**Universität Ulm** | 89069 Ulm | Germany

**Fakultät für Ingenieurwissenschaften und Informatik**
Institut für Datenbanken und Informationssysteme
Leiter: Prof. Dr. Manfred Reichert

# Business Process Quality Management

Dissertation zur Erlangung des Doktorgrades Dr. rer. nat.
der Fakultät für Ingenieurwissenschaften und Informatik der Universität Ulm

**Vorgelegt von:**
Matthias Jürgen Lohrmann, geboren in Heidenheim an der Brenz

2015

**Amtierende Dekanin:**
Prof. Dr. Tina Seufert

**Gutachter:**
Prof. Dr. Manfred Reichert
Prof.dr.ir. Hajo A. Reijers

**Betreuer:**
Prof. Dr. Manfred Reichert

# Abstract

During the past 25 years, research in the field of business process management as well as the practical adoption of corresponding methods and tools have made substantial progress. In particular, this development was driven by the insight that well-managed business processes enable organizations to better serve their stakeholders, save costs and, ultimately, realize competitive advantage. It is therefore not surprising that improving business processes ranks high on the list of priorities of organizations. In practice, this challenge is currently being addressed through approaches such as benchmarking, industry-specific best practice reference models or process reengineering heuristics. However, no systematic and generic proposition towards managing *business process quality* has achieved broad acceptance yet.

To address this gap, this thesis contributes to the field of business process quality management with the results lined out in the following. First, it defines a concise notion of business process quality based on organizational targets, and applies it to a sample real-world case. This definition is not specific to any particular application field, and thus constitutes a vital first step towards systematic and generic business process quality management. On that basis, an approach is developed to model business objectives in the sense of the requirements that shall be fulfilled by the results of a business process. In turn, this approach enables appraising if a business process achieves its business objective as one of the core criteria relevant to business process quality. Further, this thesis proposes extensions to common business process meta-models which enable quality-aware business process modeling, and demonstrates how fundamental quality characteristics can be derived from corresponding models.

At this stage, the results achieved have enabled an advanced understanding of business process quality. By means of these insights, a model of business process quality attributes with corresponding quality criteria is developed. This model complements and exceeds preceding approaches since, for the first time, it systematically derives relevant quality attributes from a business process management perspective instead of adopting these from related fields. It enables appraising business process quality independently of a particular field of application, and deriving recommendations to improve the processes assessed. To enable practical adoption of the concepts developed, the integration of procedures and functionality relevant to quality in business process management lifecycles and system landscapes is discussed next.

To establish the contribution of this thesis beyond the previous state of the art, the proposed quality model is then compared to existing business process reengineering practices as well as propositions in the area of business process quality. Further, quality attributes are employed to improve a substantial real-world business process. This experience report demonstrates how quality management practices can be applied even if quality-aware system landscapes are not in place yet. It thus contributes to bridging the gap between the research results proposed in this thesis and the conditions present in practice today. Finally, remaining limitations with regard to the research objectives pursued are discussed, and challenges for future research are lined out. Addressing the latter will enable further leveraging the potentials of business process quality management.

**Parts of this thesis have been published in the following referred papers:**

Lohrmann, M., Reichert, M.: Effective application of process improvement patterns to business processes. Software & Systems Modeling (2014) DOI 10.1007/s10270-014-0443-z

Lohrmann, M., Reichert, M.: Efficacy-aware business process modeling. In: Proc. 20th Int'l Conf. on Cooperative Information Systems (CoopIS'12). Volume 7565 of LNCS, Springer (2012) 38–55

Lohrmann, M., Reichert, M.: Demonstrating the effectiveness of process improvement patterns. In: Proc. 14th Working Conf. on Business Process Modeling, Development, and Support (BPMDS'13). Volume 147 of LNBIP, Springer (2013) 230–245

Lohrmann, M., Reichert, M.: Modeling business objectives for business process management. In: Proc. 4th S-BPM ONE – Scientific Research. Volume 104 of LNBIP, Springer (2012) 106–126

Lohrmann, M., Riedel, A.: Requirements towards effective process mining. In: Proc. On the Move to Meaningful Internet Systems Workshops (OTM Workshops '12). Volume 7565 of LNCS, Springer (2012) 64–68

Lohrmann, M., Riedel, A.: Process Quality and Performance in Shared Services Organizations. In: Finance Bundling and Finance Transformation. Springer Gabler (2013) 225–251

Lohrmann, M., Reichert, M.: Understanding Business Process Quality. In: Business Process Management: Theory and Applications. Volume 444 of Studies in Computational Intelligence, Springer (2013) 41–73

Lohrmann, M., Reichert, M.: Formalizing concepts for efficacy-aware business process modeling. Technical Report UIB-2012-05, Ulm University, Germany (2012)

Lohrmann, M., Reichert, M.: Basic considerations on business process quality. Technical Report UIB-2010-04, Ulm University, Germany (2010)

# Contents

# List of Abbreviations

| | |
|---|---|
| AHP | Analytic hierarchy process |
| BI | Business intelligence |
| BP | Business process |
| BPI | Business process intelligence |
| BPM | Business process management |
| BPMN | Business Process Model and Notation |
| BPR | Business process reengineering |
| BSD | Binary state determinant |
| BU | Business unit |
| CASE | Computer-aided software engineering |
| CFE | Control flow element |
| CFO | Chief financial officer |
| CIO | Chief information officer |
| COBIT | Control Objectives for Information and Related Technology |
| CPI | Continuous process improvement |
| DCR | Dynamic condition response |
| EC | Effectiveness criterion |
| EDI | Electronic data interchange |
| EFQM | European Foundation for Quality Management |
| EPC | Event-driven process chain |
| ERP | Enterprise resource planning |
| FTE | Full time equivalent |
| G&A | General and administrative |

| | |
|---|---|
| HR | Human resources |
| IFRS | International Financial Reporting Standards |
| IS | Information system |
| ISO | International Organization for Standardization |
| IT | Information technology |
| ITIL | IT Infrastructure Library |
| KPI | Key performance indicator |
| LTL | Linear Temporal Logic |
| OCR | Original character recognition |
| OEM | Original equipment manufacturer |
| PAIS | Process-aware information system |
| PBWD | Product-based workflow design |
| PEMM | Process and Enterprise Maturity Model |
| PEP | Possible enactment path |
| PIM | Process improvement measure |
| PIO | Process improvement objective |
| PIP | Process improvement pattern |
| PO | Process orientation |
| QA | Quality attribute |
| QM | Quality management |
| QoBP | Quality of business processes |
| QoS | Quality of services |
| RML | Referent Model Language |
| SBVR | Semantics of Business Vocabulary and Business Rules |
| SC | Service client |
| SCOR | Supply Chain Operations Reference model |
| SLA | Service level agreement |

| | |
|---|---|
| SO | Service organization |
| SPC | Statistical process control |
| SR | Semantic requirement |
| SSO | Shared services organization |
| TQM | Total quality management |
| UC | Usability criterion |
| UML | Unified Modeling Language |
| VCFE | Virtual control flow element |
| WfMC | Workflow Management Coalition |
| YAWL | Yet Another Workflow Language |

# Part I

# Foundation

# 1 Introduction

Since the early 1990s, the field of *business process management* (BPM) has gained broad acceptance and application in both industry and academe [10, 11, 12, 13]. In practical terms, BPM provides organizations with a bundle of methods and tools to implement, utilize and improve business processes as "a set of one or more linked procedures or activities which collectively realise a business objective or policy goal" [14]. This thesis aims to support the progress of BPM by developing a holistic approach towards *business process quality* as a means to control and govern the design, implementation and enactment of business processes (BPs). Following the suggestion of Benbasat and Zand to orient research issues at stakeholder's interests [15], the motivation to apply the term *quality* in the context of BPM becomes evident when considering four stipulations further lined out in the following sections:

**Motivational Thesis 1 (Process Improvement through Analysis and Control).** A clear and concise notion of BP quality in the sense of *good* or *better* processes will be a substantial means to guide improvement efforts along the entire BP lifecycle.

**Motivational Thesis 2 (Growing Demand for Flexibility and Scalability).** The growing demand for flexible and scalable business processes induces the need to substitute domain-specific practical knowledge on how to design, implement, enact, and improve processes with generalized concepts. Note that this constitutes a requirement for effective process-aware information systems (PAISs) engineering as well [16].

**Motivational Thesis 3 (Proven Effect of Quality Management Practices).** Applying quality management practices to the domain of BPM will tap potentials proven in industrial applications beyond the common field of BPM applications.

**Motivational Thesis 4 (Practitioners' Concern for Generic BP Appraisal).** Relevant research on information systems should be primarily deducted from practitioners' interests. Industry demand for generic BP appraisal has been substantiated by empirical studies.

In the remaining sections of this chapter, these motivational theses are discussed in more detail. On that basis, research objectives summarize the intentions derived for the present thesis. Finally, it is lined out how that challenge is addressed.

## 1.1 Process Improvement through Analysis and Control

Improving business processes constitutes a topic of lasting relevance in research and industry communities. For example, consider the BP re-engineering trend of the 1990s [17, 18]. The annual Gartner chief information officer (CIO) survey of more than 2,000 enterprises globally

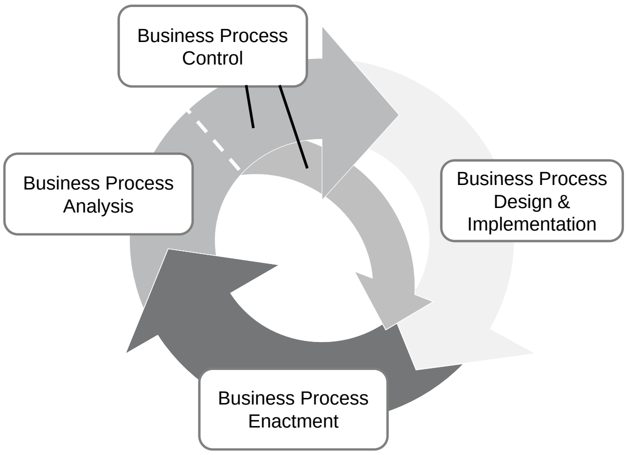



Figure 1.1: Analysis and Control in the Business Process Lifecycle

identified "improving business processes" as the number one challenge for CIOs in three consecutive years, from 2008 to 2010 [19]. The same conclusion was made in McKinsey's 2011 annual technology survey based on 927 responding subjects [20]. This indicates a sustained demand for methods, technologies and tools with respect to BP improvement which, in turn, represents a major goal of BPM [21].

Enabling continuous BP improvement[1] within a BP lifecycle [13, 22] constitutes the overall objective of process analysis and control efforts [23]. In this context, the term "control" refers to "the process of guiding a set of variables to attain a preconceived goal or objective" [24].

In the sense of a feedback cycle, insights gained when analyzing the design, implementation and enactment of business processes are, for instance, used to track down bottlenecks, allocate costs by cause, or incentivize management. In turn, these activities act as drivers towards process improvement, which can be achieved by improving process design, enhancing available infrastructure like PAISs [16, 25] or better motivating parties involved in process enactment [26, 27]. *Business process intelligence* (BPI) constitutes a current trend which, besides process discovery [28] and management of process variants [29, 30, 31, 32, 33], addresses the topics lined out above [34, 35, 36].

Figure 1.1 presents an adaptation of common BP lifecycle models [13, 22, 21] employed in this thesis. Lifecycle phases *design & implementation* and *enactment* are subject to *analysis* and resulting *control* measures. Note that control measures pertain not only to *design & implementation* adaptations, but also to *enactment*, e.g. in the provision and allocation of labor resources.[2]

[1] Note that in this thesis, the term *process improvement* is used instead of *process optimization*, since optimum business processes cannot be determined.

[2] One might argue that the *analysis* phase could also address *design & implementation* results without an intermittent *enactment* phase. This view, however, obliterates the border between *design & implementation* on the one hand in comparison to *analysis* and *control* on the other. Note that, nevertheless, *design & implementation* quality issues are valid with or without considering subsequent *enactment*.

Lifecycle-based BPM stands in notable contrast to the concept of business process reengineering (BPR) [37, 38, 39]. In BPR, actual processes are challenged by analyzing whether their results are actually required or could be replaced by something else, or even abolished in total. Contemporary BPM, in comparison, does not address this question. Instead, BPM looks into how a given process result can be best achieved through BP design and enactment. In other words, BPR asks "what", while BPM scrutinizes the "how". This thesis is committed to the BPM school of thought, because its more restricted scope is more suitable for formal and generic description, and thus to being addressed through scientific means in general and information systems research in particular.

The role of *goals* in organizational control and decision-taking has been recognized for a long time [40]. Accordingly, effective management of BP improvement through the BP lifecycle necessitates a clear notion of the goals business processes should converge to. In other words, we need to know what constitutes a good business process, and how to discern a superior process from a substandard one. This thesis addresses these questions through the concept of *BP quality*.

## 1.2 Growing Demand for Flexibility and Scalability

With the growing prevalence of PAISs [41], the field of applications where BPM concepts and technologies are applied has become broader. As example, consider the early works of Davenport and Short [17] and Hammer [18] on BPR. Notably, both articles cite the implementation of a credit note procedure in accounting at Ford Motor Corporation as a primary example. Since then, BPM adoption has spread to wholly new fields like healthcare [42, 43] or research and development [44].

This development poses a challenge to BPM practitioners and researchers alike. Traditionally, BP development and management has been guided by past experience of involved experts from the business or IT side. Similar to qualitative benchmarking [45], these specialists have contributed their knowledge of good practice to BP analysis and control. As BPM is applied to entirely new processes, however, this source of input is not available for all application areas anymore. Accordingly, BPM requires new criteria to guide initial process design and subsequent analysis and control. To transcend available areas of expertise, these criteria must be *flexible* with respect to application fields. Accordingly, *flexibility* in this context refers to sustained validity in varying application areas.[3]

In addition, BPM has not only evolved qualitatively into new application scenarios, but also quantitatively, e.g. in the sense of the adoption of workflow systems in practice. While available empirical data on this (global) trend is scarce, it can be illustrated by the growth of specialized software and services vendors (e.g., [46]). Like BPM methods and tools in general, BP analysis and control concepts must be *scalable* to accommodate this trend. In this context, *scalability* refers to the ability of a concept to cope with a growing volume of business processes it is applied to. Scalability of procedures can be achieved through automation, i.e. through independence from limited "human resources" to execute tasks.

[3] Note that the term "flexibility" can also be applied in the sense of being able to change. For example, PAISs are considered as "flexible" if they can be easily adapted to changing requirements [41, 25]. However, this does not mean that flexible PAIS must be deployable to multiple application scenarios (e.g., different organizations) at the same time.

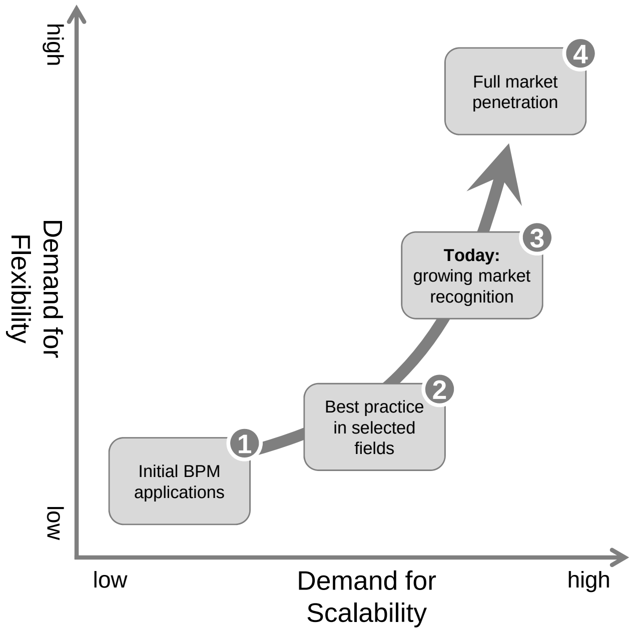



Figure 1.2: Growing Demand for Flexibility and Scalability

While it cannot be expected that this will be possible for all aspects of analysis and control with currently available means, automation can be enabled as far as possible by thoroughly formalizing applicable criteria as well as by clearly delineating what cannot be formalized and automated.

The evolution of the demand for flexibility and scalability in BPM is characterized in Figure 1.2. Initially, organizations applied BPM concepts and the related technology to particular transactional processes such as expense claims, job applications, or incoming invoices. In a second stage, these specialized applications evolved into best practices recommended across industries. Today, BPM is evolving into additional application scenarios like financial reporting, healthcare and information management. In a fourth stage yet to be reached, the full potential of the underlying concepts will be leveraged. Thus, the demand for scalability gained importance before flexibility.[4] This thesis seeks to address both issues described, flexibility and scalability, through a generic and formalized *BP quality* concept.

## 1.3 Proven Effect of Quality Management Practices

Since the 1980s, the impact of quality management (QM) practices, like design for quality or statistical process control [47], on product quality and also on business performance has been

[4] For example, consider the evolution of large near- and offshore "campuses" offering BP outsourcing capabilities for selective fields like accounting, human resources (HR) and IT support. This industry has been flourishing since the end of the 1990s.

established through empirical studies [48, 49, 50, 51, 52].[5] Although product quality and quality management practices are by far not the only determinant for business performance, most studies found that these factors have significant influence in terms of competitive advantage and, ultimately, sustained profitability. It is therefore not surprising that QM has evolved into one of the most pervasive management practices. An overview on its various forms, which are often subsumed under the term *Total Quality Management* (TQM), can be found in [47]. Traditional QM puts a strong emphasis on operations[6] as part of the primary activities of an organization according to Porter's value chain model [57]. In contrast, besides services industries [58], BPM is mostly applied to other primary and secondary activities that may be subsumed as administrative and overhead processes, i.e., the materials and labor not directly entering into end products. Since the relative weight of administrative and overhead processes has been growing for decades in terms of the total cost base of organizations to well beyond 50% in most industries [59, 60, 61], this characteristic has been an important factor to the success of BPM as a corporate practice. Moreover, BPM technology provides organizations with a wide array of IT-supported methods and tools to facilitate effective and efficient adoption [62, 22, 28, 25]. Considering the proven merits of QM with respect to competitiveness as well as profitability and the comprehensive functional scope addressed by BPM, it can be expected that bringing these concepts together will lead to substantial benefits for organizations, by extending the field QM can be leveraged in. In addition, it is promising to investigate how appropriate QM practices can be promoted by integrating them with IT-based BPM methods. It will thus be rewarding to assess how QM and BPM concepts can be aligned.

Another perspective on the relation between QM and BPM pertains to the role of BPM as enabler for improving quality. Reflecting this, the requirements posed towards QM by the International Organization for Standardization (ISO) mainly pertain to process management issues [63]. In this context, [64] cites "improving the quality of products and services" as the sole aim of BPM.

To be utilized effectively, QM always starts with specifying a notion of quality the organization aspires to achieve. Therefore, it is a crucial first step towards QM for business processes and the primary objective of this thesis to develop a notion of *BP quality* which is based on QM insights, fits well with BPM concepts and methods, and can be applied in a practical business environment.

## 1.4 Practitioners' Concern for Generic Appraisal of Business Processes

In line with the requirement to orient the initial selection of topics for relevant information systems research at the issues practitioners are interested in [15], the final motivational

[5]Comparable to these results, a literature review of contributions describing the effects of process orientation (PO) in the sense of "focusing on business processes ranging from customer to customer instead of placing emphasis on functional and hierarchical structures" [53] found that, in a predominant number of cases, positive effects of PO outweigh negative ones [54].

[6]For industrial businesses, the term "operations" refers to manufacturing and logistics. The quality of services provided to external customers is comparable to industrial operations as well as "internal" administrative functions and constitutes a distinct area of research (e.g., [55, 56]).

stipulation for this thesis is based on personal observations when working with BPM professionals.

The idea to conduct research on *BP quality* has been a result of discussions with process managers and other stakeholders when working as a consultant with clients from various industries. In general, both managers and external advisers find it fairly easy to assess the maturity – and therefore improvement potentials – of business processes which are well-understood due to personal experience or based on comparison to other organizations. However, this is much more challenging for processes not broadly used or even unique to a company. A notable example in this respect, which contributed much to the motivation for this thesis, was the market entry process of a fast-growing retail group. When asked to evaluate that process, it was difficult to come to a conclusion for lack of comparable cases.

However, as BPM technology enjoys growing prevalence, the application of BPM quickly moves beyond standard use cases. Thus, there is a growing demand to liberate techniques to assess, evaluate and improve business processes from the need to refer to anecdotal evidence. This amounts to the demand for a generic theory of *BP quality* to facilitate generic appraisal methods. Initially motivated by clients' requirements, the author of this thesis started to discuss the matter with practitioners and to look into approaches from industry and academic research. It soon became clear that it was warranted to explore the issue more thoroughly. This thesis summarizes the results of the resulting research project.

## 1.5 Research Objectives

Based on the motivational theses, this section summarizes research objectives in the sense of results this thesis aims to achieve. The research methodology in Chapter 3 will elaborate in more detail on this topic, and demonstrate how research objectives connect to research deliverables and effectiveness criteria. The concept of *managerial analysis and control* has emerged as a *leitmotif* in the motivational theses. The first two objectives thus reflect the demands of managerial analysis and control for business processes.

**Research Objective 1 (Enabling Business Process Quality Analysis).** *Providing organizations with holistic, generic and formalized concepts to analyze business process quality will facilitate evaluating the performance of parties involved, effectively incentivizing, and identifying improvement potentials while maintaining flexibility and scalability in BPM.*

**Research Objective 2 (Enabling Business Process Quality Control).** *Quality control in the sense of steering quality-relevant factors aims at quality improvement. In turn, quality improvement starts with identifying respective potentials, and is thus closely linked to quality analysis. To enable improvement, issues that hamper quality must be identified and amended with mitigation strategies. This topic should be addressed with a sufficient level of abstraction to be applicable to a broad range of application scenarios.*

The third research objective complements Research Objectives 1 and 2. Pursuing it will not result in novel concepts, but substantially enhance the relevance of achieved results.

**Research Objective 3 (Enabling Economically Reasonable Practical Adoption).** *Practical applicability and relevance of conceptual results are determined by whether or not* *they can be transferred to real-world application. In the given context, this means that organizations must be enabled to determine whether adoption and implementation of concepts is economically reasonable.*

## 1.6 Outline

The present thesis is structured in three parts. Part I provides readers with the relevant background for the subsequent development of new concepts. To this end, the part contains chapters on basic concepts (cf. Chapter 2), research methodology (cf. Chapter 3), and state of the art (cf. Chapter 4).

Part II comprises the contribution to the body of knowledge in BPM developed in this thesis. In Chapter 5, a definition framework for BP quality is derived. Chapter 6 then discusses the issue of business objectives and BP efficacy as a resulting conceptual requirement not yet solved by available approaches. Chapter 7 provides a meta-model to reconcile current BP modeling approaches to the demands of BP quality management in the sense of quality-aware BP modeling. On that basis, Chapter 8 develops a model to assess BP quality based on quality attributes, criteria and predicates. Chapter 9 integrates the results achieved with common methods in the BP lifecycle, and reconciles propositions to process-aware system landscape architecture components.

Part III seeks to substantiate the conceptual results achieved. Chapters 10-12 validate results by means of an initial field evaluation, a detailed comparison of the quality model of this thesis to available literature, and a method to determine the benefits of adopting quality management measures including its application to a real-world sample case. Chapter 13 revisits the research objectives of this thesis as well as the effectiveness criteria defined along the research methodology, and thus provides a final evaluation of the results achieved. In addition, the resulting implications are summarized, and directions for future research are discussed.

# 2 Basic Concepts

As a preliminary step to the development of concise concepts to manage BP quality, it becomes necessary to ensure a common understanding of the basic concepts employed in the areas of BPM and quality. This is of particular relevance since both terms have been subject of a great number of attempts to find a definition over time (e.g., [65]). Therefore, this chapter shortly presents basic terms and definitions adopted in this thesis.

## 2.1 Business Processes and Business Process Management

This section discusses relevant aspects of business processes and BPM with special regard to the issues relevant to BP quality and quality management.

### 2.1.1 Business Process Terminology

Early works addressing BPM topics date back to the 1930s [66]. In the 1980s, the office automation trend employed concepts attributable to BPM as well [10]. The notion of business processes and BPM gained wide-spread recognition in the early 1990s when the concept of BPR became popular with practitioners. As a proponent of this movement, Davenport defines a business process as follows:

> "A business process is simply a structured, measured set of activities designed to produce a specified output for a particular customer or market [...] A process is thus a specific ordering of work activities across time and place, with a beginning, an end, and clearly defined inputs and outputs." [37]

Concurrently, a similar view was developed by Hammer and Champy:

> "We define a business process as a collection of activities that takes one or more kinds of input and creates an output that is of value to the customer." [38]

While many authors have discussed these definitions [65], their core content has basically not changed until today, and has achieved wide-spread acceptance. Further considerations can therefore refer to the definition advocated by a well-established trade association. The Workflow Management Coalition (WfMC) defines the term business process as follows:

> "A set of one or more linked procedures or activities which collectively realise a business objective or policy goal, normally within the context of an organisational structure defining functional roles and relationships." [14]

Overall, authors agree on a set of core constituents amounting to a business process. Figure 2.1 summarizes aspects shortly described in the following:

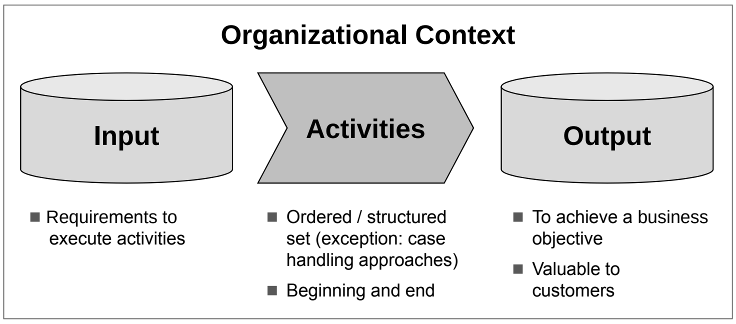



Figure 2.1: Core Constituents of Business Processes

- **Activities:** Activities describe the "content" of business processes as a blueprint for action. In a wider sense, this includes decisions taken (automatically or manually) to determine how a process continues [67]. Contrary to the definition by Hammer and Champy and in agreement with Davenport, an ordering of activities or control flow is generally assumed today. Notable exceptions are data-centric BPM approaches where additional degrees of freedom are required to cope with complexity (e.g., [68, 69, 70, 71, 72, 73]).
- **Input:** While process input is not explicitly mentioned in the WfMC definition, each activity uses resources of a kind [74]. Thus, the requirement for process input is implicitly comprised in the notion of a set of activities. Accordingly, process input is understood as anything any process activity refers to or requires to be executed.
- **Output:** Process output refers to the aspired result of a business process. In this context, many authors cite its value to an internal or external customer. In the WfMC definition, process output is replaced by the notion of a "business objective or policy goal" to be achieved [14]. The interrelations between these concepts will be discussed in Chapter 5.
- **Organizational context:** The concept of organizational context is referred to in some BP definitions. However, a more concise explanation of why this is important to business processes remains an open issue. As an explanatory approach, one may assume that the organizational context reflects the role of business processes as an instrument of co-ordination in environments based on the division of labor.[1] Newer developments extend this principle to cross-organizational division of labor (process orchestrations vs. process choreographies [22, 76]). In the context of quality management, the organizational environment is important with respect to the definition of appropriate quality standards (cf. Chapter 5).

