capturing BPI patterns on a broad scale and (ii) results in a list of requirements for language X prioritized based on five criteria, including their impact on the BPI patterns. These requirements need to be fulfilled to support the specification and matching of business process improvement patterns. In addition, we propose solutions for how some of these requirements can be met with AoURN through a profile called BPP in section 5.6. These requirements are not only applicable to the BPI space but also can be used as a guideline to improve modeling languages in general to better capture behavioral and intentional requirements of a system.

One considerable risk with such a comprehensive matching mechanism for language X is always performance and scalability in large models, which need to be investigated. Moreover, the formulation of requirements related to the capturing of instance level information, which would enable us to model even more improvement patterns, is another potential avenue for future work. Finally, while the usability and learning curve of aspect-oriented approaches is already a concern [118], this approach adds even more model elements and syntax that need to be understood by the user. Therefore, hiding the details of the languages under a higher level of abstraction and an intuitive user interface is another topic that needs to be discussed. Furthermore, while we endeavored to not be influenced by AoURN, language X requirements should be revisited in the context of at least one more modeling language to ensure they are truly independent from AoURN.

In conclusion, although currently there are no languages that address all the requirements discussed in this thesis for BPI frameworks, by using some of the enhancements suggested, e.g., the new BPP profile, AoURN is currently in the best position to be extended and become language X. Using AoURN or languages X, which address the aforementioned requirements, empowers analysts to find repeated patterns in a given context and to bundle the integration between goal, performance and process models as a self-contained unit. This enables the modeling of complex circumstances involving goal, process, and performance views together, using a unified modeling language.

In the next chapter, we expand on the definition of formula-based relationships between the KPIs used in the performance models. This approach enables analysts to better define the business context and the relationship between the KPIs that are used in both the base model and in the patterns.

Chapter 6. Enhanced Performance View

In this chapter, the algorithm and the supporting tool for monitoring of the organizations' Key Performance Indicators, together with their impact on each other and on organization goals, are studied. Using the approach given in this chapter, organizations can define the relationships between the standard KPIs as well as the new raw data values used to calculate the value of the organization's KPIs. Section 6.1 discusses industry standard KPIs and their formulas, and illustrates the value and motivation for the content covered in the rest of this chapter. Section 6.2 highlights the benefits of an enhanced performance view. Section 6.3 assesses the impact of *situations* on performance evaluations, while section 6.4 formalizes the new evaluation algorithm. Section 6.5 compares this new algorithm with the existing GRL evaluation algorithms.

6.1. Standard Key Performance Indicators

Nowadays, KPIs are so pervasive that standards enabling the computation of KPIs have started to emerge. The World Intellectual Capital Initiative (WICI), the world's business reporting network, is a private/public collaboration with the goal of improving business reporting. They believe in standard KPIs as a mean to make corporate reporting consistent and understandable. WICI has proposed a framework and a set of industry standard KPIs to achieve this purpose [149]. Gartner and the Enhanced Business Reporting Consortium (EBRC), in a recent effort [48], have extended the WICI framework, identified a set of KPIs, and proposed them as an industry standard enabling businesses to measure their performance. As part of this effort, the formulas for calculation of the KPIs are also defined. Table 15 shows a selected subset of these KPIs, together with their definition and formulas, as examples. The complete list of KPIs and their definition is available online as Extensible Business Reporting Language (XBRL) taxonomies [49].

The standard focuses on actionable activities in organizations. The activities are organized in three groups called business domains. The aggregates can be considered as measurable sub-categories whose quantitative values can be calculated by multiplying the normalized value of a selected set of KPIs in the group, using a [0..100] scale.

Such standardized KPIs and their formulas, the aggregates' values, and business domain goals, can be incorporated into our framework. The enhanced performance view and the formula-based evaluation algorithm that are discussed in this chapter allow one to use KPIs and raw data values in goal/performance models of the organization and to define their relationships using associated formulas (or any formula that the organization can mine from historical data or that it wants to explore).

Table 15: Standard industry KPI examples (Gartner and EBRC initiative) [48]

Domain / Aggregates	**KPI**	**Definition**	**Formula**
Demand Management / Product Development Effectiveness	New Product Index	Illustrates the business desire for changing its products and services to address the new market demands	= (Revenue of products & services released in last 12 months) / (Total company revenue)
	Feature Function Index	Shows the degree of the changes in the new products and services	= (New component items for products released last year) / (Total component items for those products)
	Time-to-Market Index	The ability to release new products and services on a timely basis	= Average (time from approval to launch for each product)
	R&D Success Index	The ability of the product development function to bring product to market	= (New products launched in last 12 months) / (Development projects due to complete in last 12 months)
Supply Management / Customer Responsiveness	On-Time Delivery	The ability to satisfy an order on-time	= (Order delivered on time) / (Total orders received)
	Order fill Rate	The ability to meet the quantity of the ordered items	= (Total number of orders filled correctly) / (Total number of orders)
	Material Quality	Indicates whether the materials were either damaged or defective on receipt	= (Orders with material quality within agreed tolerance) / (Total orders)

6.2. Enhanced Performance View

Although several "standard" GRL evaluation algorithms (qualitative, quantitative, hybrid, and evaluation-based) already exist, as described by [7][119], none of them provides a method for precisely defining the cause-effect relationship between the KPIs in the performance view of the organization or uses a formula to calculate the value of KPIs using raw data values or other KPIs as inputs to the formula. As discussed in section 6.1, industry standard KPIs are often defined using standard formulas. Yet, from a flexibility perspective, it is important to bring such formulas in performance/goal models in order to minimize the need to have them computed outside of the modeling environment. Therefore, the lack of formula-based evaluation and analysis is a major gap in the current set of algorithms.

The current algorithms allow modelers to specify the contribution level of a KPI on another GRL intentional element and to calculate the satisfaction level of that target element [113]. However, these algorithms prevent one KPI from driving the computation of the *current value* of another KPI. Although the current evaluation methods allow computing the impact of one KPI on another KPI in terms of *satisfaction levels*, when it comes to showing the impact of several KPIs on one KPI (e.g., their *aggregate* effect), the current evaluation methods quickly become a bottleneck and thus obstruct the cause-effect analysis. Furthermore, the current algorithms have no means to consider the *situation* impacts in calculation of the evaluation values. These three features (KPI aggregation, KPI calculation using raw data values, and situation impact) are essential as they allow modelers and managers to modify the way KPIs impact each other inside the goal model. This may happen frequently, especially when little data or knowledge about the enterprise is available or when the organization is small and does not want to have a large infrastructure in place. The alternative is to rely on heavy computing at the BI tool level and hence involves complex modifications that often require implementation by IT staff.

Other modeling languages and enterprise modeling frameworks exist that can be used to model KPIs. However, many have limited computational power and do not allow one to define proper relationships between KPIs for advanced analysis [112]. In addition, there have been recent efforts in industry to use strategy maps and measurable objectives to help with decision-making and process improvement [134]. However, influence of KPIs on one another has not been discussed. More recently, and inspired by our work on GRL and KPIs, the Business Intelligence Model (BIM) proposed by Jiang *et al.* [65] also suggests KPI aggregation and support for situations in goal models. However, BIM's formal semantics is mainly based on Description Logic [58] and, although the existence of tool support is suggested in [57] and prototyped in [33], semantics and tools are currently unable to consider the impact of situations in models in a *quantitative* way.

6.3. Formula-Based Algorithm

In order to address aforementioned shortcomings of the current evaluation algorithms, we introduce further extensions to GRL and a novel evaluation algorithm that allow analysts and decision makers to define precise mathematical formulas describing relationships between the model elements. This method extends the bottom-up quantitative propagation algorithm defined in [7]. Modelers and analysts gain full control of the model and can change the impact of one element on another as desired.

The algorithm uses current/evaluation values of the source KPIs as inputs for the formula (described as metadata, see Figure 31) and calculates the target KPI evaluation value using these inputs. Then, the satisfaction level of the KPI is calculated using the KPI's target/threshold/worst values as discussed previously. The impact of KPIs on other types of intentional elements (e.g., goals, softgoals and tasks) is computed using conventional GRL quantitative and qualitative algorithms. This unique combination allows one to have both quantifiable KPIs and strategic-level softgoals that are hard to quantify together in the same model and to show and monitor the impact of KPIs on the goals of the organization.

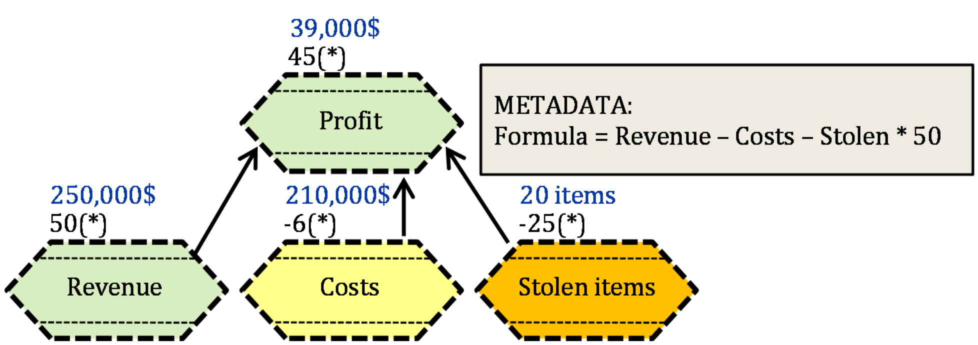


Figure 31: Formula-based KPI evaluation

Figure 31 shows a simple example where the current KPI values are displayed, with their units, above the usual satisfaction values. Note that the inputs can be of different units; the formula in the target KPI must take this into consideration. In this example, the current value of *Profit* is computed as *Revenue – Costs – Stolen*50* (the first two are in dollars and the third is a number of items). Note also that the contributions have no weight; the satisfaction of the *Profit* KPI is based on the normalization of its computed current value ($39,000) against its specified target, threshold and worse values. We have implemented this new algorithm in the jUCMNav tool.

In cases where no linear formula exists for a KPI, several approaches can be taken:

- The value of the KPI can be initialized using external BI tools.
- More sophisticated mathematical formulas from the current library can be used to calculate an approximate value.
- Experimental *what-if* formulas can be used from within jUCMNav and validated in future iterations using historical data.
- Finally, other approaches like regression analysis, data mining, and clustering can be used on historical data to infer the formulas or approximations.

6.3.1 Situations

Another benefit of this formula-based approach is the ability to account for *situations*. In organizations, cause-effect analysis and decision making usually involve an element of threat or opportunity that we both capture as situations influencing the business. Even though we can easily show situations as model elements in GRL diagrams (e.g., using softgoals stereotyped with «Situation»), it is rather hard to quantify the impact of situations on the value of a KPI and consider it in evaluation algorithms.

We propose a new model element in GRL called *situation* (a stereotyped softgoal in our profile) with an evaluation value between 0 and 100. This element allows us to easily model and evaluate threats and opportunities in the models. The situation element can be connected to KPI nodes using either positive or negative contributions. Table 16 discusses one potential example of the expected impact on a target KPI. The method is not limited to this example (where linear interpolation is used) and one can define any formula that is suitable for their situation.

The target KPI *threshold value* changes based on the evaluation level and level of contribution of the situation factor on the target element as indicated in Table 17. This enables the modeler to vary the acceptable range of values for a KPI when there is an expected situation involved.

Table 16: An example heuristic for situation impact on a target KPI

Contribution Level	**Description**
Positive	A situation with a positive contribution (i.e., an *opportunity*) positively impacts the expected result from the target KPI. Therefore it will move the impacted *threshold* closer to the *target* value. In other words, such opportunity reduces the range of acceptable values or increases the expectation about the outcome for a KPI towards the target. Hence, the KPI's current value in this new context will lead to a lower satisfaction value if it is not improved.
Negative	A situation with a negative contribution (i.e., a *threat*) negatively impacts the expected result from the target KPI. Therefore it will move the impacted *threshold* closer to the *worst* value. In other words, such situation increases the range of acceptable values for a KPI towards the worst value (i.e., it reduces the expectation about the outcome of the KPI). Hence, the KPI's current value in this new context will lead to a higher satisfaction value.

Table 17: Situation formula based on the example situation heuristic

Conditions	**Target KPI's New Threshold Value**
• Contribution positive • Target KPI values o Target > Threshold	$= Current\ threshold\ value + \frac{\lvert \ll Situation \gg Evaluation\ value\rvert \times \ll Situation \gg Contribution\ value}{10000} \times (KPI\ Target\ value - KPI\ current\ threshold\ value)$
• Contribution negative • Target KPI values o Target > Threshold	$= Current\ threshold\ value + \frac{\lvert \ll Situation \gg Evaluation\ value\rvert \times \ll Situation \gg Contribution\ value}{10000} \times (KPI\ Worst\ value - KPI\ current\ threshold\ value)$
• Contribution positive • Target KPI values o Target < Threshold	$= Current\ threshold\ value - \frac{\lvert \ll Situation \gg Evaluation\ value\rvert \times \ll Situation \gg Contribution\ value}{10000} \times (KPI\ Target\ value - KPI\ current\ threshold\ value)$
• Contribution negative • Target KPI values o Target < Threshold	$= Current\ threshold\ value - \frac{\lvert \ll Situation \gg Evaluation\ value\rvert \times \ll situation \gg Contribution\ value}{10000} \times (KPI\ Worst\ value - KPI\ current\ threshold\ value)$

Figure 32 uses the same formula-based algorithm described in Figure 31, only this time the model also contains a new situation element, with a negative contribution (–75).

Therefore the threshold value of the profit KPI has changed and increased the acceptable range for the evaluation value of the KPI. In this example, the risk has somehow been mitigated (satisfaction level of 20, which means weakly denied). Therefore, even though the evaluation value of the KPI (i.e., 39,000$) has not changed from Figure 31, the evaluation level of the KPI has slightly improved and went up from 45 to 52.

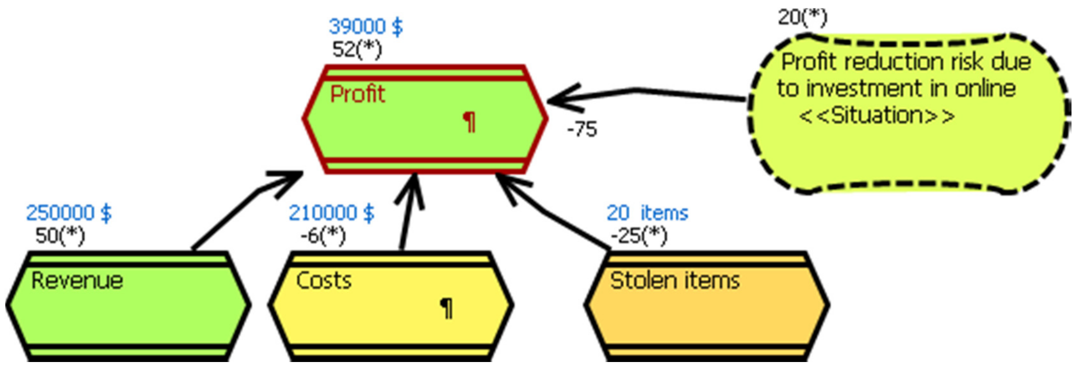


Figure 32: Situation impact on KPI evaluation

6.4. Formalized Algorithm

In this section, we discuss the formal definition of the formula-base algorithm. This algorithm is based on the evaluation algorithm described in Appendix II of the URN standard [62]. To set the context, the core part of the standard algorithm is repeated here (Figure 33 and Figure 34) before the extension points and modifications are illustrated in later figures.

The algorithm is a bottom-up forward propagation algorithm that uses the initial values set in a GRL strategy to calculate the satisfaction level of the rest of the GRL elements in the graph. Figure 33 shows the forward propagation algorithm from the standard. In this algorithm, *totalSourceLink* is total number of incoming source links to a node and *linkReady* is used to track the number of links that have been used in the propagation so far [62].

The forward propagation algorithm calls the *CalculateEvaluation* algorithm (found in Figure 34) to calculate the evaluation value of each model element. The calculation method varies depending on the type of the link that is connected to the model elements and for each type of link, a different algorithm is invoked. Prior to looking at the links, the standard algorithm calls *PreGetEvaluation* to use the initialized values in the

GRL strategies if any. If the node is already initialized by the strategy that value is used and the evaluation result is returned immediately.

```
Algorithm ForwardPropagation
Inputs GRLmodel:GRLspec, currentStrategy:EvaluationStrategy
Output newEvaluations:HashMap

elementsReady:List = ∅              // containable elements that can be evaluated
elementsWaiting:List = ∅            // containable elements that cannot yet be evaluated
newEvaluations = ∅

for each element:GRLContainableElement in GRLmodel.intElements
{
        element.linkReady = 0
        if (element in currentStrategy.evaluations.intElement) // is the element initialized?
                elementsReady.add(element)
        else
                elementsWaiting.add(element)
}

while (elementsReady.size() > 0)
{
        element = elementsReady.get()
        elementsReady.remove(element)
        newEvaluations.add(element, CalculateEvaluation(element, currentStrategy))
        for each link:ElementLink in element.linksSrc
        {
                destination = link.dest
                destination.linkReady = destination.linkReady + 1
                if (destination.linkReady == destination.totalSourceLink)
                {
                        // all source elements have known evaluation values
                        elementsWaiting.remove(destination)
                        elementsReady.add(destination)
                }
        }
}

return newEvaluations
```

Figure 33: Forward propagation algorithm, standard version [62]

```
Algorithm CalculateEvaluation
Inputs element:GRLContainableElement, currentStrategy:EvaluationStrategy
Output result: Integer

decompValue:EvaluationValue          // intermediate result
contribValue:EvaluationValue         // intermediate result
result: Integer
eval: Evaluation = evaluations.get(element)

evalMathEvaluator:MathEvaluator              // new, for KPI aggregation
situationMathEvaluator:MathEvaluator         // new, for situation computation

quickReturn: Integer = PreGetEvaluation(element, eval, evalMathEvaluator,
                                                        situationMathEvaluator)

if (quickReturn != null)
        return quickReturn.intValue();
if not(element in currentStrategy.evaluations.intElement) // is the element not initialized?
{
        // calculate based on decompositions, contributions, and dependencies
        decompValue = CalculateDecompositions(element)
        contribValue = CalculateContributions(element, decompValue)
        result = CalculateDependencies(element, contribValue)
}
result = PostGetEvaluation(element, eval, result, evalMathEvaluator,
                                                        situationMathEvaluator)

return result
```

Figure 34: Calculate evaluation algorithm – Formula-based algorithm

In the formula-based algorithm, *PreGetEvaluation* is extended (see Figure 35) to get the formulas definition from the element metadata and to define the math evaluators. Each element can have both evaluation formula and situation formula defined as metadata, i.e., *FormulaBasedGRLStrategyAlgorithm_evalFormula* and *FormulaBasedGRLStrategyAlgorithm_situationFormula* respectively. The math evaluator for each formula is implemented using a math library that gets a string math formula as well as name-value pairs as variables used in the formula as input, and returns the result of the formula as output. The implementation in jUCMNav that was done as part of this research uses the math library that was developed by Lai [82].

After the formula is defined, in the CalculateContributions algorithm (not shown here), *ComputeContributionResult* (Figure 36) is called, which is overridden in the formula-based algorithm to set the name-value pair variables of the formula using the KPI value sets of the source nodes. Therefore, KPI value sets (i.e., evaluation, target, threshold, and worst values) of each source node connected to the KPI that is being evaluated can be used in the formula. This allows for writing formulas such as those discussed in Table 15.

```
Algorithm PreGetEvaluation
Inputs element:ItentionalElement, eval:Evaluation
Modified: evalMathEvaluator:MathEvaluator, situationMathEvaluator:MathEvaluator
Outputs result:Integer

// get the evaluation formula from the metadata defined on the GRL intentional ele-
ment
evalFormula:String = MetadaHelper.getMetaData(element, Messages.getString
                        ("FormulaBasedGRLStrategyAlgorithm_evalFormula"))
// get the situation formula from the metadata defined on the GRL intentional element
situationFormula:String = MetadaHelper.getMetaData(element, Messages.getString
                        ("FormulaBasedGRLStrategyAlgorithm_situationFormula"))

if (evalFormula == null)
        evalFormula = "" // no evaluation formula defined
else
        evalMathEvaluator = new MathEvaluator(evalFormula)

if (situationFormula == null)
        situationFormula = ""// no situation formula defined
else
        situationMathEvaluator = new MathEvaluator(situationFormula)

// from the standard algorithm
result = QuantitativeGRLStrategyAlgorithm.preGetEvaluation (element, eval)

return result
```

Figure 35: PreGetEvaluation – formula-based algorithm

```
Algorithm ComputeContributionResult
Input link:ElementLink, contrib:Contribution,
Modified evalMathEvaluator:MathEvaluator
Output result: Integer
-----------------------------------------------------------------
if (evalMathEvaluator == null)
        result = QuantitativeGRLStrategyAlgorithm.ComputeContributionResult
                                                          (link, contrib)
else
{
        set: KPIEvalValueSet   = strategyManager.getActiveKPIEvalValueSet
                                                          (link.getSrc())
        evalMathEvaluator = InitFormula (set, evalMathEvaluator)
        result = 0
}

return result
```

Figure 36: ComputeContributionResult – formula-based algorithm extension

Furthermore, *ComputeSituationResult*, Figure 37, is called and initializes the situation-MathEvaluator using the elements that are marked with the Situation stereotype as input. This approach also allows using any formula, including the default ones discussed in Table 17, to define a situation that impacts a KPI.

```
Algorithm ComputeSituationResult
Inputs link:ElementLink
Modified situationMathEvaluator: MathEvaluator
-----------------------------------------------------------------
if (situationMathEvaluator != null)
{
        if ( IsSituation(link.getSrc()) )
        {
            set: KPIEvalValueSet = strategyManager.getActiveKPIEvalValueSet
                                                          (link.getSrc())
            InitFormula (set, situationMathEvaluator)
        }
}

return
```

Figure 37: ComputeSituation – formula-based algorithm extension

```
Algorithm InitFormula
Inputs set:KPIEvalValueSet
Modified mathEvaluator: MathEvaluator

if (set != null)
{
        mathEvaluator.addVariable(contrib.getName() + "_evaluation",
                                                set.getEvaluationValue())
        mathEvaluator.addVariable(contrib.getName() + "_target", set.getTargetValue())
        mathEvaluator.addVariable(contrib.getName() + "_threshold",
                                                set.getThresholdValue())
        mathEvaluator.addVariable(contrib.getName() + "_worst", set.getWorstValue())
}
return
```

Figure 38: InitFormula – formula-based algorithm extension

```
Algorithm IsSituation
Inputs element:IntentionalElement
Output srcIsSituation:Boolean

srcIsSituation = false
for each elementMetadata:Metadata in element.metadata
{
   if (elementMetadata.name == "ST_SITUATION")
   {
       srcIsSituation = true
       break
   }
}
return (srcIsSituation)
```

Figure 39: IsSituation – formula-based algorithm extension

The other extension point that is provided by the standard algorithm is *PostGetEvaluation*. The standard algorithm calls this method after the evaluation is performed but does not implement anything and just returns the values. This is the extension point (Figure 40) that is used to calculate the result of the formula and finally the satisfaction level of the KPI by calling *CalculateIndicatorEvalLevel*. CalculateIndicatorEvalLevel maps the KPI value sets to the GRL standard (-100 to 100) satisfaction level. Prior to calling CalculateIndicatorEvalLevel, the evalMathEvaluator and situationMathEvaluator, if defined, are used to calculate the evaluation value and threshold value of the KPI, respectively.

```
Algorithm PostGetEvaluation
Inputs element:ItentionalElement, eval:Evaluation, result:Integer,
        evalMathEvaluator:MathEvaluator, situationMathEvaluator:MathEvaluator
Output result:Integer

EvaluationStrategyManager  strategyManager =
                                        EvaluationStrategymanager.getInstance()
set:KPIEvalValueSet  = strategyManager.getActiveKPIEvalValueSet (link.getSrc())

if (situationMathEvaluator !=null)
{
        // change the Threshold Value if there is a situation formula
        calculatedThresholdValue: double = situationMathEvaluator.getValue()

        if (thresholdCalculatedValue != 0)
        {
                // set the calculated threshold value
                set.setThresholdValue(thresholdCalculatedValue)
        } else {
                set.setThresholdValue(0.0)
        }
}
if (evalMathEvaluator != null)
{
        // change the Evaluation Value if there is a situation formula
        calculatedEvalValue: double = evalMathEvaluator.getValue()

        if (calculatedEvalValue != 0) {
                // set the evaluation value
                set.setEvaluationValue(calculatedEvalValue)
        } else {
                set.setEvaluationValue(0.0)
        }

        // after the value sets are set, calculate the KPI satisfaction level by mapping it
        // to a -100 to 100 scale.
        strategyManager.calculateIndicatorEvalLevel(eval)

        result = eval.getEvaluation()
}
return result
```

Figure 40: PostGetEvaluation – formula-based algorithm extension

6.5. Comparison with Current Algorithms

Table 18 compares the existing GRL evaluation algorithms with the new algorithm defined in this thesis (last column).

Table 18: Comparison between GRL evaluation algorithms

Criteria	**Qualitative**	**Quantitative**	**Hybrid**	**Old formula-based**	**New formula-based**
Evaluation type	Qualitative	Quantitative	Hybrid	Quantitative	Quantitative
KPI evaluation	No	Yes	Yes	Yes	Yes
KPI aggregation	No	No	No	No	Yes
Situation impact analysis	No	No	No	No	Yes

Although this new algorithm addresses issues related to KPI aggregation and situation consideration, there is room for improvement and this will be addressed as future work. The main disadvantage of the algorithm is that the formula is part of the model for all strategies. This makes the model maintenance hard when different versions of the model with different formulas are required for what-if scenario analysis. This enhanced performance view can be significantly more powerful if the formulas could be defined as part of or in combination with the GRL strategies themselves.