In general, definitions of the term *business process* are inclusive and cover virtually everything members of an organization undertake to serve organizational purposes. As discussed in Section 2.2, however, quality management is already well established in the fields of supply chain management, production and customer service. Moreover, quality assessment will

[1] Adam Smith's 1776 description of procedures in a pins factory to illustrate this principle can be considered as an early example of process orientation [75].

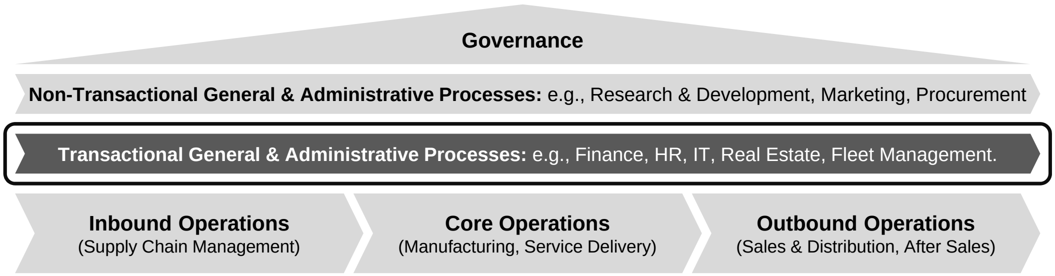



Figure 2.2: Transactional General & Administrative Processes in the Value Chain of Enterprises

be particularly effective when its results can be applied in future iterations in the sense of a feedback cycle. Therefore, this thesis focuses on *transactional*[2] *general and administrative (G&A) processes*. Note that this restricted scope is implicitly or explicitly assumed in most BPM contributions as well. Figure 2.2 puts transactional G&A processes as the major scope of this thesis into the context of a general value chain of enterprises by extending Porter's corresponding concept [57].

As an additional concept, it is possible to distinguish between a *BP definition* as an abstract notion and *BP instances* as concrete enactments thereof. According to the WfMC, a process definition "consists of a network of activities and their relationships, criteria to indicate the start and termination of the process, and information about the individual activities, such as participants, associated IT applications and data, etc." [14]. On that basis, a *BP model* is an artifact that represents a process definition in a way being useful to implement or enact the process definition [77]. Accordingly, a BP model can be given on the basis of a distinct modeling notation or meta-mode [78, 79, 80], but may also be available as a purely textual description or in any hybrid form. In the following, the first case is referred to as *explicit* process models, while the second is subsumed as *implicit* ones. In turn, a process instance is defined as "the representation of a single enactment of a process" [14]. For the more basic term business process, it remains open whether it refers to a process model or a set of one or more process instances corresponding to a common process model. BP models and BP instances are described in a model proposed by Weske [22]. In most applications, this distinction is made implicitly based on the *BP lifecycle stage* relevant to the issue at hand.

Moreover, virtually all definitions of business processes allow for processes *aggregation and dis-aggregation*, i.e., multiple process models may be consolidated into one joint process model (i.e., aggregated), or a single process model may be split into multiple process models (i.e., dis-aggregated). Generally, it is possible to consider everything that happens on purpose in a structured manner in an organization as part of a single huge business process without contradicting common definitions. Splitting up the entire organizational value chain [57] into individual business processes is thus subject to the discretion of the responsible modeler. In this context, *up- and downstream processes* denote related business processes in the sense that the output of an upstream process constitutes input for a downstream

[2]In the context of this thesis, the term "transactional" generally refers to uniform, well-structured and repetitive processes.

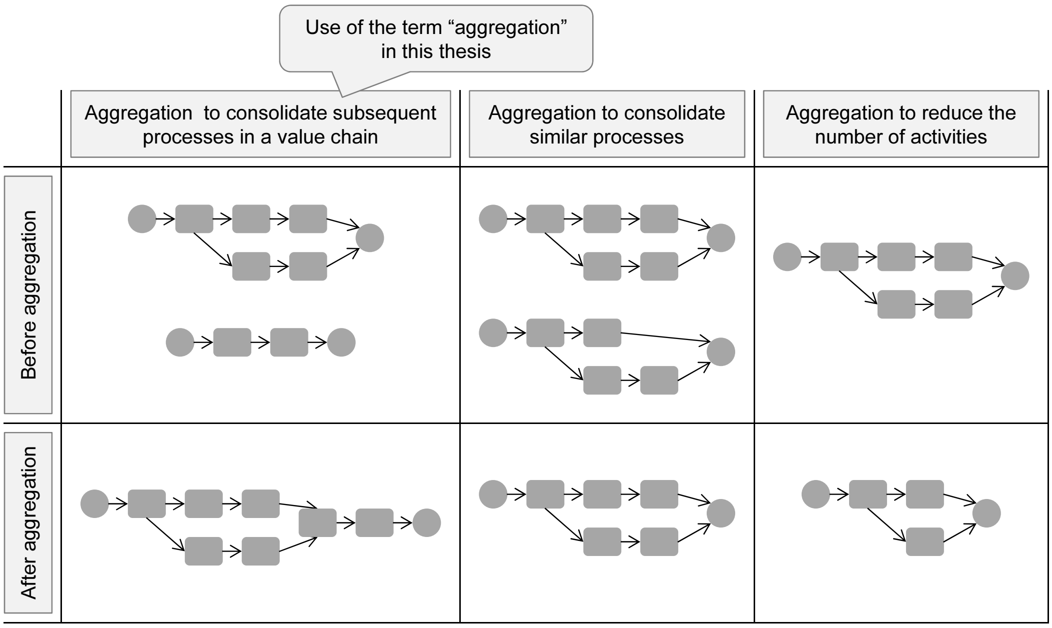



Figure 2.3: Perspectives on Process Aggregation

process. For example, the process of issuing purchase orders to suppliers constitutes an upstream process to the downstream process of handling incoming supplier invoices. Note that this understanding of processes aggregation and dis-aggregation differs from the notion presented in [81], where process aggregation describes the consolidation of multiple similar process models into a single comprehensive model. Thus, while the view used here to discuss process quality relates to subsequent processes in a value chain, the perspective proposed in [81] mainly addresses processes occurring in parallel. Another notion of process aggregation pertains to consolidating semantically related activities of process models into more abstract ones to reduce the overall number of activities in a model [82, 83, 84]. Figure 2.3 clarifies the three different perspectives on process aggregation.

### 2.1.2 Business Process Lifecycle

Figure 1.1 summarizes typical stages of a BP lifecycle comprising BP design & implementation, enactment, analysis and control.[3] Typically, the term business process is implicitly interpreted as either a *model* or a *set of instances* in within one lifecycle stage. Since this distinction is relevant to what needs to be considered in BP quality management, it is useful to define two *fundamental lifecycle stages* on that basis. Table 2.1 summarizes the fundamental lifecycle stages used in the following, and compares them to the BP lifecycle presented in [13].

[3]Note that the lifecycle model used here for the purpose of quality management excludes the diagnosis stage included in [13], because BP quality assessment is in itself part of this stage.

| Fundamental Lifecycle Stage | "BP" Interpretation | Corresponding Lifecycle Stages in [13] |
|---|---|---|
| **Lifecycle Stage I: BP design & implementation** | The business process as an abstract process model and its implementation in terms of organizational capabilities (*actual process model*) | Process design, system configuration |
| **Lifecycle Stage II: BP enactment** | The business process as a set of one or more instances of a common abstract process model | Process enactment |

Table 2.1: Fundamental Business Process Lifecycle Stages

Note that the definition of Lifecycle Stage I in Table 2.1 refers to organizational capabilities. These denote the organization's ability to actually execute a process model by ensuring the availability of required resources such as information systems, equipment or staff. On that basis, the term *actual process model* designates an explicit or implicit process model including the corresponding organizational capabilities.

BP quality assessment must address both lifecycle stages. From a management perspective, it makes sense to analyze the quality of an actual process model *as well as* the quality of the corresponding process instances. Typically, organizational responsibilities differ for the lifecycle stages. To achieve *exclusive coverage* (cf. Effectiveness Criterion EC 1 in Table 3.1), i.e. to constrain the scope covered by quality assessment procedures to the respective scope of influence, it is necessary to separate results for both analyses.

### 2.1.3 Business Process Management Objectives

To properly employ quality management in the context of BPM, it is instrumental to develop a sound understanding of the objectives pursued. This requirement reflects the characterization of BPM methods as *goal-bound artificial constructs* (cf. Chapter 3). Notably, literature on the issue exhibits a distinct paradigm shift in this respect. In the 1990s, BPR was advocated as a "radical" way of changing and improving economically-oriented organizations [17, 18, 37, 38].[4] BPR asked not only whether a given process could be improved, but also whether it was required at all, or whether it could be replaced by something else. Example 1 illustrates the basic principles.

**Example 1 (The Credit Note Procedure Example).** To illustrate the case for "radical" BP reengineering, [17, 18] describe an example for process optimization at Ford Motor Corporation . The example relates to the optimization of the accounts payable process,

[4]Interestingly, the main motivation cited by the authors was American companies' quest for a competitive response to the tremendous success of Japanese corporations at the time – this topic has also been linked to their perceived lead in quality management.

one of the best-understood processes in terms of optimization potentials in administrative functions. The authors claim that Ford's North American operations were able to reduce personnel capacity requirements in the accounts payable department by 75% by radically reengineering the process: instead of receiving and checking invoices, credit notes are issued to suppliers after goods have been delivered.

More than 20 years later, one would expect a practice that has been implemented with as much success at a well-known multinational company to have gained wide-spread acceptance. Personal observations during the last years, mainly when working with European manufacturing groups as a consultant, show that this is indeed the case, and that the practice is well-known and adopted in many companies. However, it is by no means pervasive. For instance, the practice is very prevalent at automotive OEMs (original equipment manufacturers, i.e., car makers), but not widely spread in the machine tools industry. Although empirical data is not available, it could be argued that, as an estimate based on experience with consulting clients, on average less than 10% of total purchasing volume are processed via a credit note procedure.

This outcome can be ascribed to the fact that by applying credit note procedures instead of receiving invoices, work is by no means "obliterated" [18], but merely shifted from the customer to the supplier: instead of the customer checking the invoice, the supplier checks the credit note. The new activities on the supplier side involve matching the original customer order against the delivery note and the credit note, which is rather similar to the original invoice checking process. Of course, this is only possible in industries where buyers are in a good bargaining position, hence the wide-spread adoption by automotive OEMs. As opposed to the claim of the advocates of BPR, the workload has not been obliterated but merely reassigned. Moreover, the pressure to adopt a credit note procedure has lessened with the advent of advanced process automation techniques in the field, such as electronic data interchange (EDI) and intelligent scanning of documents.

The example shows that the BPR postulation of radically re-thinking activities and processes does not always lead to tremendous results in the long run. Most projects that achieved long-term success were based on a combination of process reengineering, organizational reallocation of tasks and elimination of non-value adding fringe activities.

Thus, reengineering involved reconsidering the desired output or the *objectives* that were pursued by enacting business processes.[5] One demand made was that all processes – or, more precisely, process objectives – should be oriented towards "the customer". Of course, "customer" was a rather flexible notion as departments began to define other departments or management as their "customers". However, assessing business processes from a reengineering perspective clearly comprises assessing the business objectives pursued as well. If asked to ascertain BP quality, one can assume that the advocates of reengineering would have commenced with scrutinizing the associated business objective.

The reengineering approach, however, led to a number of issues that could not be easily resolved, such as change management and other topics related to human involvement in

[5] E.g., [37, p. 10]: "Process innovation [...] involves stepping back from a process to inquire into its overall business objective", or [38, p. 35]: "In doing reengineering, businesspeople must ask the most basic questions about their companies and how they operate: Why do we do what we do? And why do we do it the way we do?"

business processes [85]. For example, there was no clear way to support decisions on which processes serve the customer and which do not, or which business objectives are important and which are not. Overall, radical approaches towards BP optimization that did not focus on improving given processes, but involved reconsidering whether process objectives were reasonable at all quickly led to a huge scope of change. It became apparent that this tended to overtax organizations. Consequently, the question which processes are actually required and, inherently, which business objectives are to be pursued, or whether one business objective is to be preferred over another, has been omitted from today's BPM. This trend even pertains to recent work by Hammer as one of the founders of the reengineering philosophy [86].

As a result, the mentioned decisions have been delegated to strategic management such that the business objectives associated with processes are not an object of today's lifecycle-based understanding of BPM anymore [13, 22]. This corresponds to the insight that it is difficult to analytically assess business objectives (at least, with currently available methods). Thus, it makes sense to treat them as part of a discipline not as focused on formal analytics as BPM. For instance, [87] refers to "operational effectiveness" as the ability to "get more out of their inputs than others because they eliminated wasted effort, employ more advanced technology, motivate employees better or have better insight into managing particular activities or sets of activities". A large part of these topics relates to BPM. "Competitive strategy", however, is defined as "deliberately choosing a different set of activities to deliver a unique mix of value". Trade-offs between differing business objectives are thus seen as the subject of strategic management rather than BPM.

In other words, BPR and contemporary BPM differ with respect to whether process output or business objectives (cf. Figure 2.1) are treated as an object or as a constraint regarding BP design. Based on these considerations, this thesis reflects the latter point of view. Thus, contrary to the BPR standpoint, *what* an organization aims to achieve through a business process is not subject to quality management in the sense of this thesis. Rather, the following chapters focus on *how* given objectives are addressed.

## 2.2 Quality and Quality Management

Since the 1950s, QM has become one of the core management concepts adopted by organizations globally. During that time, concepts and notions for quality have evolved from the work of pioneers such as Shewhart, Deming, Crosby, Feigenbaum, Juran and Ishikawa to standardized terminologies and methods that are propagated by trade and governmental bodies [88]. In terms of practical adoption, the definition of quality most widely spread today has been developed with the ISO 9000 series of standards [89]. As a set of norms in the area of QM for business applications, ISO 9000 has achieved broad acceptance through endorsements by governmental bodies like the European Union and the ISO 9000 certification scheme [90, 91, 92]. For a fundamental definition of quality, it is reasonable to resort to the definition given in the ISO 9000 series of standards: *quality denotes "the degree to which a set of inherent characteristics fulfills requirements"* [89].

This definition duly reflects a fundamental issue relevant for all approaches towards quality management: determining quality is based on a comparison to an ideal, target or standard that sets requirements for the object in question. The following chapters refer to this concept

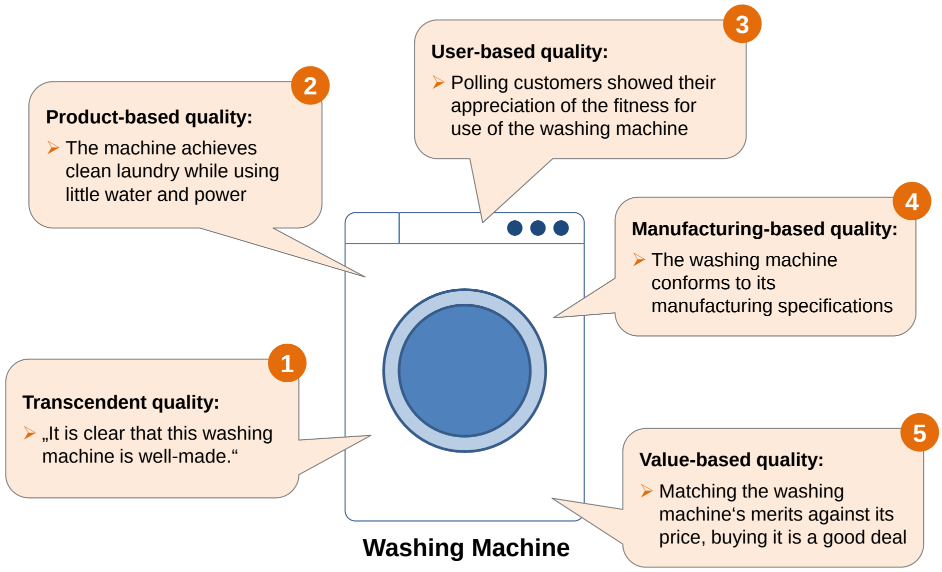



Figure 2.4: Quality Views Example

as a *quality standard*. By defining a quality standard that considers effectiveness criteria, a specific definition of quality is created, and quality management is thus enabled. Inversely, if one wants to apply quality management to a class of artifacts (e.g., business processes or units of BP input), one has to define an appropriate quality standard. This is tantamount to developing a quality definition specific to the class of artifacts.

In the course of the evolution of QM as a discipline, various views on quality have been argued. Basically, these views correspond to different classes of quality standards. As a foundation to further evolve the understanding of quality with respect to aspects of business processes, these views are reflected along a widely used classification. According to Garvin [93, 94], quality can be discussed in terms of "the *transcendent* approach of philosophy", "the *product-based* approach of economics", "the *user-based* approach of economics, marketing, and operations management", and "the *manufacturing-* and *value-based* approaches of operations management". For an initial overview, Figure 2.4 illustrates differing views on quality with a product quality example. The following paragraphs sketch each quality view, and appraise how well it fits with the effectiveness criteria defined in Section 3.1. Note that the *Cost effectiveness* criterion depends on the concrete implementation of quality measures. Therefore, it is not discussed in more detail here, but can be approximated through *Transparency and retraceability*: transparent and retraceable aspects typically lend themselves well to formalization and thus coverage via, for instance, automated appraisal methods.

### 2.2.1 Transcendent View

The transcendent view is the only approach that defines quality independently of the perceptions or requirements of individuals or organizations, such as customers, and therefore without an utilitarian or economic perspective. Instead, according to this approach, quality refers to "innate excellence" which, in principle, is applicable to all concepts in a BPM context and independent of any external factors. While this view is deeply rooted in classical and modern Western philosophy [95], it is not in line with the ISO 9000 definition. Moreover, BPM is aimed at *business* process support [13]. Obviously, this implies an economically motivated context. While its role in disruptive innovation [96, 97] is duly recognized, it must be concluded that transcendent quality is not appropriate for BPM applications, since BPM in itself is motivated by economic or business-oriented targets. This corresponds to the *Congruence to organizational targets* criterion.

Moreover, the abstract nature of transcendent quality or quality as excellence does not lend itself to practical quality assessment or measurement [98] because results cannot be considered as objective, thus violating the *Transparency and retraceability* criterion. This is another obstacle towards its practical applicability in a management context such as BPM.

### 2.2.2 Product-based View

The product-based view focuses on measuring concretely defined quantifiable and desirable attributes in products [99]. In a BPM environment, it is possible to substitute the term product with the output of a business process, which includes end products as well. The more of a quantifiable attribute is found in an unit of output, the better its quality is judged. Hence, this approach can be directly linked to process output as a central construct in BPM, and it supports management applications by enabling measurement.

**Example 2 (Product-based View).** Consider the process of dejamming telephone landlines. A short cycle time should hopefully be a desirable process output attribute for any provider, and will therefore constitute a valid quality measure in the product-based quality view.

However, the product-based approach in a BPM context addresses process output only. This is not fully satisfactory as process input, process execution and, therefore, the economic viability of business processes are not considered at all, which is clearly not sufficient in terms of *Congruence to organizational targets*. Thus, product-based quality management always needs to be suitably complemented. An additional issue relates to the relative weight of attributes considered for quality assessment if more than one product characteristic is analyzed.

**Example 3 (Relative Importance of Attributes).** Low weight and high stability are attributes that are both valued in bicycle frames. How the relative merits of a heavy but robust frame are judged against a light but fragile one, however, depends on the relative weight alloted to both characteristics. Note that this may constitute a major issue if customers judge differently.

Thus, it is possible to conclude that, according to this approach, *transparency* is typically given since only well-defined quantifiable attributes need to be considered. In contrast, *retraceability* is impeded in many cases: the relative weighing of multiple attributes can usually not be done analytically, but needs to be estimated or approximated.[6].

### 2.2.3 User-based View

The user-based view takes the product-based approach one step further by replacing measurable attributes of products (or process outputs) with the satisfaction of a user or customer.

Regarding applicability in a business context, the same conclusions apply as to the product-based view, i.e., *Congruence to organizational targets* and *transparency*. As major difference, this approach is more inclusive and leads to more effective quality assertions, i.e., all product and output attributes are considered, and criteria are weighed to reflect user satisfaction; i.e. there is only one scale of measurement. This allows for analytically well-founded normative statements on quality. In the product-based approach, this is only possible if exactly one product attribute determines quality. In contrast, the weighing of product or output attributes to effectively reflect user preferences (e.g. across a broad customer base) constitutes an additional layer of complexity that is not easily resolved. This, as well as the alternative approach of polling users directly, severely impedes *retraceability*.

### 2.2.4 Manufacturing-based View

The manufacturing-based view focuses on conformance to specifications instead of achieving optimum measures for certain product attributes or user satisfaction. Similar to the product- and user-based approaches, it is output-centered, but implicitly recognizes differences between optimum user satisfaction and aspired attribute values.

**Example 4 (Optimum User Satisfaction vs. Aspired Attribute Values).** Consider the assembly of body parts in car making. Customers generally prefer tighter clearances, i.e. more narrow gaps between parts. However, achieving minimum clearances requires much re-work at the assembly plant. Thus, manufacturers define levels of tolerance for clearances as aspired attribute values, and consider quality requirements as satisfied if manufacturing achieves these levels.

It can be argued that the ISO quality definition has been derived from this view on quality, as it also stresses conformance to requirements.

As rationale behind this, optimum attribute values (product-based view) or user satisfaction (user-based view) might not be economically sensible from the point of view of the organization. Thus, the approach indirectly incorporates economic considerations which corresponds well with aspects of *Congruence to organizational targets*. Moreover, approaches towards manufacturing-based quality mostly include engineering ("design for quality") and production control, thus managing quality not only on the basis of process output. With respect to *Transparency and retraceability*, the manufacturing-based approach exhibits the same advantages regarding *transparency* as the product-based approach, but also the same issues regarding *retraceable* quality measurement across multiple attributes and transparency.

[6] As an example for procedures developed to resolve problems related to the relative weighing of characteristics, consider the Analytic Hierarchy Process [100]

Quality in the manufacturing-based view lacks a direct link to common quality expectations since process output which does in no way satisfy the expectations of users, but fully conforms to specifications will be considered as high quality. From a business perspective, high-quality manufacturing is therefore not sufficient, but needs to be complemented by defining high-quality specifications to conform to.

### 2.2.5 Value-based View

The value-based view on quality incorporates the economic environment of organizations even more as it defines quality not only in terms of product attributes and user expectations, but puts these in relation to the cost or price involved. According to this view, good quality is achieved if a product or process output does not only meet expectations, but also comes at a reasonable cost [101]. This reflects the inclusive and economic nature of BPM in terms of *Congruence to organizational targets*, as it considers both resource consumption and output, and additionally incorporates economic considerations.

Garvin notes that this view on quality is hard to apply in practice [94] since it is difficult to comprehensively evaluate process input and output in economic terms. In turn, this leads to issues related to the *Transparency and retraceability* effectiveness criterion.

**Example 5 (Economic Evaluation of Process Input and Output).** With regard to the difficulty of economically evaluating process input and output, consider the following examples:

- If a process delivers output not directly sold in the marketplace, its value to the organization cannot be trivially measured. As an example, consider research and development processes that do not lead to new products.
- Process input procured from suppliers or obtained as output of upstream processes cannot be valued at cost without further considerations; e.g., supplier selection might not have been fully effective, or upstream processes might not be designed optimally.
- Risks associated with process input or process execution are difficult to appraise. For instance, a supplier might provide process input at a better price, but with the additional risk of not being able to deliver in time.

### 2.2.6 Summary

Based on Garvin's structuring of quality approaches, Table 2.2 summarizes initial considerations with respect to quality views and the effectiveness criteria introduced in Section 3.1. It can be concluded that *retraceability* remains an issue as an objective set of criteria to determine actual quality is difficult to determine. In the context of this thesis, it is intended to address this issue as described in the research methodology (cf. Chapter 3). The derivative character of this approach will enable practitioners to "drill down" from a definition framework based on relevant business considerations to a detailed quality model specific to the demands of BPM.

| *Quality View* | *EC 1: Congruence to Organizational Targets* | *EC 2: Transparency and Retraceability* |
|---|---|---|
| *Transcendent view* | No fit: does not correspond to a business context | Not transparent<br>Not retraceable |
| *Product-based view* | Limited consideration of economic aspects | Transparency given<br>Limited retraceability in case of multiple attributes |
| *User-based view* | Limited consideration of economic aspects | Transparency given<br>Severely limited retraceability in case of multiple attributes |
| *Manufacturing-based view* | Indirect consideration of economic aspects by definition of economically viable specifications | Transparency given<br>Limited retraceability in case of multiple attributes |
| *Value-based view* | Consideration of economic aspects by reflecting cost / value relation | Limited transparency due to complexity<br>Limited retraceability due to complexity |

Table 2.2: Quality Views vs. Effectiveness Criteria

## 2.3 Sample Processes

This section presents sample processes used in this thesis in order to illustrate and exemplify the ideas and concepts discussed. The sample processes have been selected to cover a broad spectrum of process characteristics relevant to the issue at hand. Each process is described in textual form and represented as BPMN flow chart [80].

The first sample process, *invoice checking and approval*, corresponds to a typical use case in financial accounting. This field is particularly well-suited to the approach developed in this thesis since it occurs in all industries. Invoice handling and approval is a well-understood high-volume repetitive transactional process which, due to its high degree of standardization, is often used in sourcing models such as shared services or outsourcing [102]. Note that this process has been used by Davenport [17] as well as Hammer and Champy [18] to illustrate BPR benefits.

**Example 6 (Sample Process A: Invoice Checking and Approval).** The business process starts with the receipt of a supplier invoice (activity A1). The invoice is then compared to the respective purchase order (A2). If deviations exist, these are subject to approval. In practice, this is often the case when, for instance, price data have not been maintained or no purchase order has been entered into the ERP system. If the deviation is approved (A3), the purchase order is created or adapted (A4). Otherwise, the invoice is declined (A6, A7). In the next step, the invoice is matched against goods receipt (A5) and, depending on the result, either declined (A6, A7) or passed to the next check, which is based on the invoice value. For a value of more than 5,000, senior management approval is required (A8). If this is granted, the invoice may be finally approved (A9).

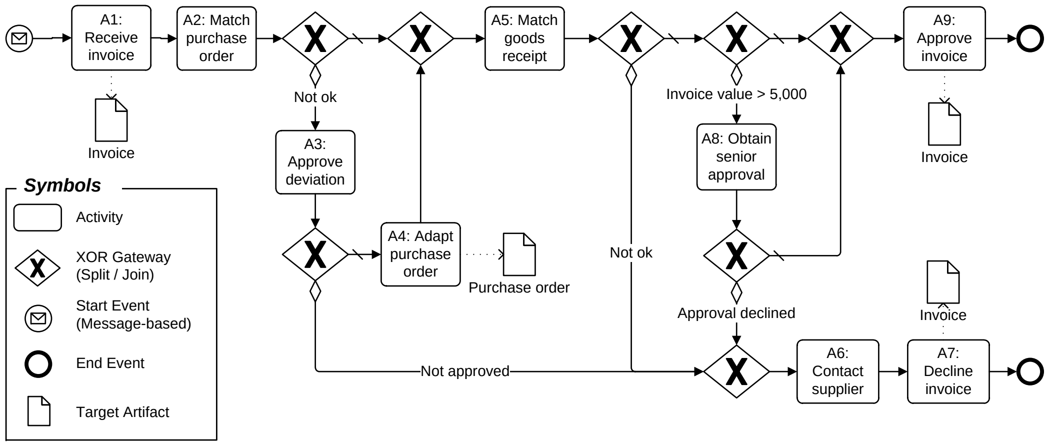



Figure 2.5: Sample Process A: Invoice Checking and Approval

Sample process B refers to the same domain: *Payment run* constitutes as a typical follow-up procedure or downstream process of *invoice checking and approval* (cf. Section 2.1). The combination of Sample Processes A and B exemplifies a characteristic that can often be observed in process chains as well: while each instance of Sample Process A addresses one incoming invoice, an instance of the payment run process will potentially address a large number of invoices at once. Accordingly, the cardinality of process input or output objects varies along chains of up- and downstream processes [73]. Moreover, while Sample Process A constitutes an example of a decision-related administrative process, Sample Process B is comparable to typical production processes requiring no decision-taking.

**Example 7 (Sample Process B: Payment Run).** The process starts with selecting appropriate open items (i.e., approved invoices) from the ERP data base (activity B1) and generating an empty payment list (B2). Then, for each open item selected, it is checked whether the due date has been reached depending on the payment terms (B3). If the invoice is due for payment, it is checked whether all master data (e.g., bank account numbers) required for the payment are available (B4). If this is not the case, the relevant data have to be collected and entered (B5). Then, the respective entry in the payment list is generated (B6). After performing these steps for each open item, the payment list is approved (B7) and sent to the bank (B8). Then, a new set of open items representing the outgoing payments is created (B9). In a follow-up process, the bank statement can be matched against these open items.

To ensure that results are also applicable to domains where PAISs are not as common yet, a third sample process from the field of healthcare is also included.

**Example 8 (Sample Process C: Medical Examinations).** In the alternate example from the healthcare field, a medical examination A is performed (activity C1). Based on its result, a drug is applied (C2), and it is decided whether to perform a second examination

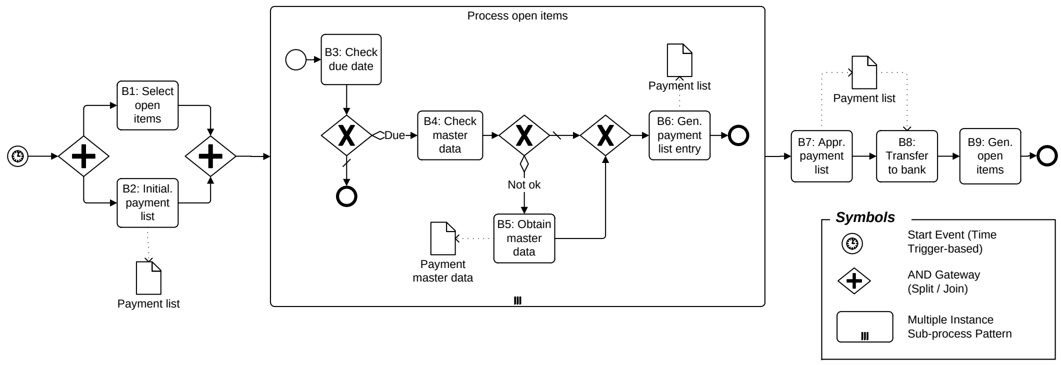



Figure 2.6: Sample Process B: Payment Run

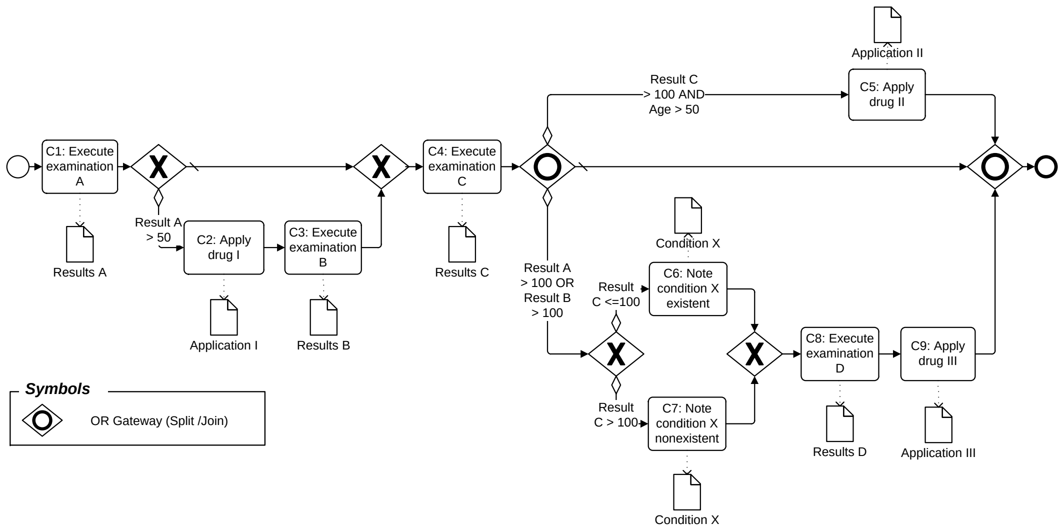



Figure 2.7: Sample Process C: Medical Examinations

B (C3). A third examination C, which may only be carried out once examination A is completed, should follow in each case (C4). Thereafter, another drug is applied depending on the result of examination C (C5) and the age of the patient. In parallel, further steps are performed depending on the results of examinations A and B: First, the existence or non-existence of condition X is noted dependent on the result of examination C (C6, C7). Then, a fourth examination D is performed (C8). After completing examination D, application of a drug is required (C9).

# 3 Research Methodology

Business processes aim at achieving business objectives of the organization in an economic context [37, 38, 14]. This characteristic, which can be described as the *utilitarian nature* of business processes, is common to major BP definitions (cf. [65]). Accordingly, the concept of *BP quality* and associated methods like quality assessment or improvement, as well as BPM in general, are considered as means to support this goal. This implies that BP quality is a *goal-bound artificial construct* that can be addressed by the *design science paradigm* [103, 77]. Therefore, the respective research principles are applied to the research methodology employed in this thesis. In this regard, certain characteristics of the design science paradigm are of particular relevance:

- Design science addresses *design problems* as opposed to *knowledge problems* in natural science. In other words, design science does not ask for the truth or describe *what is*, but it asks for solutions or describe *what should be*. For BP quality, this means that meaningful research cannot be based on building a theory on what will impact quality and validating it by empirical means – BP quality is a mental concept that cannot be readily measured by observing organizational reality. Instead, it is necessary to first find a definition reflecting the needs of organizations.
- To maintain scientific rigor, even when the theorize-and-test pattern of natural science cannot be applied, design science requires alternative means of validating results. To this end, *effectiveness criteria* are deducted from the goal pursued by the design problem. Design results are then validated by determining whether they fulfill applicable effectiveness criteria.
- Design problems and knowledge problems often exhibit a nested structure [104]. As an example, consider the definition of effectiveness criteria. These can be understood as characteristics supporting the aspired state. Accordingly, finding them is a knowledge problem, because one will ask what *does* support the goal, not what *should* support it. Being aware of this circumstance and the implications for appropriate research methodologies is another prerequisite to maintain scientific rigor.

Accordingly, the remainder of this chapter is structured as follows. First, *effectiveness criteria* for assessing results are developed. Then, the design artifacts and design procedures are aligned to the outline of the thesis, thus describing the underlying methodology. Finally, the resulting characteristics are matched against guidelines for applying design science to information systems research to ensure the validity of the approach [105].

## 3.1 Effectiveness Criteria

In design science, the value of design artifacts is to be judged "against criteria of value or utility" [77]. In this thesis, the utility of an artifact is subsumed as its *effectiveness*.

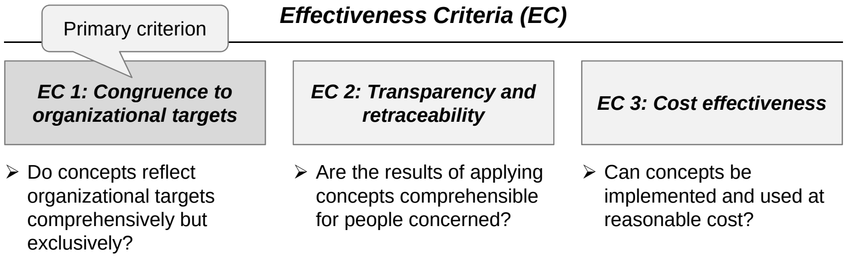



Figure 3.1: Summary of Effectiveness Criteria

Consequently, appropriate *effectiveness criteria* constitute a central part of the underlying research methodology. They are applied to evaluate existing approaches as well as the contribution of this thesis.

To obtain appropriate effectiveness criteria for BP quality artifacts, their relevant application context is considered. BP quality artifacts are to be employed in the context of BPM activities as defined in [13]: design, enactment, control and analysis of processes. Out of these, analysis and control constitute the most relevant fields: the quality of business processes is assessed and analyzed (either in the productive stage or before), and control is exercised by feeding back into design and execution, with the goal of achieving improvements. Therefore, effectiveness criteria are to be derived from general requirements towards effective managerial analysis and control, which are then narrowed down to reflect the specific demands of BPM as the relevant field of action.

According to Epstein and Henderson, the notion of performance measures as constituents of control instruments corresponds well to managerial analysis and control in the context at hand [106]. BP quality can be understood as a particular manifestation of performance measures in general. In this regard, [106] identifies “goal congruence”, “perceived fairness”, and “cost of computation” as effectiveness criteria. The following sections refine these criteria to apply them to BP quality. Figure 3.1 summarizes the resulting refined criteria including an underlying guiding question for each criterion.

### 3.1.1 From Goal Congruence to Congruence to Organizational Targets

Misalignment between performance measures and goals of the organization will invariably entail critical corporate governance issues since common management control and performance measurement systems incorporate explicit feedback loops between measures and management (re-)actions. Accordingly, measures must reflect desired actions [107]. It is not surprising that empirical results confirm that the content of performance measures impacts managerial behavior and decisions [108, 109] (“What gets measured, gets done”).

With regard to BPM, the notion of aligning performance measures and managerial control instruments to organizational goals retains its validity: analysis of BP performance, for instance, must reflect the goals the organization associates with the process. Accordingly, for the purpose of effectiveness assessment in the field of BPM, the notion of goal congruence

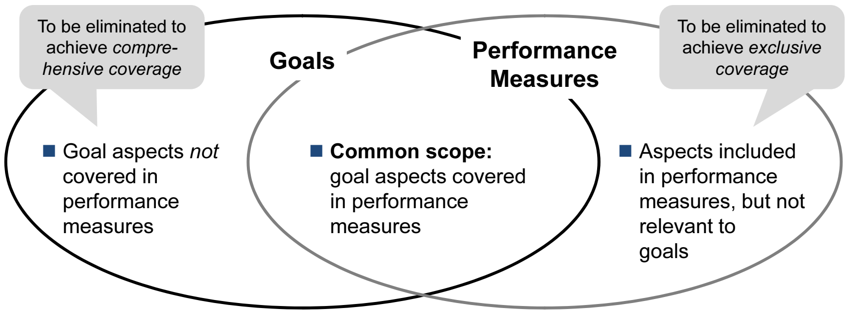



Figure 3.2: Goal Congruence: Comprehensive, but Exclusive Coverage

is refined to consider two closely interrelated aspects (cf. Figure 3.2): *comprehensive* but *exclusive* coverage:

- *Comprehensive coverage:* A notion of BP quality *must* reflect the full spectrum of goals associated with business processes.
- *Exclusive coverage:* A notion of BP quality *must not* reflect characteristics that are not related to goals associated with business processes.

Note that both comprehensive and exclusive coverage cannot always be fully achieved. For example, it might be difficult to properly delineate organizational responsibilities in quality measures, e.g. the impact of faulty IT systems developed by another department. In this case, it is important to mitigate the issue as far as possible by transparently marking shortfalls in comprehensive and exclusive coverage. In the context of BPM, it is moreover important to recognize that the concept of “goals”, “objectives” or “targets” occurs on distinct semantic levels. General *organizational targets* reflect both economic considerations common to most enterprises and non-profit organizations [110] as well as the strategy of the individual organization [87]. Examples include achieving a market leader position for a specific product category or providing best-in-class service to employees. In general, organizational targets cannot be fulfilled by a particular process alone, but require a coordinated effort across the boundaries of various processes and departments.

Moreover, the term *business objectives* can be used to represent notions like objectives, goals, or results as included in many definitions of the term “business process” (e.g., [14, 17]). Business objectives refer to the desired outcome an organization seeks to achieve with a particular business process, either through the totality of all process instances [14] or by enacting a single process instance.[1] As examples, consider the appropriate filing of all insurance contracts that occur in an enterprise or the correct posting of an individual supplier invoice. Accordingly, business objectives consider the “what” in the sense of the end result of a process. However, they do not reflect the “how” in terms of constraints or side conditions to be considered when designing and enacting a process. Most notably, this limitation applies to economic aspects. A business process may achieve its business objective, but still be far from being viable from an economic point of view. This observation clearly indicates that

[1] Note that for multiple instances of the same process definition, the desired outcome may differ. This issue occurs, for instance, if the process comprises decisions.

goal congruence as an effectiveness criterion cannot refer to business objectives only, but must consider a wider array of organizational targets like, for example, minimizing cost.

Thus, goal congruence as an effectiveness criterion for BP quality concepts refers to organization level goals. Beyond the process level goals or business objectives considered in definitions of the term business process, BP quality measures must reflect the goals an organization assigns to BPM, independent of a particular business process. To enhance clarity, the respective effectiveness criterion is thus referred to as *Congruence to organizational targets*. Business objectives specific to particular business processes will be discussed in Chapter 6.

In accordance to [111], an approach to BP quality thus satisfies the criterion of *Congruence to organizational targets* if maximizing BP quality amounts to maximizing the contribution to organization targets that can be achieved within the scope of influence of the business process.

### 3.1.2 From Perceived Fairness to Transparency and Retraceability

Organizational acceptance is a major prerequisite to actually leverage analysis results in terms of improving BP design and enactment. The most common impediment in this respect is that stakeholders may find that measurement and analysis results do not reflect the actual performance of business processes. This topic is exemplified by the acceptance issues commonly encountered in quantitative benchmarking projects.[2]

The still unbroken tendency of enterprises to link executive remuneration to performance measurement [26, 27] emphasizes the importance of designing measures to be perceived as *fair* by stakeholders. Moreover, the topic has been included the International Financial Reporting Standards (IFRS) as the concept of a “true and fair view” [112, paragraph 46].

Perceived fairness can be fostered by applying an appropriate *standard* to performance or, as in the context of this thesis, BP quality assessment [113]. An assessment standard can be regarded as appropriate if it fulfills the following requirements:

- *Comprehensive, but exclusive coverage:* To be perceived as fair by stakeholders, the delimitation of standards should reflect the scope of influence of the object or person whose performance is subject to assessment. On the one hand, performance measures should avoid focusing on particular aspects that are well measurable (e.g., cycle time) without considering more complex topics (e.g., the complexity of cases handled). On the other hand, uncontrollable factors should not influence measurement (e.g., the provision of master data by other departments) [113]. This sub-criterion is also reflected in *Congruence to organizational targets*.
- *Transparency:* To actually achieve perceived fairness, the performance measurement procedure should be transparent to stakeholders. This requirement applies to both measurement criteria and methods. For instance, the concept of “jargon” in [114] refers to using terms that are common to the respective field of application and therefore understandable for stakeholders.

[2] Acceptance issues in benchmarking projects refer to the inclination of responsible project managers to reject negative analysis results by pointing out differences to the peer group, e.g. in upstream business processes, available information systems, etc.

- *Retraceability:* Performance assessment should not be influenced by arbitrary decisions made during assessment procedures [113]. In other words, an impartial observer should be able to retrace the assessment procedure, and conclude with comparable results.

Since comprehensive, but exclusive coverage as the first sub-criterion discussed is already covered by the *Congruence to organizational targets* effectiveness criterion, *perceived fairness* can be refined to *Transparency and retraceability* for further considerations.

### 3.1.3 From Cost of Computation to Cost Effectiveness

The aspect of computation cost as a criterion to appraise performance measurement or, in general, organizational control methods has been developed with the concept of bounded rationality [115, 116]. The criterion reflects efficiency considerations generally valid for measurement and analysis in an economic context: cost and effort incurred may not exceed benefits gained. If this criterion cannot be fulfilled, the practical applicability of any approach towards performance assessment is severely impeded. This topic has, for instance, been included in the International Financial Reporting Standards as well [112, paragraph 44]. It closely corresponds to the demand for scalability, which has been discussed in the motivational theses for this work (cf. Section 1.2).

For the purpose of assessing BP quality concepts, it will not be possible to concisely determine cost of computation since results strongly depend on factors specific to an organization, such as BP complexity, available tools, and factor costs. The criterion is thus substituted with the more general *Cost effectiveness* criterion, covering the following sub-criteria:

- *Formalization:* The formalization of concepts and their relations is a major prerequisite for automating assessment procedures, which may significantly reduce manual efforts.
- *Avoidance of redundant measures:* Individual aspects considered in performance assessment should not "overlap" in the sense of being semantically interdependent (i.e. dependent on each other or on a common root cause). Note that, in this case, redundant measurement would cause unnecessary additional effort.
- *Integration with common BPM approaches:* Developed concepts should integrate well with existing BPM approaches to foster re-use of available methods and tools. Besides avoiding one-off effort incurred in the development of new methods and tools, this may increase the degree of automation achievable in practice.

### 3.1.4 Summary: Effectiveness Criteria

Effectiveness criteria constitute a central aspect to the design science paradigm pursued in this thesis. Table 3.1 summarizes considerations in this respect.

This thesis considers Effectiveness Criterion 1 (*Congruence to organizational targets*) as the primary criterion since it can be derived directly from the fundamental requirement towards managerial analysis and control, i.e., to support the organization's targets. Any misalignment between a notion of BP quality and organizational targets will entail organizational governance flaws as soon as BP quality is applied to practical use cases. *Congruence to*

| **Effectiveness Criteria (EC)** | **Rationale** | **Implications** |
|---|---|---|
| **EC 1: Congruence to organizational targets** | Common management control and performance measurement systems exhibit explicit feedback loops between measures and management (re-)actions. Accordingly, measures must reflect desired actions [107].<br><br>Empirical results confirm that the content of performance measures impacts managerial behavior and decisions. [108, 109]. | *Comprehensive coverage*: BP quality artifacts should reflect the full range of organizational targets associated with business processes.<br><br>*Exclusive coverage*: BP quality artifacts should not address issues that are not related to organizational targets associated with business processes.<br><br>As a fallback solution if full congruence cannot be achieved, *transparency on deficiencies* should be given to mitigate defective governance effects. |
| **EC 2: Transparency and retraceability** | Recognizing organizations as a social environment is prerequisite for successful staff motivation and change management [114].<br><br>Performance measures are commonly used for individual target setting and remuneration [26].<br><br>For financial reporting, an equivalent provision has been long recognized with the "true and fair view" [112]. | *Transparency*: accountable managers' and stakeholders should be able to understand the link between status, actions and assessment results.<br><br>*Retraceability*: assessment results should be reproducible for stakeholders and independent observers. |
| **EC 3: Cost effectiveness** | Practical applicability in an economic context depends on an appropriate relation between effort incurred and gains made possible by performance assessment.<br><br>This demand has been recognized for a long time in the bounded rationality approach [115].<br><br>Again, an equivalent provision exists for external (e.g., capital markets) financial reporting [112]. | *Formalization*: concepts should be formalized to allow for automation of related procedures.<br><br>*Avoidance of redundant measures*: performance measures should not overlap to eliminate redundant assessment effort.<br><br>*Integration with common BPM concepts*: new concepts should integrate well with existing and proven approaches to enable re-use of methods and tools. |

Table 3.1: Effectiveness Criteria

*organizational targets* in part predetermines the research methodology: applying organizational targets to the scope of influence of business processes is the starting point to derive BP quality artifacts.

Effectiveness Criteria 2 and 3, in turn, are concerned with the practical applicability of analysis results (*Transparency and retraceability*) and execution issues (*Cost effectiveness*), respectively. They constitute secondary effectiveness criteria since both are severely impeded if the primary effectiveness criterion is not met.

## 3.2 Course of Action

This section summarizes the course of action that ensues from the design science paradigm with regard to the research objectives of this thesis. It matches design procedures and artifacts with its respective chapters.

As a preliminary step to detail the motivation for proposing an alternative approach towards BP quality, Chapter 4 substantiates the claim that available approaches are not yet fully effective from a management perspective. To this end, a literature review based on the effectiveness criteria set out in Table 3.1 is conducted.

The further steps of the employed methodology are organized around the design processes of *build* and *evaluate*, and the research outputs (or design artifacts) of *constructs*, *models*, *methods*, and *instantiations* [77]. The additional research activities of *theorize* and *justify* are employed to properly treat emerging "nested" knowledge problems as described above, and to validate results by integrating them into an exemplary application scenarios. Figure 3.3 summarizes the interactions between design artifacts, their respective contributions, design processes, effectiveness criteria, and the research objectives described in Section 1.5 with regard to BP quality: Effectiveness criteria are applied to the design artifacts developed in this thesis, which are categorized into *constructs*, *models*, *methods*, and *instantiations*. As demanded by Hevner et al. [105], design artifacts deliver *contributions* discussed in more detail below. Each category of design artifacts is *built* on the basis of the preceding categories. In turn, *evaluation* occurs in reverse order: since design artifacts should be judged against "criteria of value or utility", they are evaluated by using them to develop subsequent artifacts while considering appropriate effectiveness criteria as defined in Table 3.1.

This structure can be matched against Research Objectives 1 to 3: To fulfill Research Objectives 1 and 2 (i.e., BP quality analysis and quality control) it becomes necessary to build constructs, models and methods. Research Objective 3 (i.e., demonstrating the effectiveness of concepts developed) additionally requires an instantiation. Sections 3.2.1 and 3.2.2 summarize how the research methodology lined out in Figure 3.3 is reflected in the structure of this thesis.

### 3.2.1 Building Business Process Quality Artifacts

Part II *builds* concepts around BP quality. In accordance to both the primary effectiveness criterion defined in Section 3.1.1 and the initial considerations of this chapter, Part II follows a deductive methodology. The deductive methodology spans both the derivation of a concise

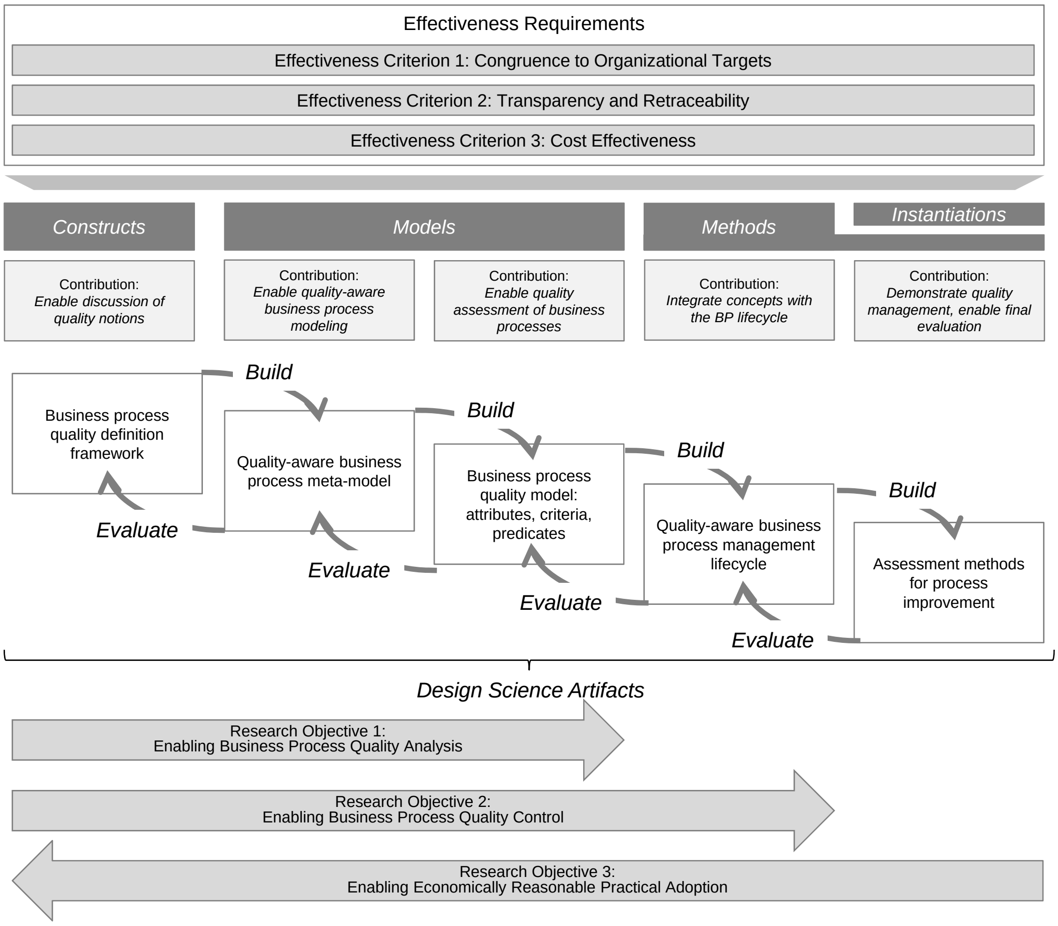



Figure 3.3: Design Methodology

notion of BP quality from organizational targets and the scope of influence of business processes (cf. Chapter 5), as well as the subsequent deduction of corresponding *model*, *method*, and *instantiation* artifacts.

Chapter 5 provides a definition framework for BP quality. In terms of design artifacts, this represents a *construct*: as its contribution, it provides common ground to discuss the underlying apprehension of the term *quality* and how it should be applied to BPM.[3] Moreover, it constitutes a means to facilitate the derivation of appropriate sets of quality characteristics which. On the one hand, these reflect the definition framework in terms of content. On the other hand, they properly consider effectiveness criteria as described in Table 3.1.

Chapter 6 addresses the formalization of business objectives as a relevant prerequisite to quality modeling. Note that this issue has not been sufficiently covered in existing BPM approaches yet. Based on this, Chapter 7 develops a meta-model for *quality-aware BP models*

[3]The special relevance of enabling this discussion will become apparent when considering the various notions of quality found in both literature and application scenarios. Chapter 2 provides a respective summary.

by deriving required semantics from the definition framework for BP quality, and extending available process modeling mechanisms accordingly. Thus, the meta-model allows us to model business processes in a way providing sufficient semantic content to assess process quality.

Chapter 8 discusses characteristics of business processes relevant to quality. These are derived from considerations on organizational targets and the scope of influence of business processes. According to the design science approach, a properly structured set of quality characteristics constitutes a *model* since it interrelates aspects of the BP quality definition framework to characteristics of both process models and process instantiations which can be evaluated in practice. Part II concludes with Chapter 9 which integrates quality concepts into the BP lifecycle in the sense of a *method*. In addition, Chapter 9 discusses the integration of requirements posed by BP quality management into process-aware system landscape architectures. This is achieved by matching necessary and desirable capabilities against common components such as workflow management systems [14] or process mining tools [28].

Altogether, Chapters 6-9 provide the required concepts to achieve Research Objectives 1 and 2 (i.e., BP quality analysis and control): Analysis requires an understanding of BP quality as a *construct* as well as *model* and *method* artifacts enabling quality assessment.[4] Control can be executed if analysis capabilities are integrated with BP lifecycle concepts to enable feedback into process design & implementation, and process enactment.