6.6. Chapter Summary

In this chapter, the enhanced performance view was discussed, which mainly consist of a new evaluation algorithm and modeling approach that relies on cause-and-effect relationships between the performance model KPIs. A relationship is defined using a formula and a new algorithm was defined to allow evaluation of these formula-based models. Furthermore, the value and necessity of the formula-based approach was discussed and industry standard examples for KPI formulas were pointed out in Table 15. Finally, the algorithm was compared with the existing GRL evaluation algorithms and potential improvements were pointed out.

The next two chapters provide case studies exploiting the BPP profile and the new formula-based evaluation algorithm, first in the healthcare domain and then in the retail business domain.

Chapter 7. Healthcare Example

In this chapter, we illustrate the application of the framework using an example based on the real *Data Warehouse Approval Process* (DWAP) of a health care provider in Ontario, Canada. This DWAP assesses requests for access to the health information in the data warehouse based on patient privacy concerns, ethical concerns, as well as technical feasibility and impact. This example focuses on the use of AoURN-based detection/improvement patterns.

7.1. Modeling the Current State of the Business

We modeled the DWAP after interviews with the stakeholders in the hospital who were in charge of executing the process. The modeling helped the process owners to better understand the steps they have in the process as well as the goals of the process and KPIs required for monitoring its efficiency.

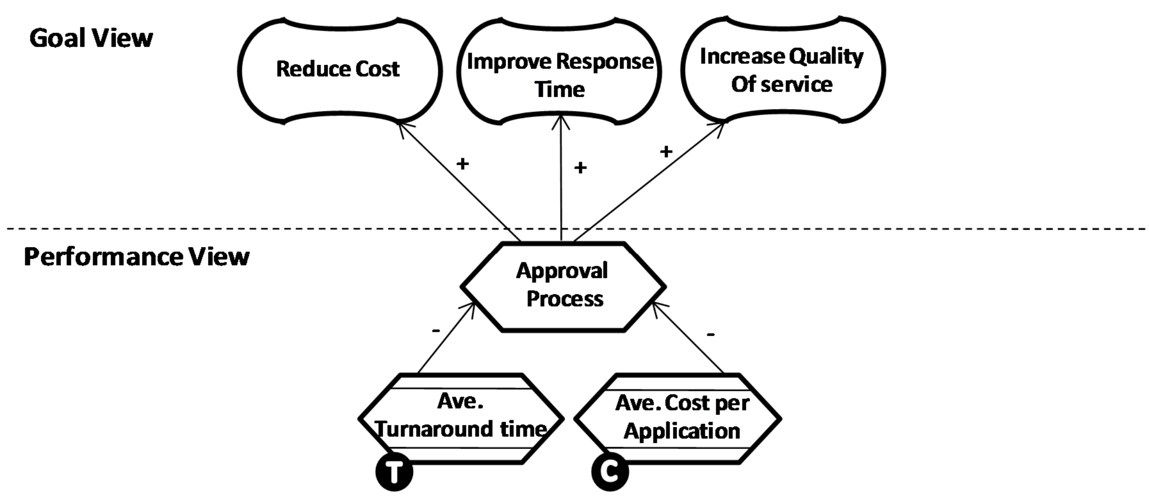


Figure 41: DWAP – goal and performance views – first step

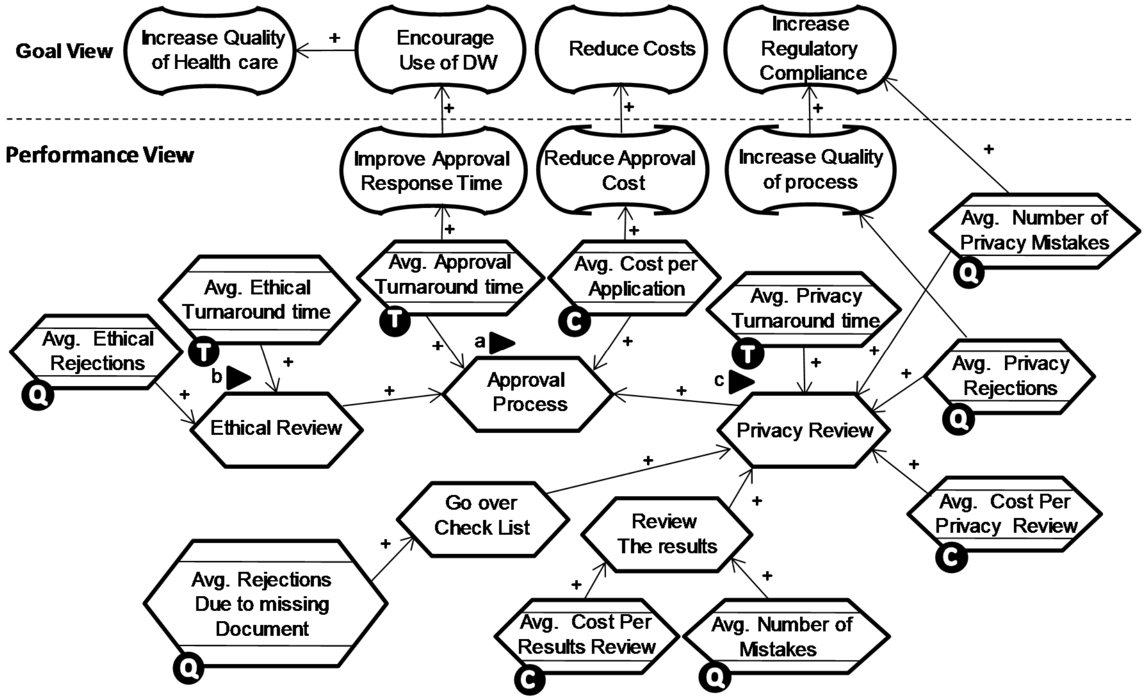


Figure 42: DWAP – goal and performance views – second step

As illustrated in Figure 42, the main performance goals of the process are to improve the approval response time, reduce the cost of approval, and increase the quality of the process, which in this case helps increasing the regulatory compliance business goal. The goal view shows the high-level business goals linked to this process while the performance view shows the specific goals of the process, which mainly tend to focus on efficiency of the process with respect to the ultimate business goals that the process is supposed to satisfy.

In the first step of our analysis lifecycle (Table 9), when the goal view and performance view are modeled, process owners have often a hard time defining what needs to be measured. However having business goals and performance views of the process side by side, as in Figure 41, helps process owners to better understand what aspects of the process are important and will help with achieving the business goals.

After some exploration, in the second step of the analysis lifecycle, new KPIs required to measure the performance of the process are added to the model (leading to Figure 42). In many cases the data required to measure performance related KPIs of the process is not available or just gathered informally using spreadsheets or other personal tools. From what we experienced, the modeling process and the KPI definitions help with specifying the required data and with defining a method for systematically collecting them.

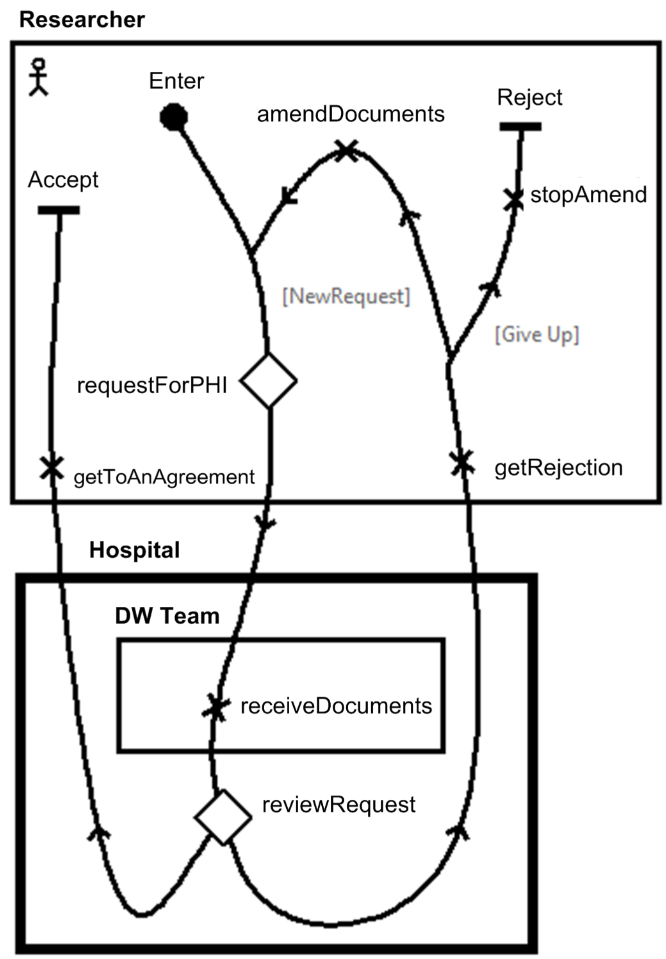


Figure 43: DWAP – process view

Figure 43, shows the approval process at the highest level of abstraction. Researchers who wish to use the health information for their research must submit access requests. The Privacy Review Board, the Ethics Board, and Data Warehouse Administrator examine the requests, as illustrated in review request sub process in Figure 44.

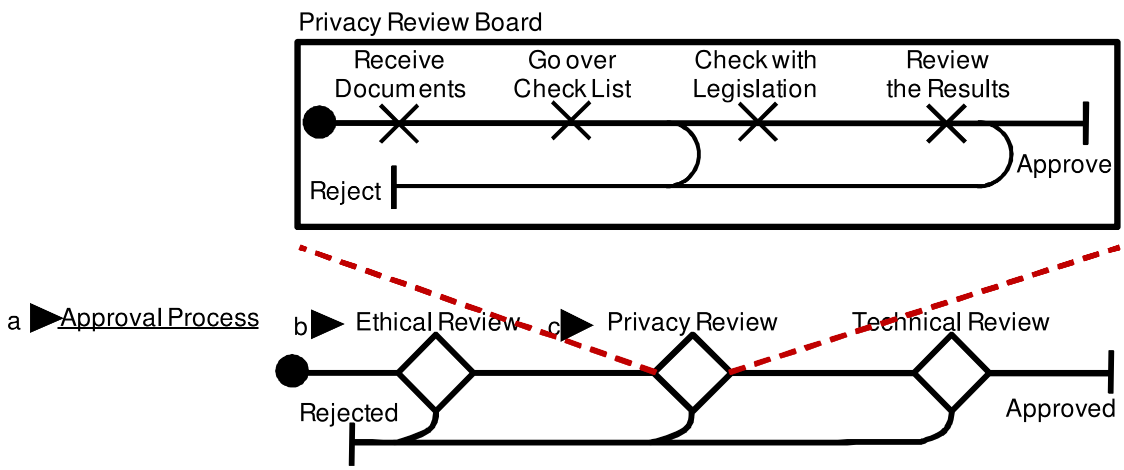


Figure 44: DWAP, review request sub-process – process view

7.2. Exploring, Discovering, and Adapting the Business

Figure 45 shows the evaluated performance model in the current, as-is state. The evaluation was done using the GRL quantitative evaluation algorithm and by initializing a GRL strategies using the values in Table 19. As illustrated in this model, some of the high-level goals (e.g., increase quality of health care and reduce costs) are not satisfied.

Table 19: KPI evaluations before applying the three patterns

KPI	EV	TV	ThV	WV
Avg. Approval Turnaround Time	20	10	19	30
Avg. Cost per Application	200	100	150	250
Avg. Ethical Rejection	10	11	0	0
Avg. Ethical Turnaround Time	10	8	12	15
Avg. Privacy Rejection	20	23	8	0
Avg. Number of Privacy Mistakes	0.5	0	2	3
Avg. Privacy Turnaround Time	5	3	6	8
Avg. Cost per Privacy Review	1000	750	1250	1500
Avg. Number of Mistakes	2	2	3	5
Avg. Cost per Results Review	200	100	150	250
Avg. Rejections due to Missing Docs	10	1	5	15

EV: Evaluation Value, TV: Target Value, ThV: Threshold Value, WV: Worst Value

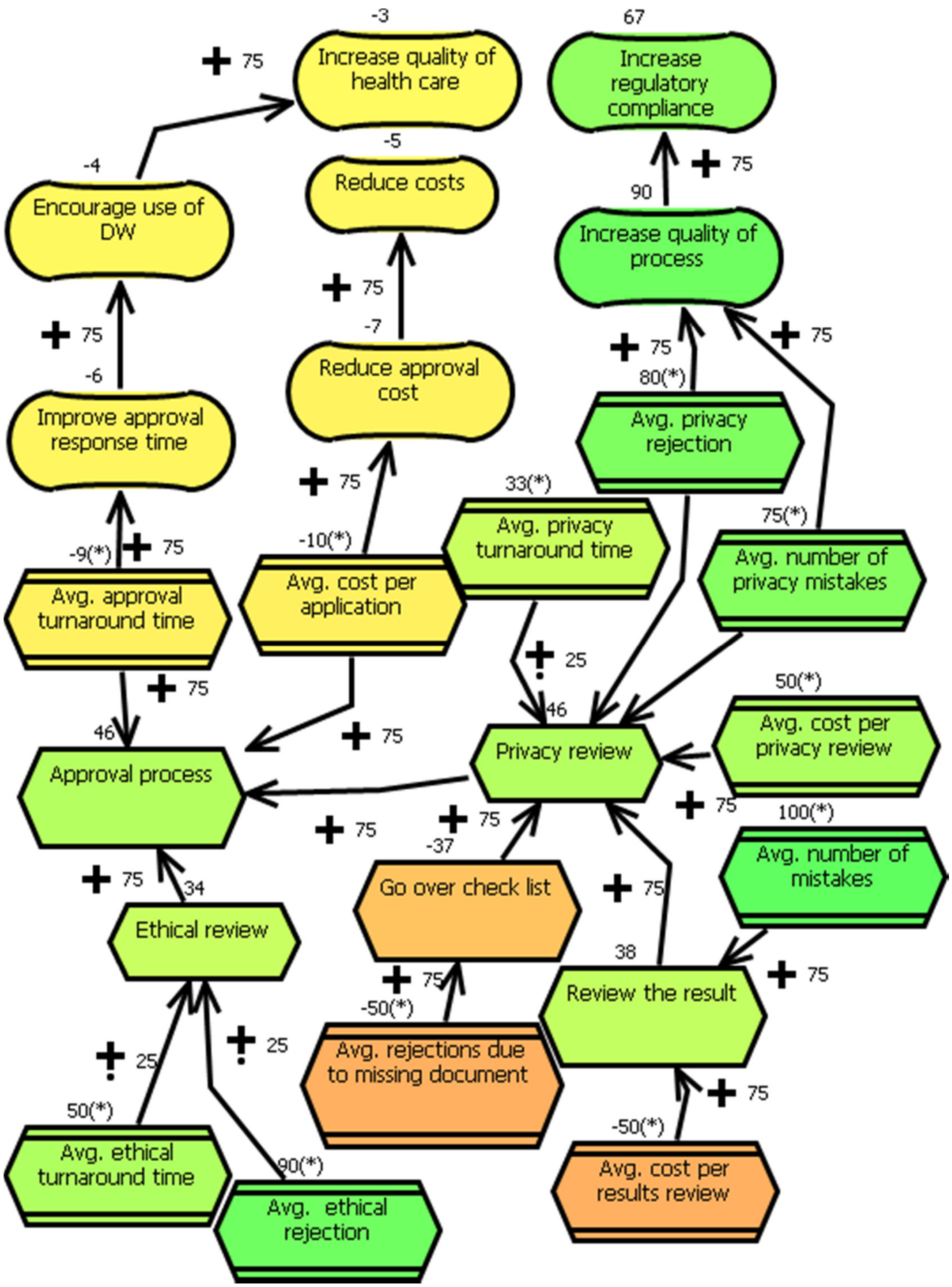


Figure 45: Evaluated goal and performance views before applying the patterns

Therefore, improvement to the processes is required to increase the satisfaction level of the goals. The initial review of the process model shows that some of the redesign patterns may be applicable to this process. The adaptation lifecycle, presented in Table 10, can be used in the next step to apply the patterns to the process as discussed in the rest of this chapter.

7.2.1 The Knockout Pattern

The Knockout redesign pattern, introduced earlier in section 5.3.1, reorders a sequence of tasks based on their failure rate and effort. It can therefore be applied to any approval process with multiple sequential approval steps including the DWAP. For instance, if the average number of rejections caused by the Privacy Review is higher than for the Ethical Review and the Ethical Review takes longer to complete than the Privacy Review, then the pattern is applicable to DWAP. In that case, the Privacy Review should be moved ahead of the Ethical Review to become the first task in the sequence. Figure 46 illustrates this desired change from the "As-Is" process to the "To-Be" process. The "To-Be" process results from applying the Knockout redesign pattern to the DWAP.

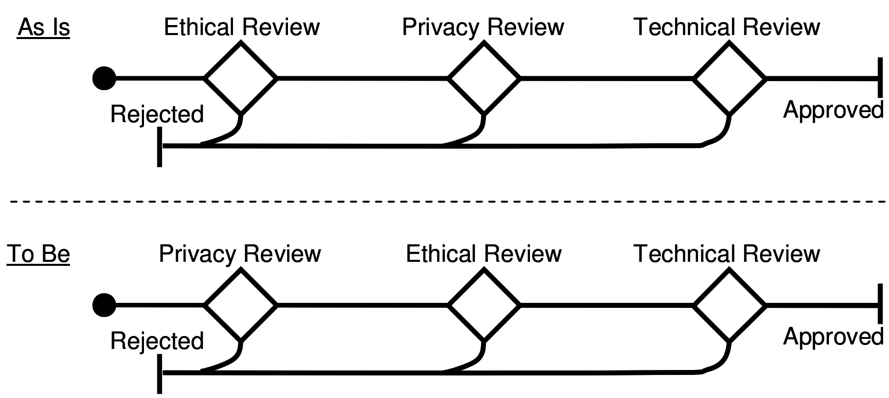


Figure 46: DWAP – As-Is vs. To-Be process request review sub-process

When the Knockout redesign pattern (Figure 23) is applied to the URN model of the DWAP, a match is found in the performance model. Approval Process, Ethical Review, and Privacy Review match against the three tasks. Avg. Approval Turnaround time, Avg. Ethical Turnaround time, Avg. Ethical Rejections, Avg. Privacy Turnaround time, Avg. Privacy Rejections, and Avg. Cost per Application match against the six KPIs. Furthermore, the map linked to Approval Process and the two stubs linked to Ethical Review and Privacy Review match the AoUCM pointcut expression – behavioral detection model. Note that the dashed portion of the pointcut expression matches against any sequence of

UCM modeling elements and therefore can be matched against the join after the Privacy Review stub in the as-is process, Figure 46.

The pattern also has an impact on the expected values for the KPIs and can be used to monitor the expected outcome of applying the pattern to the process. The changes could be either defined as part of the generic pattern if possible or defined based on the historical data available in the context that the pattern is being applied to. For example, assuming that 20% of the submitted requests that fail ethical review also fail the privacy review, then the matched quality KPI of the ethical review (KPI2) will decrease by 20% (R: *0.8) while the matched quality KPI of the privacy review (KPI4) will increase by the same number (R: +KPI2*0.2), if the order of the two reviews is reversed. Furthermore, there is also an impact on the average cost of the approval process and turnaround time, as more requests will now be rejected earlier in the process leading to a cost decrease of 2% (R: *0.98) and a reduced backlog that makes the process more efficient (R: * 0.96).

The pointcut expression for the Knockout redesign pattern is very generic and therefore uses only parameterized elements. This may lead to undesired, false matches. If this is the case, then the pointcut expressions can be tailored to the specific needs of the current condition. For example, the * for the task with the URN link number 3 could be replaced with Approval Process to narrow down the search space.

Since the pointcut expression is matched in the DWAP, the aspectual behavior is added to the process (see Figure 47). Therefore, aspect markers are added before and after the matched model elements as defined on the Knockout pattern map. AoURN uses slightly different aspect markers, i.e., *tunnel entrance and exit aspect markers, to indicate that the aspect replaces existing model elements.* Solid bars and triangles are added to standard aspect markers to denote tunnel entrances and exits. The aspect marker before the Ethical Review stub is a *tunnel entrance* (*triangle points away from the aspect marker*) as the behavior does not continue with the Ethical Review but with the aspectual behavior and only returns to the map at the *tunnel exits* (i.e., the two other aspect markers; *triangle points towards the aspect markers*). Note how the two stubs on the Knockout Pattern map have been replaced by the matched model elements from the pointcut expression. The resulting model in Figure 47 is semantically equivalent to the desired model in Figure 46.

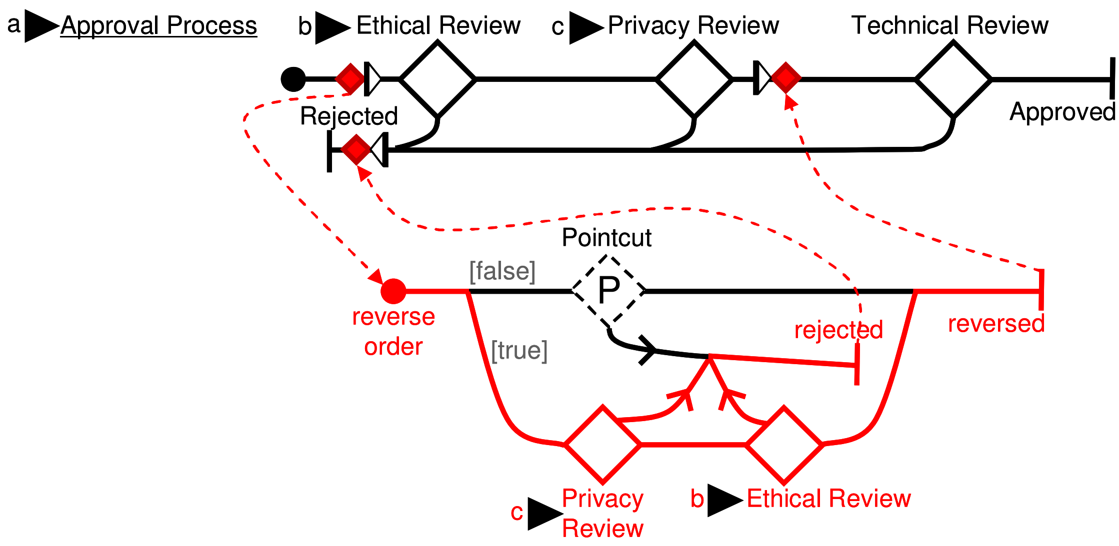


Figure 47: Applied Knockout pattern

7.2.2 The Task Elimination Pattern

The Task Elimination pattern, formalized in section 5.3.2, may be applied to the sub-process for the privacy review by the privacy review board. Figure 49 shows the "As-Is" process and the desired "To-Be" process after the application of the Task Elimination pattern. This pattern is often used to remove process steps without significant value [123]. In this case, we have removed the responsibility called "Review the Results" done by the Privacy Review Board since the average number of mistakes found by this review is low at 1% (see the evaluation value in the third column (EV K) of Table 20).

As can be noticed in Figure 48, the generic pattern has been tailored to be more specific to avoid false positives. In this case the KPIs "Avg. Number of Mistakes" and "Avg. Number of Pr*" are specific to the process being monitored if compared with * in the generic model of the pattern described in Figure 24.

Since a match can be found in the base model, the Task Elimination pattern can be applied and the responsibility Review the Results is removed with the help of tunnel entrance and tunnel exit aspect markers (see Figure 50). Figure 50 is semantically equivalent to the "To-Be" process in Figure 49, except for the relocation of the "Go over Check List" responsibility, to be explained in the next section.