The course of action lined out exemplifies the nesting of design and knowledge problems as discussed above: while the overall concept of BP quality constitutes a goal-bound artifact, the deduction of lower-level *constructs* (i.e., an appropriate definition framework) and their evolution into *models*, *methods*, and *instantiations* can be viewed as knowledge problems since each artifact is derived from preceding concepts. However, note that the deductive approach is subject to effectiveness considerations. For instance, if Effectiveness Criterion 3 (i.e., *Cost effectiveness*), is impeded, it is possible to deliberately deviate from finding the "true" *model* interrelating given *constructs*. This freedom to choose is a typical characteristic of design science.

### 3.2.2 Validating Results

Chapter 9 concludes Part II with a discussion on how BP quality management can be integrated in day-to-day BPM practice and the corresponding application landscape. This constitutes a *model*, but also an *instantiation* in the sense of a "realization of an artifact in its environment" [77]. As the "value or utility" of an artifact must always be considered in terms of integration with application conditions, the contribution of Chapter 9 also lies in facilitating an initial validation of results.

Beyond Chapter 9, Part III documents additional *theorize* and *justify* procedures with respect to the results developed. In Chapter 10, the results of a field experiment addressing the implications of the BP quality definition framework are discussed. Note that this field experiment was conducted early on, on the basis of the quality definition framework, but

[4]Thus, analysis transcends assessment since it requires understanding of the underlying notions and terms to enable interpretation of results. Assessment, however, can be executed routinely *without* understanding the underlying concepts.

not under consideration of subsequently developed artifacts. Thus, it exemplifies a characteristic often encountered in design science research: *theorize* and *justify* procedures are nested, which leads to a step-by-step evolution of solution approaches [104]. Chapter 11 tests whether the employed top-down methodology leads to contributions beyond available BP reengineering and BP quality approaches in terms of aspects covered by the quality model. Chapter 12 develops an approach for validating the effectiveness of individual process improvement measures in given application scenarios. Since quality attributes can be directly converted to process improvement measures based on quality assessment results, this is an important step to ensure practical applicability of the results presented, in particular with respect to Research Objective 3 (i.e., *Enabling Economically Reasonable Practical Adoption*) (cf. Section 1.5). Chapter 13 finally evaluates the results presented in this thesis against the effectiveness criteria developed in Section 3.1, and concludes the thesis with an outlook on future potentials.

## 3.3 Evaluation against Guidelines for Design Science in Information Systems Research

To ensure the viability of the approach underlying this thesis, its characteristics are matched against the guidelines for applying design science principles to information systems (IS) research [105]. This procedure also provides valuable pointers to maintain standards relevant to design science research in further chapters.

**Guideline 1 (Design as an Artifact).** Design science research is required to "produce a viable artifact in the form of a construct, a model, a method, or an instantiation." The structure lined out in Figure 3.3 meets to this demand.

**Guideline 2 (Problem Relevance).** Design science research should address business issues. In this regard, the relevance of the topic of this thesis is demonstrated through exemplary application scenarios (cf. Chapter 1), and through the evaluation of results based on business-oriented effectiveness criteria.

**Guideline 3 (Design Evaluation).** The requirement for rigorous demonstration of "utility, quality, and efficacy" is addressed through the application of concise effectiveness criteria deducted from management demands (cf. Section 3.1).

**Guideline 4 (Research Contributions).** Based on the presented motivational theses, research objectives are derived in Chapter 1. in turn, these are reflected in the research methodology described in Section 3.2. The individual contributions of design artifacts, as mapped in Figure 3.3, are consistently relevant to the research objectives of this thesis.

**Guideline 5 (Research Rigor).** In the *build* procedure towards the design artifacts, a rigorous deductive approach is employed to ensure conformance to management requirements. In addition, evaluation of both available work and final results is executed based on concise effectiveness criteria.

**Guideline 6 (Design as a Search Process).** The search process of the design science paradigm should be based on "utilizing available means" and "satisfying laws in the problem environment". These topics are addressed by leveraging a deductive approach under consideration of the relevant outer environment [103]. A more detailed discussion on this issue is included in Chapter 5.

**Guideline 7 (Communication of Research).** Effective presentation of research results is part of the challenge for Part III. Multiple discussions of emerging topics with BPM practitioners have been employed to validate results. In particular, this applies to the described methodology to validate potentials for concise application scenarios (cf. Chapter 12).

Summarizing the considerations lined out above, the methodology employed in this thesis is aligned to accepted principles of design science research. This characteristic supports the validity of results in terms of relevance, contribution, and practical applicability.

# 4 State of the Art

Both quality management and BPM constitute wide and well-established fields of research and practical adoption. Accordingly, this thesis touches – and builds upon – a broad body of available results ranging from considerations on the governance and control of organizations to BP modeling languages. This chapter focuses on available work related to BP quality as a whole in the sense of a state of the art discussion with respect to the research objectives of this thesis.

Work related to the quality of business processes can be broadly divided into three categories as summarized in Figure 4.1: general management approaches that are also applicable to BP quality, BPM frameworks, and BPM research addressing individual aspects related to quality. Note that related work pertaining not to BP quality as a whole, but to individual aspects such as business objectives is discussed in the respective chapter for the sake of readability.

As stated in the motivation to this thesis, it is postulated that existing approaches towards BP quality are not yet fully effective from a management perspective. Accordingly, this chapter discusses related work with respect to Effectiveness Criteria EC 1 to EC 3 (cf. Table 3.1), and presents a final summary including conclusions inferred for this thesis.

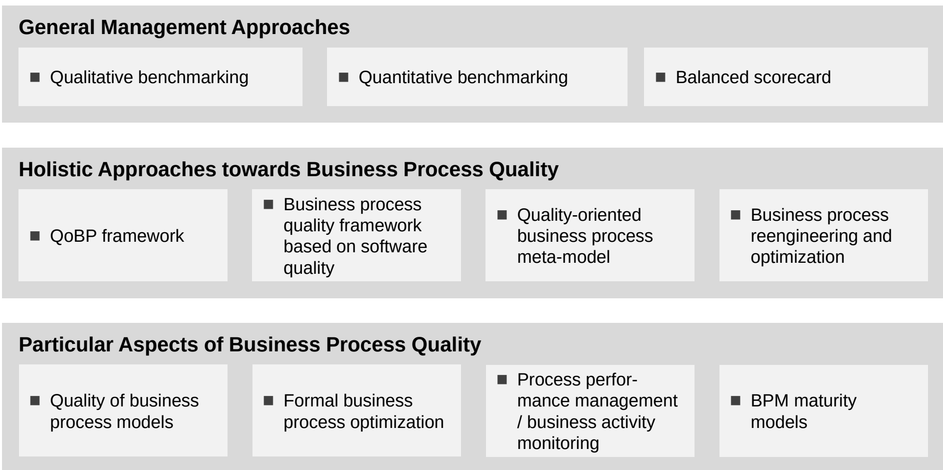



Figure 4.1: State of the Art Overview

## 4.1 General Management Approaches

There are many management concepts that are not specific to the BPM field, but might be adapted to it. Two approaches are discussed due to their widespread adoption as well as their special relevance to BP quality.

*Benchmarking* utilizes experience and knowledge available from comparable business processes. *Qualitative benchmarking* matches the actual situation against known good practices. Depending on the concrete application area, good practices may be documented in frameworks pertaining to a particular industry or organizational function. As examples consider the Control Objectives for Information and Related Technology (COBIT, [117]) or the IT Infrastructure Library (ITIL, [118]) in the field of information management, or the Supply Chain Operations Reference model (SCOR, [119]) in supply chain management. Practices may relate to general organizational structures or to concrete aspects of business processes or information systems (cf. Example 9).

In turn, *quantitative benchmarking* uses key performance indicators (KPIs) to measure process aspects. In particular, this approach enables the comparison with results from peer organizations [45]. It can be traced back to the analysis of financial indicators [120]. An approach to rank an available set of KPIs for a particular business process (in this case, strategy development) has been proposed by Nestic et al. [121]. The approach is based on fuzzy sets and genetic algorithms, and requires a substantial set of empirical data as input for the ranking procedure.

**Example 9 (Good Practice in Process Design and Key Performance Indicators).** Consider Sample Process A from Figure 2.5, i.e., the process of handling supplier invoices. Here, good practices for qualitative benchmarking include the use of early scanning (also known as “intelligent scanning”) and EDI as IT-based practices, and credit note procedures as an organizational practice. The use of credit note procedures has been described in detail as an example of BP reengineering in [17, 18].

Quantitative key performance indicators include the number of invoices processed per full-time personnel resource and year, the processing cost per invoice, and average cycle time.

The *balanced scorecard* approach is used to measure and control organizational performance based on multiple dimensions: the “financial”, “customer”, “innovation and learning”, and “internal business” perspectives [122]. KPIs are specifically developed for the organization and assigned to each dimension. Compared to traditional financial performance measures, the balanced scorecard recognizes that financials are always backwards-oriented, and provide little clarity on an organization’s future perspectives. Moreover, organizational goals are often contradictory, e.g., when considering cash flow maximization against the need for investments. This issue has been acknowledged in literature for a long time [116], and has been addressed via the multiple dimensions of the balanced scorecard, i.e., the approach does not try to combine everything into one single perspective. Applying the original concept to business processes would require substantial adaptation starting with the fundamental scorecard perspectives, as these are defined to encompass all performance aspects of an organization instead of focusing solely on business processes (which are considered as part of the “internal business” perspective). However, the basic idea of treating multiple performance

| **Business Process Quality Definition** | **Quality Attributes / Criteria** | **Evaluation vs. Effectiveness Criteria** | | |
|---|---|---|---|---|
| | | Congruence to Organizational Targets | Transparency & Retraceability | Cost Effectiveness |
| *Qualitative Benchmarking* [117, 118, 119] | | | | |
| Implicit: degree to which good practices are implemented | Criteria: implementation of good practices known from peer organizations | Low: focus on copying peer strategies without considering the specific environment | Low: failure to consider organizational constraints (e.g. capital expenditures) | High: easy assessment of good practices implementation |
| *Quantitative Benchmarking* [45, 121] | | | | |
| Implicit: degree to which peer key performance indicator values are achieved | Criteria: comparison to key performance indicator values achieved at peer organizations | Low: focus on efficiency measures without considering capital expenditures or quality of process input | Low: efficiency measures typically do not reflect non-manageable factors (e.g. capital expenditures or quality of process input) | High: key performance indicators are typically chosen to be easily assessable |
| *Balanced Scorecard (with adaptations to BPM application)* [122] | | | | |
| Degree to which objectives in target dimensions (typically, four) are achieved | Criteria: achievement of objectives defined for measures | High: objectives and measures are derived from organizational targets | Dependent on definition of manageable scorecard dimensions (classic dimensions appropriate for business units) | High: measures are typically chosen for high assessability |

Table 4.1: Evaluation of General Management Approaches

dimensions as orthogonal instead of trying to find an absolute single measure of quality may be unavoidable for practical application.

Table 4.1 summarizes general management approaches with respect to effectiveness criteria.

## 4.2 Holistic Approaches towards Business Process Quality

BPM research has led to a number of proposals that might be applied to BP quality as well. As opposed to benchmarking and the balanced scorecard, respective approaches abstract from the business content of the processes in question. In other words, a person charged with executing a corresponding approach does not necessarily need to be a business subject matter expert (i.e., be familiar with the application field of the business process). In addition, these propositions aim at a holistic view of BP quality instead of focusing on individual aspects.

An attempt to develop a *"Quality of Business Processes (QoBP) framework"* focusing on process models was made by Heravizadeh et al. [123, 124]. BP quality is defined in terms of 41 quality dimensions derived from literature, e.g. in the field of software engineering. At first glance, the topics addressed in [123] resemble the subject of this thesis. However, there is an important difference regarding research objectives. While [123] focuses on a *quality-aware BP lifecycle*, the primary scope of this thesis is to assess the *quality of business processes per se*. In line with the characteristics of design science, this difference leads, firstly, to different criteria that are used to evaluate resulting design artifacts, and thus, ultimately, to a widely diverging view of BP quality. Hence, the difference in underlying research objectives is to be kept in mind when considering the evaluation of the QoBP framework in terms of the effectiveness criteria used in this thesis.

The QoBP framework approach does not show the interrelation of the quality dimensions to organizational targets or to a formal definition of BP quality in the sense of a design science *construct*. This means that it is not possible to determine whether the dimensions are complete or how to actually evaluate overall process quality. The quality dimensions are arranged along the categories of function quality, input / output quality, non-human resource quality, and human resource quality. Regarding effective quality assessment, in the context of this thesis this is questionable since it mixes up the quality of a process with factors not controllable by process management. In practical settings, this leads to issues with the *Congruence to organizational targets* (consider, in particular, the *exclusivity* sub-criterion), and the *Transparency and retraceability* effectiveness criteria. As an example, consider the inclusion of input quality in the QoBP framework: input quality is not governed by the process manager. Therefore, effective assessment of BP quality requires properly delineating the quality of process input. In other words, BP quality must be judged *independently* from input quality.

The QoBP approach has been presented in more detail in [124]. It defines quality as "non-functional, but distinguishing" characteristics of a business process. This thesis does not concur with that view. From the perspective presented in Section 3, excluding the business objective of a process would neglect the goal-bound character of any BP quality construct as a design science artifact. This means that, from the point of view assumed in this thesis, a business process not achieving its given business objective cannot be considered as a high-quality process.

Heinrich and Paech proposed a *BP quality framework* based on software quality [125]. While work on software quality is not the only source used, the eight main "activity characteristics" (as well as 27 sub-characteristics) were derived from this field. The "activity characteristics" are amended by four characteristics in the areas of "resource" and "actor". Similar to QoBP, this approach lists various quality characteristics, but does not integrate them into a comprehensive formal quality definition, leading to similar issues as described above. Moreover, it is stipulated that the applicability of software engineering results to design problems in the BPM area still requires in-depth analysis and discussion.

An approach towards including quality factors in BP models by defining a *quality-oriented BP meta-model* has been proposed by Heidari et al. [126, 127]. Similar to the approaches described above, this proposition lists quality factors grouped into dimensions, but does not base these on a concise definition of BP quality. Again, this leads to issues regarding the delineation between the quality of a business process and characteristics of its organizational or BMP environment (cf. Example 10). Moreover, there still seem to be unresolved overlaps between the quality factors identified (e.g., *cost* vs. *cost efficiency* or *execution time* vs. *timeliness* vs. *time efficiency*). In addition, there are no quality factors referring to process output or the business objective of the process, i.e., whether the process enables achieving a given objective does not affect its quality assessment according to [126].

**Example 10 (BP Quality and the Organizational Environment).** Consider the "cost" quality factor as defined in [126]: in reality, the cost of enacting process instances is not only determined by BP design, implementation and enactment, but also by the cost of production factors made available by the organization. This renders the absolute cost of enacting a business process inappropriate as a measure of BP quality. Manufacturing processes in the automotive industry provide an example for this issue. Today, these processes are highly standardized across plants throughout the globe based on "production systems" specific to the various manufacturers [128]. Nevertheless, cost differences will occur due to locally specific factor costs. Following the approach of treating cost as an important quality factor will thus result in widely differing quality assessments for one and the same business process. Similar considerations apply to other quality factors proposed in [126] such as "timeliness" or "availability".

Dumas et al. included quality as one out of four *process performance dimensions* besides time, cost, and flexibility [21]. The authors' view of *external quality* as the degree to which clients of the process are satisfied with the resulting service or product corresponds to Garvin's product-based quality view (cf. Section 2.2). However, their conception of *internal quality* can be viewed as more in line with Garvin's user-based quality perspective. In comparison, this thesis is based on a more comprehensive notion of quality, including, for example, cost as a quality issue as well.

*BPR and optimization* constitutes an area closely related to the optimization of BP quality. Hammer and Champy [38] as well as Davenport [37] provide good examples for the "traditional" reengineering view. BPR approaches commonly comprise best practices and other informal methods mostly based on anecdotal evidence. Reijers et al. systematized and evaluated BPR practices based on literature reviews and empirical research [129, 130, 131]. When evaluating the results of this thesis, this synopsis of BPR allows comparing corresponding aspects developed from a BP quality perspective to existing process improvement methods (cf. Chapter 11).

To support "the process of process redesign itself", Nissen developed a tool-based method to promote "process measurement, pathology diagnosis, and transformation matching" as the key intellectual activities required in process reengineering [132, 133]. In this context, "transformations" refer to abstract process model reengineering practices that can be used to address "pathologies" diagnosed for the process [129]. The main goal of this effort was to reduce both the cost and duration of reengineering projects. Nissen's use of so-called measurement-driven inference techniques to automatically derive conclusions on desirable process adaptations from process measurement can be viewed as anticipating major characteristics of later process intelligence approaches (cf. Section 4.3). Note, however, that

Nissen's approach is based on measuring the process model instead of the process enactment logs used in process intelligence. The process measures defined by Nissen can thus also be considered as a set of *quality drivers* as described in Chapter 8.3.

Speck and Schnetgöke [134] and, with a focus on process models, Becker et al. [135] constitute additional examples for optimization based on informal methods. This view is also reflected in the OMG Business Process Maturity Model [136] and other BPM maturity models [137], which suggest criteria to allocate business processes to maturity levels without, however, providing evidence on how this structure has been devised.[1] While this informal character fits well with practical applicability, there is still a lack of an overarching comprehensive model to ensure causal relations between measures recommended and intended results as well as completeness with regard to the coverage of quality aspects.

Table 4.2 summarizes holistic approaches towards BP quality with respect to effectiveness criteria.

## 4.3 Particular Aspects of Business Process Quality

In the field of BPM, a great number of approaches have been developed to address particular quality aspects of business processes. While they do not aim at an overarching construct of BP quality, they provide important methods for practical BP quality management.

There exists work that deals with the *quality of BP models*. In this context, Lindland et al. discern syntactic, semantic and pragmatic model quality as distinct aspects [139].[2] These aspects have been used by Reijers et al. to develop the SIQ framework[3], which integrates a broad range of propositions from the field of BPM that can be used to manage BP model quality [142]. *Syntactic model quality* deals with the question whether a model is valid in terms of the modeling language or meta-model it refers to. With respect to syntactic quality of process models, van der Aalst introduces soundness of Workflow Nets [143]. Mendling assesses formal errors in event-driven process chain (EPC) BP models [144] in an automated approach. Hallerbach et al. discuss how to ensure soundness for an entire process family [145]. Reichert et al. enhance these considerations by also considering soundness in the context of dynamic process changes during run-time [41].

*Semantic model quality* addresses the issue whether a model appropriately describes its topic. In this context, there exist BPM approaches which aim at enhancing the description of a domain through process models: Weber et al. develop process model refactorings [146, 147]. Li et al address reference model discovery by model merging [148, 149]. Weber et al and Rinderle et al describe quality issues in respect to a case-based capturing of knowledge about reusable process adaptations which can be applied in dynamic environments [150, 151]. Research from the field of BP compliance deals with the question whether the specification

[1] An overview comprising additional maturity models in the area of BPM is provided in [138]. Additional models describing the maturity of BPM in organizations instead of the maturity of particular processes are discussed in Section 4.3.

[2] This framework has later been refined to nine aspects of model quality [140, 141]. This extension is not used here since it does not allow clearly allocating individual BPM approaches to model quality aspects addressed.

[3] The SIQ framework "is *S*imple enough to be practically applicable, yet *I*ntegrates the most relevant insights from the BPM field, while it deals with *Q*uality" [142].

| **Business Process Quality Definition** | **Quality Attributes / Criteria** | **Evaluation vs. Effectiveness Criteria** | | |
|---|---|---|---|---|
| | | Congruence to Organizational Targets | Transparency & Retraceability | Cost Effectiveness |
| *QoBP Framework* [123, 124] | | | | |
| Implicit: degree to which requirements in quality dimensions are fulfilled | Attributes / criteria: fulfillment of requirements in 41 quality dimensions (requirements are not defined) | Low: quality dimensions are not systematically linked to organizational targets, no consideration of target interdependencies | Low: quality requirements do not recognize organizational environment | Low: real-world measurability of attributes not proven, may lead to protracted assessment effort as measures are developed |
| *Business Process Quality Framework Based on Software Quality* [125] | | | | |
| Implicit: degree to which requirements towards quality characteristics are fulfilled | Attributes: twelve main quality characteristics | see QoBP framework | see QoBP framework | see QoBP framework |
| *Quality-oriented Business Process Meta-Model* [126, 127] | | | | |
| Implicit: degree to which requirements towards "quality factors" are fulfilled | Attributes: eleven quality factors organized in five dimensions | see QoBP framework | see QoBP framework | see QoBP framework |
| *Business Process Reengineering and Optimization* [38, 37, 129, 130, 131, 134, 135, 136, 137] | | | | |
| Implicit: all optimization policies have been leveraged | Criteria: implementation of optimization policies / maturity level definitions (similar to qualitative benchmarking, but independent of functional content) | Low: similar to qualitative benchmarking, but peer strategies are replaced with general optimization policies | Low: similar to qualitative benchmarking | High: easy assessability of implementation of recommended practices |

Table 4.2: Evaluation of Holistic Approaches towards Business Process Quality

and the enactment of business processes are compliant to rules and regulations [152, 153, 154, 155, 156, 157]. Work from this area is discussed in Section 6.3 since it is particularly relevant to the notion of *business objectives*, which is developed in the respective chapter. Gebhart et al. applied an initial set of quality attributes developed in the course of this thesis (cf. Chapter 10) to agile process development by integrating corresponding quality gates into modeling procedures [158].

*Pragmatic model quality* pertains to the understandability and usability of models for human actors. In this regard, Becker et al. discuss BP model quality focusing on the requirements of particular stakeholder groups and applications [135]. Based on available literature (e.g., quality in conceptual modeling [139, 159]), Mendling et al. describe a framework of informal BP modeling guidelines mainly aimed at improving the understandability (and hence the robustness of the modeling procedure against introducing errors) of the resulting models [160]. Additional propositions to enhance BP model understandability include the use of appropriate labels and icons [161] as well as syntax highlighting in BP models with the respective tool support [162, 163, 164]. Empirical studies of pragmatic model quality comprise an assessment of the overall factors impacting the understandability of BP models by Reijers and Mendling [165] as well as an empirically validated proposition to structure BP models into modules by Reijers et al. [166].

The *communication flow optimization model* was developed by Kock and Murphy as an alternative to activity-based process representations [167, 168]. The communication flow representation focuses on the flow of information between process participants and information systems instead of sequences of activities. It could be empirically established that this representation better supports human users in identifying process improvement opportunities and apply process redesign guidelines [169, 170]. It thus deals with a particular aspect of pragmatic model quality.

The effectiveness of process models to guide PAIS implementation constitutes another particular aspect of pragmatic model quality. In this regard, Guceglioglu and Demirors apply selected software quality characteristics to business processes to guide PAIS implementation [171]. Cardoso analyzes workflow complexity as one possible measure for BP model quality [172], and Vanderfeesten et al. as well as Mendling discuss quality metrics in BP modeling [173, 174, 175].

The increasing spread of process mining techniques [28] has given rise to the challenge of dealing with the quality of *mined process models* as a particular aspect of process model quality. As an alternative perspective to syntactic, semantic, and pragmatic model quality, Buijs et al. have identified *replay fitness*, *precision*, *generalization* and *simplicity* as the four dimensions relevant to the quality of process discovery algorithms or, respectively, the quality of the resulting process models [176]:

- *Replay fitness* deals with the question whether the process instances given in the execution log that is used for process mining can be reproduced using the resulting mined process model. This can be considered as a sub-aspect of *semantic model quality.*
- *Simplicity* addresses the issue of complexity in mined process models. The simpler the process model, the more accessible it will be to human users. Hence, this aspect corresponds to *pragmatic model quality.*

- *Precision* covers the topic of restricting potential process instances conforming to the mined BP models to the behavior of the business process actually covered in the process log. Accordingly, precision reflects another aspect of *semantic model quality.*
- *Generalization* addresses the question whether mined process models will be able to cover *future* behavior of the process considered, i.e., whether the mined model reflects just the instances given in the execution log, or the process in general. Generalization can thus be viewed as another aspect of *semantic model quality.*

Note that syntactic model quality is considered as implicitly given, since process mining algorithms are expected to deliver valid models "such as Petri nets, BPMN-models, EPCs, YAWL-models etc." [176]. The authors also defined metrics for each of the four quality dimensions, and proposed a process mining algorithm that allows optimizing process discovery results towards any quality dimension.

There also exist approaches to *formally optimize processes or workflows based on models* [177]. As an example, consider an approach developed by van der Aalst and van Hee [178, 143] which proposes Petri nets to leverage existing analysis methods. Beyond that, Oliveira et al. [179] apply the concept of "generalized stochastic Petri nets" to verify the correctness and evaluate BP performance aspects. In this context, the treatment of simultaneous process instances concurring for the availability of shared resources is of particular interest.[4] In addition, Hofacker and Vetschera [180] discuss various optimization strategies for process designs with given input and output sets per activity. These approaches are mainly suited to optimize control flow and resource scheduling as they do not address individual activities in terms of necessity, effort or alternatives. Thus, they constitute important tools, but cover only aspects of optimum BP design.

*Process intelligence, process performance management and business activity monitoring* are closely linked to the quality of process execution. Research in this area is often driven by industry and tool vendors. It is oriented at practical requirements and tends to take an operational, short-term view as opposed to the rather structural, long-term perspective of BP quality (e.g., [34, 36, 181]). In the context of process enactment, Grigori et al. have developed a proposal to monitor and manage exceptions in process execution [182]. Heckl and Moormann, in turn, provide a broader overview on process performance management which, beyond BPM, considers related work from the more general field of performance management and control as well [183].

Besides approaches addressing quality in terms of particular BP aspects, there exist propositions dealing with the quality of the organization in which BPM takes place. Rosemann et al. proposed a *BPM maturity model* consisting of five dimensions: *information technology and systems*, *culture*, *accountability*, *methodology*, and *performance* [184]. Each dimension should be evaluated for five distinct BP lifecycle phases: *align*, *design*, *execute*, *control*, and *improve.* This leads to an overall assessment of BPM maturity in a particular organization. Rosemann and de Bruin also proposed *strategic alignment*, *governance*, *method*, *information technology*, *people*, and *culture* as an alternative structure of factors relevant to BPM maturity [185]. For each factor, an organization may achieve one of the following stages: *initial*

[4]Note that this topic is not comprised in the understanding of BP quality as propagated in this thesis (cf. Chapter 5) since it rather pertains to the quality of underlying BPM methods as a meta-framework to manage multiple individual processes. This consideration exemplifies ideas to be clarified in the later chapters of this thesis.

*state*, *defined*, *repeated*, *managed*, or *optimised*. An overview comprising additional maturity models in the area of BPM is provided in [138].

Based on working with a group of companies and personal experience, Hammer defined the *Process and Enterprise Maturity Model (PEMM)* [86]. The PEMM identifies *design*, *performers*, *owner*, *infrastructure*, and *metrics* as *process enablers* specific to a particular business process. For each process enabler, Hammer provides criteria that can be used to evaluate a process, and to derive improvement measures. Hammer also recognizes that the notion of a "weakest link" applies to process enablers: if, e.g., there is no strong process owner, it will be difficult to successfully implement even the best process design. In addition, *enterprise capabilities* relevant to BPM in the organization as a whole comprise *leadership*, *culture*, *expertise*, and *governance*. Hence, the PEMM acknowledges the need to discern between factors to be ensured for each individual process and the maturity of the organization as a whole in terms of aspects relevant to process performance or quality. Similar to process enablers, assessment criteria are provided for the enterprise capabilities.

De Bruin and Rosemann also worked towards aligning BPM to organizational strategy by describing five relevant *capability areas* [186]. The capability areas have been identified by having potential candidates rated by researchers and practitioners, and subsequently refining the results by additional literature research and a case study. Relevant capability areas comprise *process improvement plan*, *strategy and process capability*, *enterprise process architecture*, *process measures*, and *process customers and stakeholders*. They allow operationalizing strategy in terms of BPM, i.e., transferring strategic requirements into the management of business processes.

Table 4.3 summarizes approaches from the field of BPM with implications towards particular quality aspects regarding effectiveness criteria. For approaches covering only particular aspects, evaluation against the primary Effectiveness Criterion 1, *Congruence to organizational targets* (cf. Table 3.1), as a whole is obviously not meaningful and therefore omitted.

## 4.4 Conclusion

This section summarizes conclusions based on the evaluation of existing approaches to BP quality with regard to Effectiveness Criteria 1-3 (cf. Table 3.1). Thus, it provides guidance on aspects to be considered in particular when proceeding further.

From Tables 4.1 to 4.3 it can be concluded that most approaches do not explicitly state a concise definition of BP quality. Instead, they employ either quality attributes or quality criteria. *Quality attributes* are properties that may be used to evaluate quality. Amending quality attributes with target or threshold values results in *quality criteria*. *Statements on quality* can be made based on an assessment whether quality criteria are fulfilled or not.

There are crucial implications when utilizing a formal quality definition and quality criteria as opposed to quality attributes solely. Figure 4.2 summarizes the considerations discussed in the following:

- A short and concise *definition of BP quality* allows discussing the underlying quality view (cf. Section 2.2), e.g., by matching it against the strategy of the organization (e.g., being an innovation leader). It reduces the risk of misinterpretations and makes

| **Business Process Quality Definition** | **Quality Attributes / Criteria** | **Evaluation vs. Effectiveness Criteria** | | |
|---|---|---|---|---|
| | | Congruence to Organizational Targets | Transparency & Retraceability | Cost Effectiveness |
| *Quality of Business Process Models* [142, 143, 144, 145, 146, 147, 148, 149, 150, 151, 152, 153, 154, 155, 156, 157, 158, 135, 160, 161, 162, 163, 164, 165, 166, 171] | | | | |
| Implicit: optimization levers for formal model quality are fully utilized | Attributes: measures for model quality (formal definition but coverage of individual aspects only) | n/a | Medium: formal measures allow for objective assessment, but non-manageable factors are not made transparent | Medium: assessment automatable, but formal modeling required first |
| *Formal Business Process Optimization* [177, 178, 143, 179, 180] | | | | |
| Implicit: formal control flow optimization levers are fully utilized | Attributes: measures for process quality with respect to control flow optimization | n/a | Low: aspects beyond control flow (i.e., ordering of activities) are not considered | Medium: assessment automatable, but formal modeling required first |
| *Process Performance Management / Business Activity Monitoring* [34, 36, 181, 183] | | | | |
| Implicit: target values for process enactment performance criteria have been achieved | Attributes / criteria: process enactment performance measures without / with target values | n/a | Low: non-manageable factors important e.g. for cycle times are mostly not considered | High: automated assessment tools available to support workflow management systems |
| *BPM Maturity Models* [184, 185, 138, 86, 186] | | | | |
| n/a, focus on BPM instead of BP quality | Attributes / criteria: various dimensions relevant to BPM maturity | n/a | High: maturity dimensions with qualitative evaluation and low complexity | High: low complexity, evaluation for the entire organization instead of particular business processes |

Table 4.3: Evaluation of BPM Approaches towards Particular Aspects of Business Process Quality

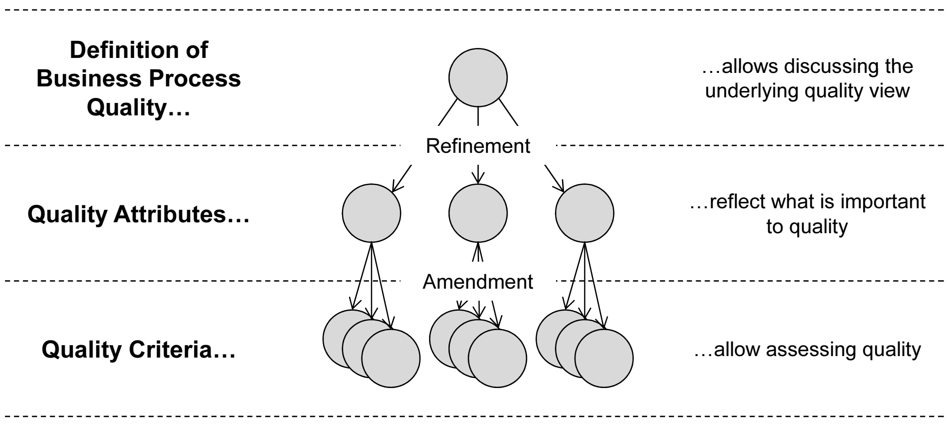



Figure 4.2: Quality Definition, Attributes, and Criteria

the underlying quality view accessible for people involved. This reflects the role of *constructs* as defined by March and Smith [77].

- *BP quality attributes* reflect what is important to quality. If a concise definition of BP quality is available, quality attributes can be deducted from this. It is possible to discuss the link of each attribute to the underlying definition of BP quality, but difficult to judge whether a set of quality attributes completely represents the definition. "Productivity" and "the capability [...] to enable users to expend appropriate amounts of resources in relation to the effectiveness achieved" constitute examples of quality attributes in [123]. Quality attributes are not sufficient to deduct statements on the level of quality achieved, since they do not state requirements associated with achieving certain quality levels.
- *BP quality criteria* are defined by amending quality attributes with explicit or implicit threshold values enabling to distinguish between poor and high quality in terms of the respective attribute. Implicit threshold values may, for instance, be provided by computing average values for a peer group of comparable organizations. Quality criteria are required to assess quality. The "productivity" example for a quality attribute evolves into a quality criterion when "appropriate amounts of resources" are further specified.

When comparing the approach employed in this thesis to the state of the art, the use of a concise BP quality definition in the sense of a *construct* is particularly relevant. This characteristic is reflected in the deductive methodology described in Section 3.2. Besides this major differentiation, which substantially impacts results in terms of *models*, *methods*, and *instantiations*, there is a number of additional observations regarding existing approaches:

- Generally, BPM approaches tend to employ quality attributes instead of quality criteria. The classic reengineering and optimization approaches constitute an exception. Approaches not defining quality criteria are not sufficient to evaluate the concrete quality of a business process. This characteristic negatively impacts practical relevance.

- Assuming proper adaptation to BPM, the balanced scorecard is the only approach ensuring high *Congruence to organizational targets* since it was explicitly developed to accommodate the potentially conflicting target dimensions encountered in real-world business strategies.
- All approaches fail to recognize the *organizational environment* of business processes by distinguishing between manageable and non-manageable factors. Non-manageable factors in the organizational environment of a business process comprise, for instance, process input delivered by other business processes. This characteristic impacts *exclusive coverage* as an aspect of *Congruence to organizational targets*. In practice, this can often be observed when benchmarking results are challenged by management if, for instance, much different organizations are chosen as peers. In this case, the perceived lack of consideration for the individual organizational environment leads to impaired acceptance of the entire assessment.

These conclusions provide guidance to further progress towards alternative artifacts:

- The current lack of concise definitions of *BP quality* encourages developing such a construct as a first *build* step (cf. Section 3.2). This construct shall be congruent to organizational targets.
- To actually achieve *Congruence to organizational targets*, a deductive approach based on organizational targets for business processes is employed. This methodology differs from existing approaches and will allow verifying congruence to targets at each development stage.
- In the *build model* step, special regard is placed on developing assessable quality criteria instead of mere quality attributes.

# Part II

# Business Process Quality Concepts

# 5 Defining Business Process Quality

As discussed in Section 2.2, quality in itself is an abstract term subject to differing interpretations. However, to be applied in a business context, it should be defined in a way to make it a useful construct for management purposes. Based on the research methodology described in Section 3.2 and the conclusions made when reviewing existing approaches in Chapter 4, this chapter derives a definition of business process quality which aims at achieving that goal. In terms of the design science paradigm underlying this thesis, this definition constitutes a *construct* (cf. Chapter 3).

Analysis of related work in Chapter 4 implied deducting a business process quality definition from *organizational targets* in order to overcome limitations of existing approaches regarding the effectiveness criteria defined for this thesis. Accordingly, this chapter's reasoning is built along three steps:

1. In terms of design science as described by Simon [103], not only artifacts in the domain of business process quality as a means of managerial analysis and control, but also business processes *per se* constitute artifacts designed to attain goals by acting within their "outer environment". It is stipulated that these goals correspond to the organizational targets referred to in Effectiveness Criterion 1 (cf. Table 3.1). Accordingly, the outer environment of business processes is discussed to focus and structure the relevant field of analysis.
2. Organizational targets for the outer environment of business processes are identified and described. This enables discussing how business processes affect the achievement of these targets during fundamental lifecycle stages.
3. Based on the outer environment of the business process, the associated organizational targets, and the respective impact of the business process in the course of its fundamental lifecycle, a ***definition framework*** for business process quality is specified.

Steps 1 and 2 are addressed in Sections 5.1 and 5.2. Step 3 is presented in Section 5.3.

## 5.1 The Outer Environment of the Business Process

When following the methodology set out above, *Congruence to organizational targets* as the most pressing concern is primarily a matter of properly structuring the outer environment of the business process to be able to consider organizational targets comprehensively, but exclusively. Figure 5.1 summarizes a variety of options in this respect.

This chapter is based on the following referred papers:

Lohrmann, M., Reichert, M.: Understanding Business Process Quality. In: Business Process Management: Theory and Applications. Volume 444 of Studies in Computational Intelligence, Springer (2013) 41–73

Lohrmann, M., Reichert, M.: Basic considerations on business process quality. Technical Report UIB-2010-04, Ulm University, Germany (2010)

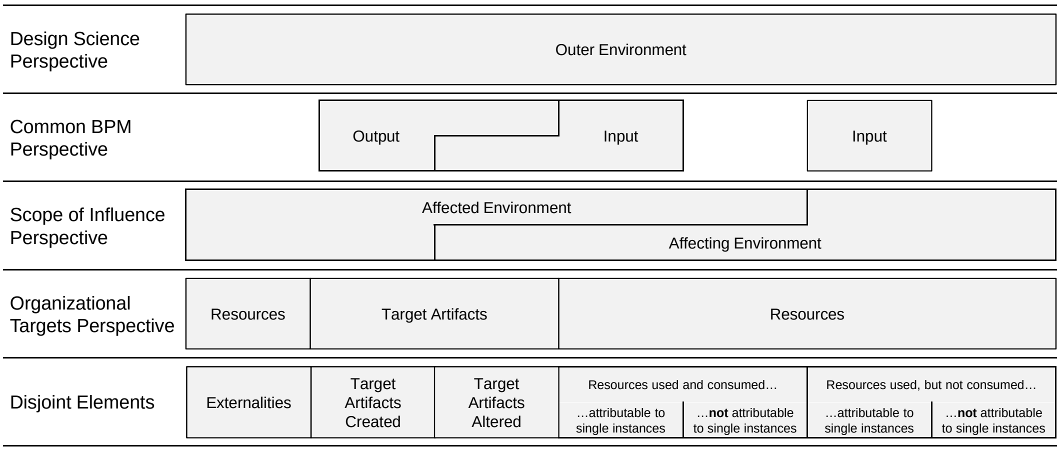



Figure 5.1: Outer Environment Perspectives

An initial *common BPM perspective* on the outer environment is based on the concepts of process *input* and process *output* used by many authors (cf., for instance, [37, 38]). For the purposes of this thesis, however, these concepts are not apt to properly structure the outer environment: *First*, input and output generally overlap if input objects are altered to assume a role as output object as well. This phenomenon is encountered in many business processes (cf. Example 11). This is, however, an issue with respect to organizational targets, because organizational targets cannot be unambiguously associated to things based on their categorization as an object of input or output.

**Example 11 (Business Process Input and Business Process Output).** Consider Sample Process B from Figure 2.6. This process relies on supplier master data, e.g. with regard to bank details. Accordingly, supplier master data constitutes business process input. If required data in this respect is missing, however, the process will also comprise completing supplier master data. Thus, supplier master data constitutes output of the process as well – input and output overlap.

*Second*, interpretations of the term process input are prone to omit resources that are not attributable to individual process instances, such as capital goods (cf. [187]), or the availability of staff to execute activities. Usually, there is also no consideration of things affected *unintentionally* like exposure to litigation risks or pollution. In this case, the outer environment and, consequently, organizational targets are not considered comprehensively. Effectiveness Criterion 1 is thus impaired.

To obtain a more comprehensive view on the outer environment of a business process, a *scope of influence perspective* can be assumed. The business process acts on a part of its environment, and a part of its environment acts on the business process. In the following, these two parts are referred to as the *affected environment* and the *affecting environment* of the business process, respectively. As an example, consider a document which is edited and thus *affected* in the course of the business process, and a piece of information which is *affecting* the business process because it is used to reach a decision, e.g. on how certain fields in the document are to be filled in. The two parts of the overall environment overlap, but things that belong to neither part are no component of the relevant outer environment. For quality assessment, the affecting and the affected environment assume distinct roles: The state of the affected environment induced by the business process can be a matter relevant to its quality, but the state of the merely affecting environment *cannot* since it is beyond the scope of influence of the business process. Considering the latter as a quality attribute would thus contradict the *exclusivity* sub-criterion as part of Effectiveness Criterion 1, *Congruence to organizational targets*.[1] It is, however, still not possible to state organizational targets for the affected environment without further analysis, because it comprises the intended results of the business process as well as the consumption of economic resources – organizational targets obviously differ for these two categories.

Therefore, an additional *organizational targets perspective* made up of two concepts is proposed: A business process interacts with its outer environment by manipulating (i.e., creating and/or altering) *target artifacts* and by using *resources*. The target artifacts involved in a business process are given by way of its business objective as defined by the WfMC (cf. Section 2.1). The resources involved are given by way of the business objective as well as business process design, implementation, and enactment. Target artifacts are the part of the outer environment the business process *strives to alter* while resources are the part the business process *needs to employ or unintentionally affect* to achieve its business objective. Anything beyond these two categories is not relevant to the business process and therefore not part of its relevant outer environment.

Note that target artifacts may evolve into resources in the context of another business process, and that resources required are not necessarily consumed. Hence, a resource is considered as consumed if it is made unavailable to other uses, either permanently or only temporarily (e.g., a plot of land used is consumed temporarily). Resources not consumed are merely part of the affecting, but not of the affected environment. Resources consumed and target artifacts are part of the affected environment. Information generally constitutes a resource which is not consumed.

**Example 12 (Target Artifacts and Resources).** To illustrate some of the concepts set out in this section, reconsider Sample Process A, the handling of supplier invoices, already used in Example 9. The *business objective* of this process is to approve or reject incoming invoices. They thus constitute the *target artifacts* of the process. *Resources* involved are determined by business process design, implementation, and enactment.

According to Example 9, available design options comprise early scanning and EDI. These clearly differ in terms of *resources* such as information systems or labor required. Hence, the resources involved in the business process are determined by the chosen design option and its implementation. However, both options pursue the same business objective, and work with the same target artifacts.

In the course of the process, invoices are not created, but merely altered – in this case, an information item whether the invoice is approved or rejected is added. This information in turn constitutes a resource for the outgoing payments process which occurs downstream in the overall process chain.

[1]Note that this proposition is not recognized in existing quality frameworks for business processes which include, for instance, process input characteristics as quality attributes ([123, 125], cf. Chapter 4).

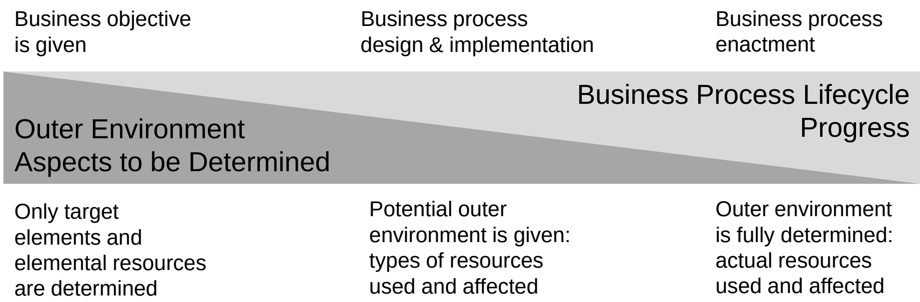



Figure 5.2: Progressive Degree of Determination of the Outer Environment

The *disjoint elements* line in Figure 5.1 depicts a categorization of the outer environment where each thing in the outer environment belongs to exactly one category. It is thus comprehensive, free of overlaps and sufficiently expressive to build all other perspectives (for instance, $\mathit{Output} = \mathit{Target\ Artifacts\ Created} \cup \mathit{Target\ Artifacts\ Altered}$).

While the basic content categories as comprised in the disjoint elements line are universally valid, their concrete content in terms of things comprised partially evolves over the lifecycle of the business process. With respect to the *organizational targets* perspective in Figure 5.1, the target artifacts part of the environment remains stable because the target artifacts of the business process are pre-determined by the business objective.[2] The resources part, however, is subject to process design & implementation. It therefore evolves with the business process lifecycle. This occurs in two ways:

- Resources used and affected condense and solidify in the course of the business process lifecycle. Before process design starts, only the general availability of resources to the organization and resources that are *elemental* (see below) to the business objective are determined. When process design & implementation are completed, the types of resources used and affected are designated. Thus, the potential outer environment of the process is given. Only when the enactment of the business process has been completed, the environment of the business process is fully determined. Figure 5.2 summarizes the progressive degree of determination with regard to the outer environment.

- The share of resources not only used, but consumed by a business process diminishes with the further advance of the business process lifecycle. That is, the further a process lifecycle has progressed, the less resources can be "saved" by aborting the lifecycle.[3] As an example, consider the implementation of PAISs: if a process has been designed and implemented, the corresponding resources are to be considered as "sunk cost", regardless of whether the process will be actually enacted. This issue generally pertains to many capital investments made.

[2] In the context of this thesis, the decision on proper business objectives is not considered as part of the business process lifecycle. Contrary to that, the reengineering advocates of the 1990s proposed to rethink the business objectives of an organization as part of process design and optimization. While this view is not shared here, a more detailed discussion on this topic is included in Section 2.1.3, and in [9].

[3] Note that this closely resembles the concept of marginal cost accounting mostly used in German enterprises [188].

In general, parts of the affected environment during business process design & implementation become parts of the solely affecting environment during business process enactment. This issue will have to be considered in the course of further investigation. In this context, resources are denoted as *elemental* if their necessity is not determined by process design, implementation or enactment, but pre-determined by the underlying objective of the process. As an example, consider Sample Process C (cf. Figure 2.7). In the course of the process, drugs are applied. These constitute elemental resources since the necessity of the drugs' availability is based on the business objective, regardless of the design, implementation and enactment of a corresponding business process.

**Example 13 (Resources in the Business Process Lifecycle).** Consider Sample Process A from Figure 2.5. The underlying objective of the business process is to assess and approve or reject incoming supplier invoices. When embarking on the design of a corresponding business process, a number of options to achieve the business objective can be considered:

- One might manually send the invoices to the purchasing department and to the department which received goods or services for approval.
- One might implement one or more of the IT-based practices from Example 9.

At this stage, it is still open whether organizational resources are employed to implement an IT-based process, or whether the organization simply sticks with more manual effort to distribute and recollect paper documents. However, if the business objective is to check invoices against purchase orders and goods receipts, purchase order information is an elemental resource and will be required regardless of process design. Likewise, if organizational resources are not sufficient to implement IT solutions, this might have to be considered as a constraint as well.

Once the business process is implemented, however, it is known what types of resources will be needed for enactment. The actual quantity per resource type will still depend on the actual number of process instances and their concrete enactment.

Regarding the diminishing share of resources that are actually consumed, consider the implementation of an EDI system. At deploy time, the system is in place regardless whether the business process is executed or not. The business process does not "consume" the EDI system as a resource. At design time, one gets a different picture: whether and how the EDI system has to be implemented depends on the design of the business process, and will surely impact the consumption of resources.

Note that the business objective determines *what* is to be achieved by the business process in terms of target artifacts, but not *how* this should be accomplished. Moreover, per definition, direct materials (including information items) are the only kind of process input to be "embodied" into target artifacts [189]. Accordingly, elemental resources determined by the business objective always relate to direct materials.

## 5.2 The Impact of the Business Process on Organizational Targets

The quality of a business process as an artifact needs to be assessed in terms of its impact on its outer environment. Based on considerations on the environment of a business process (cf. Section 5.1), the set of organizational targets which are impacted by the business process and thus relevant to business process quality can be identified. It is possible to readily determine *"what the organization would want to achieve"* with respect to both target artifacts and resources:

- With respect to *target artifacts*, achieving the business objective of the process by definition constitutes an organizational target. This aspect is typically addressed by the focus of conventional quality management approaches on the quality of products and services *delivered* by business processes. It corresponds to the notion of *efficacy* as "the ability to produce a desired or intended result" [190].
- With respect to *resources*, it can be assumed that the organization aims to act economically (as may be inferred from the term *business* process). Accordingly, resources should be impacted as little as possible. This aspect is typically addressed by the focus of process performance management approaches on capacity management, cost and time. It corresponds to the common management notion of *efficiency*.

Note that discussing organizational targets for the common BPM concept of process input would be much more difficult: both resources and target artifacts to be altered constitute process input. With regard to process input, it is thus not possible to specify whether organizations aim at changing or upholding process input.

Assessing business process quality on the basis of relevant organizational targets amounts to appraising the impact of the business process on the achievement of the respective targets. To this end, one has to consider that a business process is enacted within an outer environment which comprises not only affected, but also affecting elements, i.e. resources used and target elements to be altered. Thus, the business process cannot "achieve" organizational targets on its own, but merely contribute to their achievement in the sense of a method or tool. In other words, the affecting environment constrains the business process with respect to achieving organizational targets. This becomes clear when considering the issue of resource availability in Example 14.

**Example 14 (Resource Availability and Business Process Quality).** Consider Sample Process B from Figure 2.6. The process relies on the availability of supplier bank data, which should usually be entered into the respective information system in a preceding "upstream" business process. If the respective data are not available, they need to be amended while enacting the sample process, which will drive the consumption of labor resources as well as cycle time. Nevertheless, the ready availability of supplier master data cannot be ensured by the business process or the employees enacting it. Therefore, the impact of this factor should not influence quality assessment results – otherwise, the effectiveness of quality assessment as a means of managerial control would be diminished (cf. *Transparency and retraceability* as an effectiveness criterion discussed in Section 3.1).

To obtain a meaningful assessment of business process quality, it is necessary to delineate the impact of the affecting environment from the impact of the business process. Moreover, the affecting environment and the affected environment evolve with the business process lifecycle as discussed above. Thus, the impact of the business process on organizational targets needs to be reflected specifically for differing lifecycle stages as well.

**Example 15 (The Impact of Business Processes and the Affecting Environment).** Consider Sample Process A from Figure 2.5. When automating comparable processes, EDI systems for incoming invoices typically try to match invoices against purchase orders and goods receipts to determine whether the invoice can be posted and approved for payment. In this case, purchase order and goods receipt data constitute process input or resources employed. If one or both elements are missing, the ability of the business process to check the invoice in time will be impeded. As a result, it may not be possible to obtain an early payment discount or, worse, the supplier may decline to make new deliveries. In this case, the achievement of organizational targets is clearly impeded, but this is not the "fault" of the business process. Instead, elements of the affecting environment prevent achieving organizational targets. In other words, the EDI process alone cannot ensure timely payments because effective input of purchasing and goods receipt data is required as well. To effectively assess the quality of the EDI process, these effects have to be delineated accordingly.

As an example for differing requirements to delineate the affecting environment in the course of the business process lifecycle, consider that EDI operations are often outsourced to service providers subject to service level agreements. During design & implementation, this is a deliberate decision reflecting the quality of service required [56]. Whether this decision is taken reasonably should enter quality assessment. During enactment, however, the availability of the EDI service becomes part of the affecting environment. When assessing enactment quality, one needs to make sure that results are not biased by EDI service failures.

To fulfill Effectiveness Criterion 2, *Transparency and retraceability*, distinct organizational responsibilities for process design and process enactment as encountered in most organizations have to be recognized. Quality assessment results for business process design & implementation should therefore not depend on the quality of business process enactment, and vice versa. This implies that the business process design & implementation lifecycle stage not only determines the types of resources employed and affected in business process enactment, but also that business process design & implementation *in itself* is to be considered as part of the affecting environment during business process enactment. In a strict interpretation, this means that business process enactment will in actually not impact the achievement of organizational targets, because the behavior of the business process is fully determined by its design, its implementation, the resources used and the target artifacts to be altered.

Of course, this does not match practical requirements because assessing business process enactment quality is usually understood as assessing the quality of the human effort involved. Although human effort in principle constitutes a resource to the business process, further considerations will follow this interpretation because of its practical relevance. However, one has to be aware that this decision implies a certain deviation from a fully stringent approach based on the business process as an artifact in the sense of the design science paradigm.

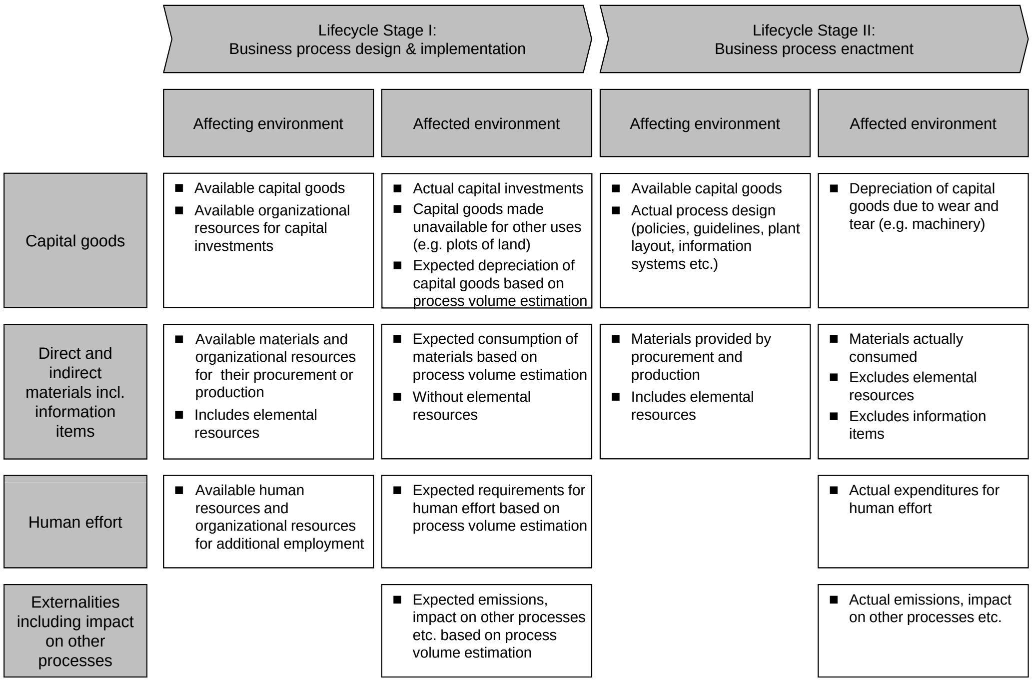



Figure 5.3: Affecting and Affected Resources in the Business Process Lifecycle

To summarize and exemplify the evolution of the outer environment in terms of resources, Figure 5.3 illustrates the affecting and the affected environment within fundamental lifecycle stages in terms of common business administration concepts.