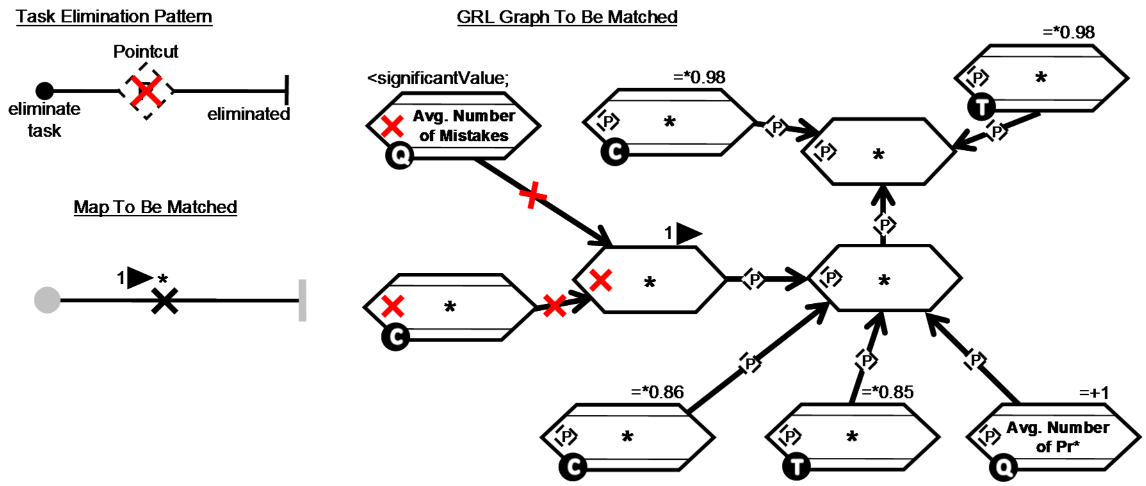


Figure 48: Task Elimination tailored to avoid false positives

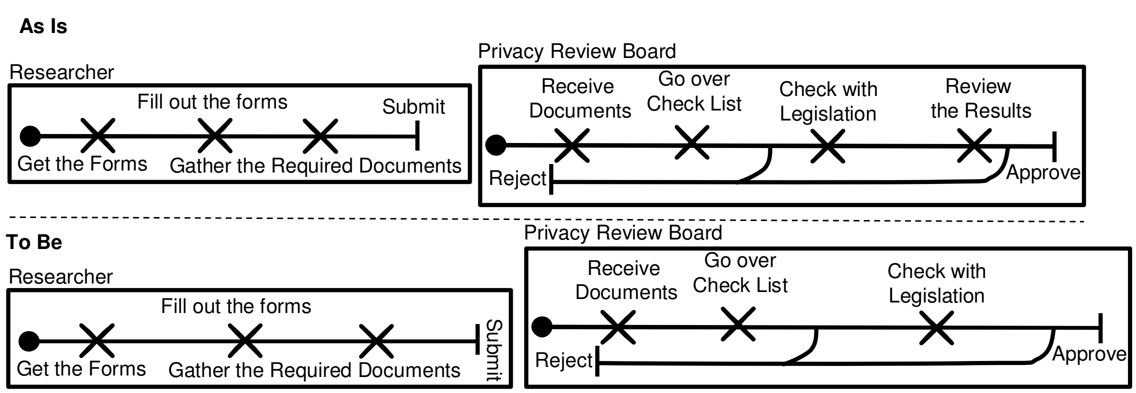


Figure 49: Further improvements to the DWAP – Task Elimination

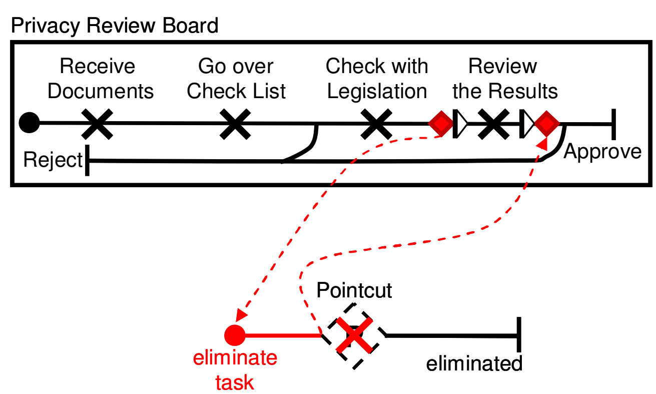


Figure 50: Applied Task Elimination pattern

7.2.3 The Control Relocation Pattern

The Control Relocation pattern introduced in section 5.3.3 (see Figure 51, duplicated from Figure 25 for convenience) may be applied to the sub-processes for submitting an application by the researcher and the privacy review by the privacy review board. This pattern is often used to move input validation checks to the client side [123]. In this process, we have moved the responsibility Go over Check List to the researcher's application submission process because too many applications (10%) were rejected due to missing documents (see the evaluation value in the fourth column (EV E) in Table 20), while this could easily be checked by the Researchers themselves.

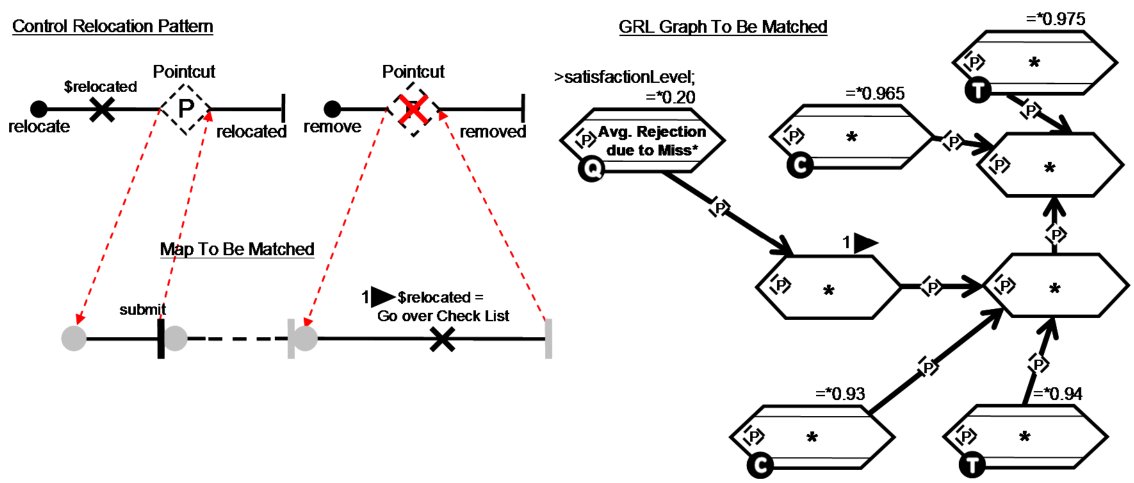


Figure 51: Generic aspectual model for the Control Relocation pattern

Since a match can be found in the base model, the Control Relocation pattern can be applied and the responsibility Go over Check List is removed with the help of tunnel entrance and tunnel exit aspect markers from its original location and then inserted with the help of a standard aspect marker at its new location (see Figure 52). Figure 52 is again semantically equivalent to the "To-Be" process in Figure 53.

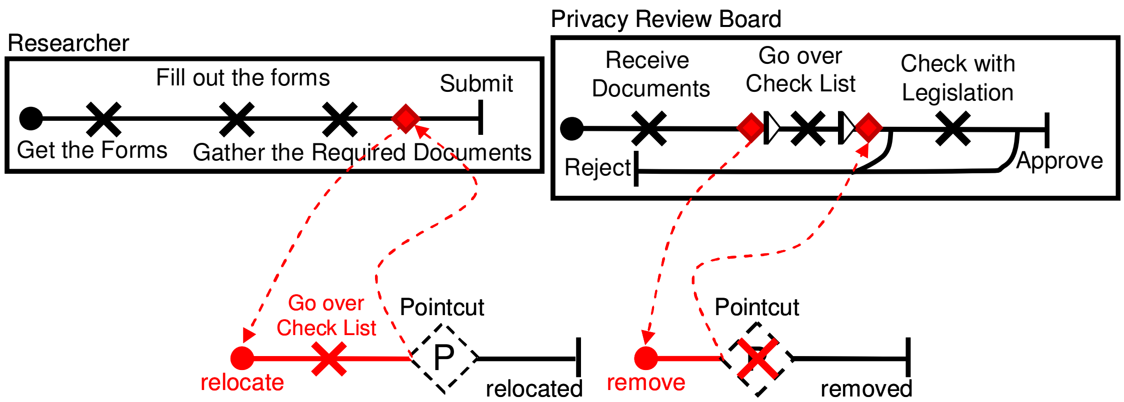


Figure 52: Applied Control Relocation pattern

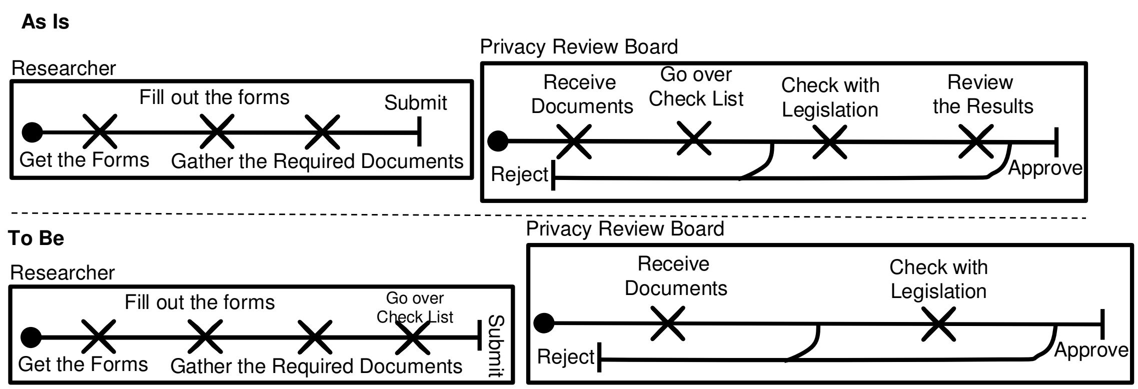


Figure 53: Further improvements to the DWAP – Control relocation

7.3. Monitoring, Results Evaluation, and Refinement

Table 20 shows the impact of all three patterns on the KPIs, business goals, and tasks as calculated by the GRL evaluation mechanism (the quantitative evaluation mechanism with a tolerance of 1 was used). Although the applied patterns have a positive impact on the Approval Process and Privacy Review performance, they have also a negative impact on the business goals Increase Regulatory Compliance and Increase Quality of Process. In most cases when redesign patterns are applied, positive and negative impacts occur simultaneously [123]. Our suggested approach equips business analysts with the ability to observe and explore these impacts, leading to more informed decisions about the process.

In addition, the evaluation of GRL models is no longer limited to bottom-up algorithms, and constraint-based approaches could be used to enable top-down, optimization-based evaluations. Luo has provided a prototype of such a mechanism in jUCMNav [88], but indicators are not yet supported.

Table 20: KPI evaluations before and after applying the three patterns

KPI	EV	EV K	EV E	EV CR	Pattern
Avg. Approval Turnaround Time	20 d (-9)	19.2 d (-1)	18.8 d (2)	18.3 d (7)	K – E – CR
Avg. Cost per Application	4100$ (-10)	4018$ (-1)	3938$ (6)	3800$ (20)	K – E – CR
Avg. Ethical Rejection	10% (90)	8% (72)	8% (72)	8% (72)	K
Avg. Ethical Turnaround Time	10 d (50)	10 d (50)	10 d (50)	10 d (50)	N/A
Avg. Privacy Rejection	20% (80)	22% (93)	22% (93)	22% (93)	K
Avg. Number of Privacy Mistakes	0.5% (75)	0.5% (75)	1.5% (25)	1.5% (25)	E
Avg. Privacy Turnaround Time	5 d (33)	5 d (33)	4.25 d (58)	4 d (66)	E - CR
Avg. Cost per Privacy Review	1000$ (50)	1000$ (50)	860$ (78)	800$ (90)	E - CR
Avg. Number of Mistakes	1% (100)	1% (100)	N/A	N/A	E
Avg. Cost per Results Review	200$ (-50)	200$ (-50)	N/A	N/A	E
Avg. Rejections due to Missing Docs	10% (-50)	10% (-50)	10% (-50)	2% (75)	CR

Satisfaction values in parentheses – EV: Evaluation value before applying redesign pattern
EV K: Evaluation value after applying Knockout (K) Pattern – EV E: Evaluation value after applying Elimination (E) Pattern– EV CR: Evaluation value after applying Control Relocation (CR) Pattern
Note: Pattern column indicates which pattern has caused changes in the KPI.

Table 21: Goal and Task satisfaction values before and after applying the three patterns

Goal/Task	SV	SV K	SV E	SV CR	Pattern	Impact
Increase Quality of Health Care	-3	0	0	2	K – E – CR	Positive
Encourage Use of DW	-4	0	0	3	K – E – CR	Positive
Improve Approval Response Time	-6	0	1	5	K – E – CR	Positive
Reduce Cost	-5	0	3	11	K – E – CR	Positive
Reduce Approval Cost	-7	0	4	15	K – E – CR	Positive
Increase Regulatory Compliance	74	74	65	65	E	Negative
Increase Quality of Process	99	99	87	87	E	Negative
Approval Process	30	43	47	90	K – E – CR	Positive
Ethical Review	12	12	12	30	N/A	N/A
Privacy Review	46	46	45	90	E – CR	Positive

SV: Satisfaction value before applying redesign pattern – SV K: Satisfaction value after applying Knockout (K) Pattern –SV E: Satisfaction value after applying Elimination (E) Pattern – SV CR: Satisfaction value after applying Control Relocation (CR) Pattern
K: Knockout – E: Elimination – CR: Control Relocation
Note: Pattern column indicates which pattern has caused changes in the KPI.

Figure 45 and Figure 54 respectively show the evaluated GRL graphs (computed by jUCMNav) for the DWAP that correspond to the second (EV / SV) and fifth (EV CR / SV CR) columns of Table 20 and Table 21.

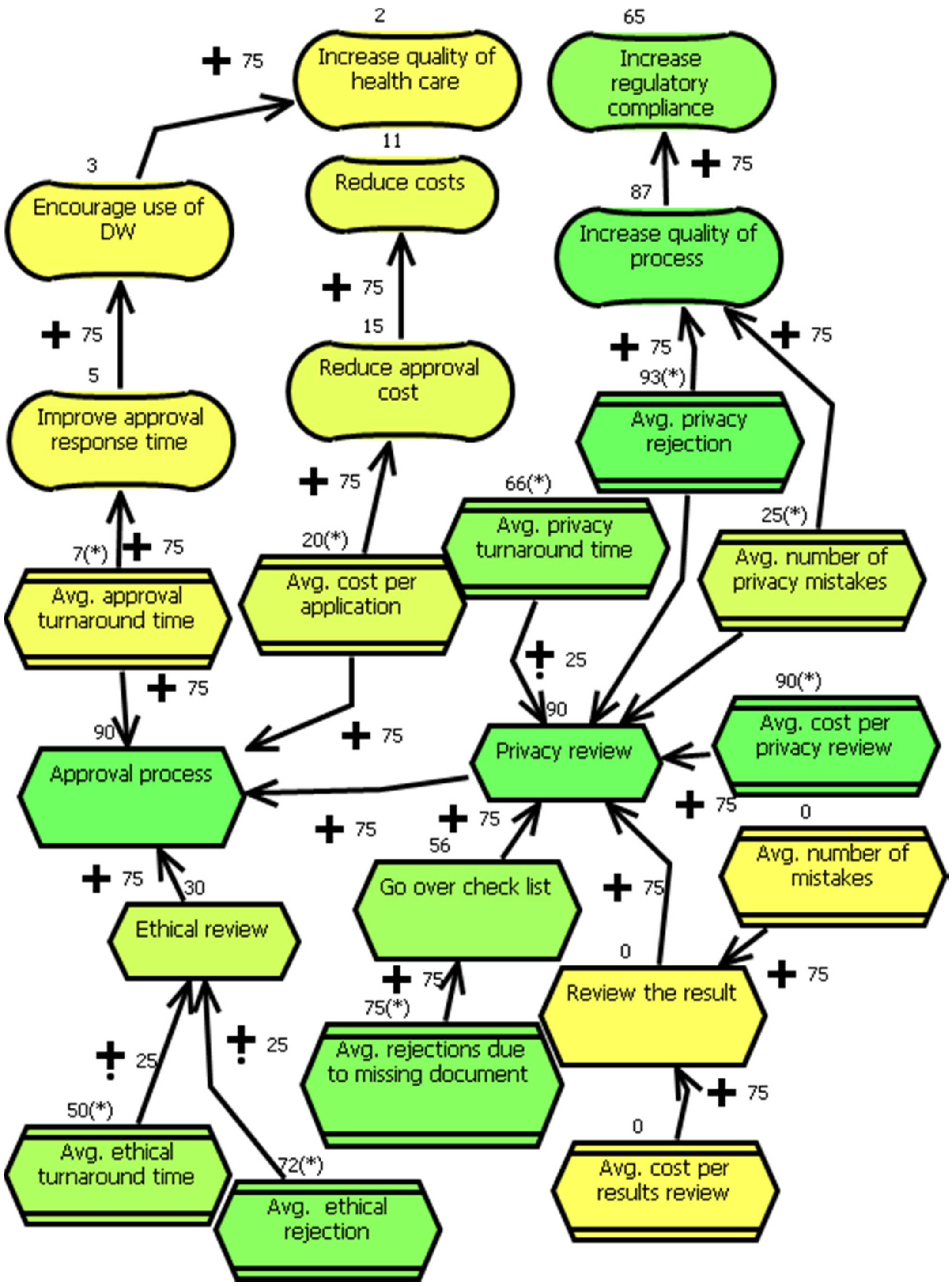


Figure 54: Evaluated goal and performance views after applying the patterns

7.4. Chapter Summary

In this chapter, we applied the framework to a healthcare domain example. The example shows the application of the framework from the beginning to the end by starting from modeling the context, applying the improvement patterns and then illustrating how the results can be monitored. One of the issues that this example shows is the need for customizing the patterns to produce fewer false positive matches. This validation through a real-life example was done prior to and was a deriving factor for the proposal to have a partial matching approach (section 5.4), which reduces the need for pattern customization and makes the application of library of generic patterns much more compelling for the business. The example shows the approach can be used in real-life examples and help such organizations optimize their processes to get closer to their targets.

Chapter 8. Retail Business Example

In this chapter, we describe the application of the framework defined in Chapter 4 to a real Ontario-based retail business. We explore the iterative and incremental modeling of the business goals, performance model, and decision mode. We also use the enhanced performance view discussed in Chapter 6. Finally, we show how the process improvement patterns from Chapter 5 can be applied to the business processes of the retail store.

The retail business would be categorized as a small enterprise (revenues less than $50 million) with 4 local stores and planned expansion nationally using a franchise business model. The business has existed for over 15 years, establishing a strong foothold in one neighbourhood. Three years prior to the study, new owners who set national growth as a key strategic objective purchased the business. As part of the expansion plans, market and competitive studies were conducted. In addition, the owners had created a scorecard that tracked key operational indicators and provided the ability to conduct an assessment of business results. Some data however, for example, the flow of customers through each of the locations, was not yet available in the scorecard.

8.1. Modeling the Current State of the Business

At the time of the study, most revenues were earned through *consignment* sales. The business, however, had started selling new items as well and was planning to invest in an *online* business. Revenue was driven by ensuring that stock was properly displayed, which in turn depended on assuring that enough staff were available to sort, tag, and lay out the products. The supply side of the business depended on the number of consigners available, the amount of product they brought to each store, and the speed at which these products could be displayed. The demand side depended on local advertising and word of mouth to stimulate traffic flow. All stores were situated in prominent locations with good visibility that stimulated walk-in traffic.

We started implementing the framework by going through the analysis lifecycle (see section 4.3) and playing the analyst role (see Table 9) in step 1 of that lifecycle. To begin our investigation, we interviewed the CEO, as analysts would do, in order to identify the high-level goals as well as KPIs and drivers of organizational success. As depicted in Figure 55, the goals of the principals, i.e., business owners, were related to market growth: they wanted to be the number one distributor within their geographical market. Store managers were aware of the growth objective, but on the short term, they focused on increasing revenues and the number of items sold in their stores.

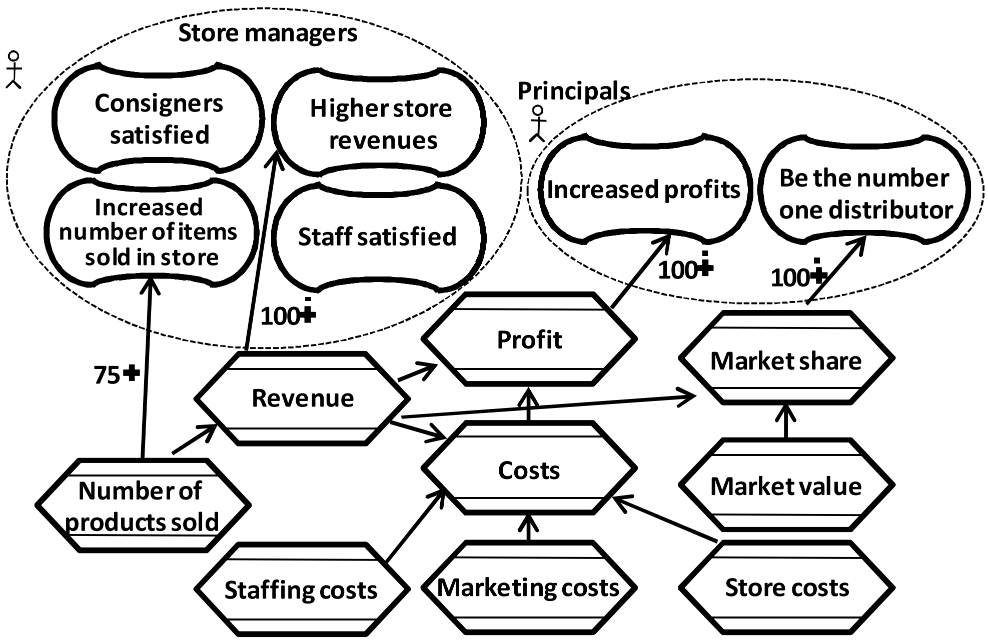


Figure 55: First iteration model (aggregation formulas not shown here for simplicity)

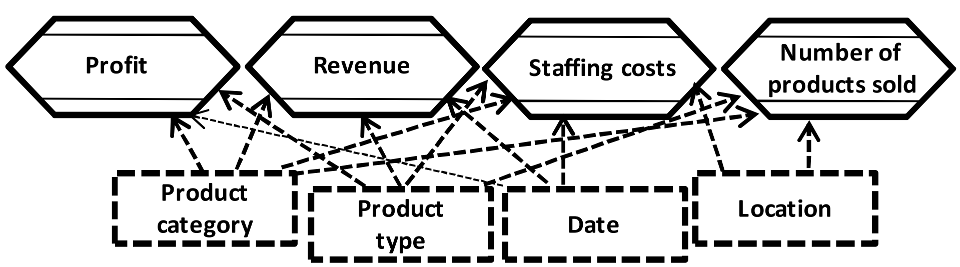


Figure 56: First iteration dimensions

As discussed above, the first iteration of the model provides an initial alignment of higher-level goals and KPIs. We started with a minimal decision model and limited set of data just to illustrate the business goals and financial targets and to identify the indicators and driver KPIs required to monitor the business and to make informed business decisions. Figure 55 illustrates the first iteration of the model. At this stage, we also developed a rough dimensional model (Figure 56) in order to ensure that the data needed for the performance model would be available. The dimensional model helps the store to analyze the impact of the KPIs on goals based on their different store locations, in each period of time, by product type (e.g., clothing, electronics, etc.), and by product category (i.e., retail and consignment).

8.2. Analysing the Business

In step 2 of the analysis lifecycle, Table 9, more KPIs were added to the model as drivers and then linked to the high-level financial KPIs and organizational goals. In addition, we added the new KPIs as well as a new dimension called "marketing type" (e.g., outreach, online advertising, etc.) to the dimensional model. This new dimension allows decision makers (i.e. consumers and analysts in the defined roles for the framework) to analyze which marketing initiative has a more significant impact on the goals. In this step, we also created a decision diagram (Figure 57) illustrating the specific actions consumers can take to improve goal accomplishment. The decision model helps with analysis of the business situations and comparing alternative options in a particular case. If the data is available, GRL strategies can be used to show the impact of various alternatives on the performance model. In this example, one of the decision options available to the business managers is to invest in an online business by increasing the advertising budget for the website and increasing the website maintenance budget, which are reflected in Figure 57. We consider this option as the decision made by managers and update the models accordingly, which are considered the transition steps between Step 2 to Step 3 of the analysis lifecycle as described in Table 9.

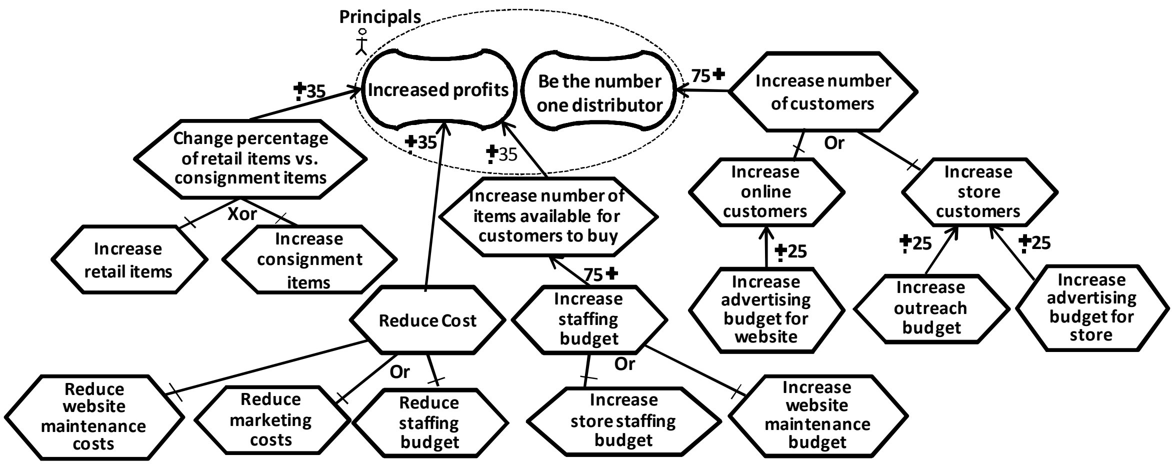


Figure 57: Second iteration decision diagram

8.3. Monitoring, Evaluating Results, and Refinement

In step 3 of the analysis lifecycle, we added situations, new KPIs and started the monitoring effort. Figure 58 depicts the third iteration model, which defines the expected impact of the actions identified in Figure 57 along with the KPIs and acceptable ranges for each of the relevant goals based on the *situations* associated with each goal. Table 22 shows the formulas defined between the KPIs in this model. There is one new situation element in the model reflecting the initial cost of *investment in the online business*. This situation is a threat at this point. Therefore, the situation increases the acceptable range of the profit values by moving the threshold value closer to the worst value, as explained in Table 17. Furthermore, there is also a new KPI used to monitor the investment made in the online business and its impact on the costs. The GRL strategy used for the evaluation here focuses on the use of the online business investment. Note that after the cost of investment is absorbed by the business, the online business investment KPI can be eventually removed from the model. In addition, due to the expected online revenue stream, the associated threat can eventually be replaced by an opportunity situation. This opportunity (where the contribution will become positive) will work in the exact opposite way of the currently modeled threat and will increase the expectation from the profit KPI.