Consider the following explanatory notes:

- As discussed in Section 5.1, target artifacts do not evolve with the business process lifecyle as they are pre-determined by the business objective. They are therefore not included in Figure 5.3.
- *Capital goods* refer to "property, plant and equipment" such as machinery, information systems, etc. [187]. In general, this corresponds to resources not attributable to individual process instances. Capital goods employed are an outcome of the business process design & implementation lifecycle stage.
- *Direct materials* correspond to resources attributable to individual process instances. For the purposes of this thesis, this includes information items (as well as special cases like dies for casting, i.e., resources used, but not consumed). Indirect materials correspond to supplies not attributable to individual process instances.
- *Human effort* refers to the quantity and quality of labor employed. Note that, as stated above, human effort is not included in the affecting environment at the enactment stage.

- *Externalities* refer to unintended impacts caused including emissions and effects on other processes, e.g., when shared resources like machinery are made unavailable. Per definition, externalities are part of the affected environment, but not of the affecting environment.
- Note that the affecting environment for business process enactment also comprises the actual process design, i.e., the results of the process design & implementation stage. This ensures that quality assessment of the enactment stage is not impacted by process design & implementation. The actual process design is included with the capital goods category of resources, because it comprises machinery and implemented information systems as well as intellectual property such as policies and guidelines. This inclusion also links both lifecycle stages in terms of their environments: the affected environment of process design & implementation also comprises the affected environment of process enactment, and the affecting environment of process enactment comprises the affecting environment of process design & implementation. The respective impact is "funneled" through the results of the design & implementation stage.

## 5.3 Business Process Quality based on Organizational Targets

In the previous sections, a number of conclusions to guide the definition of business process quality have been made:

1. Business process quality has to be assessed in terms of the impact of the business process on its outer environment. For this purpose, its outer environment can be analyzed in two dimensions: the *affecting* vs. the *affected environment*, and *target artifacts* vs. *resources*.
2. There are differing *organizational targets* with respect to the target artifacts and resources parts of the affected environment. These targets correspond to *business process efficacy* and *business process efficiency*, respectively. As the affected environment will be determined by the business process *and* the affecting environment, the business process cannot "achieve" these organizational targets, but merely contribute to their achievement.
3. Affecting and affected resources evolve with the *business process lifecycle*. To reflect differing organizational responsibilities, business process quality must be assessable separately for *business process design & implementation* and for *business process enactment*.

Based on these considerations and the ISO quality definition (cf. Section 2.2), a definition framework for business process quality can be derived:

**Definition 1 (Business Process Quality Framework). Business process efficacy** *means the effectiveness of a business process with respect to achieving its business objective. A business process is efficacious if and only if its business objective is achieved for a reasonable set of states of its affecting environment.*

**Business process efficiency** *means the effectiveness of a business process with respect to limiting its impact on resources. A business process is efficient if and only if it reasonably limits its impact on resources considering the state of its affecting environment.*

**Business process design & implementation quality** *is the degree to which an actual business process model enables business process efficacy, achieves business process efficiency during design & implementation, and enables business process efficiency during its enactment. Provided that the respective context prevents ambiguity, business process design & implementation quality is also referred to as* design quality *in the following.*

**Business process enactment quality** *is the degree to which a set of business process instances achieves business process efficacy and business process efficiency. Provided that the respective context prevents ambiguity, business process enactment quality is also referred to as* enactment performance *in the following.* □

Considering the outer environment of the business process and the associated organizational targets, business process efficacy and efficiency constitute the two dimensions of business process quality requirements for both fundamental lifecycle stages. They both take into account the affecting environment, either by demanding achievement of the business objective only for “a reasonable set of states” of the affecting environment, or by considering the affecting environment when evaluating the impact on resources. A reasonable set of states in this context relates to what can be assumed regarding the affecting environment, presuming effective upstream processes. This means that the business process, to be efficacious, must be able to function in common and expectable business circumstances. Similarly, reasonably limiting the impact on resources refers to avoiding waste and diligently managing resources. A more detailed analysis of these topics (for instance with regard to a particular area of application) is a core subject of business process quality *modeling* (cf. the methodology set out in Chapter 3).

Note that a business process can be efficacious, but not efficient, whereas efficiency is only possible if a measure of efficacy is achieved as well: if the business objective is not achieved, any resources consumed have not been used reasonably. Table 5.1 resolves the dimensions of business process quality in terms of efficacy and efficiency requirements, and in relation to fundamental business process lifecycle stages with their respective affecting environment.

The definition framework discussed above is rather plain and simple. This characteristic is required to enable straightforward discussion in a business context, for instance with respect to Garvin’s five basic quality notions (cf. Section 2.2). It corresponds to the ISO definition of quality as *“the degree to which a set of inherent characteristics fulfills requirements”* [89]: “inherent characteristics” reflect the design and implementation of the business process during the respective lifecycle stage and the human effort involved during enactment, and the “requirements” are reflected by the quality stipulations that have been made with respect to business process efficacy and efficiency.

## 5.4 Conclusion

A sound understanding of business process quality is a major prerequisite for effective BPM as is provides guidance to activities along the BP lifecycle ranging from process design to

| | | Quality Requirements | |
|---|---|---|---|
| Fundamental Lifecycle Stage | Affecting Environment Constraints | Business Process Efficacy | Business Process Efficiency |
| **Lifecycle Stage I: Business process design & implementation** | Available organizational resources | Enable achievement of the business objective with respect to the target artifacts | Limit the impact on resources during design & implementation, and enable limiting the impact on resources during enactment |
| **Lifecycle Stage II: Business process enactment** | Actual process design, target entities to be altered, capital goods, direct materials | Achieve the business objective with respect to the target artifacts | Limit the impact on resources |

Table 5.1: Business Process Quality Requirements

analysis and control. Matching existing approaches against the effectiveness criteria derived in Section 3.1 has shown that an optimum solution for management purposes in this regard has not been achieved yet. More specific, a general lack of a concise definition of business process quality or related terms like business process performance inhibits discussion and evaluation of the underlying notion of quality. Instead, BPM approaches in this area mostly confine themselves to adopting results from other areas without developing a meaningful definition upfront.

Thus, quality characteristics on a more detailed level tend to appear arbitrary, and their validity cannot be demonstrated. Moreover, many approaches provide attributes, but not criteria for quality, performance, etc. Thus, they are not sufficient to *evaluate* business process quality, which, in turn, impedes practical relevance. Finally, existing approaches mostly do not recognize differing organizational responsibilities for BPM activities and within a process chain. This also limits practical applicability.

To address these topics, this chapter has applied a rigorous methodology based on a notion of business processes as design artifacts in the sense of Simon's design science paradigm [103] as well as appropriate effectiveness criteria to develop a *business process quality definition framework*. Accordingly, the outer environment of business processes has been analyzed as a first step. Then, organizational targets with regard to components of the outer environment and the respective impact of business processes have been discussed. A business process lifecycle perspective has been applied to appropriately consider organizational structures.

These steps resulted in a concise definition of business process quality as a construct in line with the design science paradigm. However, due to the high level of abstraction adopted, it still remains difficult to concisely apply the definition to practical examples, so a more detailed model of business process quality extending Definition 1 is required. Accordingly, the definition facilitates deriving appropriate quality attributes and criteria on a more detailed level.

The following chapters will elaborate a more formal and detailed quality *model* as well as corresponding prerequisites, which will also integrate available results from related aspects of BPM research.

# 6 Business Objectives and Business Process Efficacy

In Chapter 5, *BP efficacy* has been identified as one of the two dimensions of BP quality. Business process efficacy deals with the question whether a given business process achieves its underlying *business objective*. Since business objectives are not covered in common process modeling approaches yet, today's process models are not sufficient to address the efficacy of business processes. To address this issue, this chapter presents an approach to the topic of business objectives and efficacy in the context of BPM.

The management of business objectives and efficacy constitutes an issue still unresolved but potentially valuable for various application scenarios beyond BP quality, as will be discussed in the following section. Therefore, this chapter is structured as a self-contained elaboration on the matter at hand.

## 6.1 Application Scenarios and Motivation

Business goals or objectives have been a core concept to the understanding of business processes since BPM has emerged as a discipline [17, 18]. Nevertheless, objectives still pose a notable exception to the progress towards formal BP semantics, and are only rudimentarily considered in common modeling approaches [80, 191]. The effectiveness of processes in regard to achieving business objectives can be subsumed as *BP efficacy*. Accordingly, a BP model is efficacious if it enables achieving a corresponding business objective.

As discussed in Chapter 5, efficacy constitutes one of the two major perspectives of BP quality. However, an effective approach to business objectives and efficacy can be leveraged for additional BPM purposes as well. This is best illustrated by considering exemplary application scenarios:

**Scenario 1 (Automated BP Optimization).** *PAISs collect data on process execution that could be leveraged for automated business process optimization [25]. Consider, for instance, process abortions: if a process instance cannot be completed, it should abort as early as possible to avoid unnecessary consumption of resources. Next-generation PAISs might re-arrange control flow to foster this behavior based on the execution logs of past instances. However, this must be done in a way to maintain the overall efficacy of the business process.*

---

This chapter is based on the following referred papers:

Lohrmann, M., Reichert, M.: Modeling business objectives for business process management. In: Proc. 4th S-BPM ONE – Scientific Research. Volume 104 of LNBIP, Springer (2012) 106–126

Lohrmann, M., Reichert, M.: Efficacy-aware business process modeling. In: Proc. 20th Int'l Conf. on Cooperative Information Systems (CoopIS'12). Volume 7565 of LNCS, Springer (2012) 38–55

Lohrmann, M., Reichert, M.: Formalizing concepts for efficacy-aware business process modeling. Technical Report UIB-2012-05, Ulm University, Germany (2012)

*To fulfill this requirement, formal joint interpretation of business objective and BP models must enable determining whether an (adapted) business process (still) achieves its business objective.*

**Scenario 2 (Identification of BP Variants).** *The management of BP variants has emerged as an important BPM issue [30, 29, 147, 33]. However, criteria to determine whether two process models are variants of the same reference process remain a "missing link". In this respect, modeling business processes in a way that enables tracing to common business objectives can provide an effective characteristic to assess the "equivalence" of process variants [192].*

**Scenario 3 (Benchmarking).** *Qualitative benchmarking deals with good practices to identify opportunities for process improvement [45]. This often meets the resistance of practitioners as the equivalence of process alternatives with respect to their outcome is doubted. Formalizing efficacy can help to alleviate this issue. Similar considerations apply to more recent approaches like process performance management [36].*

Generally, efficacy must be considered in relation to the environmental conditions the business process encounters. On the one hand, its *outer environment* (cf. Chapter 5) will determine required activities. On the other hand, it must provide sound conditions for process execution. As examples, consider the requirement to conduct medical treatments based on the results of diagnostic procedures, or the availability of raw materials. It is therefore not sufficient to simply consider the set of activities comprised in a BP model. Rather, efficacy assessment requires careful modeling of the intended and actual interaction between business process and environment.

Present process modeling languages are mostly oriented at task sequences as required to manage workflows, but not at modeling preconditions and results. In general, there is no distinction between what a process actually seeks to achieve and what is merely accepted as part of the effort, like resources consumption. This leads to limitations regarding business objectives and efficacy assessment.

The presented approach towards the issue at hand builds on a clear distinction between the modeling of business objectives, in the sense of a formal requirements definition, and the modeling of business processes. Recognizing that business objectives exist independently from concrete BP implementations is a key element here. This approach reflects the role of requirements engineering in software development [193, 194], but differs from present BPM approaches where objectives are either considered not at all or annotated to individual elements of BP models (cf. Section 6.3).

The remainder of this chapter is structured as follows: Section 6.2 develops a self-contained methodology, basic terminology, and effectiveness criteria to evaluate results. Sections 6.3-6.5 implement the research methodology, ranging from a review of available approaches to a refined business objectives meta-model and the application of the latter to a sample case. Section 6.6 concludes the chapter and builds a link to further chapters of this thesis.

## 6.2 Preliminary Considerations and Methodology

In general, business objectives exist independently of business processes. A particular process constitutes just one of many alternatives to achieve a particular business objective. For

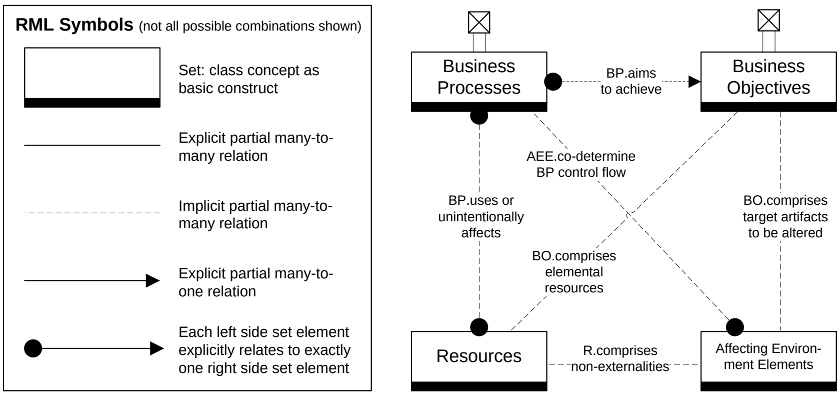



Figure 6.1: High-level Quality-relevant Business Process Concepts

example, an alternative business process to achieve the same business objective can be implemented by using another IT system or re-arranging the order of activities. In other words, a business objective is achieved by inducing a state of the outer environment that satisfies certain criteria – no matter *how* this is done. Accordingly, Figure 6.1 presents interrelations between business objectives, business processes, and outer environment. The Referent Model Language (RML) notation defined in [195] is used since it provides a concise graphical notation for set theory constructs. In particular, the following topics are considered:

- *Relations between basic constructs:* Each business process aims at achieving exactly one business objective. However, a business objective may be addressed by an arbitrary number of business processes. Business objectives refer to at least one target artifact, and target artifacts *altered* by the business process are elements of the affecting environment (target artifacts *created* are not). Each business process uses resources, and resources are always used or unintentionally affected by at least one business process (otherwise, an element of the environment is not a resource). Note that the business objective might require that particular resources are (physically or logically) incorporated into the target artifacts. For example, this is generally the case for manufacturing processes. Resources are generally elements of the affecting environment with the sole exception of externalities.[1] All relations have been modeled as implicit. They will be amended by explicit relations on a more detailed level as a more detailed meta-model is defined in the following sections.

- *Aggregation of business processes:* Formally, there is no limitation to the composition and decomposition of business processes [22]. For example, Sample Processes A and B might easily be merged, or the admission of individual open items in the middle part of Sample Process B might be modeled as a separate business process. In principle, the entire value chain of an enterprise might be represented as single integrated business process, or it might be decomposed into elemental business processes where each process solely consists of a single task. In practice, structuring the value chain of an organization into distinct business processes is based on organizational responsibilities and the wish to make process models as simple and understandable as possible by

[1] Externalities represent things that are not required to execute the process, but affected by incident only.

limiting the number of artifacts involved and by avoiding switches between different levels of cardinality.

- *Aggregation of business objectives:* Per definition, each business process is linked to a business objective. Accordingly, business objectives can be composed and decomposed along with the respective business processes.[2]

Example 16 illustrates the concept of business processes aggregation and, in turn, business objectives aggregation.

**Example 16 (Aggregation of Business Processes).** Consider Sample Process B in Figure 2.6. The box entitled "Process open items" constitutes a multiple instance workflow pattern, i.e. it is executed multiple times for each instance of the overarching payment run process. Its content could also have been modeled as a lower-level sub-process instead. Similarly, it would have been possible to model Sample Process A from Figure 2.5 and Sample Process B from Figure 2.6 as one process (e.g., as "accounts payable management"). In that case, Sample Processes A and B would have to be enclosed in a multiple instance workflow pattern each to properly consider the differing "cardinality" of process instances (i.e., there might be any number of invoice checking instances for any number of payment run instances). Accordingly, it may be assumed that while process aggregation and disaggregation is, in principle, arbitrary, a process change is in many cases divided into individual processes to avoid changes in cardinality.

Generally, assessing and controlling BP efficacy in the sense of business processes appropriately implementing their business objectives require two modeling facilities:

1. A *business objective meta-model* that is sufficiently expressive to model business objectives independently of corresponding business processes in the sense of a formal requirements definition.
2. An *efficacy-aware BP meta-model* that is sufficiently expressive to relate business objective models to common BP modeling constructs in a way that allows assessing the efficacy of BP models.

Out of these two modeling facilities, this chapter addresses the first one since a business objective meta-model is required as a self-contained artifact to enable the use cases set out above. Efficacy-aware modeling, however, exhibits requirements that are mostly shared with efficiency-aware modeling. Therefore, this thesis summarizes both topics in the form of *quality-aware BP modeling* as described in Chapter 7. Besides building relevant artifacts, the design science paradigm demands for the evaluation of results based on "criteria of value or utility" [77]. Since this chapter intends to reflect the broader appeal of business objectives modeling as a self-contained approach, it requires proprietary effectiveness criteria beyond the ones defined in Section 3.1, as well as self-contained evaluation of results. Accordingly, the sub-methodology employed in this chapter is structured along the following steps:

1. Define *effectiveness criteria* to assess the utility of design artifacts in the field of business objectives modeling (cf. Section 6.2.2).

[2]Depending on the application scenario, modeling business processes can result in large and complex models if a process is not decomposed into smaller individual lower-level processes. Appendix A provides another practical example how large and complex processes can be addressed in business objective modeling.

2. Assess the *state of the art* based on the defined effectiveness criteria to determine gaps and obtain pointers towards a refined solution (cf. Section 6.3).
3. *Build* required terminology for business objectives based on effectiveness criteria and research into available approaches (cf. Section 6.4).
4. *Build* a meta-model for business objectives (cf. Section 6.5).
5. *Evaluate* the solution with respect to effectiveness criteria (cf. Section 6.5.2).
6. Discuss implications and further steps to leverage results (cf. Section 6.6).

The remainder of this section discusses required preliminary terms, and develops effectiveness criteria to evaluate present approaches as well as the results presented here.

### 6.2.1 Basic Terminology

Business processes constitute artifacts in the sense of design science [103]. Hence, they operate within an *affecting and affected outer environment*. The outer environment of a business process consists of *target artifacts* and *resources*, i.e., things the process strives to create and alter, and things required to properly do so. Note that this perspective differs in some regards from the classic BPM concepts of process input and process output as it includes things usually not considered, e.g., capital goods. See [4] for a more detailed discussion of this topic.

In the BPM field, business objectives represent the targets an organization aims to achieve with a business process. As illustrated in Example 17 and Figure 6.2, this can be understood on a strategic, collective operational or transactional level.

**Example 17 ("Business Objective" Interpretation Alternatives).** As another example of a business process, consider the handling of job applications in an enterprise. On a *strategic level*, the business objective of this process may be understood as providing the organization with the right "human resources". On a *collective operational level*, the business objective may be understood as properly handling the overall occurring cases of job applications. Depending on the required service level, the business objective may, for instance, be fulfilled if 90% of cases are managed correctly. On a *transactional level*, it may be understood as properly handling an individual application.

For the purpose of business objectives modeling, the term business objective is defined on the *transactional level* to achieve consistency with common BP modeling approaches: In BP modeling, models are generally defined on a process instance [14] level without considering the cardinality of cases or instances, i.e., a task that occurs many times for the business process, but one time per process instance is modeled as an individual activity, not as a set of activities.

Remember that an affecting environment may determine what actually needs to be induced to fulfill a business objective, e.g., when considering decision processes (cf. Example 18).

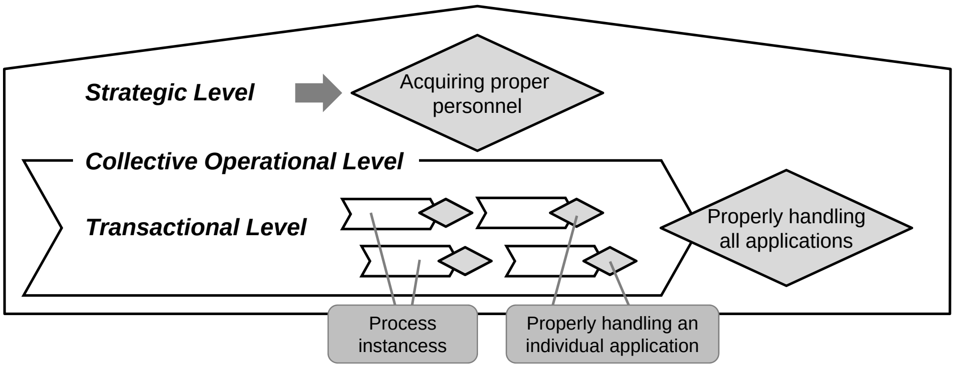



Figure 6.2: Semantic Business Objective Levels

**Example 18 (The Affecting Environment of Business Objectives).** Re-consider the job application process from Example 17. The business objective cannot be achieved by simply approving or disapproving an application. Rather, the respective hiring criteria must be considered. Thus, they constitute the affecting environment of the business objective. As another example consider medical treatments. In many cases, tests are required to find out which drugs are required. In this case, the test results are part of the affecting environment of the business objective.

When considering the affecting environment of process instances, achieving the business objective on the transactional level for all process instances (or, in other words, achieving the business instance on the collective operational level) does not mean that the business objective on the strategic level has been achieved as well (cf. Example 19).

**Example 19 (Business Objective Levels and the Affecting Environment).** When handling incoming job applications, the strategic level business objective will be to fill the respective positions. However, the transactional level business objective for an individual process instance may be fulfilled if an applicant is declined because her qualifications (as part of the affecting environment) are not sufficient.

In summary, this leads to the following basic definition for business objectives to be further elaborated in the modeling approach:

**Definition 2 (Business Objective).** *A **business objective** in the sense of BPM constitutes a refinement of organizational targets to the transactional level. It pertains to an affecting and affected environment. The affecting and affected environment represent the things to be considered and the ones to be manipulated to achieve the business objective. The business objective relates each state of its relevant affecting environment to a set of aspired states of the affected environment.* □

### 6.2.2 Effectiveness Criteria

Considering the scenarios lined out in Section 6.1, business objective models constitute requirements definitions for business processes. They will generally be used to

- determine what needs to be done to achieve a business objective (e.g., as a starting point for BP design, or as in Scenario 1 from Section 6.1),
- assess whether a modeled business process enables achieving its business objective (e.g., to evaluate design options, or as in Scenario 2), and
- assess whether a concrete BP instance has actually achieved its business objective (e.g., in PAIS implementation testing, or in the way described in Scenario 3).

Accordingly, the notion of an *achieved* function reflecting whether an aspired state of the affecting and affected environment of a business objective is reached is central to business objectives modeling.

Recapitulating the terms introduced in Section 6.2.1, business objectives are *achieved* by propagating target artifacts to an *aspired state*. However, which target artifacts need to be created or altered, and which states are considered as aspired may depend on other elements of the affecting environment.[3] Thus, business objectives cannot be recorded solely in terms of attributes of targets artifacts, but in terms of a set of consistency rules to be satisfied in respect to the relevant environment. This set of rules must be complete and free of overlaps to ensure that conformance can be assessed for each state of the outer environment.

Table 6.1 summarizes *effectiveness criteria* towards business objectives modeling. *Semantic Requirements* SR1 to SR3 are based on the issues discussed above. They reflect the semantic content an approach needs to address to properly model business objectives. In addition, an effective modeling approach will fulfill *Usability Criteria* UC1 to UC3 to support both modelers and users. The usability criteria are based on the considerations about model quality [196]. In this context, a meta-model level is addressed instead of the model level in [196]. Hence, special regard is assigned to the quality types of “physical quality”, “semantic quality”, and “pragmatic quality”.

## 6.3 Related Work

Models for business objectives or goals[4] have been proposed by Kueng and Kawalek [197], Neiger and Churilov [198], Soffer and Wand [199], and Lin and Sølvberg [200]. Markovic and Kowalkiewicz [201] presented a business goal ontology as part of the SUPER project on semantic BPM (cf., e.g., [23]). An approach to integrate goals into the BP lifecycle has been developed by Cardoso [202]. Ponnalagu et al. developed an approach to manage service-oriented process variants oriented at goals [192]. Table 6.2 matches these approaches against Semantic Requirements SR1 to SR3 (cf. Table 6.1). For reasons of brevity, Usability Criteria UC1 to UC3 are not considered here.

[3] Note that the affecting environment of a business objective may differ from the affecting environment of an associated business process – the affecting environment of an efficacious business process will encompass, but possibly not be limited to, the affecting environment of its business objective (cf. [4]).