Figure 59 depicts the third iteration dimensional model (including its new "Sales method" dimension) used to ensure that decision makers can compare and analyze the numbers for the online part of the business separately. In addition, this model now also supports the location dimension on more KPIs, allowing business owners to compare the stores. Note that, in jUCMNav, such figures can be split over many diagrams when they become complex.

Another enlightening aspect of the third iteration model, Figure 58, is the evaluation value of the defined KPIs as well as the relationship of the KPIs with various stakeholder goals. Note also in this figure that the performance view in jUCMNav was enhanced to show the KPI values themselves (in blue), with their units. As shown, the evaluation value of "Average turnover days", which has significant impact on the satisfaction of both Revenue and Consigners, is 55. In this case, the lower the number of turnover days, the better the outcome. Therefore, according to the model, business owners have to reduce the average turnover time to increase the consigners' satisfaction and store revenues. According to Figure 60, the latest version of the decision model, the only option was to increase the number of staff. However, when the new location dimension added to Figure 59 is used, one could easily point out that the average number of turnover days is significantly higher, with negative impact on the business, in one of the stores even though the numbers of staff in all stores were the same. After some brainstorming, owners realized that the design of that specific store as well as how the drop-off is handled in that store is contributing to the higher number of turnover days. Therefore, their decision model was updated (Figure 60) to show process improvement as an alternative to increasing the number of products available to customers, which according to the model directly impacts the average turnover days. This process improvement will be specified with Use Case Maps.

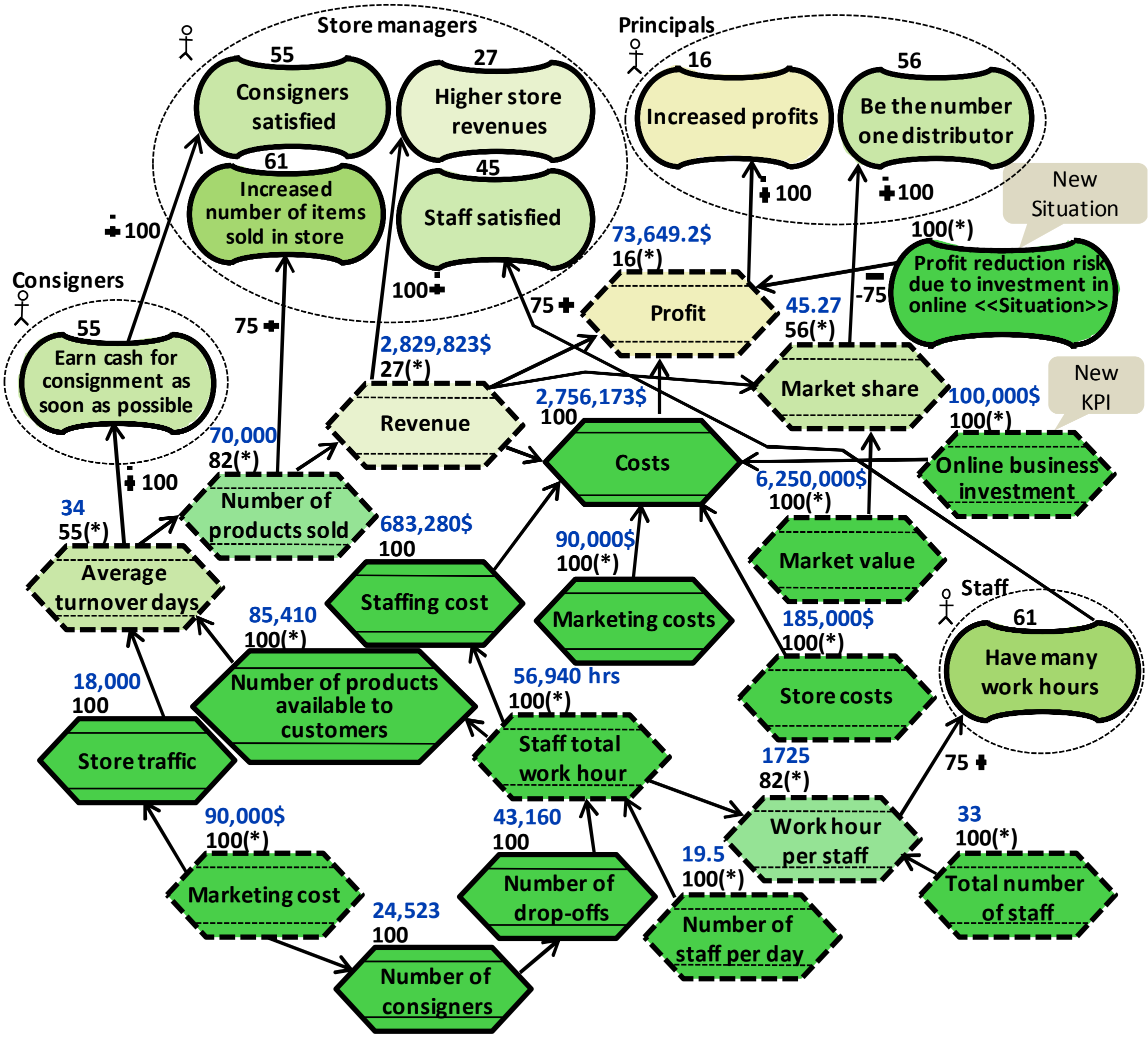


Figure 58: Third iteration model – evaluated

Table 22: Third iteration model formulas

Formula	*Target KPI*	*Source KPIs*
2*MC/10	Store traffic	MC: Marketing cost
floor(MC/3.67)	Number of consigners	MC: Marketing cost
floor(1.76*NC)	Number of drop-offs	NC: Number of consigners
365*8*NSPD	Staff total work hours	NSPD: Number of staff per day
STWH/TNS	Work hour per staff	STWH: Staff total work hour TNS: Total number of staff
STWH*8 + STWH*4	Staffing cost	STWH: Staff total work hour $8 per hour + $4 overhead
STWH*1.5	Number of products available for customers	STWH: Staff total work hour
StaffC+MarketingC+StoreC+R*0.6+OBI	Costs	StaffC: Staffing cost MarketingC: Marketing cost StoreC: Store cost OBI: Online business investment R: Revenue
R-C	Profit	R: Revenue C: Cost
(R*100)/MV	Market share	MV: Market value

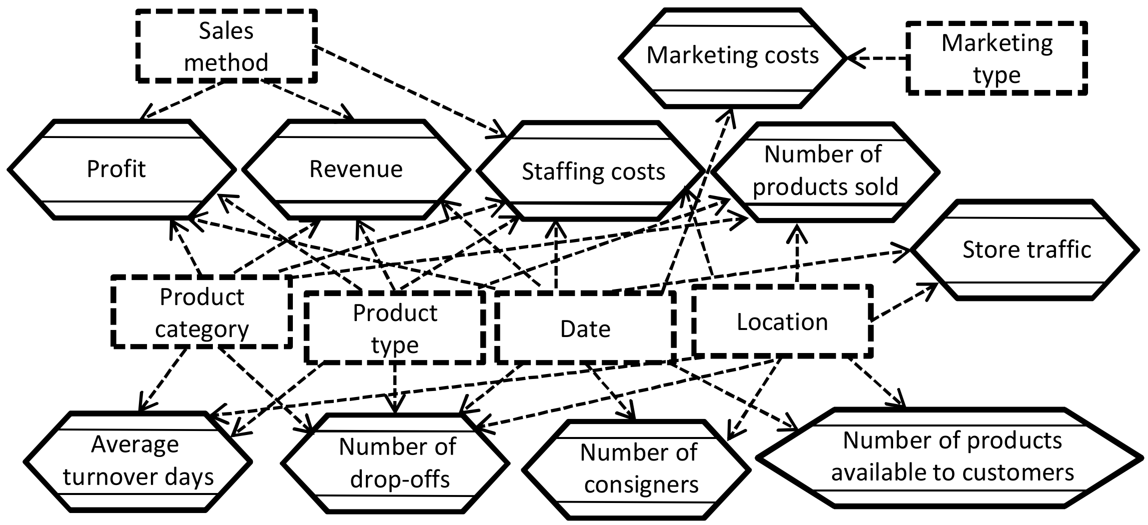


Figure 59: Third iteration dimensions

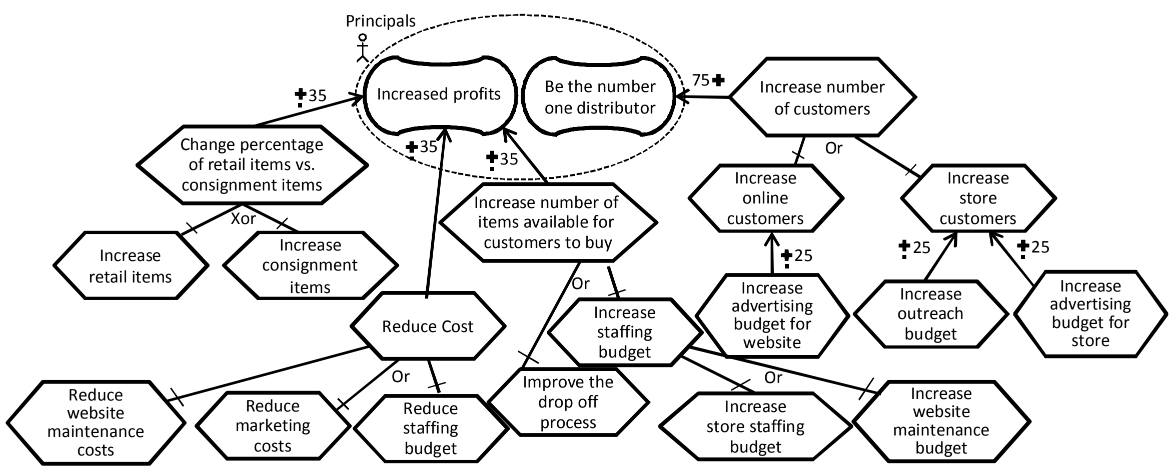


Figure 60: Fourth iteration decision model

The change in the decision model also highlighted a potential enhancement in the third iteration performance monitoring model as well as the defined GRL strategy and target values for the KPIs (Figure 58). Although the impact of “Number of products available to customers” on “Average turnover days” is illustrated on the model, the target value for the earlier is not accurate. To elaborate, the discussed KPI evaluation value is 100, which means it fully satisfies the expected value, whereas we know we have to improve this number to improve the average turnover days. This is another example that shows how not having enough information about the context and relationships at the beginning is not a showstopper for using the framework. Analyzing the model in each iteration can help us both to understand the business more accurately and to improve the model itself and change the business targets accordingly.

In the adaptation lifecycle, the owners decided to improve the drop-off (of second-hand goods to sell) process model. The as-is drop-off process was modeled using UCM, as shown in Figure 61.

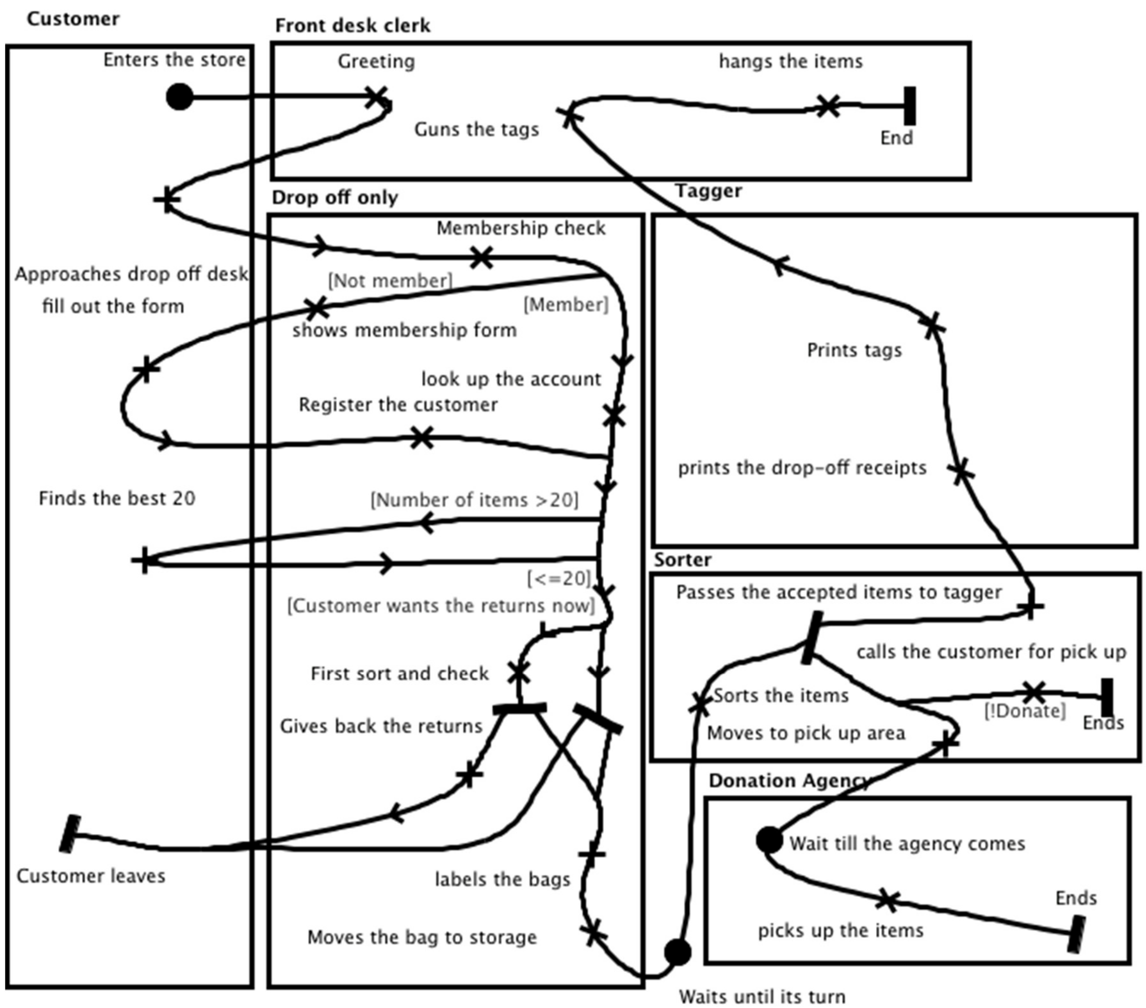


Figure 61: Drop-off as-is process model

After some brainstorming and considering some process redesign patterns as a reference (i.e., task reduction, task composition, and technology integration), a to-be process was proposed (Figure 62). This to-be process was designed to minimize the time staffs spend on the drop-off process as well as reducing the wait time between drop-off and the time that the items are on display for customers to buy. Note that in the first discussions around adaptation, the semi-automated patterns where not used and we only used the re-design patterns as best practices. This is evidence that the framework is flexible as if the smaller businesses or the analysts are not ready to commit to every single step by the book, they can choose what works for them, and help the business. In smaller businesses with fewer processes, this approach may prove to be less costly. However, in larger businesses with thousands of processes, pinpointing the problem areas is much harder and using the tooling and semi-automated patterns to detect the potential improvements can be more efficient.

As illustrated in Figure 61, in the as-is process, all the interactions with customers are done manually, including the registration of new customers and the entire process for dropping off items to be sold. This problem was addressed in the to-be process, Figure 62, by adding kiosks to the process to automate part of this work and reduce the staff workload, technology integration redesign pattern. In addition, in the as-is process, there are two sorting steps in the process, one in the presence of the customer and another one later on. If a customer does not want to donate the items deemed “unacceptable”, then there is a follow up step to call the customer to pick up these rejected items. This sequence was streamlined in the to-be process using task reduction and composition. This was achieved by sorting all the items in the presence of the customer or mandatory donation as the two possible options for the customer. This change not only reduces the staff workload but also removes a backlog of items to be sorted from the entire process and reduces the time lag between drop-off and display significantly.

One may argue that the workload and performance of the staff is impacted by store traffic and therefore the current process may perform very well in low traffic hours but not so much in high traffic hours. This issue is applicable to all systems and processes where workload intensity affects outcomes (e.g., by increasing faults or by reducing productivity). To distinguish the conditions where systems are under significant workloads from normal conditions, either a different set of KPIs or a new workload dimension can be used. Such approaches can help analysts study the system under various conditions and distinguish the issues that are associated to high intensity workload from those that happen under normal conditions.

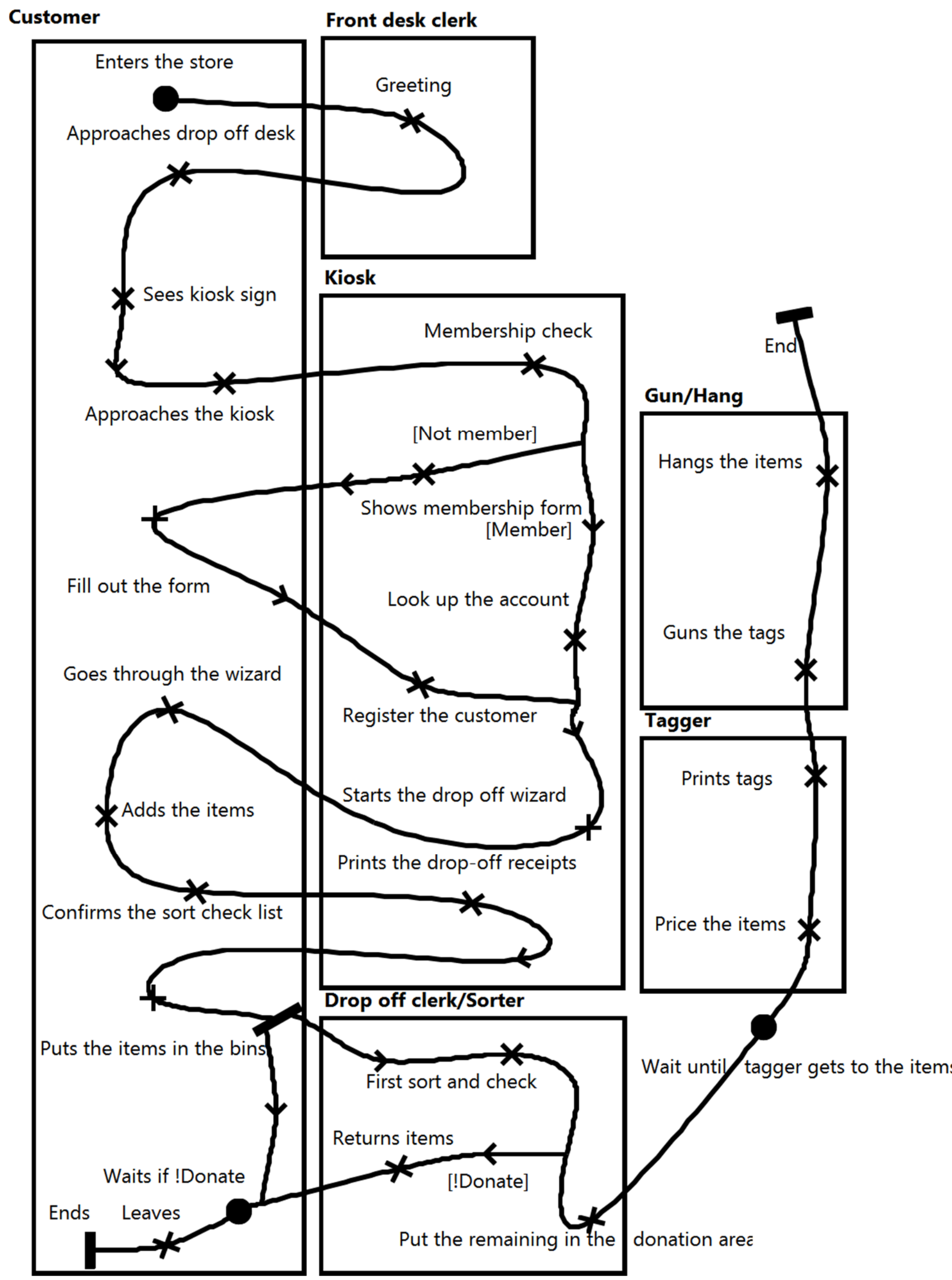


Figure 62: Drop-off to-be process model

After the suggested changes to the process model are implemented, the model allows the owners to monitor the results and see the impact on the KPIs that were assumed to be improved using the new process model (i.e., average turnover days in this example).

8.4. Pattern Matching and Application Examples

As discussed in the previous section, the adaptation can be done manually. However, if the businesses are ready and have a large enough process set that cannot be handled manually, they can use the semi-automated patterns and the supporting tool to make the process more efficient. In this section, we describe the application of the patterns to the retail example. Figure 63 and Figure 64 respectively show the high-level business process and performance model of the retail story.

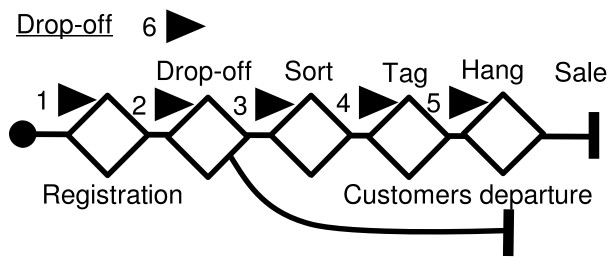


Figure 63: Retail store high-level process

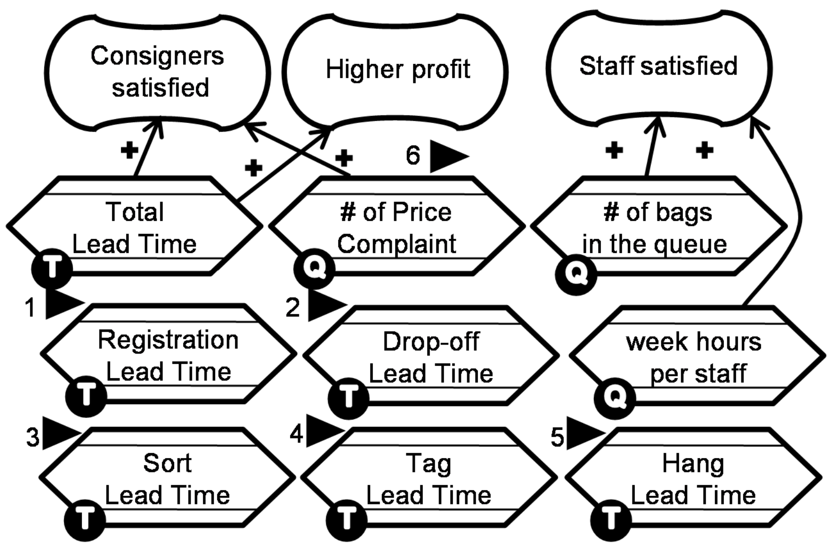


Figure 64: Retail store goals and KPIs subset

8.4.1 The Task Composition Pattern

In the first example, "Task Composition" pattern introduced in section 5.3.4 is applied against the process described in Figure 63. As illustrated in Figure 65 – Option B (duplicate of Figure 26 for convenience), the behavioral detection model matches against a repetition of stubs with any names. Hence, the stubs Registration, Drop-off, Sort, Tag, and

Hang are all matched. The behavioral detection model matches against any combination of two or more consecutive stubs, i.e., ten matches of the behavioral detection model are possible and reported as the result of the first stage of the partial matching algorithm (see section 5.4).

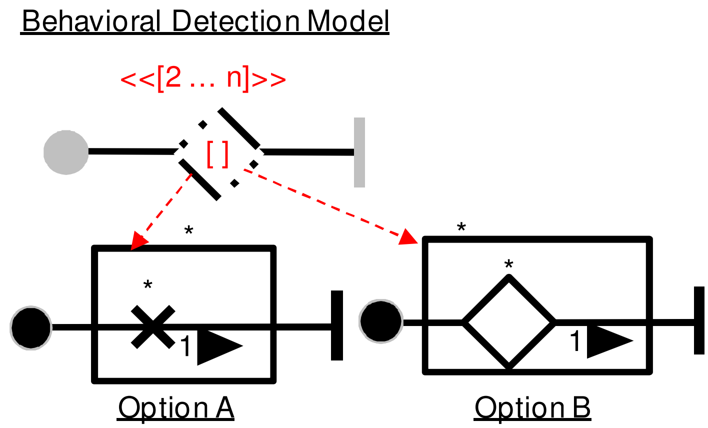


Figure 65: Task Composition – Behavioral detection model

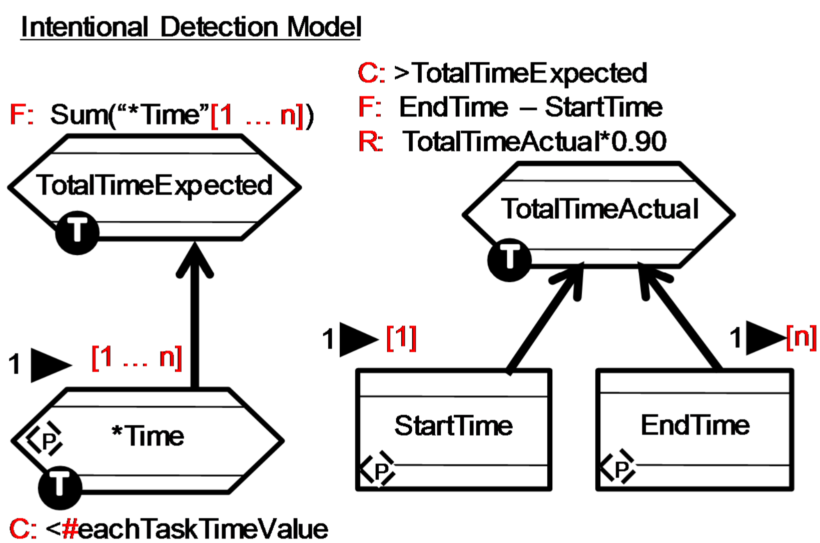


Figure 66: Task Composition – Intentional detection model

In the second stage of the matching algorithm, the intentional detection model in Figure 66 is used, requiring a KPI of type time to be found in the model (i.e., the **Time* KPI with the  and  icons). In this case, the lead time KPIs measuring the execution duration of a sub-process are present for each stub in the GRL model in Figure 64. However, the StartTime and EndTime data model elements do not exist in the GRL model. Therefore, the resulting matches can only be partial. In the third stage, the conditions are evaluated. Therefore, only those combinations where each lead time for each stub is less than the #eachTaskTimeValue constant will partially match against the intentional detection model. The other condition is not attempted to be matched, because not all structural model elements required for the condition could be matched during the second stage. Assuming that the lead times of the first three stubs do not violate the condition, the partial matching algorithm hence reports that there are no overall successful matches, seven unsuccessful matches, and three potential matches.

The potential matches are illustrated in Figure 67 using conventional AoURN aspect markers (◆) to denote the beginning (i.e., a1, b1, and c1) and end of a match (i.e., a2, b2, and c2). Therefore, the stub combinations Registration/Drop-off (a), Registration/Drop-off/Sort (b), and Drop-off/Sort (c) are potential matches.

Given that these are partial matches, the modeler may now turn his/her attention to what is required for a full match. Therefore, the intentional detection model must be composed with the base GRL model in a way that helps the modeler with this task. In AoURN, the matched, missing, and added model elements are highlighted in the composed model with AoURN aspect markers (◆), in dark red color, and in light gray color, respectively, as shown in Figure 68. The figure shows all three potential matches simultaneously with the help of an array notation and clearly indicates that the StartTime and EndTime data model elements are missing in the current GRL model and need to be added to it. At this point, there are three options. Either the data for these model elements have already been collected by the organization, in which case they can simply be added to the GRL model, or they are not available. The latter case is an indication for the organization to gather more measurements to be able to improve its business processes, or alternatively, the organization may decide not to collect further measurements, maybe because doing so is too expensive, and hence the decision is made not to improve the business process in this case.

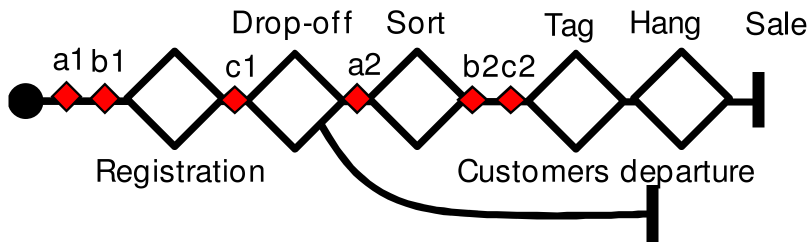


Figure 67: Task Composition – Matches in the UCM base model

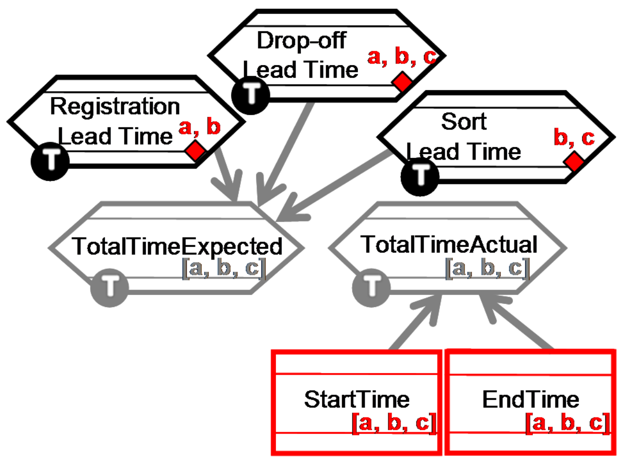


Figure 68: Task Composition – Matches in the GRL base model

In Figure 68, the model elements are marked by similar labels (i.e., a, b, and c) as are used for the matches in the UCM model to identify the matches that belong to the corresponding aspect markers in Figure 67. Each label is associated with exactly one matched instance of the intentional detection model, e.g., only those elements that are relevant to the Registration/Drop-off (a) match are labeled with (a) in the composed GRL model. Note that if a modeler wants to focus on a single match, then it is straightforward to show this match individually instead of all matches at once.

After all the required model elements have been added to the GRL model, only those cases among the three potential matches found earlier that now satisfy also the condition of the added KPIs (i.e., TotalTimeActual>TotalTimeExpected) will be matched. After the matches are identified, the associated tasks and roles in the process that are performing the task will be merged into one task and role to improve the turnaround time. This step requires human smarts and will be done manually.

8.4.2 The Case Manager Pattern

In the second example, the Case Manager pattern introduced in section 5.3.5 is applied to the retail store high-level business process. Figure 70 shows where the "Case Manager" pattern, Figure 69 (duplicated from Figure 28 for convenience), is detected in the process model assuming that the #MinRoleCount constant is five. The suggested improvement is inserted into the process model again with the help of the conventional aspect marker (◆), which is automatically linked to the improvement model by the conventional AoURN composition mechanism. The behavioral detection matches the base model because there is a sequence of activity exists in the base model without any managerial check and the intentional detection model matches because the quality KPI (i.e. # of Price Complaint in the base goal model, See Figure 64) is not meeting the defined #QualityTarget constant.

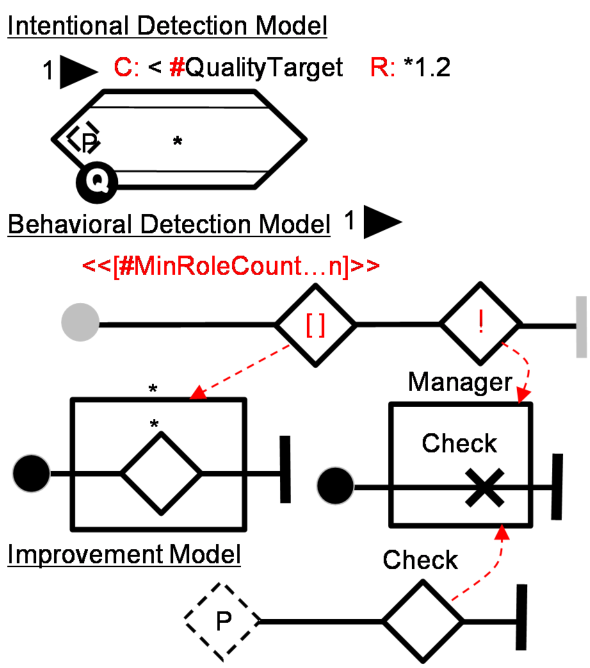


Figure 69: Case Manager – Intentional and behavioral detection models, as well as improvement model

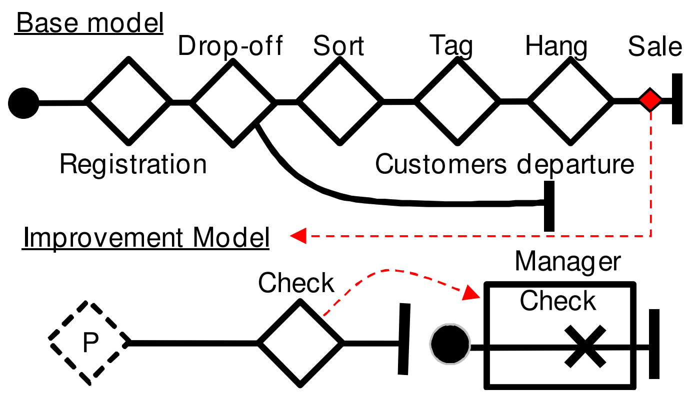


Figure 70: Case Manager – Detection and suggestion in UCM base model

8.4.3 The Task Automation Pattern

In the third example, the Task Automation pattern introduced in section 5.3.6 is applied to the retail store high-level business process. Figure 72 shows where the "Task Automation" pattern, Figure 71, is detected in the base model. This example shows only potential matches because the Turnaround KPI does not exist in the base GRL model. Two responsibilities in the registration sub-process (i.e., Fill out the form and Registers) are in the automation dictionary and hence are matched. These matches are again indicated with the help of aspect markers and missing KPIs in red color with the array [a, b] to indicate KPI instances as explained for previous examples. This is a good example to show how the Task Automation pattern can help detect automation possibilities and hint analysts for potential improvements.

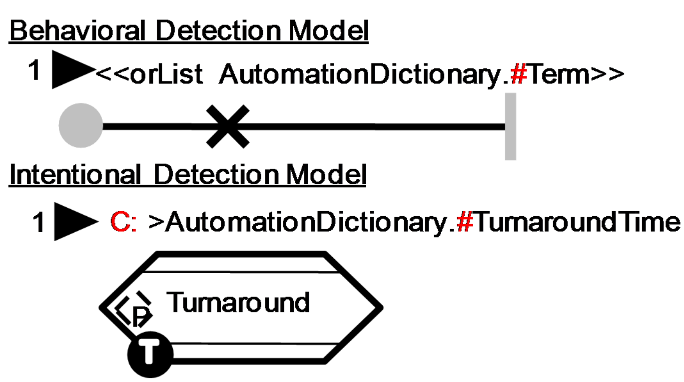


Figure 71: Task Automation – Behavioral and intentional detection models

Ultimately, the decision is up to the analyst and the consumers (here, the retail chain owners) to decide if they want to do any automation. In this case, the retail store changed the manual registration process to use either their website or kiosk in the store because they realized that this approach can both modernize their store and reduce the time that staff spend on registration, which allows them to work on the backlog of the drop-off items and get the items on the floor faster, which would in turn have a positive effect on both the customer satisfaction and the business bottom line.

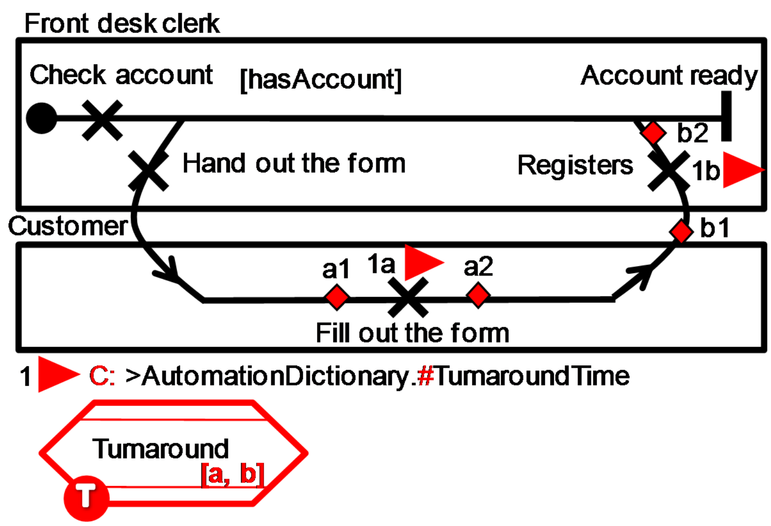


Figure 72: Task Automation – Detection in UCM and GRL base models

8.5. Chapter Summary

In this chapter, we illustrated the application of the two lifecycles of the framework (i.e. Analysis and Adaptation – see sections 4.3 and 4.4 respectively) for creating the required models to monitor and improve a business in a relatively small retail store business without a lot of pre-existing monitoring infrastructure. This approach showed how using the framework and going through steps 1 and 2 of the analysis lifecycle can increase the awareness of the business owners about the business drivers and the required KPIs to be monitored to improve the business. In the first phase of this study, the models were used to allow owners to think about potential process improvements and, using the patterns, a manual adaptation of the processes was done. In addition, after further progress on the redesign patterns modeling in this research, we illustrated how patterns and partial matching can be used to detect the potential improvements as well as the missing data and model elements that will allow the business to enhance their performance models. This real-life example proved to be valuable in assessing the validity of this thesis' framework.

Chapter 9. Validation and Evaluation

This chapter presents the iterative validation and evaluation that was done throughout this research. Since the research method used in this thesis, described in section 1.4, is an adaptation of the design science research method, the validation was not left to the end. Some level of validation was done at each stage of the research and the output was used in a feedback loop, which improved the intermediate and final outcomes. In summary, we validate the framework in this thesis in five ways:

1. *Systematic literature review*: We compare the framework with the state of the art and validate it from novelty and feasibility points of view.
2. *Mandatory elements of method*: We use the structured method suggested by Zellner [155] to validate that the framework satisfies the requirements of a business process improvement method.
3. *Real life examples*: We apply the framework to real-life examples and illustrate that it can be applied in two different domains.
4. *Pilot experiment*: We perform a pilot experiment as an initial evaluation on the usability of the notation.
5. *Tool support implementation*: The formula-based algorithm is implemented to validate the proposed GRL evaluation of aggregated KPIs as one of the important parts of the framework.

The rest of this chapter will review the validation results, one section per way.

9.1. Evaluation Based on the Systematic Literature Review

In Chapter 3, we conducted a systematic literature review on the application of aspect-oriented techniques to business process adaptation. We tried to evaluate our framework from novelty and feasibility points of view. We investigated the novelty and feasibility of the framework by seeking answers to the following questions:

- Have aspect-oriented methods been used for business processes adaptation?
- Is our proposal for an AoURN-based framework realizable given prior research and existing infrastructures?

After reviewing 60 papers (54 in the first version and six additional papers in 2013) , whose analysis results are discussed in section 3.3, we concluded that our framework is both novel and realizable. The framework is novel because we did not find any other method that uses aspect-oriented methods for the detection of undesirable conditions and the adaptation and improvement of goal, process, and performance models at the same time using predefined patterns. Our framework is realizable because there is much existing research that shows the existence of technological infrastructure supporting aspect-oriented techniques.

In the existing research, significant attention was devoted to the application of aspect-oriented techniques to business process and service-based system adaptation. Much of the research is focused on adaptation at the process execution level and in the following areas: i) composing and swapping services based on QoS, cost, and rules, and in the event of failure, ii) extracting roles and crosscutting concerns from composite services, iii) customizing process instances based on user profiles or Service Level Agreements, iv) adapting service composition and collaboration policies, and v) using monitoring aspects to detect undesired conditions.

Current research mainly focuses on *instance-level* adaptation [72] as opposed to considering the overall performance of the process and improving the *process model* itself. Furthermore, the impacts on the goals of business are often not considered. As illustrated in Table 23 (already shown as Table 8 in the literature review), only the framework proposed in this thesis considers business goals, performance models (KPIs), constraints, and redesign patterns (for process improvement) as parts of its monitoring and adaptation steps.

Table 23: Reviewed papers compared with our framework

		Ref.	Author - Year	Adaptive systems	Aspects used?	Process improvement?	Redesign patterns?	Business goals & KPIs?
A	CS	[5]	Algahtani and Zedan - 2009	Y	Y	Y	N	N
		[56]	Hermosillo *et al.* - 2010	Y	Y	Y	N	N
		[53]	Hermosillo *et al.* - 2010	Y	Y	Y	N	N
		[54]	Hermosillo - 2012	Y	Y	Y	N	N
		[107]	Narendra *et al.* - 2007	Y	Y	N	N	N
		[43]	Erradi *et al.* - 2005	Y	Y	N	N	N
	E	[120]	Rahman *et al.* - 2008	Y	Y	N	N	N
		[107]	Narendra *et al.* - 2007	Y	Y	N	N	N
		[128]	Sánchez and Villalobos - 2008	Y	Y	N	N	N
		[136]	Stearns - 2002	Y	Y	N	N	N
	CP	[148]	Wan, *et al.* - 2008	Y	Y	N	N	N
	MA	[24]	Charfi *et al.* - 2009	Y	Y	Y	N	N
		[144]	B. Verheecke *et al.* 2003	Y	Y	N	N	N
		[109]	Patiniotakis *et al.*- 2013	Y	Y	Y	N	N
O		[50]	Graml *et al.* - 2007	Y	N	N	N	N
		[127]	Sawyer *et al.* - 2007	Y	N	N	N	N
		[126]	Ruy *et al.* - 2007	Y	N	N	N	N
		[71]	Kazhamiakin *et al.* - 2009	Y	N	Y	N	N
		[42]	Erradi *et al.* - 2006	Y	N	N	N	N
		[73]	Kalavathy *et al.* 2010	Y	N	N	N	N
		[83]	Lian *et al.* - 2010	Y	N	N	N	N
		[106]	Na *et al.* - 2010	Y	N	N	N	N
		[121]	Ramakrishnan - 2009	Y	N	N	N	N
		[86]	Lu - 2011	Y	N	Y	N	N
		[151]	Xiao *et al.* – 2011	Y	N	N	N	N
		[32]	Conforti *et al.* – 2012	Y	N	N	N	N
		[41]	Eleonu and Oruh - 2013	Y	N	N	N	N
G		[70]	Kapuruge *et al.* - 2010	Y	N	N	N	N
		[26]	Cheng *et al.* - 2009	Y	N	N	N	N
		[30]	Courbis and Finkelstein - 2005	Y	Y	N	N	N
		[72]	Kazhamiakin *et al.* - 2010	Y	N	N	N	N
P		[117]	Pourshahid *et al.* - 2011	Y	Y	Y	Y	Y

A: Aspect-oriented-based methods, **G:** Generic studies, **O:** Other methods **P:** Proposed method
CS: Composing and swapping services based on QoS, cost, rules, and failures
E: Extracting roles and crosscutting concerns from composite services
CP: Customizing process instances based on user profiles or SLAs
MA: Using monitoring aspects to detect undesired conditions

Furthermore, researchers have the following two main concerns about the complexity of aspect-oriented approaches:

- Learning curve in aspect-oriented approaches could be (too) high. This was also verified by our pilot study in section 9.4.3.
- Interaction between aspects when multiple aspects are applied to a process could increase the complexity of the system.

In addition, common issues with aspect-oriented modeling such as pointcut fragility [108] (when models evolve) are still present. Considering the lessons learned from this research, in our future work, we are going to extend our framework to at least support monitoring of the process models adaptation impact on all layers of the system including process execution layer. Since there has been much research in this area and since the existing work will likely make our proposed holistic modeling framework realizable in the future, we are going to continue on this research. However, we have to address the complexity issues associated with aspect-oriented methods to make the framework usable by business people. A potential solution to address this problem is to hide the complexity of aspect-oriented models behind a layer of abstraction that simplifies the interaction of the users with the system. Another potential and complementary solution avenue is that of explicit interaction handling at the modeling level through the use of Concern Interaction Graphs, as suggested by Mussbacher [93][98].

9.2. Evaluation of Mandatory Elements of a Method

Zellner introduces "mandatory elements of a method (MEM)" in his structured evaluation of business process improvement approaches [155], a recent approach that is already well cited. Zellner believes a process improvement methodology can be complete if it describes all these elements: 1) Procedure model, 2) Technique, 3) Results, 4) Role, and 5) Information Model. In addition, after evaluating 14 state-of-the-art business process improvement methods selected using a systematic approach, he concluded that no existing method possesses all the mandatory elements. We performed two rounds of evaluation on our approach based on these mandatory elements. In the first round, which was done midway through the research, two shortcomings were identified that were addressed later on in the research. The second round of evaluation shows that these shortcomings have been addressed. Table 24 illustrates the result of this evaluation.

Table 24: Evaluation of the methodology using Mandatory Elements of a Method

MEM	**First round**	**Second Round**	**Discussion**
Procedure	◎	◎	The proposed method consists of defined steps that are discussed and illustrated using examples.
Technique	◎	◎	Several specific techniques are defined for different steps, including formula-based cause-effect analysis, aspect-oriented modeling and application of re-design patterns, where examples are given for each.
Results	◎	◎	The expected inputs and outputs of each method step are discussed, with examples.
Role	○	◎	Roles involved in the framework were not initially included in the framework. This shortcoming was later on addressed in section 4.5.
Information Model	○	◎	Information model (meta-model) that shows the relationship between different elements of the method was not available in the first round. This issue was addressed later in section 4.2.

Accomplished:◎, Partially Accomplished: ⊙, Not Accomplished: ○

9.3. Evaluation of the Method using Examples

Although we have evaluated our framework using a systematic review and MEM, we also believe using the method in examples better illustrates the application of the framework, helps us detect shortcomings, and helps us come up with ideas for potential future improvements. Therefore, we have also applied our framework to two real-life examples. The details on these examples are available in Chapter 7 and Chapter 8, and a summary of what we learned in these studies is presented here.

9.3.1 Healthcare Example

We validated our approach partially using a healthcare example, discussed in Chapter 7 of this thesis. Although we tried to make the patterns we used as generic as possible, we realized that having very generic patterns can be challenging and can potentially cause false positives. In other words, the right level of genericity of redesign patterns represented as aspects needs to be determined, as balancing reusability with the ability to avoid incorrect matches and false positives remains a challenge for now. Therefore, although a library of generic patterns can be useful as a starting point, we came to the conclusions that in many cases the users of these patterns need to perform some level of customization to adapt the patterns to their models. This conclusion was used as a feedback loop and partial pattern matching was introduced in section 5.3.7 in order to address this issue.

In addition, although we believe much of the complexity of the patterns and the underlying aspect-oriented techniques can be hidden from business analysts, managers, and other potential end users of this framework, more research on the usability of this approach is also required. Even though we performed a pilot experiment (to be discussed in section 9.4), which confirmed the complexity of the notation, a more comprehensive usability study in the future would be beneficial. Our examples show that patterns can become quite complex and one requires to have advanced knowledge of AoURN to be able to use the patterns. In our methodology, we try to address this problem by defining a specific technical role (i.e., Analyst) who needs to deal with this level of complexity if an organization decides to use our framework.

9.3.2 Retail Business Example

The development of our framework and its application to the retail business led to several lessons learned. From a business management perspective, we observe the following:

- Modeling goals and defining drivers and KPIs (i.e., creating the cause-effect decision model) not only helps to document the known aspects of the business but also helps to clarify unknown factors that might be driving goal accomplishment. Validation of the model through interviews with decision makers ensures that data and KPIs included are indeed relevant to the business. This can have a great impact especially for small businesses where initial goals might not be clear.

- Even though modeling the indicators helps define the required information and the relationships between variables, we are still unsure about where to draw the line regarding the data we need to show in the model versus the data maintained in source systems (e.g., databases or BI reports). We still need to explore how to find the appropriate balance so we do not omit important information in the model for decision making while preventing the inclusion of too much data/indicators, which can complicate the decision-making environment. We believe however that getting feedback on the right balance is facilitated by the use of graphical goal models with rapid evaluation feedback as provided by GRL strategies, which provide better cognitive fit than conventional BI reports. Note however that the goal-oriented view introduced here is complementary to what is found in BI tools, not a substitute.
- Defining relationships between the model elements without historical data is difficult. In some cases, managers (i.e., consumers) themselves are not aware of the linkages because they have not had the historical data available to create cause-effect models. In this case, analysts first create the models using industry standards (e.g., as seen in section 6.1) or "best guesses" and then use the different iterations of the framework to improve the cause-effect decision model.
- The ability to adjust the range of acceptable values for a KPI is useful for registering threats and opportunities (i.e., situations). For example, one might establish a wide range of acceptable values for an objective in a specific condition, such as expected sales for a new product. On the other hand, objectives with lower profiles, such as sales of products once they become well established, might have a narrower range of acceptable values.

From a technical point of view, we have learned that:

- Our new extensions to GRL and the new formula-based algorithm provide much flexibility for model evaluation, especially as they are combined with standard goal satisfaction evaluation, hence offering the best of both worlds. However, our new algorithm still has room for improvement, especially when it comes to using other intentional elements (e.g., goals) as contributors to KPIs. We have had limited experience with this idea by considering situations as input to KPIs, but this type of modeling may be useful in other circumstances that require further investigation.

- Creating different versions of a model in different iterations and keeping them consistent for comparison purposes can be painful with current tool support. Saving separate files for each version of the model quickly becomes a maintenance issue that requires a better technical solution. Luckily, recent jUCMNav features such model differences and contribution overrides [10] help mitigate that issue.

9.4. Pilot Experiment

In this section, we discuss the pilot experiment that we performed to do an initial evaluation on the usability of the notation. This pilot was not a formal experiment. However, the result is promising and allows us to plan a more comprehensive study as future work.

9.4.1 Experiment Approach

The experiment questions were designed to evaluate the usability of the notation and patterns by users who are familiar with URN but who are not experts in AoURN. The six participants were selected to support this intention (several other volunteers were invited but did not have the time or the desire to participate). The questions were designed to be simple at the beginning (with guidelines and hints) and harder towards the end (by reusing already described patterns and by using more complex concepts). The intention was to evaluate and compare the participants' ability to comprehend introduced concepts and patterns. The expectation was that participants could understand and use the patterns after their introduction using hints and guidelines that are provided as documentation. Appendix A lists the questions that were sent to the participants.

9.4.2 Participants' Experience with URN and AoURN

The six participants were asked to identify their level of expertise in URN and AoURN. As illustrated in Figure 73 and Figure 74, the participants had a high level of experience with URN but had less experience with AoURN. None of the participants were advanced

AoURN users who have contributed to the research and used AoURN regularly for their work or research.