[4] In the field of BPM, the terms are generally used as synonyms.

| | |
|---|---|
| *SR1* | *Consideration of the affecting environment:* Whether a business objective is achieved or not must be determined in terms of target artifacts *and* additional properties of the outer environment. For example, in Sample Process A (cf. Example 6), the approved or disapproved invoice as a target artifact and the defined conditions for invoice approval as additional properties of the outer environment must be jointly considered to determine whether the process achieves its business objective. |
| *SR2* | *Varying target environment:* The set of target artifacts to be created or altered as well as the concrete operations to be carried out on them may vary; e.g., in Sample Process A (cf. Example 6), the purchase order may have to be adapted or not, but the invoice must always be approved or disapproved. |
| *SR3* | *Order constraints:* There may be constraints regarding the order in which the activities of a process need to be executed in conformance with the business objective. For example, consider Sample Process C from Example 8: drug application and examinations must occur in a specific order. It is important to note that these constraints actually represent constraints with respect to target artifacts manipulation, since by definition executing activities cannot constitute a business objective. |
| *UC1* | *Semantic interdependencies:* The approach should be apt to transparently capture semantic interdependencies between elements of the outer environment, e.g., mutual exclusivity or correlation. As example of mutual exclusivity, consider the approval or disapproval of invoices in Sample Process A from Example 6 (cf. "pragmatic quality" in the sense of comprehension in [196]). |
| *UC2* | *Model compaction:* The approach should lead to a compact result in the sense of avoiding unnecessary content which might "hide" the relevance of model elements. For example, in Sample Process A, it would be obstructive to model the effect of senior management approval for invoices below a value of 5,000 (cf. "semantic quality" in the sense of validity or relevance to the problem in [196]). |
| *UC3* | *Knowledge externalization:* The approach should leverage implicitly available knowledge of the modeler (cf. "physical quality" in the sense of externalization in [196]). |

Table 6.1: Effectiveness Criteria for Business Objective Modeling Approaches

In the field of *BP compliance*, compliance requirements are generally understood not only as regulatory topics imposed by actors external to the organization (such as national legislation), but also as internal rules and regulations. Accordingly, whether a business process addresses its business objective might be understood as a compliance requirement as well. Exemplary BP compliance approaches have been developed by Sadiq, Governatori et al. [152, 153] and Ly et al. [154, 155]. Note that Governatori and Sadiq explicitly discern between "business objectives" and "normative objectives", with the latter being in the focus of BP compliance [153]. Knuplesch et al. extend BP compliance management to cover the requirements of cross-organizational business processes [156], and provide an enhanced approach for visual modeling of compliance rules [157]. Note that BP compliance is closely related to declarative BP modeling approaches (cf. Section 7.2) since both fields are based on the idea of restricting the range of valid BP instances through applicable rules. In this context, Goedertier and Vanthienen developed a language to express compliance constraints relevant to sequence and timing of activities [203]. This approach, in turn, has contributed

| | *Evaluation against Semantic Requirements* (cf. Table 6.1) | | |
|---|---|---|---|
| *Source / Focus* | *SR1* | *SR2* | *SR3* |
| *Kueng and Kawalek* [197]: Goals-based modeling, design evaluation | *Not fulfilled:* No formal measurable definition of goals. | *Not fulfilled:* Goals are discussed on an abstract level only. | *Not fulfilled:* Goals are discussed on an abstract level only. |
| *Neiger and Churilov* [198]: "Value-focused thinking" to structure objectives | *Not fulfilled:* No formal measurable definition of objectives. | *Not fulfilled:* "Functional objectives" on a more abstract level. | *Not fulfilled:* "Functional objectives" on a more abstract level. |
| *Soffer and Wand* [199]: Formalizing processes' contribution to "soft goals" | *Not fulfilled:* Business goals as any possible process termination state, goal achievement only pertains to target artifacts. | *Partially fulfilled:* Implicitly considered: only one relevant process path required per target artifact. | *Partially fulfilled:* Order constraints implicitly considered via consistent process paths. |
| *Lin and Sølvberg* [200]: Goal ontology for semantic annotation in distributed environments | *Partially fulfilled:* Goals are seen as states of activities or artifacts, but no specification of respective artifact states. | *Not fulfilled:* Goals are defined for activities instead of processes, no concept of goals changing with the environment. | *Partially fulfilled:* Constraints are comprised in the metamodel, but not further specified as state of activities or the environment. |
| *Cardoso* [202]: Integrating goals into a BP lifecycle | *Partially fulfilled:* Use of "key performance indicators" (KPIs) that might represent artifact states. | *Partially fulfilled:* KPIs defined in sufficient detail might represent environment-related artifact states. | *Not fulfilled:* No notion of order constraints. |
| *Ponnalagu et al.* [192]: Using goals to manage service-oriented process variants | *Not fulfilled:* Definition of goal achievement based on annotated services "capability library" only. | *Not fulfilled:* No representation of interrelations between the environment and goals. | *Not fulfilled:* No notion of order constraints. |
| *Markovic and Kowalkiewicz* [201]: Integrating goals into BP modeling | *Not fulfilled:* No concise definition of when a goal has been achieved. | *Partially fulfilled:* the notion of dependencies between organizational actors might be abstracted to cover environmental relations. | *Not fulfilled:* No notion of order constraints. |

Table 6.2: Business Objective Modeling Approaches

| | *Evaluation against Semantic Requirements* (cf. Table 6.1) | | |
|---|---|---|---|
| *Source / Focus* | *SR1* | *SR2* | *SR3* |
| *Sadiq et al.* [152, 153]: Business process compliance | *Partially fulfilled:* Compliance controls may refer to the state of environmental elements. | *Partially fulfilled:* Whether compliance controls are required may be modeled based on the state of environmental elements . | *Partially fulfilled:* Order constraints can be modeled as "control rules". |
| *Ly et al.* [154, 155]: Business process compliance | *Not fulfilled:* Compliance requirements in the sense of task enactment constraints, no notion of compliance in the sense of state. | *Partially fulfilled:* Not considered in the sense of states of the external environment, but complex interaction between constraints. | *Fulfilled:* Order constraints are explicitly integrated. |
| *Knuplesch et al.* [156]: Cross-organizational BP compliance, BP modeling | *Not fulfilled:* Compliance requirements in the sense of task enactment constraints considering cross-organizational processes, but no consideration of environmental elements such as data or resources. | *Not fulfilled:* States of the external environment which impact the relevance of constraints are not considered. | *Fulfilled:* Order constraints, even beyond organizational borders, are explicitly integrated. |
| *Goedertier and Vanthienen* [203]: Language to express sequence and timing constraints | *Not fulfilled:* Focus on activities and actors, but no consideration of environmental elements such as data or resources. | *Not fulfilled:* States of the external environment which impact the relevance of constraints are not considered. | *Fulfilled:* Order constraints are explicitly integrated. |

Table 6.3: Business Process Compliance Approaches

to the development of the EM-BrA²CE vocabulary in the field of declarative BP modeling [204]. Table 6.3 reflects BP compliance approaches along SR1 to SR3.

The field of *requirements engineering* provides approaches related to business objectives modeling as well: Yu developed the *i** approach to document actors' goals and dependencies in "early-phase requirements engineering" [205], and Dardenne et al. proposed *KAOS* to deduct technical requirements from goals [206]. Table 6.4 matches requirements engineering approaches against SR1 to SR3. For comparison, an approach by Engelsman et al. towards goals modeling in *enterprise architectures* is considered as well.

| | *Evaluation against Semantic Requirements* (cf. Table 6.1) | | |
|---|---|---|---|
| *Source / Focus* | *SR1* | *SR2* | *SR3* |
| *Yu* [205]: "Early-phase requirements engineering" | *Not fulfilled:* Goals are not formalized, therefore, no concise definition of when a goal has been achieved. | *Not fulfilled:* No notion of goals evolving with the environment. | *Not fulfilled:* No notion of order constraints. |
| *Dardenne et al.* [206]: "Goal-directed requirements acquisition" *(KAOS)* | *Not fulfilled:* No concise definition of when a goal has been achieved. | *Not fulfilled:* No notion of goals evolving with the environment. | *Not fulfilled:* No notion of order constraints. |
| *Engelsman et al.* [207]: Enterprise architecture goals modeling language | *Not fulfilled:* Hard goals concept, but no formal notion of goal achievement. | *Not fulfilled:* No affecting environment concept. | *Partially fulfilled:* Goal aggregation might be extended to include ordering. |

Table 6.4: Requirements Engineering

The desired output or *product* of a business process constitutes a construct closely related to the business objective of a process. In this context, Reijers et al. developed the *product-based workflow design (PBWD)* approach to analytically derive process models from given product or data specifications [69] based on earlier work by van der Aalst [208]. The approach has also been successfully applied in practice [209, 69]. PBWD is based on a "product/data model" covering the information involved in administrative processes, both in terms of process input and process output. The product/data model comprises one "top element" modeling a data element as the desired output of the process in a way similar to a "bill of materials" in manufacturing. Hence, the product/data model can be considered as covering the business objective of the process. "Production rules" are used to model tasks which determine the value of data elements based on other data elements. Modeling the concrete derivation of data element values is not required for the product/data model since only "production rule" relations in the sense of predecessor / successor links are necessary for the purpose of PBWD.

Similarly, *PHILharmonic Flows* as proposed by Künzle and Reichert enables object-aware process management through a tight integration of business processes on the one hand and business objects on the other [72, 73]. Thereby, the approach distinguishes between object behavior and object interactions relevant in the context of a business process. Hence, the separation between BP objectives and the business processes themselves is substituted by a separation between objects and business processes. Furthermore, the processing of objects must be data-driven, i.e., a business object only achieves a particular state if corresponding attribute values are provided. Table 6.5 summarizes approaches dealing with output / data-oriented BP design with regard to SR1 to SR3.

| | *Evaluation against Semantic Requirements* (cf. Table 6.1) | | |
|---|---|---|---|
| *Source / Focus* | *SR1* | *SR2* | *SR3* |
| *Reijers et al.* [69]: Product-based workflow design | *Partially fulfilled:* Goal achievement defined as determining a top data element value. Production rules allow linking the affecting environment to the top element, but values leading to particular top element values do not need to be specified. | *Partially fulfilled:* Only one "top element" to represent a target aspect. Multiple or varying aspects can be modeled through workarounds, however. | *Fulfilled:* Order constraints can be modeled through corresponding production rules. |
| *Künzle and Reichert* [72, 73]: PHILharmonic Flows | *Partially fulfilled:* Required states of objects are modeled through object attributes. However, there is no distinction between target objects and other properties of the affecting environment, and no explicit notion of objectives. | *Partially fulfilled:* Varying target objects are not explicitly considered, but may be modeled through corresponding (i.e., unmodified) object states. | *Fulfilled:* Order constraints are considered with regard to changes of object states. |

Table 6.5: Output- / Data-oriented Business Process Design Approaches

Generally, the discussed business objective modeling approaches aim at amending BP models with a descriptive goals perspective, and not necessarily at using business objectives as a formal requirements definition in a BPM context. Moreover, most current approaches from the field of BPM do not separate business objectives from business processes, i.e., a business objective is (implicitly) defined as the enactment of a business process or a certain set of tasks. Hence, these approaches do not enable comparison of multiple BP implementation alternatives towards one common business objective in the sense of BP variants [29, 33]. Output- and data-oriented BP design approaches have accomplished some progress in this respect (cf. Table 6.5). Still, it becomes clear that additional work is needed to develop a *business objectives meta-model* to fully address the criteria set out in Table 6.1.

## 6.4 Extended Business Objective Modeling Terminology

According to Semantic Requirement SR1 (cf. Table 6.1), an effective approach to business objectives modeling must relate aspired states of target artifact properties to states of additional properties of the outer environment. Only if the latter are fulfilled, the related states of target artifact properties become relevant. In the following, the respective environmental properties will be referred to as *elements of the target environment* (or, in short, *target elements*) and *elements of the conditional environment* (or, in short, *conditional el-*

*ements*). Both sets of *environmental elements* may overlap, i.e., an environmental element may constitute a target element, a conditional element, or both. Environmental elements may be conceived of as “metering points” that, taken together, suffice to determine whether a business objective has been *achieved*. Note that the conditional elements correspond to the additional properties of the outer environment cited in Semantic Requirement SR1 in Table 6.1. The relevant “metering points” may be expressed as *binary state determinants*. Let $E$ be the set of environmental elements of a BP model, and let $e \in E$ be an individual environmental element with its domain $\mathcal{D}(e)$ as the range of possible values it may assume. Binary state determinants are then defined as follows:

**Definition 3 (Binary State Determinant).** *A binary state determinant (BSD)* $\gamma = \langle E_\gamma, \Lambda_\gamma \rangle$ *is defined by a sequence of environmental elements* $E_\gamma = (e_{\gamma_1}, \ldots, e_{\gamma_n})$, $e_{\gamma_i} \in E$, *which may not be empty, and a sub-set of the cartesian product of their domains* $\Lambda_\gamma \subseteq \mathcal{D}(e_{\gamma_1}) \times \ldots \times \mathcal{D}(e_{\gamma_n})$. $\Lambda_\gamma$ *describes the set of value tuples or states of affecting elements for which the BSD is* fulfilled.

*A BSD is a conditional BSD or a target BSD if it is used used to determine the state of the conditional or of the target environment, respectively.* □

Note that ordering the affecting elements of BSDs is necessary to maintain the significance of the value tuples comprised in $\Lambda_\gamma$: each individual value obtains its semantic meaning out of the association with the environmental element at the respective position in the sequence of affecting elements.

As illustrated in Example 20, most BSDs can be described as an equation or inequation of a term consisting of arithmetically combined environmental elements and a static value. In most cases, it can even be assumed that a BSD will refer to just a single environmental element. However, there are cases requiring a more complex transcription.[5]

**Example 20 (Binary State Determinants).** In a manufacturing process, parts are often procured from multiple suppliers. Thus, it may be necessary to check minimum inventory levels with a conditional BSD as follows: “Inventory A + Inventory B + Inventory C $\geq$ 10”. If the conditional BSD is fulfilled, production may commence to achieve a target BSD as follows: “Part D available = true”. As an example of a more complex BSD, consider tolerances for the size of the manufactured part: “15 $\leq$ Diameter D $\leq$ 17”. Tolerances, of course might be modeled as multiple BSDs as well. As discussed, this would, however, ”hide“ the interrelation between the BSDs.

On that basis, it would be possible to list all conceivable conditional BSDs (many conditional BSDs relate just one conditional element to a fixed value range), enumerate the possible states, and relate them to the corresponding set of aspired states of the target elements, which are, in turn, represented by target BSDs. This approach would link aspired target states to the affecting environment as demanded by Semantic Requirement SR1 (cf. Table 6.1). However, there would still be major issues regarding the effectiveness criteria, as presented in Table 6.6.

[5]A simpler notion of BSDs might refer to comparing an arithmetic combination of environmental elements to a value range, and splitting up more complex BSDs into multiple BSDs. However the above “complex” definition of BSDs has been chosen to allow semantically rich BSDs in modeling since there are cases where the complex BSD is more common to the field of application, and to maintain the semantic interrelation between environmental elements comprised in one BSD.

| | |
|---|---|
| *SR2* | As all target BSDs are enumerated for each conditional state, the potentially limited relevance of individual target artifacts is "hidden". |
| *SR3* | Order constraints are not addressed, and still require an additional construct. |
| *UC1* | Interrelations between elements of the outer environment, such as mutual exclusivity or correlation, are not captured. |
| *UC2* | For an individual target BSD, only a (typically small) part of the conditional environment is relevant. Hence, a relation matrix between conditional and target BSDs would only be sparsely populated. For instance, in Sample Process C (cf. Figure 2.7), the age of the patient is not relevant to examination B. This characteristic is not utilized which leads to a unnecessarily bloated model. |
| *UC3* | From a modeler's perspective, it is much easier to determine (e.g. by discussion with stakeholders) what the prerequisite conditions for a target BSD are than which target BSDs are determined by a conditional element, let alone *a priori* enumerating relevant conditional BSDs. Moreover, interrelations or mutual relevance of BSDs (cf. UC1-2) are not addressed. Capturing available knowledge is thus impeded. |

Table 6.6: Basic Modeling vs. Effectiveness Criteria

To address these topics, a business objectives modeling approach is introduced. It (i) reflects distinct types of target BSDs, (ii) commences the modeling procedure with target BSDs instead of conditional BSDs, and (iii) avoids redundancies in its modeling of both the target and the conditional environment. Terms used to this end are summarized in the remainder of this section.

**Definition 4 (Target BSD Types).** ***Target BSDs*** *are constituents of the business objective. To achieve a business objective, all respective target BSDs must assume* ***target values****. Depending on the range of target values, various target BSD types are discerned.*

*To achieve the business objective,* ***monovalent target BSDs*** *must assume a "true" value (target BSDs that may only assume a "false" value are to be rephrased accordingly). There is no condition attached. Note that target BSDs subject to order constraints must include "false" in their value range.*

*To achieve the business objective,* ***fully determinate bivalent target BSDs*** *may assume either a "true" or a "false" value. Thus, only one condition attached to either "true" or "false" is required.*

*To achieve the business objective,* ***partially determinate bivalent target BSDs*** *may assume either a "true" or a "don't care" value ("false" target BSDs are to be rephrased).*[6] *"True" is bound to a respective condition.*

*To achieve the business objective,* ***trivalent target BSDs*** *may assume a "true", "false", or "don't care" value. Trivalent target BSDs differ from bivalent ones as there are two condi-*

[6] "Don't care" implies that the business process needs to do nothing – consider, for instance, the target BSD "Purchase order value = invoice value" from Sample Process A in Figure 2.5, where the purchase order either needs to be adapted or simply left as it is. Semantically, this represents the characteristic that the set of relevant target artifacts may change with the conditional environment.

| ***Target BSD Types*** | ***Condition States*** | ***Aspired target BSD states*** Not Fulfilled | Fulfilled | ***Example*** |
|---|---|---|---|---|
| *Monovalent* | n/a | | X | Examination A in Sample Process C |
| *Fully determinate bivalent* | Not fulfilled | X | | Invoice approval in Sample Process A |
| | Fulfilled | | X | |
| *Partially determinate bivalent* | Not fulfilled | X | X | Senior management approval in Sample Process A |
| | Fulfilled | | X | |
| *Trivalent* | Only 1st condition fulfilled | | X | Marking of condition X in Sample Process C |
| | Only 2nd condition fulfilled | X | | |
| | No condition fulfilled | X | X | |
| | Both conditions fulfilled | May not occur | | |

Table 6.7: Target BSD Types

*tions attached to "true" and "false", respectively. The conditions are mutually exclusive, but not comprehensive (i.e. one or none of the two can evaluate to "true" at the same time).* ☐

Table 6.7 and Figure 6.3 provide an overview on the target BSD types and the state they must assume to enable achieving the business objective depending on the state of their relevant conditional environment.

Note that trivalent target BSDs can also be understood as two partially determinate bivalent target BSDs referring to the same target element. However, modeling a trivalent target BSD as two bivalent target BSDs results in a loss of model content because the two respective bivalent target BSDs' mutual exclusivity is not visible in the model.

**Definition 5 (Conditional Propositions).** *Conditions attached to target BSDs can be expressed as* ***conditional propositions*** *consisting of conjunctively and / or disjunctively interlinked* ***conditional BSDs****. Unlike target BSDs, the value range of conditional BSDs is confirmed to "true" and "false". A target element may also act as a conditional element within one business objective.*

***Absolute conditional BSDs*** *compare one conditional element to an absolute value range.* ***Relative conditional BSDs*** *compare two conditional elements to each other.*

*Target BSDs are considered as* ***conditionally equivalent*** *if the attached conditional propositions are equivalent or if, for fully determinate bivalent target BSDs, the attached conditional propositions are a negation of each other. Target BSDs are considered as* ***conditionally dependent*** *on each other if a BSD's conditional proposition comprises the value another target BSD has assumed or should assume by way of a relative conditional BSD.* ☐

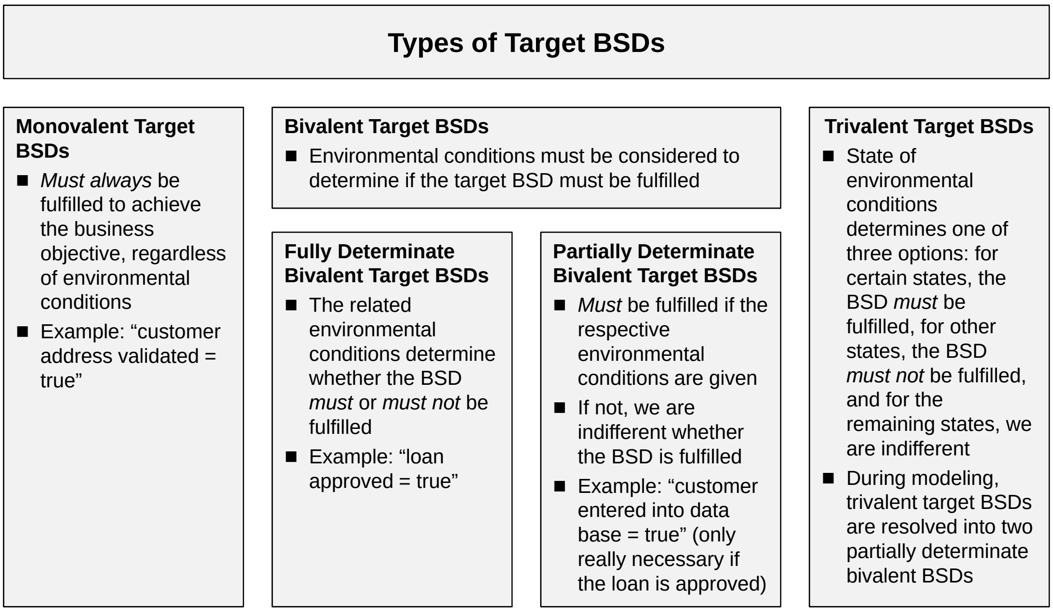



Figure 6.3: Target BSD Types

The treatment of order constraints has been identified as a requirement of business objective modeling (see Semantic Requirement SR3 in Table 6.1). To address this issue, a number of characteristics of conditional propositions (cf. Definition 5) must be considered:

- As shown in Example 21, a conditionally dependent target BSD shares the conditional proposition of the “parent” BSD.
- A conditional dependency exists for any two target BSDs where an *order constraint* applies; i.e., the dependent target BSD must be fulfilled before, after, or at the same time as the “parent” BSD.
- From a modeling perspective, it does not make a difference which BSD is the “dependent” one, because both are required to achieve the business objective.

Based on these characteristics, Table 6.8 describes a convention is introduced to model conditional dependencies and order constraints.

Note that conditionally dependent target BSDs that shall be fulfilled *at the same time* should be merged with their “parent” BSD. That is, the two underlying target elements should be treated as one as they must be manipulated concurrently anyway.[7]

**Example 21 (Order Constraints Modeling).** Consider Sample Process C in Figure 2.7. Examination B (Task C3) must be *prepared* by applying a drug (Task C2) while another drug (Task C9) is required *after* examination D (Task C8). The applications of both drugs thus become elements of the target environment which are conditionally dependent on the respective examination.

[7]Note that this issue is also not addressed in common process modeling approaches.

| ***Order Constraint:*** *the conditionally dependent target BSD must be fulfilled...* | ***Modeled Conditional Propositions*** | |
|---|---|---|
| | *"Parent" target BSD* | *Conditionally dependent target BSD* |
| ... before | Dependent target BSD only | Shared conditional proposition only |
| ... after | Shared conditional proposition only | "Parent" target BSD only |
| ... at any time (no order constraint) | Shared conditional proposition only | Shared conditional proposition only |

Table 6.8: Order Constraints Modeling

In the first case, the application of drug I depends on whether examination B *shall happen*. In the second case, the application of drug II is dependent on whether examination D *has happened*. Regardless of the requirements with respect to the order of activities, both drug applications depend on the relevant examination and thus share the examination's conditional environment. However, they differ in terms of their order constraint in regard to the respective examination. Nevertheless, both are part of the business objective, which – given the respective conditional environment – cannot be fulfilled unless the drugs are applied properly.

The considerations discussed above enable defining business objective achievement on the basis of target BSDs and conditional propositions. Thus, Semantic Requirement SR1 (*Consideration of the affecting environment*, cf. Table 6.1) is addressed.

**Definition 6 (Business Objective Achievement).** *A business objective is achieved iff each target BSD comprised in the business objective has assumed a state reflecting its conditional propositions. Thus, a business process has to approve or disapprove each conditional proposition and manipulate target artifacts accordingly.* □

Based on Definitions 5 and 6, BP models must enable approving and disapproving conditional propositions. As shown in Example 22, approving or disapproving conditional propositions *as early as possible* will contribute to BP optimization since enacting unnecessary tasks can be avoided [210].

**Example 22 (Early Approval and Disapproval of Conditional Propositions).** Consider Sample Process A in Figure 2.5. To approve an invoice, it is necessary to check whether a corresponding purchase order has been issued, whether a goods receipt has been posted, and, depending on the invoice value, whether senior management approval is given. As soon as one of these conditional propositions is not approved, the other conditional propositions do not have to be checked anymore. Instead, the invoice can be declined immediately without incurring further processing effort.

To facilitate designing business processes that consider this principle on the basis of business objective models, *necessary and sufficient sub-conditions* are discerned as possible constituents of conditional propositions.

**Definition 7 (Necessary and Sufficient Sub-conditions of Conditional Propositions).** *For conditional proposition $CP := NC_1 \wedge NC_2$, $NC_1$ and $NC_2$ constitute **necessary sub-conditions**. Any part of a conditional proposition that is conjunctively linked to the entire remainder of the conditional proposition (e.g. any subterm in a conjunctive normal form) constitutes a necessary sub-condition. If any one necessary sub-condition is not fulfilled, the conditional proposition is disapproved.*

*For conditional proposition $CP := SC_1 \vee SC_2$, $SC_1$ and $SC_2$ constitute **sufficient sub-conditions**. Any part of a conditional proposition that is disjunctively linked to the entire remainder of the conditional proposition (e.g. any subterm in a disjunctive normal form) constitutes a sufficient sub-condition. If any sufficient sub-condition is fulfilled, the conditional proposition is approved.* □

Sufficient and necessary sub-conditions can be identified by building minimal conjunctive and disjunctive normal forms for each conditional proposition (e.g., by way of a Karnaugh-Veitch diagram). The respective subterms provide minimal ways to either approve or disapprove a target BSD. As they are relevant for any BP implementation of a business objective, they are included in the semantic business objectives meta-model.

To fully capture the meaning of business objectives either formally or based on *a priori* knowledge, interrelations between target BSDs beyond conditional equivalence or dependency (cf. Definition 5) are considered as well. Target BSDs may be correlated or mutually exclusive. Correlation of target BSDs infers that if a BSD is required to achieve the business objective, all correlated BSDs will be required as well.[8] Mutual exclusivity implies that the business objective cannot be fulfilled if two respective target BSDs are both fulfilled. This is caused by "overlaps" in the conditional environment, i.e. conditional BSDs that are relevant for multiple target BSDs or in themselves correlated or mutually exclusive. Table 6.9 summarizes the possible interrelations between two fully determinant bivalent target BSDs that occur with common sub-conditions.

**Example 23 (Correlation and Mutual Exclusivity of Target BSDs).** Re-consider Sample Process C in Figure 2.7. If examination D is executed, drug III must be applied as well. Hence, the target BSDs "Examination D executed" and "Drug III applied" are correlated. Moreover, Sample Process C addresses the two target BSDs of "Condition X noted as existent" and "Condition X noted as nonexistent". The latter two target BSDs are mutually exclusive.

Besides common sub-conditions, correlation and mutual exclusivity may also be caused by interdependencies between conditional BSDs. Beyond the simple case of non-overlapping value ranges for conditional BSDs referring to a common conditional element, it is, however, not practical to capture these characteristics in business objective modeling. Hence, an effective business objective model will reflect ...

[8]Note that temporal concurrency would be an even more strict requirement, as it would demand that target BSDs are fulfilled *at the same time*.

| | Type of Common Sub-condition X | | | |
|---|---|---|---|---|
| | Target BSD A | | Target BSD B | |
| **Interrelation** | Necessary | Sufficient | Necessary | Sufficient |
| Common conditional branch | $X$ | | $X$ | |
| Mutually exclusive (fully determinate bivalent only) | $X$ | | $\overline{X}$ | |
| B is correlated to A | $X$ | | | $X$ |
| *[No proposition]* | $X$ | | | $\overline{X}$ |
| Possibly common approval | | $X$ | | $X$ |
| *[No proposition]* | | $X$ | | $\overline{X}$ |

*Note: $\overline{X}$ refers to an inversed sub-condition*

Table 6.9: Target BSD Interrelations

- multiple occurrences of individual necessary or sufficient sub-conditions in various conditional propositions linked to target BSDs as well as
- concurrent and mutually exclusive conditional BSDs referring to a common conditional element.

## 6.5 Business Objectives Meta-Model

The semantic concepts discussed in the previous section can be integrated into the RML meta-model presented in Figure 6.4.

### 6.5.1 Modeling Approach

The following modeling steps illustrate how a business objectives model which is compliant with the presented meta-model can be obtained. Following these steps is a possibility suggested with regard to Usability Criterion UC3 (cf. Table 6.1). The numbering included in Figure 6.4 reflects the order of modeling steps. Relevant explanatory notes regarding modeling concepts and their interrelations are comprised as well. Capital letters represent sets of constructs where all elements are of the same type. Each step is followed by an example which applies the respective step to Sample Process C from Figure 2.7.

**Step 1 (List Target BSDs).** Based on the target artifacts of the *business objective*, all relevant target BSDs including their types are listed. The respective *conditional propositions* may be modeled in a later step to make use of implicitly available knowledge on the business process first and limit modeling effort.

A *business objective bo* comprises a set of *target BSDs* $TB_{bo}$. The *business objective* is achieved iff all comprised *target BSD*s have assumed a target value.