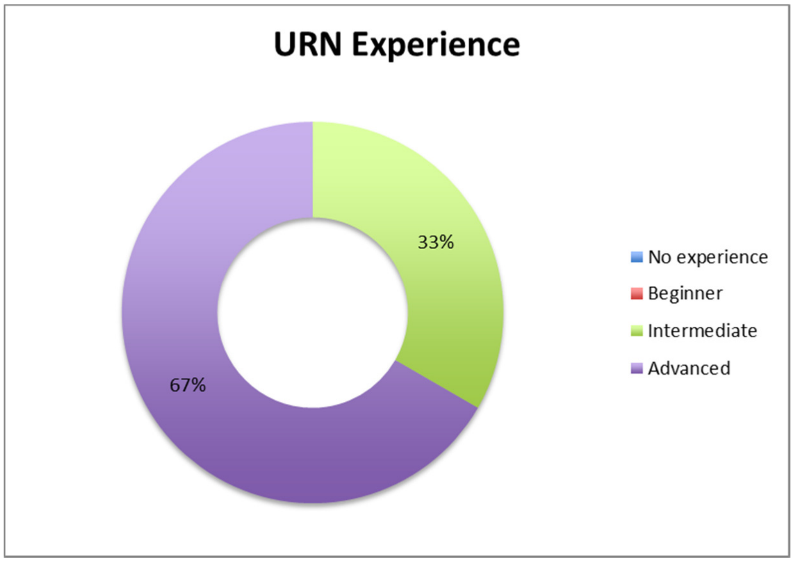


Figure 73: Pilot study – Participants' URN experience level

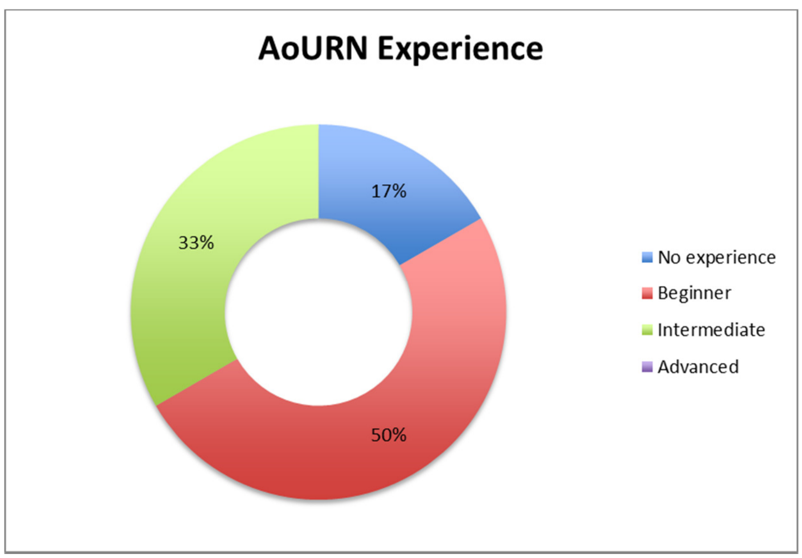


Figure 74: Pilot study – Participants' AoURN experience level

9.4.3 Pilot Experiment Results

The results of the pilot experiment suggest that when the notation concepts and patterns are explained to the users, they can be understood and used to complete other similar tasks. However, the notation is not self-explanatory and requires guidelines and documentation to be used even by people who are experts in URN. To have more accurate results with fewer numbers of threats to validity (see section 9.4.4), a more comprehensive and formal study has to be performed as part of future work.

As illustrated in Figure 75, the best possible result for the technical questions is when the total score for each question is 18, which is calculated by multiplying the highest possible score for each question (3) by number of participants (6). The possible scores for the technical questions are:

0 – No answer, or answer showing no comprehension of the concepts

1 – Mostly incorrect answer, but shows partial comprehension of the concepts

2 – Acceptable answers with some errors

3 – Correct answer with no or minor errors

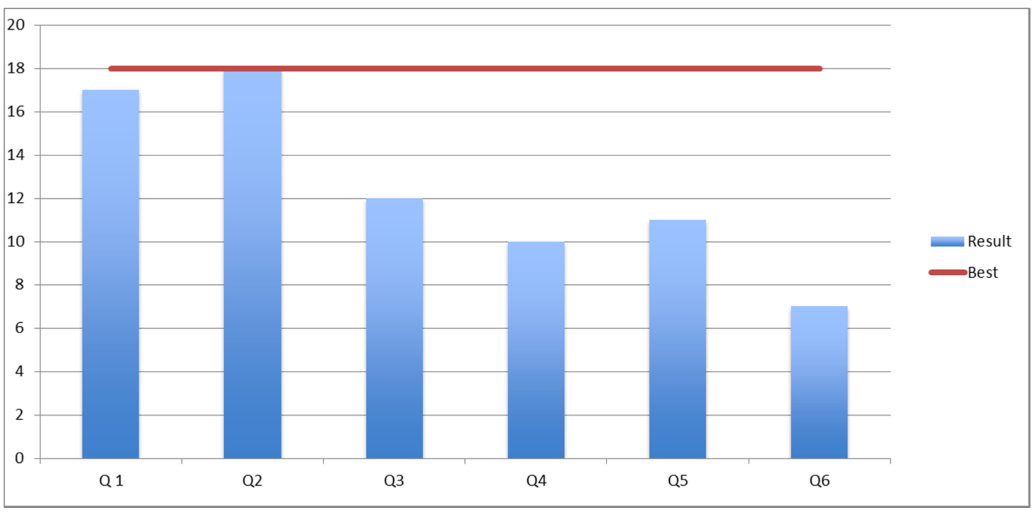


Figure 75: Pilot study – Technical questions results

The aggregated results in Figure 75 show that the participants had better scores in the first few questions, where the concepts were explained with examples and where guidelines and hints were provided. As the participants reached the questions that were not used in existing examples and did not have enough guidelines, they were not able to gain as high scores as they did in the first few questions. In fact, the aggregated score has gone down towards the end. Hence, we conclude that the notation and concepts require explanation and examples for users to be able to use them.

When the participants were asked about their opinions, only one of the participants felt that the notation is hard to understand, where three other participants suggested that the notation is not self-explanatory. This vote supports the aforementioned conclusion based on participants' answers to the technical questions to some extent, but it is slightly more positive. Table 25 shows the number of participants who voted for each choice in this set of multiple choice questions while Figure 76 shows the aggregate results. These aggregate results were calculated by multiplying the number of responses for each choice by the choice number. For all questions except for question 7.4, the higher aggregate value is in favor of the proposed approach.

Table 25: Pilot study – Opinion questions results

Questions (Appendix A)	1-Strongly Disagree	2-Disagree	3-No Opinion	4-Agree	5-Strongly Agree
7.1. Easy to understand?		1	1	4	
7.2. Self-explanatory?	1	2		3	
7.3. Has other applications?			3	2	1
7.4. Simpler ways to achieve the same result?		2	4		

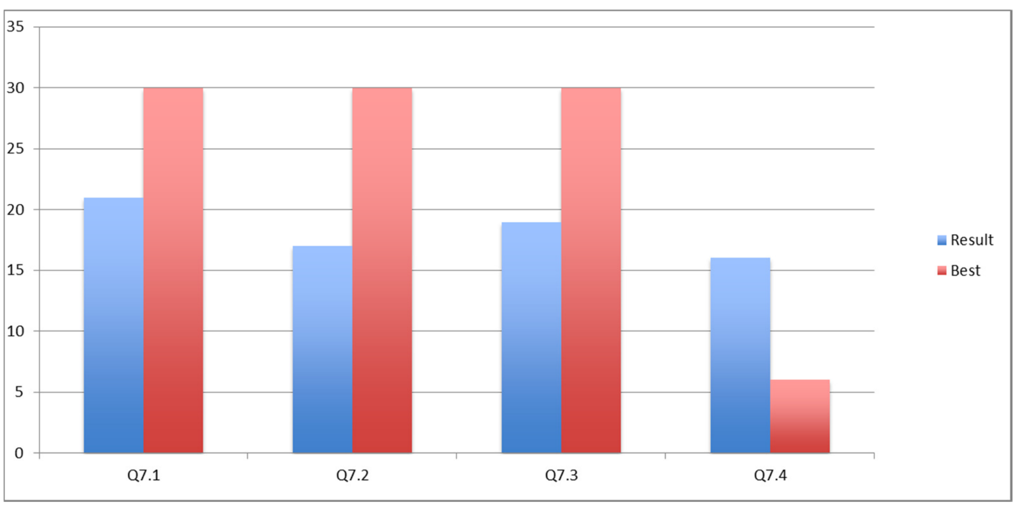


Figure 76: Pilot study – Opinion questions results aggregation

9.4.4 Threats to Validity

This pilot experiment was done to guide the research, do an initial validation, and learn from the process for a future, more comprehensive study. The following are important threats to the validity of this experiment that should be mitigated in the future.

- Sample group size: Only 6 participants were used in this sample group, which is not large enough for the results to be statistically significant. A larger sample size should be used in the future study.
- Participants' skill set: The participants' skill set in the domain (Figure 73 and Figure 74) can be considered too high even if the target audience for this work are business process engineers. Even though most of those users would have knowledge of various process modeling notations, assuming that they are familiar with URN and AoURN is too optimistic. The future study should include users without any knowledge of URN and AoURN so that the results can be generalized.
- Researcher bias: The technical questions are open ended and descriptive, which opens them to interpretation. Since the scoring is done by the researcher, there could be a risk of bias. In the future study, the questions could be designed as multiple-choice with specific answers to reduce this risk.

9.5. Tool Support

To validate the possibility of using the enhanced performance view, a prototype of the formula-based algorithm was implemented by extending the current tool, jUCMNav. The implementation was tested in simple test scenarios (e.g., Figure 31) as well as in the retail real-life example (Figure 58). As this algorithm has now been available in jUCMNav for 4 years[1], it has also been used by many others, including by students in a graduate course at the University of Ottawa, and in Johari's thesis [66]. Although the implementation performs the evaluations as expected and demonstrates that the framework is realizable, since formulas are defined at the model level, it is very hard for analysts to investigate different scenarios using different formulas with the same model. At this point, the model file has to be saved in a different version if different formulas are required with the same model. This issue can be addressed by moving the formulas definition to the GRL strate-

[1] http://jucmnav.softwareengineering.ca/ucm/bin/view/ProjetSEG/JUCMNavRelease420#New_and_Improved_GRL_Propagation

gies so that one model can have different formulas defined as the relationship of the various model elements.

9.6. Chapter Summary

This chapter summarizes the validation that was done throughout this research and used to recalibrate the research direction and outcome. The following list recalls the five validation approaches used to evaluate the research from various perspectives:

- *Systematic literature review*: A repeatable, step-by-step literature research was used to identify the state of the art in the application of aspect-oriented techniques for business process adaptation. A comparison of our work with methods from 60 papers highlighted the originality and suitability of our framework.
- *Mandatory elements of method*: This structured method was used to improve the framework (in a first iteration) and then validate that it satisfies the requirements of a business process improvement method.
- *Real life examples*: Two real-life examples from healthcare and retail sectors were used to illustrate that the framework can be used and be beneficial in various domains.
- *Pilot experiment*: A pilot experiment involving 6 URN users was used as an initial evaluation of the usability of the notation, highlighting good points and points to improve (including point about future studies).
- *Tool support implementation*: The formula-based algorithm implementation in jUCMNav validated the proposed GRL evaluation algorithm as one of the important parts of the framework.

Chapter 10. Conclusions

The final chapter of this thesis summarizes its contributions, the validation outcomes and threats, and future work items.

10.1. Contributions

The contributions of the thesis are in two groups, major and minor, and are captured respectively in Table 26 and Table 27.

Table 26**:** Major thesis contributions

Deliverable	**Value**
An iterative and incremental **framework** for bringing organizations to a state that allows them to monitor and improve their business processes. See Chapter 4	Organizations can use this framework to: • Better understand how their business operates; • Monitor their goals, KPIs, processes; and • Improve their processes. Researchers can use this framework as a stepping stone to progress their research towards a more comprehensive automated process improvement framework.

Deliverable	Value
An **aspect-oriented approach** that allows organizations to detect the potential problems in their business processes, business goals, and performance models using predefined generic or custom **redesign patterns** and to apply the predefined solutions to those problems accordingly. See Chapter 5	Allows organizations to save time and move towards a better business state by using a predefined library of best practices. More specifically, using AoURN allows for analysts to find repeated patterns in a given context and bundle the integration between goal, performance and process models as a self-contained unit. This enables the modeling of complex circumstances involving goal, process, and performance views together with analysis techniques for what-if analysis and process adaptation, all using a unified and integrated modeling language. Allows researchers to consider potential new applications of aspect-oriented modeling and cross-language pointcut expressions, which open up new research directions.
A method for defining **relationships between the Key Performance Indicators** of organizations on performance models using mathematical formulas. See Chapter 6	Allows organizations to: • Combine the benefits of visualization and of accuracy of mathematical formulas; and • Use predefined industry standards or custom formulas for their KPIs.
A new **evaluation algorithm** for URN's Goal-oriented Requirement Language, which allows the evaluation of the models defined using the approach mentioned above. See section 6.3	Allows organizations to: • Evaluate their performance models based on their defined formulas and see the impact of their performance indicators and defined relationships on their business objectives; and • Observe the impact of defined situations on their business objectives. On the research front, this innovative work contributes to the goal modeling body of knowledge and may inspire the creation of other evaluation algorithms.

Deliverable	Value
Modeling of several generic process **redesign patterns** using AoURN. See section 5.3	Allows organizations to: • Use the modeled patterns as is or customize them according to their business needs; and • Use the patterns as examples to model their own custom patterns for their business. Allows researchers to use these patterns as examples to: • Model more generic patterns in the same modeling language; • Model the same generic patterns in other languages; and • Investigate the requirements for other languages to support a similar approach.
Identification of **requirements for a process modeling language** to capture the models to support the aforementioned approach. See section 5.5	Allows researchers who are focused on modeling languages other than AoURN to use the same requirements to enhance their modeling language of interest.
A set of **proposed changes to AoURN** as a profile called *Business Process Pattern (BPP) Profile* to realize the discussed requirements. See section 5.6	Enhances AoURN to not only be a competitive modeling language in the process improvement domain but also to advance the aspect-oriented modeling capabilities of the language in a way that can be used in other domains.

Table 27**:** Minor thesis contributions

Deliverable	Value
Comparison of several modeling languages using a set of relevant criteria. See section 2.4.1	Can be used by other researchers interested in business process modeling and management.
Repeatable systematic review on aspect-oriented	The current results can be used by other researchers interested in business process and aspect-oriented domains. The

Deliverable	Value
methods applied to business process adaptation. See Chapter 3	availability of the methodology enables the repetition of the review in the future, to update the results.
Models of processes, goals and indicators for two real-life examples. See Chapter 7and Chapter 8	Can be used by businesses and researchers who are interested in the framework as examples and guidelines to better learn the method and understand its potential applications.
Evaluation of the framework based on *Mandatory Elements of a Method*. See 9.2	Improved the outcome of this research through an evaluation against external criteria.

Many of the major contributions can be used in isolation: the new performance view and the support for KPI aggregation does not depend on the BPP profile, and the profile does not necessarily rely on the KPI aggregation either. KPI aggregation can be used in any domain, and many enhancements coming with BPP can be used in multiple domains as well. However, to support a framework for monitoring and adapting business processes, BPP and KPI aggregation, together with redesign patterns, bring synergy to the table and essentially have to come together.

In terms of other potential impact (beyond the many publications already extracted from this thesis), this thesis brings new opportunities for the automatic and dynamic adaptation of business processes. The framework also highlights the need and the usefulness of heterogeneous pointcut expressions by exploiting aspects that combine goals, scenarios, and indicators. Before this work, nobody had combined such views in aspects. Many of the new enhancements to URN (e.g., raw data element, situations, and repetition/option stubs) and AoURN (e.g., absent stub) proposed in the profile could also eventually become new first-class concepts in URN (ITU-T Z.151 [62]), and provide further arguments for the formal support of aspects in the standard.

10.2. Validation Outcomes

The research in this thesis was validated using an iterative and incremental method adapted from the design science research method. The validation steps were used in a feedback loop to build the framework and they improved the outcomes of the research along the way. Table 28 summarizes the validation methods used in this research and their outcomes, and highlights major threats to validity.

Table 28**:** Validation approaches and their outcomes

Validation Methods	**Description, Outcomes, and Threats**
Systematic literature review	A repeatable, step-by-step literature research involving 60 papers was used to identify the state of the art in the application of aspect-oriented techniques for business process adaptation. The outcomes of this validation method are: • The thesis framework is novel because we did not find any other similar method using aspect-oriented approach to improve business process, goal, and performance models at the same time. • Much of the research has a narrow focus on adaptation at the process execution level and instance-level handling of faults and exception cases. • The underlying research in this area, at a process execution level, makes us confident that our framework is realizable in practice. • The learning curve in aspect-oriented approaches could be (too) high. • Interaction between aspects when multiple aspects are applied to a process could increase system complexity. Important threats to validity, already discussed in section 3.4 together with their mitigations, include: • The scope of the research might not have been wide enough. This could further be mitigated in the future with more abstract queries and a larger selection of papers. • The selection of publications could have been biased. This could be further mitigated by having different people do the selection and cross-check their results, before the analysis.

Validation Methods	Description, Outcomes, and Threats
Mandatory elements of method (MEM)	A structured method to validate that the proposed framework satisfies the requirements of a business process improvement method. The outcomes of this validation method are: • In the first validation iteration, two missing mandatory elements were detected: definition of roles, and information model. • The missing elements were added to the framework and therefore the framework was identified as compliant in the second validation iteration. Some important threats to validity include: • The evaluation was solely done by the thesis author (internal validity). The assessment could be redone by other, neutral people. • Zellner's MEM approach [155] is still new and may not be perfect. However, as of December 2013, it has already been cited 27 times in 2 years, which is a good sign of popularity.
Real life examples	Two real-life examples from healthcare and retail sectors were used to illustrate that the framework can be used and be beneficial in various domains. The outcomes of this validation method are: • The framework can be applied to real-life examples from different domains. • While we were working on the healthcare example, we came to the conclusion that if we rely on the existing pattern matching approach, the generic patterns will either not match due to missing elements in the base model or have to be modeled without specific details, which would cause many false positives. Therefore, generic patterns need to be customized for the context where they are being used. • The learning from first the example was used in a feedback loop to the framework and the partial matching approach (section 5.4) was added, which not only addresses the issues mentioned above but also provides an added value for the organization by identifying the missing elements in their model and helping them to improve their models for better monitoring.

Validation Methods	Description, Outcomes, and Threats
	• The second example shows the value of enhanced performance view as well as the partial pattern matching. Some important threats to validity include: • In terms of generalization, only two domains were covered, and further case studies from these domains and others would be beneficial. • The feedback collected from the industrial partners was positive but very informal, and not statistically significant.
Pilot experiment	A pilot experiment with 6 participants to perform an initial study on the usability of the notation and patterns with the following outcome: • The results of the pilot experiment suggest that when the notation concepts and patterns are explained for the users, it can be understood and used to complete other similar tasks. • The notation is not self-explanatory and requires guides and documentation to be used even by people who are familiar in URN. • Given the sample size and informal nature of the experiment, a more formal and comprehensive study is required in the future to allow any form of reliable generalization. Important threats to validity, already discussed in section 9.4.4 together with their mitigations, include: • The small sample size; a larger group is needed. • The participant's skill set, which needs to be more varied. • Researcher bias (internal validity); questions should be multiple-choice, or marked by someone else.
Implementation	The formula-based algorithm implementation had the following outcomes: • The proof-of-concept implementation shows that the proposed evaluation approach works as expected and can address the intended use cases. • The current implementation relies on adding the formu-

Validation Methods	Description, Outcomes, and Threats
	las to the model level, which prevents one from using various formulas using the same model. This implementation needs to be improved in the future to address what-if scenario analysis. Some important threats to validity include: • We have not done formal testing, although the tool has been successfully used for 4 years. • The functions provided by the math library we selected [82] may not be sufficient.

10.3. Future Work

The research that was done in this thesis provides enough value to be used as is. However, there are some areas of the research that can be further enhanced to mitigate the threats identified in the previous section. In addition, this research opens up some new research directions and opportunities. The following subsections describe a list of potential future research areas and enhancements.

10.3.1 Framework

The current framework only works at the model level. The framework can be extended to support monitoring and adaptation of the processes at all layers of a BPMS, including the process execution layer and process instances. The extension to accommodate the execution layer would allow the analysts to trigger the approved changes to be automatically applied to the rest of the system. Furthermore, gathering information about process instances will allow analysts to use more design patterns that require such information. In addition, process-mining approaches [142] can be used in the analysis lifecycle on instance level logs and events in order for analysts to discover the process instances that are not compliant with the process models with respect to steps or expected behavior, which is another area that organizations can improve upon. Process-mining methods can also be used to extract the initial version of the process models based on what happens in reality and to gain an initial understanding of the organization behavior.

Furthermore, given the wide adoption of balanced scorecards (BSS) and Strategy Maps (SM) [69], and given prior attempts to combine goal modeling with scorecards [133], an area of interest for future research is about combining the framework with BSS and SM. Such combination could help improve the adoption rate of the framework by providing an entry point for those organizations that already use BSS and SM.

10.3.2 Notation Complexity

Our systematic literature review, our pilot study, and the sample patterns that were modeled in the thesis show that the models and notation can be complex and that the complexity of aspect-oriented concepts and notations can be hard to grasp for users. We believe this study exposed this issue and can help improve AoURN and other languages in the future. This issue can be addressed in several complementary ways:

- By performing a formal usability study on the modeling notation in order to improve the usability of the modeling language. The observations made during our pilot study will help construct such a study.
- By using visual notation evaluation frameworks such as Moody's Physics of Notations [90].
- By hiding the complexity of aspect-oriented models behind a layer of abstraction and by simplifying the interaction of the users with the system.

Each of the above approaches represents interesting research areas from conceptual and technological points of view.

10.3.3 Integration with Languages other than AoURN

Many of the concepts and ideas put forward by this thesis are applicable outside the URN/AoURN space, and hence researchers not working with AoURN can still exploit some of our results.

For example, many languages in Table 2 have support for goals, processes or performance, but not all three. Some also support aspects. The approaches in Table 8 support aspects and adaptation, but not necessarily patterns. Our framework provides motivation and rationale for augmenting or combining some of these approaches to cover goals, processes, performance, aspects, patterns and adaptation. More punctual and small scale improvements are also possible; for example, Business Intelligence Model (BIM) [57][58] can easily be extended to support our performance view and algorithm for KPI aggregation. Furthermore, there may be opportunities to use some of the concepts in other languages to complement the ideas presented in this thesis. For instance, BIM's *situations* have significant overlap with the situation concept introduced in this thesis and perhaps even have a broader coverage by considering all SWOT (Strengths, Weaknesses, Opportunities, and Threats) situations as opposed to only Opportunities and Threats.

In addition, given the evident complexity and learning curve of AoURN, we would like to explore other options for detecting and applying patterns, either visually or textually.

Finally, one could use our requirements for language X (Table 13) to enhance an existing language or to create one from scratch.

10.3.4 Implementation

On the implementation front, there are several opportunities to improve tool support that can be explored:

- The AoURN pattern matching algorithm requires a major enhancement to support the concepts and capabilities proposed in this thesis. The most important improvement that is not currently supported is a coordinated matching algorithm between AoGRL and AoUCM. Implementing the coordinated pattern matching will provide tool support for patterns that rely on both goal and process models. Other enhancements that were proposed in this framework are the partial, multiple, and detection-only matching approaches that have to be implemented as well. The details of the requirements for the improved algorithm are available in Table 14.
- Furthermore, the current pattern matching relies on textual and lexical wildcard matching, which may not be practical in all situations. The matching engine can be augmented with concept matching approaches using knowledge bases or ontologies to detect concepts rather than exact lexical match. For instance, the concept of time can be matched regardless of the name used for a KPI.

- The BPP (Business Process Pattern) profile was proposed to implement the language X requirements in AoURN. This profile needs to be implemented in the supporting tool (i.e., jUCMNav). The tool already supports the stereotypes and metadata input, and the well-formedness constraints from Appendix B can be formalized as OCL rules [11], but the new graphical symbols are not yet supported.
- In the current proof-of-concept implementation, the formulas are defined at the model level in GRL. Moving the formula definitions to the GRL strategies so that one model can have different formulas defined as the relationship of the various model elements can improve the flexibility and analysis power of the framework.
- With the current supporting tool, creating snapshots for trend and decision trail analysis is only doable by saving versions of the models. This approach is not effective and does not scale. Supporting model versioning and snapshots will improve the usability of the tool not only for the framework but also for other areas of research.

10.3.5 Integration with BI

An important challenge for the integration of this framework with BI tools is the alignment between the performance model and BI data sources. A very interesting research area that can be explored in the future concerns building the required BI artefacts from the performance model and vice-versa, automatically. This approach would simplify the integration with BI tools significantly.

10.3.6 Validation Case Studies and Scalability

Only two major case studies were performed so far. Although they are from different domains, there are many other domains and types of processes that could be explored, especially as some might highlight the need for other or more customizable patterns.

One of the major areas that need further validation is the scalability of the approach on all fronts, including the framework, the notation, the visual presentation of large models in the tool, and the algorithm performance.

References

[1] Aalst, W.M.P. van der and Hofstede, A.H.M ter: "YAWL: yet another workflow language". *Information Systems*, vol. 30, 2005, pp. 245–275 (2005)

[2] Abdelaziz, T., Elammari, M., and Unland, R.: "Visualizing a Multiagent-Based Medical Diagnosis System Using a Methodology Based on Use Case Maps". Lindemann-v. Trzebiatowski, G., Denzinger, J., Timm, I.J., and Unland, R. (Editors). In: *Multiagent System Technologies (MATES 2004)*, Springer, LNCS 3187, pp. 198–212 (September 2004)

[3] Alencar, F., Moreira, A. M. D., Araújo, J., Castro, J., Silva, C., and Mylopoulos, J.: "Towards an Approach to Integrate *i** with Aspects". In: *Eight International Bi-Conference Workshop on Agent Oriented Information System (AOIS'06)*, Luxembourg (2011)

[4] Alexander, C.: *A Pattern Language.* Oxford University Press (1979)

[5] Algahtani, A., and Zedan, H.: "Aspectual Interactions for Adaptive Behavioral Web-Services with tailored Maude-based Certification". In: *Proc. of E-Business and Information System Security (EBISS'09)*, Wuhan, China, pp. 1–15 (May 2009)

[6] Amyot, D.: "Introduction to the User Requirements Notation: Learning by Example". *Computer Networks*, Vol. 42(3), Elsevier, pp. 285–301 (June 2003)

[7] Amyot, D., Ghanavati, S., Horkoff, J., Mussbacher, G., Peyton, L., and Yu, E.: "Evaluating Goal Models within the Goal-oriented Requirement Language". *International Journal of Intelligent Systems (IJIS)*, John Wiley & Sons, Vol. 25, Issue 8, pp. 841–877 (August 2010)

[8] Amyot, D., Horkoff, J., Gross, D., and Mussbacher, G.: "A Lightweight GRL Profile for *i** Modeling". In: *3rd Int. Workshop on Requirements, Intentions and Goals in Conceptual Modeling (RIGiM 2009)*, ER Workshops, Springer, LNCS 5833, pp. 254–264 (November 2009)

[9] Amyot, D., and Mussbacher, G.: "User Requirements Notation: The First Ten Years, The Next Ten Years". Invited paper, *Journal of Software (JSW)*, Academy Publisher , Vol. 6, No. 5, pp. 747–768 (May 2011)

[10] Amyot, D., Rashidi-Tabrizi, R., Mussbacher, G., Kealey, J., Tremblay, E., and Horkoff, J.: "Improved GRL Modeling and Analysis with jUCMNav 5". In: *6th International i* Workshop (iStar 2013)*, Valencia, Spain, CEUR-WS, Vol-978, pp. 137–139 (June 2013)

[11] Amyot, D., and Yan, J.B.: "Flexible Verification of User-Defined Semantic Constraints in bbing Tools". In: *18th Int. Conf. of Computer Science and Software Engineering (CASCON 2008)*, Toronto, Canada, ACM, pp.81–95 (2008)

[12] Ansar, Y., Giorgini, P., Ciancarini, P., Moretti, R., Sebastianis, M., and Zannone, N.: "Evaluation of Business Solutions in Manufacturing Enterprises". *International Journal of Business Intelligence and Data Mining*. Vol. 3, Issue 3, pp. 305–329 (December 2008)

[13] Babar, A., Zowghi, D., and Chew, E.: "Using Goals to Model Strategy Map for Business IT Alignment". In: *International workshop on Business/IT Alignment and Interoperability (BUSITAL10)*, in conjunction with CAiSE'10, Hammamet, Tunisia, CEUR Workshop Proceedings, Vol-599*,* pp. 16–31 (June 2010)

[14] Benbasat, I., and Dexter, A.S.: "An experimental study of the human/computer interface". *Communications of the ACM Magazine*, ACM, Vol. 24, Issue 11, pp. 752–762 (November 1981)

[15] Benbasat, I., and Dexter A.S.: "Individual differences in the use of decision support aids". *Journal of Accounting Research*, Vol. 20, Issue 1, pp. 1–11 (Spring 1982)

[16] Benbasat, I., Dexter, A.S., and Todd, P.: "An experimental program investigation color-enhanced and graphical information presentation: an integration of the findings". *Communications of the ACM Magazine*, ACM, Vol. 29, Issue 11, pp. 1094–1105 (November 1986)

[17] Benbasat, I., and Nault, B.R.: "An evaluation of empirical research in managerial support systems". *Decision Support Systems*, Vol. 6, Issue 3, pp. 203–226 (August 1990)

[18] Birukou, A., and Weiss, M.: "A service for software pattern selection". In: *14th European Conference on Pattern Languages of Programs (EuroPLoP 2009)*, Bavaria, Germany, CEUR, Vol-566 (July 2009)

[19] Bresciani, P., Perini, A., Giorgini, P., Giunchiglia, F., Mylopoulos, J.: "Tropos: An agent-oriented software development methodology". *Autonomous Agents and Multi-Agent Systems*, Vol. 8 Issue 3, pp. 203–236 (2004)

[20] Bruce Silver Associates (2009) *BPMS Watch Ratings for Q2 2008*, January. http://www.bpminstitute.org/bpmsreport.html (accessed December 2011)

[21] Bruggen, G.H., Smidts, A., and Vierenga, B.: "Improving decision making by means of a marketing decision support". *Management Science*, Vol. 44, Issue 5, pp. 645–658 (May 1998)

[22] Cappelli, C., Leite, J. C., Batista, T., and Silva, L.: "An aspect-oriented approach to business process modeling". In: *Proceedings of the 15th Workshop on Early Aspects (EA'09)*, Charlottesville, USA, ACM, pp.7–12 (March 2009)

[23] Charfi, A., and Mezini, M.: "An Aspect-Based Process Container for BPEL". In: *Proc. of the First Workshop on Aspect-Oriented Middleware Development (AOMD05)*, Grenoble, France, ACM (November-December 2005)

[24] Charfi, A., Dinkelaker, T., and Mezini, M.: "A plug-in architecture for self-adaptive web service compositions". In: *Proc. of the 2009 IEEE International Conference on Web Services (ICWS'09)*, IEEE CS, pp. 35–42 (July 2009)

[25] Chen, P.: *Goal-oriented business process monitoring: An approach based on user requirement notation combined with business intelligence and web services.* M.C.S. thesis, Computer Science, Carleton University, Canada (2008)

[26] Cheng, B.H.C, De Lemos, R., Giese, H., Inverardi, P., and Magee, J.: "Software engineering for self-adaptive systems: a research roadmap". In: *Software Engineering for Self-Adaptive Systems*, Springer-Verlag Berlin Heidelberg, LNCS, Vol. 5525, pp. 1–26 (2009)

[27] Chitchyan, R. *et al.*: "Survey of Analysis and Design Approaches". *AOSD-Europe Report ULANC-9*, http://www.aosd-europe.net/deliverables/d11.pdf, 259 pages (May 2005)

[28] Chuck, B., Ahmed, A.H., Robert, F., Fabio, H., Julio, L., Pietro, L., and Jo. R.: *Improving Business Performance Insight ... with Business Intelligence and Business Process Management*, IBM Redbooks publication (August 2006)

[29] Colombo, E., and Mylopoulos, J.: "A Multi-perspective Framework for Organizational Patterns". In: *25th International Conference on Conceptual Modeling (ER06)*, Tucson, AZ, USA, Springer, Heidelberg, LNCS, Vol. 4215, pp. 451–467 (2006)

[30] Courbis, C., and Finkelstein, A.: "Towards aspect weaving applications". In: *Proc. of the 27th International Conference on Software Engineering (ICSE '05)*, St. Louis, Missouri, USA, pp. 69–77 (May 2005)

[31] Cui, Z., Wang, L., Li, X., & Xu, D.: "Modeling and integrating aspects with UML activity diagrams". In: *Proceedings of the 2009 ACM symposium on Applied Computing*, ACM, pp. 430–437 (2009)

[32] Conforti, R., Hofstede, A.H.M., Rosa, M., Adams, M.: "Automated Risk Mitigation in Business Processes" In: *On the Move to Meaningful Internet Systems: OTM 2012*, LNCS 7565, Springer, pp. 212–231 (2012)

[33] Dalpiaz, F., Barone, D., Horkoff, J., Jiang, L., and Mylopoulos, J.: "BIM-Tool: Modeling and Reasoning Support for Strategic Business Models". In: 6th Int. *i** Workshop (iStar 2013), CEUR-WS, Vol. 978, pp. 134–136. Online tool: http://www.cs.toronto.edu/~jm/bim/ (2013)

[34] Dardenne, A., van Lamsweerde, A., and Fickas, S.: "Goal Directed Requirements Acquisition". *Science of Computer Programming*. Vol. 20, pp. 3–50 (1993)

[35] Davis, R., and Brabänder, E.: *ARIS Design Platform: Getting Started with BPM*. Springer (June 2007)

[36] De Sanctis, G.: "Computer graphics as decision aids: Directions for research". *Decision Sciences*, Vol. 15, Issue 4, pp. 463–487 (October 1984)

[37] Debevoise, T.: "Business Process Management with a Business Rules Approach: Implementing the Service Oriented Architecture". *Business Knowledge Architects*, Canada (June 2005)

[38] *Early Aspects: Aspect-Oriented Requirements Engineering and Architecture Design*, http://www.early-aspects.net/ (accessed December 2011)

[39] Eden, A.H., Yehudai, A., and Gil, J. "Precise specification and automatic application of design patterns". In: *12th IEEE Int. Conf. Automated Software Engineering*, pp. 143–152 (1997)

[40] El Boussaidi, G., Mili, H.: "Understanding design patterns — what is the problem?: *Software Practices and Experience*, Vol 42, Issue 12, pp. 1495–1529 (2012)

[41] Eleonu, H., Oruh, J.: "Adaptive Process in a Pervasive System – A Holistic Hybrid Approach. *IJCSI International Journal of Computer Science*, Vol 10, Issue 3, No 2, pp. 208–217 (2013)

[42] Erradi, A., Maheshwari, P., and Tosic, V.: "Policy-Driven Middleware for Self-adaptation of Web Services *Com*positions". In: *Proc. of the 7th International Middleware Conference (Middleware)*, Melbourne, Australia, LNCS, Vol. 4290, pp. 62–80 (November-December 2006)

[43] Erradi, A., Maheshwari, P., and Padmanabhuni, S.: "Towards a policy driven framework for adaptive web services composition". In: *Proc. of Next Generation Web Services Practices (NWeSP)*, Seoul, Korea, IEEE, pp. 261–266 (August 2005)

[44] Falck, M., and Karlsson, F.: *Key Performance Indicators, The key to success?, A study of how KPIs are used at Intersport*. Master thesis, Tutor Klas Gäre, Jönköping International Business School, Sweden (May 2011)

[45] Feng, X., Richards, G., and Raheemi, B.: "The Road to Decision-Centric Business Intelligence". In: *2nd Int. Conf. on Business Intelligence and Financial Engineering (BIFE 2009)*, Beijing, China, IEEE CS, pp. 514–518 (July 2009)

[46] Forster, F.: "The Idea behind Business Process Improvement: Toward a Business Process Improvement Pattern Framework". *BPTrends*, http://www.bptrends.com/publicationfiles/04-06-ART-PatternFramework-Forster.pdf (April 2006)

[47] Gamma, E., Helm. R, Johnson, R., and Vlissides. J.: *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley (1994)

[48] Gartner, EBRC: *Gartner/EBRC KPI Initiative*, http://www.aicpa.org/InterestAreas/FRC/AccountingFinancialReporting/EnhancedBusinessReporting/Pages/GartnerEBRCKPIInitiative.aspx, July, 2013

[49] Gartner, EBRC: "Standard KPIs Taxonomies". http://tinyurl.com/njzgu3f

[50] Graml, T., Brachtand, R., and Spies, M.: "Patterns of Business Rules to Enable Agile Business Processes". In: *Proc. of the 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC07)*, Annapolis, Maryland, USA, IEEE Computer Society, pp. 365–375 (October 2007)

[51] Hackathorn, R.: "Making Business Intelligence Actionable". *DM Review*, 32 (2002)

[52] Heβ, H.: "From Corporate Strategy to Process Performance – What Comes after Business Intelligence?". *Corporate Performance Management*., Springer (2006)

[53] Hermosillo, G., Seinturier, L., and Duchien, L.: "Using Complex Event Processing for Dynamic Business Process Adaptation". In: *Proc. of the 2010 IEEE International Conference on Services Computing*, Miami, Florida, USA, IEEE Explore, pp. 466–473 (July 2010)

[54] Hermosillo, G. *Towards Creating Context-Aware Dynamically-Adaptable Business Processes Using Complex Event Processing*, PhD Thesis, Université de Lille Science et Technologies, France (2012)

[55] Henver, A.R., March, A.T., Park, J., and Ram, S.: "Design science in information systems research". *MIS Quarterly*, Vol. 28, No.1, pp. 75–105 (March 2004)

[56] Hermosillo, G., Seinturier, L., and Duchien, L.: "Creating Context-Adaptive Business Processes". In: *Proc. of the 2010 IEEE International Conference on Services Computing*, Miami, Florida, USA, IEEE Explore, pp. 228–242 (July 2010)

[57] Horkoff, J., Barone, D., Jiang, L., Yu, E., Amyot, D., Borgida, A., and Mylopoulos, J.: "Strategic Business Modeling - Representation and Reasoning". *Software and Systems Modeling (SoSym)*, to appear (Online first: October 26, 2012)

[58] Horkoff, J., Borgida, A., Mylopoulos, J., Barone, D., Jiang, L., Yu, E., and Amyot, D.: "Making Data Meaningful: The Business Intelligence Model and its Formal Semantics in Description Logics". *Proc. 11th Int. Conf. on Ontologies, DataBases, and Applications of Semantics (ODBASE 2012)*, LNCS 7565, Springer, pp. 700–717 (2012)

[59] Horkoff, J. and Yu, E: "Comparison and evaluation of goal-oriented satisfaction analysis techniques". *Requirements Engineering*, 18(3), Springer, pp. 1–24 (2012)

[60] Huang, W., Eades, P., and Hong, SH. "Measuring effectiveness of graph visualizations: A cognitive load perspective". *Information Visualization*, ACM, Vol. 8 Issue 3, pp. 139–152 (June 2009)

[61] IBM: *Monitor, manage and perform,* http://www-01.ibm.com/software/analytics/business-intelligence/scorecards/ (accessed September 2013)

[62] ITU-T, Recommendation Z.151 (10/12) *User Requirements Notation (URN) – Language definition*, Geneva, Switzerland (2012)

[63] Isik, O., Jones, M.C., and Sidorova, A.: "Business Intelligence Success: An Empirical Evaluation of the Role of BI Capabilities and the Decision Environment". *Business Intelligence Congress II*, Virginia, USA (August 2010)

[64] Jarvenpaa, S.L.: "The effect of task demands and graphical format on information processing strategies". *Management Science*, ACM, Vol. 35, Issue 3, pp. 285–303 (March 1989)

[65] Jiang, L., Barone, D., Amyot, D., and Mylopoulos, J.: "Composite Indicators for Business Intelligence". In: *Proceedings of the 30th international conference on Conceptual modeling (ER'11)*, Brussels, Belgium, Springer-Verlag Berlin, Heidelberg, pp. 448–458 (October-November 2010)

[66] Johari, I.: *Combining Business Intelligence, Indicators, and the User Requirements Notation for Performance Monitoring*. M.Sc. thesis, Systems Science, University of Ottawa, Canada (November 2012)

[67] Johnson-Laird, P.N. *The Computer and the Mind: An introduction to cognitive science*. Cambridge: Harvard University Press (April 1988)

[68] *jUCMNav*, version 5.4, University of Ottawa. Canada, http://softwareengineering.ca/jucmnav (accessed September 2013)

[69] Kaplan, R. S., & Norton, D. P.: "Having trouble with your strategy?: Then map it.". In: *Focusing Your Organization on Strategy – with the Balanced Score-card*, 2nd Edition, Harvard Business School Publishing Corporation, pp. 49 – 61 (2013)

[70] Kapuruge, M., Hanand, J., and Colman, A.: "Support for business process flexibility in service compositions: An evaluative survey". In: *Proc. of the 21st Australian Software Engineering Conference (ASWEC10)*, Hyatt Regency Auckland, New Zealand, IEEE Xplore, pp. 97–106 (April 2010)

[71] Kazhamiakin, R., Pistore, M., and Zengin, A.: "Cross-Layer Adaptation and Monitoring of Service-Based Applications". In: *Proc. of ICSOC/ServiceWave*, Stockholm, Sweden, Springer-Verlag Berlin, Heidelberg, LNCS, Vol. 6275, pp. 325–334 (November 2009)

[72] Kazhamiakin, R., Benbernou, S., Baresi, L., Plebani, P., Uhligand, M., and Barais, O.: "Adaptation of Service-Based Systems". In: *Proc. of Service Research Challenges and Solutions for the Future Internet*, Springer Berlin Heidelberg, LNCS, Vol. 6500, pp. 117–156 (2010)

[73] Kalavathy, G.M., Rathinam, N.E., and Seethalakshmi, P.: "Self-adaptable media service architecture for guaranteeing reliable multimedia services". *Multimedia Tools and Applications*, pp. 1–18 (December 2010)

[74] Kiczales, G., Lamping, J., Mendhekar, A., Maeda, C., Lopes, C., Loingtier, J.-M. and Irwin, J.: "Aspect-Oriented Programming". In: *Object Oriented Programming, 11th European Conference (ECOOP'97)*, Jyväskylä, Finland, Springer, LNCS, Vol. 1241, pp. 220–242 (June 1997)

[75] Kimball, R., and Ross, M.: *The Data Warehouse Toolkit*. John Wiley and Sons, Inc. USA (2003)

[76] Ko, I.S., and Abdullaev, S.R.: "A Study on the Aspects of Successful Business Intelligence System Development". In: *Int. Conf. on Computational Science (ICCS 2007)*, Beijing, China, Springer, Heidelberg, LNCS, Vol. 4490, pp. 729–732 (May 2007)

[77] Ko, R.K.L., Lee, S.S.G., and Lee, E.W.: "Business process management (BPM) standards: a survey". *Business Process Management Journal*, Vol. 15, No. 5, pp.744–791 (2009)

[78] Korhonen, P., Mano, H., Stenfors, S., and Wallenius, J.: "Inherent Biases in Decision Support Systems: The Influence of Optimistic and Pessimistic DSS on

Choice, Affect, and Attitudes". *Journal of Behavioral Decision Making*, 21, Vol. 58, pp. 45–58 (2008)

[79] Krogstie, J.: *EEML2005: Extended Enterprise Modeling Language*. Norwegian Unviersity of Science and Technology, Norway (2005)

[80] Kronz, A.: "Managing of Process Key Performance Indicators as Part of the ARIS Methodology". In: *Corporate Performance Management*, Springer-Verlag, pp. 32–44 (2006)

[81] Kitchenham, B.: *Procedures for performing systematic reviews*. Technical Report, Keele University and NICTA, Staffordshire, UK (2004)

[82] Lai, T., com.primalworld.math.MathEvaluator (Math Evaluator), http://goo.gl/z3KIEA (November 2013)

[83] Lian, W., Zeng, Q., Liang, Y., and Yan, J.: "A Framework to Improve Adaptability in Web Service Composition". In: *Proc. of the 2nd International Conference on Computer Engineering and Technology (ICCET)*, Chengdu, China, IEEE Explore, pp. 616–621 (April 2010)

[84] List B., and Korherr, B. "An Evaluation of Conceptual Business Process Modelling Language". *Proc. 21st ACM Symposium on Applied Computing*, ACM, pp. 1332–1539 (2006)

[85] Liu, L., and Yu, E.: "Designing information systems in social context: a goal and scenario modelling approach". *Information Systems*, Vol. 29, No. 2, pp.187–203 (2004)

[86] Lu, Q., "Autonomic Business-Driven Decision Making for Adaptation of Web Service Compositions". In: *Proc. of the 2011 IEEE World Congress on Services*, pp. 73-76, Washington, USA, July 2011.

[87] Lu, R., and Sadiq, S.: "A Survey of Comparative Business Process Modeling Approaches". In: *Business Information Systems (BIS)*, LNCS, Vol. 4439, pp. 82–94 (2007)

[88] Luo, H., Amyot, D.: "Towards a Declarative, Constraint-Oriented Semantics with a Generic Evaluation Algorithm for GRL". In: *5th International i* Workshop*, Trento, Italy, CEUR Workshop Proceedings, Vol-766, pp. 26–31 (August 2011)

[89] Mayer, R., Menzel, C., Painter, M., Witte, P., Blinn, T., and Perakath, B., *Information Integration for Concurrent Engineering - IDEF3 Process Description Capture Method Report*, Knowledge Based Systems, Incorporated, USA (2005)

[90] Moody, D.L.: "The "Physics" of Notations: Toward a Scientific Basis for Constructing Visual Notations in Software Engineering". *IEEE Trans. Software Eng*. 35(6), pp. 756–779 (2009)

[91] Mylopoulos, J., Chung, L., and Yu, E.: "From object-oriented to goal-oriented requirements analysis". *Communications of the ACM*, Volume 42 Issue 1, pp. 31–37 (January 1999)

[92] Muehlen, M.Z., and Léonard, M.: “How Much Language Is Enough? Theoretical and Practical Use of the Business Process Modeling Notation”. In: *CAiSE 2008*, LNCS 5074, Springer, pp. 465–479 (2008)

[93] Mussbacher. G.: *Aspect-oriented User Requirements Notation*. Ph.D. thesis, SITE, University of Ottawa, Canada (November 2010)

[94] Mussbacher, G., and Amyot, D.: “Assessing the Applicability of Use Case Maps for Business Process and Workflow Description”. In: *3rd International MCETECH Conference on e-Technologies*, Montréal, Candada, IEEE CS, pp. 219–222 (January 2008)

[95] Mussbacher, G., Amyot, D., and Weiss, M.: “Formalizing Architectural Patterns with the Goal-oriented Requirement Language”. In: *Proc. Fifth Nordic Pattern Languages of Programs Conference*, pp. 13–36 (2006)

[96] Mussbacher, G., Amyot, D., and Weiss, M.: “Formalizing Patterns with the User Requirements Notation”. T. Taibi (ed.), *Design Pattern Formalization Techniques*, IGI Global, pp. 302–323 (2007)

[97] Mussbacher, G., Barone, D., and Amyot, D., “Towards a Taxonomy of Syntactic and Semantic Matching Mechanisms for Aspect-oriented Modeling”. In: *Proc. 6th Workshop on System Analysis and Modelling (SAM 2010)*, LNCS 6598, Springer, pp. 241–256 (2007)

[98] Mussbacher, G., and Amyot, D.: “On Modeling Interactions of Early Aspects with Goals”. In: *Early Aspects at ICSE 2009: Workshop on Aspect-Oriented Requirements Engineering and Architecture Design*. Vancouver, Canada, IEEE CS, pp. 14–19 (May 2009)

[99] Mussbacher, G., and Amyot, D.: “Extending the User Requirements Notation with Aspect-oriented Concepts”. In: *SDL'09 Proceedings of the 14th international SDL conference on Design for motes and mobiles*, Bochum, Germany, Springer-Verlag Berlin, Heidelberg, LNCS, Vol. 5719, pp. 115–132 (September 2009)

[100] Mussbacher, G., Amyot, D., and Weiss, M.: “Visualizing Early Aspects with Use Case Maps”. *Transactions on Aspect-Oriented Software Development III,* LNCS, Vol. 4620, Springer, pp. 105–143 (2007)

[101] Mussbacher, G., Amyot, D., Whittle, J., and Weiss, M.: “Flexible and Expressive Composition Rules with Aspect-oriented Use Case Maps (AoUCM)”. *Early Aspects: Current Challenges and Future Directions*, LNCS, Vol. 4765, Springer, pp. 19–38 (2007)

[102] Mussbacher, G.: “Aspect-Oriented User Requirements Notation: Aspects in Goal and Scenario Models”. In: *Models in Software Engineering: Workshops and Symposia at MoDELS 2007*, Nashville, USA, LNCS, Vol. 5002, Springer, pp. 305–316 (September-October 2007)

[103] Mussbacher, G., Amyot, D., Araújo, J., Moreira, A., and Weiss, M.: “Visualizing Aspect-Oriented Goal Models with AoGRL”. In: *Second International Workshop on Requirements Engineering Visualization (REV’07)*, New Delhi, India, IEEE, pp. 1–10 (October 2007)

[104] Mussbacher, G., and Amyot, D.: "Goal and scenario modeling, analysis, and transformation with jUCMNav". In: *31st International Conference on Software Engineering (ICSE 2009),* Vancouver, Canada, Companion Volume, IEEE CS, pp. 431-432 (May 2009)

[105] Mussbacher, G., Ghanavati, S., and Amyot, D.: "Modeling and Analysis of URN Goals and Scenarios with jUCMNav". In: *17th IEEE Int. Requirements Engineering Conf*. Atranta, USA. IEEE CS, pp. 383–384 (August-September 2009) http://jucmnav.softwareengineering.ca/jucmnav/

[106] Na, J., Li, G., Zhang, B., Zhang, L., and Zhu, Z.: "An Adaptive Replanning Mechanism for Dependable Service-Based Systems". In: *Proc. of the IEEE 7th International Conference on e-Business Engineering (ICEBE)*, Shanghai, China, IEEE Computer Society, pp. 262–269 (November 2010)

[107] Narendra, N.C., Ponnalagu, K., Krishnamurthy, J., and Ramkumar, R.: "Run-Time Adaptation of Non-functional Properties of Composite Web Services Using Aspect-Oriented Programming". In: *Proc. of the 5th International Conference on Service-Oriented Computing*, Vienna, Austria, Springer Verlag Berlin, pp. 546–557 (September 2007)

[108] Noguera, C., Kellens, A., Deridder, D., and D'Hondt, T.: "Tackling pointcut fragility with dynamic annotations". In: *Proceedings of the 7th Workshop on Reflection, AOP and Meta-Data for Software Evolution* (RAM-SE '10). ACM, pp. 1–6 (2010)

[109] Patiniotakis, I., Papageorgiou, N., Verginadis, Y., Apostolou, D., Mentzas, G.: "An Aspect Oriented Approach for Implementing Situational Driven Adaptation of BPMN2.0 Workflows". In: *6th International Workshop on Event-Driven Business Process Management*, LNBIP, Vol. 132, Springer, pp. 414–425 (2013)

[110] Pedrinaci, C., Domingue, J., and Medeiros, A.: "A Core Ontology for Business Process Analysis". In: *ESWC'08 Proceedings of the 5th European semantic web conference on The semantic web: research and applications*, Tenerife, Spain, LNCS, Vol. 5021, Springer, pp.49–64 (June 2008)

[111] Petri C., and Reisig, W.: "Petri net". *Scholarpedia*, 3(4):6477 doi:10.4249/scholarpedia.6477 (2008)

[112] Popova, V., and Sharpanskykh, A.: "Modeling organizational performance indicators". In: *Information Systems, Vocabularies, Ontologies and Rules for Enterprise and Business Process Modeling and Management*, Elsevier Science Ltd, Vol. 35, Issue 4, pp. 505–527 (June 2010)

[113] Pourshahid, A., Peyton, L., Ghanavati, S., Amyot, D., Chen, P., and Weiss, M.: "Model-Based Validation of Business Processes". Book chapter in: V. Shankararaman, J.L. Zhao and J.K. Lee (Eds) *Business Enterprise, Process, and Technology Management: Models and Applications*, IGI Global, USA, pp. 165–183 (2012)

[114] Pourshahid, A., Chen, P., Amyot, D., Peyton, L., Ghanavati, S., Weiss, M., and Forster, A.: "Business Process Management with User Requirement Notation". *Electronic Commerce Research*, Springer, Vol. 9, No. 4, pp. 269–316 (2009)

[115] Pourshahid, A., Chen, P., Amyot, D., Weiss, M., and Forster, A.: "Business Process Monitoring and Alignment: An Approach Based on the User Requirements Notation and Business Intelligence Tools". In: *10th Workshop on Requirement Engineering (WER'07)*, Toronto, Canada, pp. 80–91 (May 2007)

[116] Pourshahid, A., Richards, G., and Amyot, D.: "Toward a Goal-Oriented, Business Intelligence Decision-Making Framework". In: *E-Technologies: Transformation in a Connected World*, Springer Berlin Heidelberg, LNBI, Vol. 78, pp. 100–115 (2011)

[117] Pourshahid, A., Mussbacher, G., Amyot, D., and Weiss, M.: "Toward an Aspect-Oriented Framework for Business Process Improvement". In: *International Journal of Electronic Business (IJEB)*, Vol. 8, No. 3, pp. 233–259 (2010)

[118] Pourshahid, A., Mussbacher, G., Amyot, D., Shamsaei, A. and Weiss, M.: "A Systematic Review and Assessment of Aspect-oriented Methods Applied to Business Process Adaptation". *Journal of Software (JSW)*, 7(8), Academy Publisher, pp. 1816–1826 (2012)

[119] Pourshahid, A.: *A URN Methodology for Business Process Monitoring*. M.Sc. thesis, E-Business Technology, University of Ottawa, Canada (March 2008)

[120] Ur Rahman, S. S., Aoumeur, N., and Saake, G.: "An adaptive ECA-centric architecture for agile service-based business processes with compliant aspectual .NET environment". In: *Proc. of the 10th International Conference on Information Integration and Web-based Applications & Services (iiWAS'08)*, Linz, Austria, ACM, pp. 240–247 (November 2008)

[121] Ramakrishnan, S.: "On self-adaptive process-based dependable web service composition". In: *Proc. of the 2009 Computation World*, Athens, Greece, IEEE Explore, pp. 173–179 (November 2009)

[122] Reichert, M., Dadam, P.: "A Framework for Dynamic Changes in Workflow Management Systems". In: *Proc. of 8th Int. Workshop on Database and Expert Systems Applications (DEXA'97)*, IEEE CS, Washington, USA, pp. 42–48 (1997)

[123] Reijers, H.: "Process Design and Redesign". Dumas, M., Van der Aalst, W.M.P., Ter Hofstede, A.H.M. (Editors). *Process-Aware Information Systems: Bridging People and Software Through Process Technology,* Chapter 9, John Wiley and Sons, New Jersey, pp. 207–234 (2005)

[124] Reijers, H., and Liman-Mansar, S. "Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics". *Int. J. Manage. Sci.*, Vol. 33, pp. 283–306 (2005)

[125] Roy, J.-F., Kealey, J., and Amyot, D.: "Towards Integrated Tool Support for the User Requirements Notation". In: *System Analysis and Modeling: Language Profiles*, Kaiserslautern, Germany, Springer, LNCS, Vol. 4320, pp. 198–215 (June 2006)

[126] Ruy, S., Benatallah, B., and Casati, F.: "A Framework for Managing the Evolution of Business Protocols in Web Services". In: *Proc. of the Asia-Pacific Confer-*

ence on Conceptual Modelling (APCCM'07), Victoria, Australia, Australian Computer Society, Vol. 67, pp. 49–59 (January-February 2007)

[127] Sawyer, P., Bencomo, N., Hughes, D., Grace, P., Goldsby, H.J., and Cheng, B.H.C.: "Visualizing the Analysis of Dynamically Adaptive Systems Using *i** and DSLs". In: *Proc. of the Second International Workshop on Requirements Engineering Visualization* (REV), New Delhi, India, IEEE Computer Society Washington (October 2007)

[128] Sánchez, M., and Villalobos, J.: "A flexible architecture to build workflows using aspect-oriented concepts". In: *Proc. the 2008 AOSD workshop on Aspect-oriented modeling*, Brussels, Belgium, ACM, pp. 25–30 (March-April 2008)

[129] Shamsaei, A., *Indicator-based Policy Compliance of Business Processes*, PhD Thesis, University of Ottawa, Canada (October 2012)

[130] Shamsaei, A., Pourshahid, A., and Amyot, D.: "Business Process Compliance Tracking Using Key Performance Indicators". In: *Business Process Management Workshops (BPD)*, Hoboken, New Jersey, USA, LNBIP, Vol. 66, pp. 73–84 (September 2010)

[131] Scheer, A.W., Thomas, O., and Adam, O.: "Process modeling using event-driven process chains". *Process-Aware Information Systems*, Wiley, pp. 119–146 (2005)

[132] Siegel, J.: *In OMG's OCEB Certification Program, What is the Definition of Business Process?* (2008) http://www.omg.org/oceb/defbusinessprocess.htm (last accessed February 2012)

[133] Siena, A., Bonetti, A., Giorgini, P.: "Balanced Goalcards - Combining Goal Analysis and Balanced Scorecards". In: *Third Int. Conf. on Evaluation of Novel Approaches to Software Engineering (ENASE 2008)*, Funchal, Madeira, Portugal, INSTICC Press, pp. 107–114 (May 2008)

[134] Silver, B.: *The 2006 BPMS Report: Understanding and Evaluating BPM Suite*. Bruce Silver Associates in collaboration with BPMInstitute.org (2006)

[135] Slovic, P.: "Psychological Study of Human Judgment: Implications for Investment Decision Making". In: *The Journal of Finance*, Vol. 27, No. 4, pp. 779–799 (September 1972)

[136] Stearns. M., and Piccinelli, G.: "Managing Interaction Concerns in Web-Service Systems". In: *Proc. of the 22nd International Conference on Distributed Computing systems Workshop (ICDCSW'02)*, Vienna, Austria, IEEE CS, pp. 424–429 (November 2002)

[137] Subramanian, M., Larry, W., and Hossein, C.: "Business Process Reengineering: A Consolidated Methodology". In: *Proc. 4th Annual Int. Conf. on Industrial Engineering Theory Applications and Practice*, U.S. Department of the Interior, pp. 8–13 (1999)

[138] Suleiman, B., Tosic, V., Jeffery, D.R., and Liu, J.: "Models and algorithms for business value-driven adaptation of business processes and software infrastructure". In: *31st International Conference on Software Engineering (ICSE*

2009), Companion Volume, Vancouver, Canada, IEEE CS, pp. 387–390 (May 2009)

[139] Taibi, T.: *Design Pattern Formalization Techniques*. IGI Global (2007)

[140] Tarr, P., Ossher, H., Harrison, W., and Sutton, S.M.: "N degrees of separation: Multidimensional separation of concerns". In: *21st International Conference on Software Engineering (ICSE 99)*, Los Angeles, USA, ACM Press, pp.107–119 (May 1999)

[141] Van der Aalst, W.M.P., Ter Hofstede, A.H.M., and Weske, M.: "Business Process Management: A Survey". In: *Business Process Management*, LNCS, Vol. 2678, Springer, pp.1–12 (2003)

[142] Van der Aalst, W.M.P.: "What makes a good process model?". In: Software & Systems Modeling, Vol 11, Issue 4, pp. 557-569, (2012)

[143] Van Lamsweerde, A.: *Requirements Engineering: From System Goals to UML Models to Software Specifications*, John Wiley & Sons (2009)

[144] Verheecke, B., Cibran, M.A., and Jonckers, V.: "Aspect-oriented programming for dynamic web service monitoring and selection". In: W*eb Services*, LNCS, Vol. 3250, pp. 15–29 (2004)

[145] Vessey, I.: "Cognitive Fit: A Theory-Based Analysis of the Graphs versus Tables Literature". *Decision Sciences Journal*, Vol. 22, Issue. 2, pp. 219–240 (1991)

[146] Wada, H., Suzuki, J., and Oba, K.: "Early Aspects for Non-Functional Properties in Service Oriented Business Processes". In: *2008 IEEE Congress on Services - Part I*, Honolulu, USA, IEEE CS, pp. 231–238 (2008)

[147] Weiss, M., Amyot, D.: "Business Process Modeling with URN". *International Journal of E-Business Research*, Vol. 1, No. 3, pp. 63–90 (2005)

[148] Wen, J., Ren, L., Longmore, L., and Yu, Q.: "Applying Aspect Oriented Service Control in IMS Network". In: *Proc. of the Network Operations and Management Symposium (NOMS 2008)*, Salvador, Bahia, Brazil, pp. 742–764 (April 2008)

[149] WICI: *WICI KPI*, http://www.wici-global.com/kpis (July 2013)

[150] Workflow Management Coalition: *Glossary of terms*, http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf (Feb 1999)

[151] Xiao, Z., Cao, D., You, C., and Mei, H., "Towards a Constraint-based Framework for Dynamic Business Process Adaptation". In: *2011 IEEE Int. Conf. on Services Computing (SCC 2011)*, IEEE CS, pp. 685–692 (2011)

[152] Yu, E.: "Towards Modelling and Reasoning Support for Early-Phase Requirements Engineering". In: *3rd IEEE Int. Symp. on Requirements Engineering*, Annapolis, MD, U.S.A, IEEE CS, pp. 226–235 (January 1997)

[153] Yu, E., Horkoff, J., Mylopoulos, J., Richards, G., and Amyot, D. "Business Modeling for Business Intelligence". *Perspectives on Business Intelligence*, Synthesis Lectures on Data Management, 5(1), Morgan & Claypool Publishers, pp. 19–33 (2013)

[154] Yu, E., and Mylopoulos, J.: “Why Goal-Oriented Requirements Engineering”. In: *Fourth Intl. Workshop on Req. Eng.: Foundation for Software Quality (REFSQ'98)*, Pisa, Italy (June 1998)

[155] Zellner, G.: “A structured evaluation of business process improvement approaches”. *Business Process Management Journal*, Vol. 17, No. 2, Emerald Group Publishing Limited, pp. 203–237 (2011)

Appendix A: Pilot Experiment Questions

A.1 Expertise Level Questions

Question 1: What is your expertise level in URN?

- I don’t know anything about URN
- Beginner (e.g. read one paper or two)
- Intermediate (e.g. took a course and used it)
- Advanced (e.g. contributed to the research)

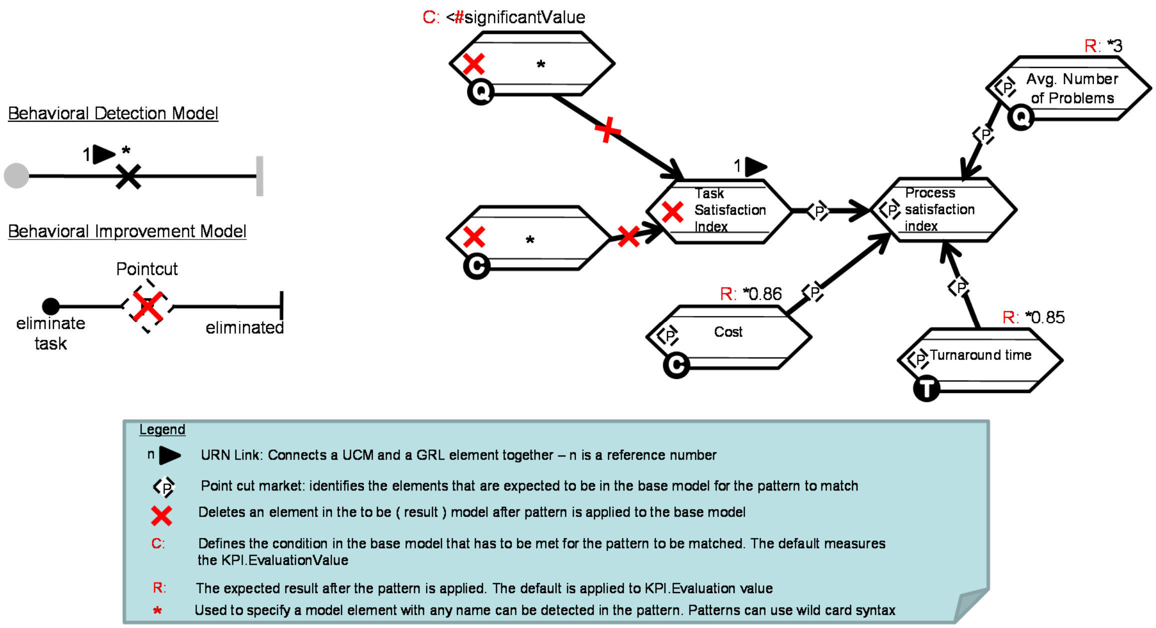
Question 2: What is your expertise level in AoURN?

- I don’t know anything about AoURN
- Beginner (i.e., read one paper or two)
- Intermediate (i.e., took a course)
- Advanced (i.e. contributed to the research)

A.2 Technical Questions

Question 1

Explain the detection and improvement parts of the patterns:



Question 2

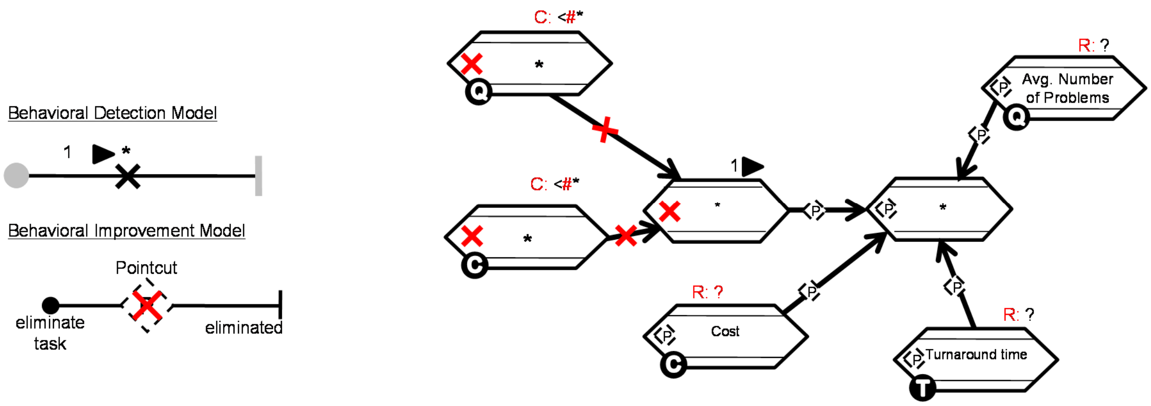
Use the generic pattern to create a pattern that does the following:

Detects:

- Detects a sub-process with a responsibility called “Integration testing”
- When “Integration testing” is connected to a GRL KPI called “System integration quality index”
- AND When the Evaluation value of a KPI called “Quality issues found” is < than a constant called “significantValue”
- AND When “System integration quality index” contributes to “System quality index” along with several other KPIs (i.e., Cost, Turnaround time, Avg. Number of Problems)

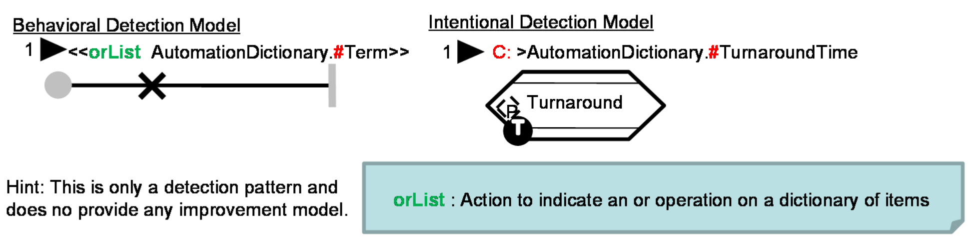
Improves:

- Removes the integration testing
- Reduces the overall cost and turnaround time by 10%
- Does not have any impact on the overall quality of the product



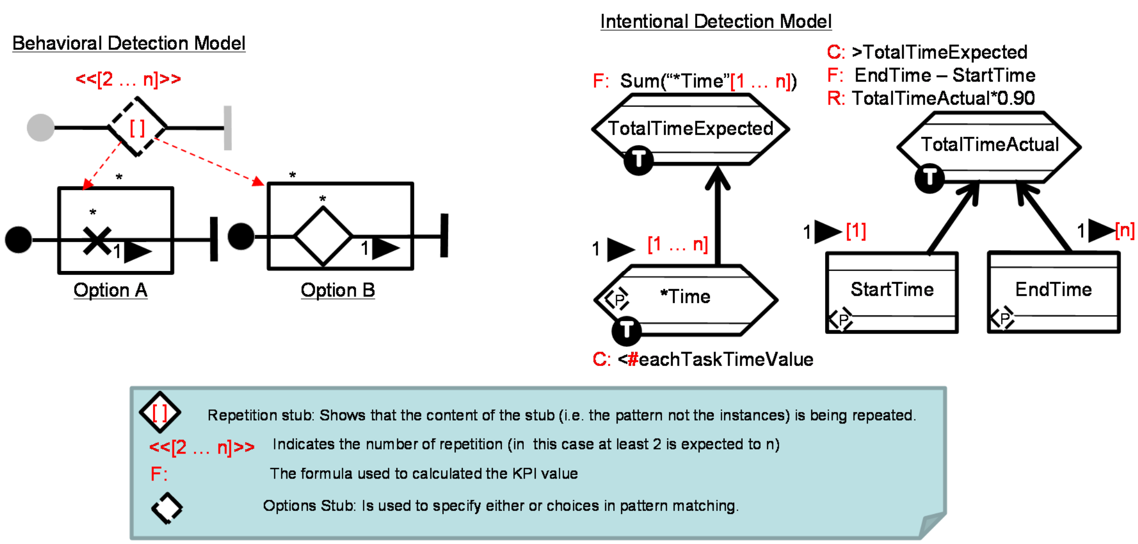
Question 3

Please explain what the pattern detects.



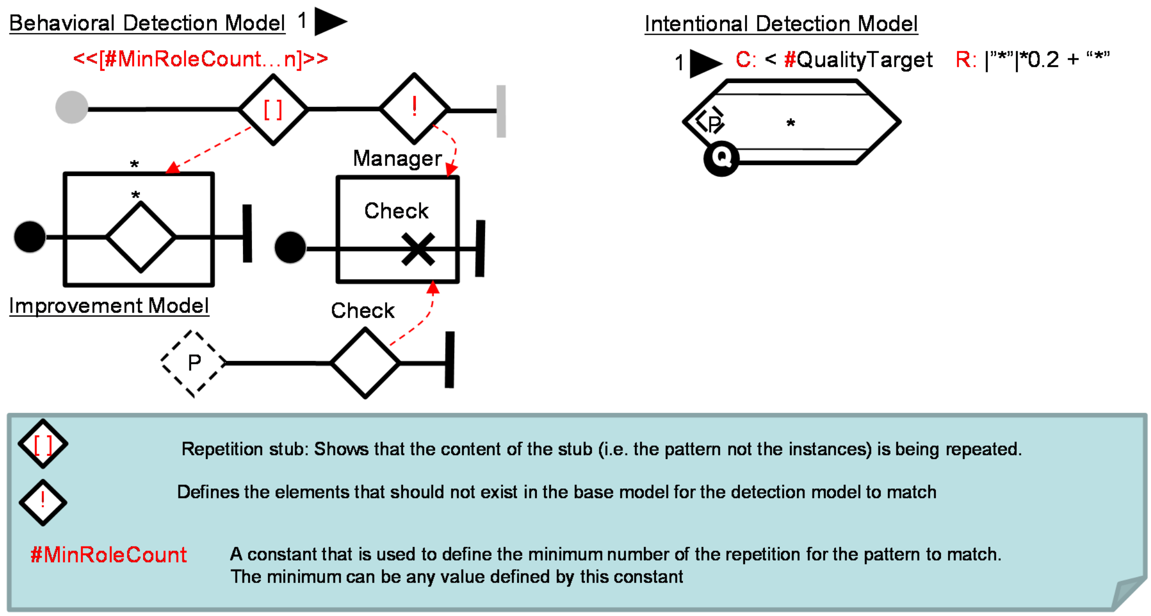
Question 4

Please explain what the pattern detects.



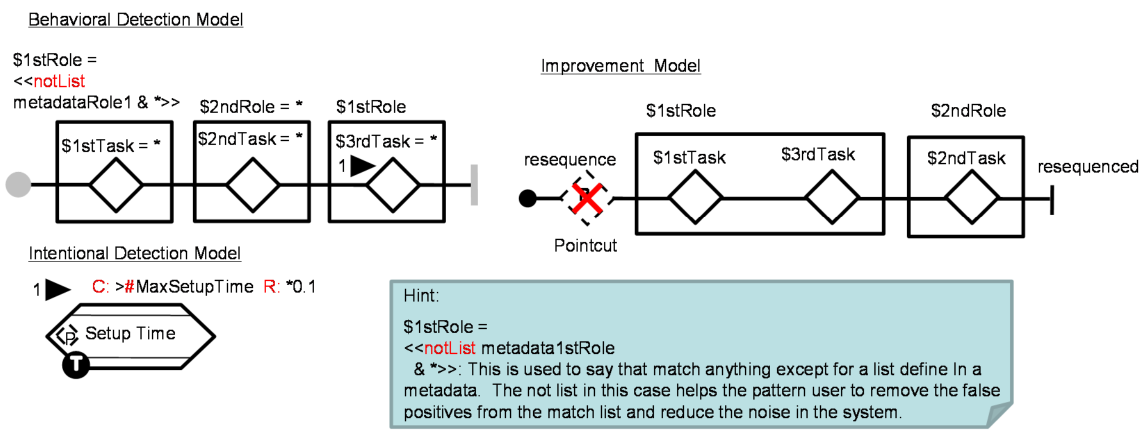
Question 5

Please explain what the pattern detects and improves.



Question 6

Please explain what the pattern detects and improves.



A.3 Opinion Questions

What do you think about the modeling notation?

Question 7.1: The modeling notation was easy to understand

1. Strongly disagree 2. Disagree 3. no opinion 4. Agree 5. Strongly agree

Question 7.2: The notation is self-explanatory

1. Strongly disagree 2. Disagree 3. no opinion 4. Agree 5. Strongly agree

Question 7.3: I can think of other cases where this notation would be useful

1. Strongly disagree 2. Disagree 3. no opinion 4. Agree 5. Strongly agree

Question 7.4: I can think of simpler ways to achieve the same goal

1. Strongly disagree 2. Disagree 3. no opinion 4. Agree 5. Strongly agree

Question 7.5: Do you have any other comments or want to elaborate on any of the above?

Appendix B: Business Process Pattern Profile

This appendix summarizes the stereotypes (Table 29) and other categories of metadata (Table 30) defined by the Business Process Pattern (BPP) profile for AoURN, which aims to implement Language X. Required well-formedness constraints are also provided.

Table 29**:** BPP profile stereotypes

Stereotype Name	**Value**	**AoURN Element**	**Visualization**
ST_BPP_MODELTYPE	“Goal”, “Decision”, or “Performance”	GRLGraph	
ST_BPP_MODELTYPE	“Process”	UCMmap	
ST_BPP_PATTERN	“IntentionalDetection” or “IntentionalImprovement ”	GRLGraph (Pointcut Graph)	
ST_BPP_PATTERN	“BehavioralDetection”	UCMmap (Pointcut Map)	
ST_BPP_PATTERN	“BehavioralImprovement”	UCMmap (Aspect Map)	
ST_BPP_KPITYPE	“Time”, “Cost”, “Quality”, or “Flexibility”	Indicator	
ST_BPP_STUBTYPE	“Repetition”	Stub	
	“Option”		
	“Absent”		
ST_BPP_RAWDATA	*(unused)*	Indicator	
ST_BPP_SITUATION	“Situation”	IntentionalElement (Softgoal)	

Table 30**:** BPP additional metadata types

Metadata Name	**Metadata Value**	**AoURN Element**	**Example**
FormulaBasedGRLStrategyAlgorithm_evalFormula	arithmetic expression	Contribution	Revenue - Costs
FormulaBasedGRLStrategyAlgorithm_situationFormula	arithmetic expression	Contribution	X * 0.8
BPP_CONSTANT	constant name	GRLGraph	TurnaroundTime
BPP_RESULT	arithmetic expression	Indicator	* 0.96
BPP_#<*label*>	label	Indicator	KPI3
BPP_CONDITION	Boolean expression	Indicator	> KPI3

Well-formedness-constraints:

- A situation can only have contributions to indicators as links.
- FormulaBasedGRLStrategyAlgorithm_situationFormula only applies to a contribution coming from a situation softgoal.
- FormulaBasedGRLStrategyAlgorithm_evalFormula only applies to a contribution between two indicators where the quantitativeContribution equals 0.
- The value for ST_BPP_STUBTYPE is a list of elements in {Repetition, Option, Absent}.