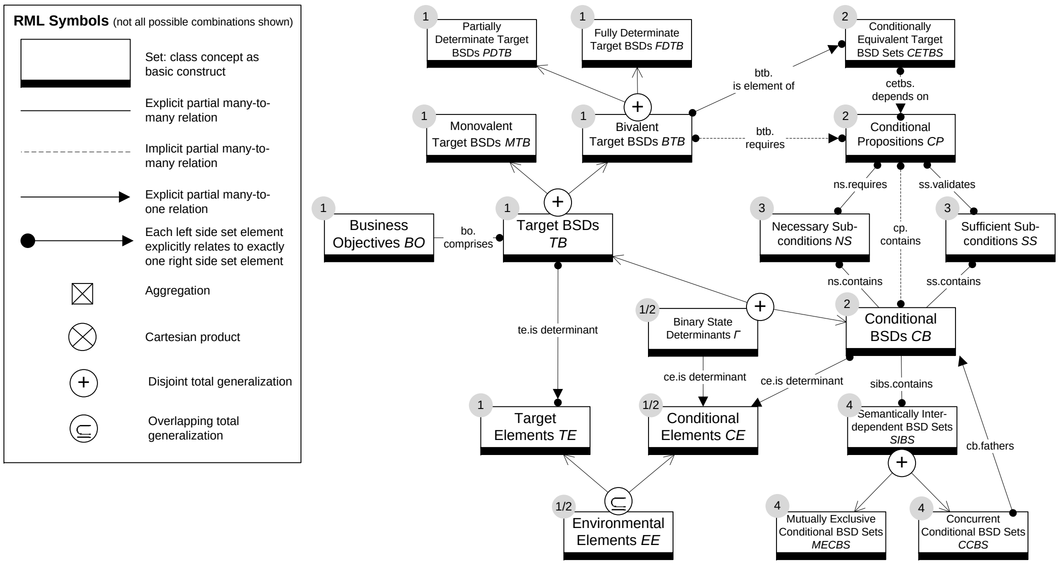



Figure 6.4: Business Objective Meta-Model

According to Definition 4, a *target BSD* might be a *monovalent target BSD* $mtb \in MTB$ or a *bivalent target BSD* $btb \in BTB$, i.e.,

$$TB = MTB \,\dot{\cup}\, BTB$$

A *bivalent target BSD* might be a *fully determinate target BSD* $fdtb \in FDTB$ or a *partially determinate target BSD* $pdtb \in PDTB$, i.e.,

$$BTB = FDTB \,\dot{\cup}\, PDTB$$

Note that *trivalent target BSDs* are modeled as two *partially determinate target BSDs* as described in Section 6.4. *Target BSDs* and *conditional BSDs* $cb \in CB$ are *BSDs* $\gamma \in \mathit{\Gamma}$, i.e.,

$$\mathit{\Gamma} = TB \,\dot{\cup}\, CB$$

A *BSD* $\gamma = \langle E_\gamma, \Lambda_\gamma \rangle$ is defined by a non-empty sequence of *environmental elements* $E_\gamma = (e_{\gamma_1}, \ldots, e_{\gamma_n})$, $e_{\gamma_i} \in E$, and a sub-set of the cartesian product of their domains $\Lambda_\gamma \subseteq \mathcal{D}(e_{\gamma_1}) \times \ldots \times \mathcal{D}(e_{\gamma_n})$ (cf. Definition 3). Each *target BSD tb* refers to a *target element* $te \in TE$, and each Conditional BSD *cb* to a *conditional element* $ce \in CE$ as its compulsory *environmental element*, i.e.,

$$\exists\, e \in E_\gamma : e \in \begin{cases} TE & \text{if } \gamma \in TB \\ CE & \text{if } \gamma \in CB \end{cases}$$

Each *BSD* $\gamma$ may refer to additional *conditional elements* $CE_\gamma \subseteq CE$ as part of its sequence of *environmental elements*. *Target elements* $te \in TE$ and *conditional elements* $ce \in CE$ are

| Target BSDs | Target BSD types |
|---|---|
| Result A available | Monovalent |
| Drug I applied | Fully determinate bivalent |
| Result B available | Fully determinate bivalent |
| Result C available | Fully determinate bivalent |
| Drug II applied | Fully determinate bivalent |
| Condition X marked | Partially determinate bivalent |
| Condition X not marked | Partially determinate bivalent |
| Result D available | Fully determinate bivalent |
| Drug III applied | Fully determinate bivalent |

Table 6.10: Sample Target BSDs

*environmental elements* $ee \in EE$. A *target element* may also be a *conditional element*, i.e.,

$$EE = TE \cup CE$$

**Example 24 (List Target BSDs Including Types).** For Sample Process C (cf. Figure 2.7), "Examination C executed" is not monovalent due to order restrictions (Examination C can only be executed after Examination A). Moreover, it is assumed that medical examinations as well as medications are not arbitrary, i.e. they should only be executed in case of a clear indication. Note that the originally *trivalent target BSD* "Condition X marked" is deconstructed into two *partially determinate target BSDs*. In accordance to these considerations, results of listing *target BSDs* including the respective types are presented in Table 6.10.

**Step 2 (Normalize Bivalent Target BSDs).** To "normalize" *bivalent target BSDs*, conditionally equivalent sets are built. To limit modeling effort, normalization can initially be conducted based on implicit knowledge without formally considering *target BSDs' conditional propositions*.[9]

According to Definition 5, *fully determinate target BSDs* are "rephrased" (i.e., negated) to join a conditionally equivalent set if the respective *conditional proposition* is a negation of a set's joint *conditional proposition*. Note that this is not possible for *partially determinate target BSDs*. Each *bivalent target BSD* $\gamma \in \Gamma$ is an element of one *conditionally equivalent target BSD set* $cetbs_{btb}$ sharing one *conditional proposition* $cp_{cetbs_{btb}}$. *Conditional propositions* are then made explicit as logical expression of *conditional BSDs* considering the convention for order constraints in Table 6.8. A *conditional proposition cp* is fulfilled iff its logical expression is fulfilled, i.e.,

$$fulfilled(cp) := \begin{cases} true & \text{if the logical expression for } cp \text{ is fulfilled} \\ false & \text{else} \end{cases}$$

[9]As an example for implicit available knowledge, consider Sample Process C: a physician will know that examination B requires drug I without modeling conditions first.

On that basis and according to Definition 6, a *business objective bo* is fullfilled iff the states of its *target BSDs* and the respective *conditional propositions* are coherent considering *target BSD* types, i.e.,

$$\begin{aligned} achieved(bo) := \forall mtb \in MTB_{bo} : \ & fulfilled(mtb) \wedge \\ \forall fdtb \in FDTB_{bo} : \ & fulfilled(cp_{fdtb}) \Leftrightarrow fulfilled(fdtb) \wedge \\ \forall pdtb \in PDTB_{bo} : \ & fulfilled(cp_{pdtb}) \Rightarrow fulfilled(pdtb) \end{aligned} \quad \square$$

**Example 25 (Normalize Bivalent Target BSDs).** There are no *conditionally equivalent target BSD sets* containing more than one *target BSD* in the example, as illustrated in Table 6.11. For comparison, Table 6.11 also shows how the normalized target BSD sets would change when not considering order constraints.

**Step 3 (Resolve Conditional Propositions).** *Conditional propositions* are resolved into *necessary and sufficient sub-conditions* according to Definition 7. Each *conditional proposition* can be decomposed into a set of *necessary sub-conditions* $NS_{cp}$ and a set of *sufficient sub-conditions* $SS_{cp}$, i.e.,

$$\begin{aligned} fulfilled(cp) &\Leftrightarrow \forall\ ns \in NS_{cp} : fulfilled(ns) \\ &\Leftrightarrow \exists\ ss \in SS_{cp} : \ fulfilled(ss) \end{aligned}$$

Each *necessary sub-condition ns* and each *sufficient sub-condition ss* contain a set of least one *conditional BSD* $CB_{ns}$ or $CB_{ss}$. A *necessary sub-condition ns* is fulfilled iff at least one of its *conditional BSDs* is fulfilled, i.e.,

$$fulfilled(ns) \Leftrightarrow \exists\ cb \in CB_{ns} : \ fulfilled(cb)$$

A *sufficient sub-condition ss* is fulfilled iff all of its *conditional BSDs* are fulfilled, i.e.

$$fulfilled(ss) \Leftrightarrow \forall\ cb \in CB_{ss} : \ fulfilled(cb)$$

*Necessary and sufficient sub-conditions* are modeled in consolidated form, i.e., equivalent sub-conditions for multiple *conditional propositions* are modeled only once. The decomposition of *conditional propositions* into sub-conditions can also be used to identify conditional equivalences not recognized yet. □

**Example 26 (Resolve Conditional Propositions).** Table 6.12 shows the resolution of *conditional propositions* into *necessary and sufficient sub-conditions*.

Figure 6.5 presents a graphical notation of the results up to now based on the exemplary content for Sample Process C. The format is simplified as it presents either necessary or sufficient sub-conditions (in case of only one conditional BSD comprised in a conditional proposition, the differentiation is unnecessary). Since modeling is executed in a consolidated form, there is just one “column” for each conditional BSD or sub-condition comprised in

Result with consideration of order constraints:

| $CETBS_{bo}$ | BSD Types | Conditional Propositions |
|---|---|---|
| Result A available | Monovalent | none |
| Drug I applied | Fully determinate bivalent | [Result A > 50] |
| Result B available | Fully determinate bivalent | [Drug I applied] |
| Result C available | Fully determinate bivalent | [Result A available] |
| Drug II applied | Fully determinate bivalent | [Result C > 100] AND [Age > 50] |
| Condition X marked | Partially determinate bivalent | ([Result A > 100] OR [Result B > 100]) AND [Result C ≤ 100] |
| Condition X not marked | Partially determinate bivalent | ([Result A > 100] OR [Result B > 100]) AND [Result C > 100] |
| Result D available | Fully determinate bivalent | [Result A > 100] OR [Result B > 100] |
| Drug III applied | Fully determinate bivalent | [Result D available] |

Alternative result without consideration of order constraints:

| $CETBS_{bo}$ | BSD Types | Conditional Propositions |
|---|---|---|
| Result A available, Result C available | Monovalent | none |
| Drug I applied, Result B available | Fully determinate bivalent | [Result A > 50] |
| ... | ... | ... |
| Result D available, Drug III applied | Fully determinate bivalent | [Result A > 100] OR [Result B > 100] |

Table 6.11: Sample Normalization of Target BSDs

Figure 6.5. Conditional elements which also constitute target elements are comprised in the "line" of the respective target BSD. This is the case for all conditional elements except the patient's age. The figure is to be read as follows: to achieve the business objective,

- the monovalent target BSD set must be fulfilled,
- all elements of bivalent target BSD sets for which necessary sub-conditions have been modeled must be fulfilled if all sub-conditions for the set are fulfilled, and
- all elements of bivalent target BSD sets for which sufficient sub-conditions have been modeled must be fulfilled if at least one sub-condition for the set is fulfilled.

Note that circular relations between target BSDs (i.e., one target BSD as conditional element of another which is also a conditional element of the first target BSD etc.) must not occur,

| | Conditional Propositions | |
|---|---|---|
| $CETBS_{bo}$ | Necessary Sub-conditions | Sufficient Sub-conditions |
| Drug I applied | [Result A > 50] | [Result A > 50] |
| Result B available | [Drug I applied] | [Drug I applied] |
| Result C available | [Result A available] | [Result A available] |
| Drug II applied | ([Result C > 100])<br>([Age > 50]) | ([Result C > 100] AND [Age > 50]) |
| Condition X marked | ([Result A > 100] OR [Result B > 100])<br>([Result C ≤ 100]) | ([Result A > 100] AND [Result C ≤ 100])<br>([Result B > 100] AND [Result C ≤ 100]) |
| Condition X not marked | ([Result A > 100] OR [Result B > 100])<br>([Result C > 100]) | ([Result A > 100] AND [Result C > 100])<br>([Result B > 100] AND [Result C > 100]) |
| Result D available | ([Result A > 100] OR [Result B > 100]) | ([Result A > 100])<br>([Result B > 100]) |
| Drug III applied | [Result D available] | [Result D available] |

Table 6.12: Sample Target BSDs with Resolved Conditional Propositions

because in that case the business objective could not be achieved by any business process. Figure 6.5 can thus be read from the top down.

**Step 4 (Consolidate Conditional BSDs).** To consolidate *conditional BSDs*, *semantically interdependent BSD sets* are identified. A *semantically interdependent BSD set sibs* comprises a number of *conditional BSDs* $CB_{sibs}$ and may either be a *mutually exclusive conditional BSD set mecbs* or a *concurrent conditional BSD set ccbs*. Each *mutually exclusive conditional BSD set* comprises at least two *conditional BSDs* with:

$$fulfilled(cb) \mid cb \in CB_{mecbs} \Rightarrow \nexists \; ecb \in (CB_{mecbs} \setminus cb) : \; fulfilled(ecb)$$

Each *concurrent conditional BSD set* comprises at least one *conditional BSD* and refers to one *conditional BSD* $cb_{parent_{ccbs}}$ which is the "parent" of the set:

$$fulfilled(cb_{parent_{ccbs}}) \Rightarrow \forall \; ccb \in CB_{ccbs} : \; fulfilled(ccb)$$

Mutual exclusivity of *conditional BSDs* propagates to *necessary sub-conditions* that consist of just the one *conditional BSD*, rendering the respective *conditional propositions* and hence *target BSDs* mutually exclusive as well. Correlation propagates to *sufficient sub-conditions* that consist of just the one *conditional BSD*, rendering the respective *conditional propositions* and hence *target BSDs* correlated as well.[10]

[10]See Table 6.9 for relations caused by common sub-conditions.

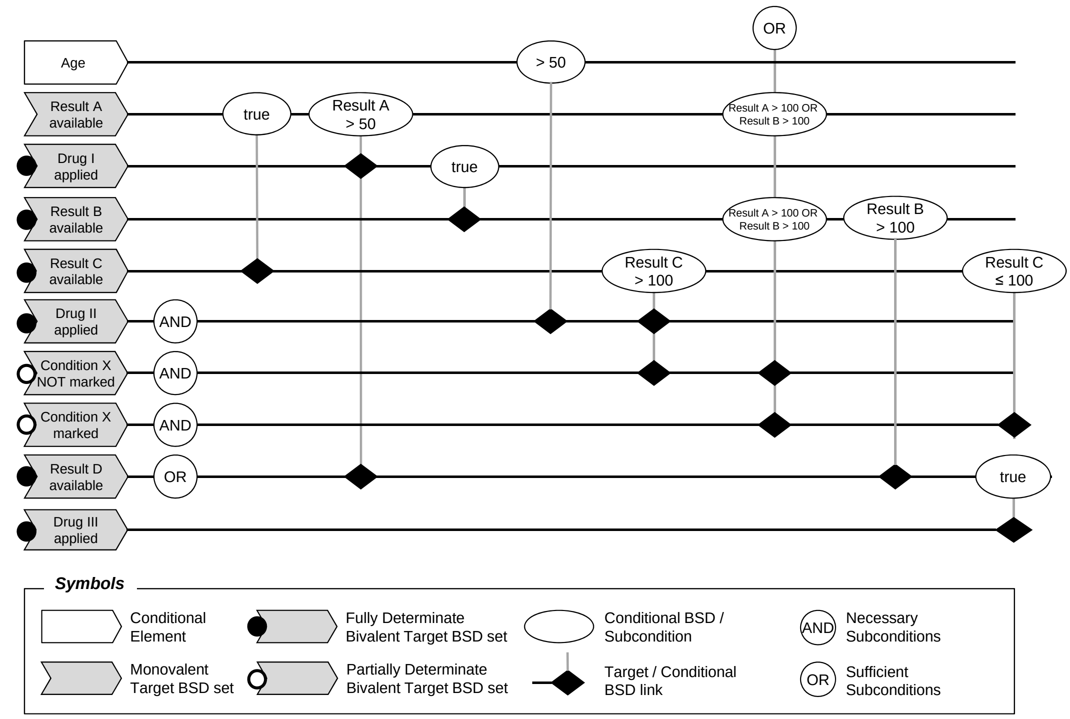



Figure 6.5: Sample Conditional Consolidation

Mutual exclusivity and correlation are obvious if the respective BSD set relates to the same *conditional element*. In that case, mutual exclusivity is caused by non-overlapping value ranges, and correlation is caused by partial quantity relations in value ranges. However, this is not a strict prerequisite.

Note that usually not all interdependencies in the outer environment are known to the modeler. Hence, this modeling step may lead to a partial result reflecting best knowledge. □

**Example 27 (Consolidate Conditional BSDs).** In the sample case, consolidation is done on the basis of *conditional elements* shared between *conditional BSDs* only, i.e., no further semantic interrelations between *conditional BSDs* are assumed. Thus, consolidation results can easily be derived from Figure 6.5 by considering line by line:

- Concurrent Conditional BSD Set: [Result A $>$ 50] $\Rightarrow$ [Result A available]
- Mutually Exclusive Conditional BSD Set: [Result C $>$ 100] $\Leftrightarrow \neg$ [Result C $\leq$ 100]

Accordingly, application of Drug I and Examination C are correlated, while marking Condition X is mutually exclusive with application of Drug II and – obviously – *not* marking Condition X. □

### 6.5.2 Evaluation against Effectiveness Criteria

To evaluate results, the criteria defined in Table 6.1 are considered:

- *SR1:* The approach builds on target and conditional elements. Accordingly, both relevant aspects of the outer environment are covered effectively.
- *SR2:* The relevance of target BSDs is determined considering the conditional environment. Together with partially determinate bivalent target BSDs, this enables target artifact sets varying with the conditional environment.
- *SR3:* Order constraints can be modeled via a convention (cf. Table 6.8).
- *UC1:* Interdependencies are captured via the normalization of target BSDs and conditional consolidation. Necessary and sufficient sub-conditions can directly be used to optimize control flow via approval / disapproval strategies.
- *UC2:* The resulting model is compact and apt for graphic presentation (cf. Fig. 6.5). Imagine, for comparison, full enumeration of the conditional environment and the related aspired states. There are no redundant model elements.
- *UC3:* By setting out with target elements, modeling becomes intuitive and less prone to errors of omission. The approach also allows capturing available relevant knowledge before formal modeling. Available modeler knowledge could be captured through the "guided" modeling steps. However, this topic is obviously subject to individual preferences.

## 6.6 Conclusion

In this chapter, an approach to business objective modeling has been developed by deriving a semantically enriched meta-model and a corresponding modeling methodology. The approach fulfills semantic requirements derived from typical application scenarios as well as additional usability criteria for practical adoption. Most prominently, and as opposed to related work, it addresses both the affecting *and* the affected environment of business objectives. To this end, it relates states of the affecting environment to aspired states of the affected environment, thus modeling conditions that determine the actually desired results of enacting a business process. Future work in this area might address the promising application scenarios facilitated by the presented approach to business objectives. As an example, consider automated ongoing optimization of control flow from Scenario 1. Leveraging the concept of necessary and sufficient sub-conditions might be beneficial in this respect. Beyond the use cases lined out already, additional areas of application such as formal control of BP chains in functionally structured organizations or in service-oriented architectures may be explored.

Chapter 7 will complement the business objectives modeling approach by developing a quality-aware BP modeling approach which addresses efficacy and efficiency of business processes.

# 7 Quality-aware Business Process Modeling

Chapter 5 discussed aspects relevant to BP quality on the basis of the outer environment of processes and their potential impact on organizational targets. It concluded with a definition framework for BP quality considering the lifecycle stages of *design & implementation* and *enactment* as well as the quality dimensions of *efficacy* and *efficiency*. In practical settings, however, this definition framework will not suffice to appraise and manage BP quality since it does not provide concepts to determine the level of quality actually achieved. Therefore, a more detailed consideration of BP models is required.

Accordingly, this chapter develops a *quality-aware BP modeling* approach considering the semantics needed to assess corresponding models in terms of process quality and to define individual quality attributes (cf. Section 3.2). Moreover, the underlying analysis of process model aspects relevant to quality constitutes, in itself, a first step towards an elaborate set of quality attributes.

This chapter is structured as follows: Section 7.1 discusses requirements to be addressed in quality-aware modeling, and the approach taken. Section 7.2 provides an overview on related work. Section 7.3 describes required terminology and integrates the respective constructs into a meta-model. Section 7.4 presents steps to deduct *quality relations* from a BP model compliant to the meta-model of Section 7.3 on the basis of a sample process. To improve readability, the corresponding formal concepts are presented separately in Section 7.5. Section 7.6 summarizes the results achieved.

## 7.1 Preliminary Considerations and Methodology

According to Definition 1, BP quality refers to the relation between the achievement of a business objective and the resources required for this purpose. As shown in Figure 7.1, this can pertain to the mere *availability* or to the *consumption* of resources. Note that resources to be consumed are comprised in the resources required to be available. However, resources required to be available are not necessarily consumed by enacting the business process (cf. Example 28). In line with Definition 1, these aspects can be understood as the *efficacy relation* and *efficiency relation*, respectively.

**Example 28 (Resource Availability vs. Resource Consumption).** Consider Sample Process C from Figure 2.7. In the course of the medical examinations and treatments,

This chapter is based on the following referred papers:

Lohrmann, M., Reichert, M.: Efficacy-aware business process modeling. In: Proc. 20th Int'l Conf. on Cooperative Information Systems (CoopIS'12). Volume 7565 of LNCS, Springer (2012) 38–55

Lohrmann, M., Reichert, M.: Formalizing concepts for efficacy-aware business process modeling. Technical Report UIB-2012-05, Ulm University, Germany (2012)

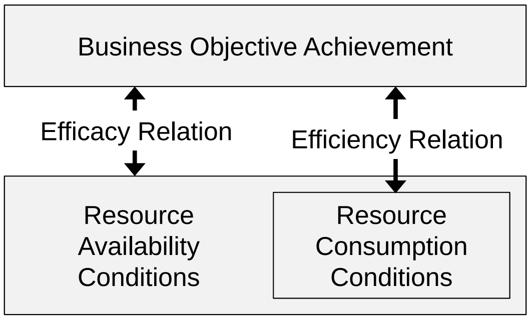



Figure 7.1: Quality Relations

various resources are required: e.g., medications to be applied, the availability of physicians and nurses, laboratory equipment. These categories of resources can be classified as follows:

- Medications to be applied need to be available to enact the treatment, and will be consumed after the treatment has been enacted.
- Similarly, physicians and nurses need to be available to enact examinations and treatments. Their available working time is consumed by process enactment since they cannot work on more than one patient at a time.
- Laboratory equipment needs to be available for certain examinations, but will not be consumed when used. Thus, this is a resource required to be available, but not consumed by the business process.

Regarding resources to be available but not consumed, a certain degree of individual judgment is involved in the classification. If the laboratory would be fully utilized throughout the day, lab time could be considered as a resource which, similar to working time, is consumed by enacting a process. Information resources constitute an exception in this regard: since they can be used in parallel by many business processes without being "spent", they constitute a class of resources generally subject to availability requirements only.

This chapter develops concepts to enable *quality-aware BP modeling* in the sense of extracting the relations described above from BP models. Figure 7.2 summarizes the intentions behind this approach: business objective and BP models allow deducting resource availability and consumption requirements (i.e., the efficacy and efficiency relations) which, in turn, enable quality assessment. For the purposes of this chapter, the effectiveness criteria lined out in Section 3.1 can be further refined:

- Since the efficacy and efficiency quality requirement dimensions have been designed to fully accommodate organizational targets as impacted by business processes, *Congruence to organizational targets* translates to the full coverage of the respective quality relations through quality-aware BP modeling. This amounts to a *functional requirement* for quality-aware BP modeling.

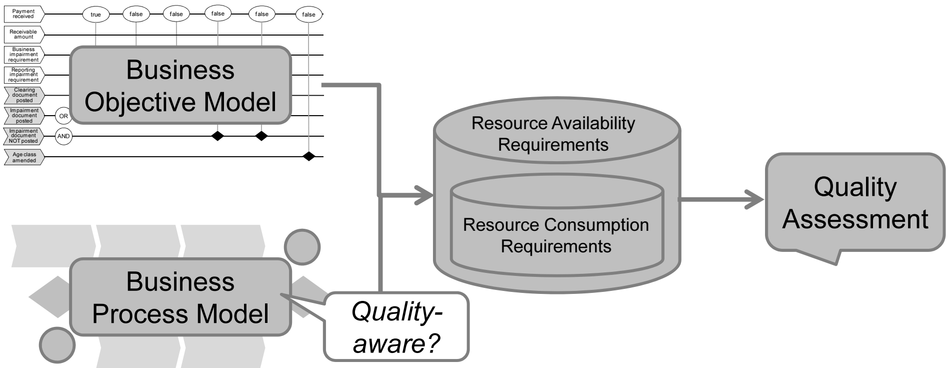



Figure 7.2: Quality-aware Models to Enable Quality Assessment

- *Transparency and retraceability* as well as *Cost effectiveness* can be addressed by closely relating constructs designed to prevalent and broadly accepted BP modeling languages. This enables using existing knowledge, methods and tools as far as possible which, in turn, amounts to a *practical applicability requirement.*

In principle, assessing BP quality refers to assessing the efficacy and efficiency relations given by a business process. By formalizing these relations, the quality-aware BP modeling approach presented here thus provides the “language” to express a refined BP quality model in the following chapters.

## 7.2 Related Work

To address the quality relations from Figure 7.1, it is necessary to integrate the aspects of business objective achievement, resource availability, and resource consumption into BP models. Accordingly, Table 7.1 looks into common BP modeling approaches in regard to these aspects. [22] presents a representative collection of BP modeling approaches adopted for this assessment.

Further to the more common paradigm of imperative process modeling, which is basically oriented at the sequence of tasks in a process, *Declare* was included to cover the alternative paradigm of declarative process modeling [211, 212].[1] Declarative process modeling does not prescribe permitted task sequences, but rather defines sets of tasks as well as constraints with regard to their permitted combination into a process instance. This is achieved by using Linear Temporal Logic (LTL) [214], which is translated into graphical representations for various constraint patterns. The declarative paradigm may prove advantageous in settings where a high degree of flexibility is required. The results of a empirical investigations of declarative process modeling in terms of understandability can be found in [215, 216, 217]. Table 7.1 comprises Declare to represent declarative process modeling.

[1]Note that the Declare language is referred to as “ConDec” and “DecSerFlow” in [211] and [212], respectively. This naming has been changed by now [213].

Additional examples of declarative process modeling approaches comprise dynamic condition response (DCR) graphs and the EM-BrA²CE vocabulary. Both are shortly described in the following. *DCR graphs* constitute another proposition towards declarative process modeling [218]. In comparison to Declare, DCR graphs restrict the set of available constraint patterns to only four patterns instead of permitting the full expressiveness of LTL, with the aim of improving the runtime efficiency of corresponding process enactment tools.

In turn, the *Semantics of Business Vocabulary and Business Rules* (SBVR) meta-model defined by the Object Management Group seeks to specify business domains through a corresponding vocabulary and a set of applicable business rules [219]. Based on earlier work [203], Goedertier et al. propose to extend SBVR to integrate declarative process modeling capabilities by way of the EM-BrA²CE vocabulary ("enterprise modelling using business rules, agents, activities, concepts and events") [204]. In this context, 16 business rule types are used to constrain the "state space" that is permitted when enacting a business process.

Considering the overview on available BP modeling approaches in Table 7.1, one can conclude that quality-aware BP modeling requires additional modeling semantics since resource requirements are in general not fully considered. In the following sections, the relevant concepts are elucidated and shaped into a corresponding extended BP modeling concept.

## 7.3 Terminology and Meta-Model for Quality-aware Business Process Models

This section builds constructs to integrate business process and business objective models to enable analysis of the quality relations between business objective achievement and resource requirements. Chapter 6 lined out the business objectives meta-model used as a basis. With respect to BP modeling, this chapter refers to BPMN [80] as a broadly applied language covering common modeling concepts. This ensures that the results presented here can be integrated with common BP modeling languages. In this respect, the status of BPMN as an industry standard is highly relevant. Combining the constructs developed provides a *quality-aware BP meta-model*. Considering the notion of business objectives discussed in Chapter 6, it is possible to define:

**Definition 8 (BP Model Efficacy).** *A BP model is* formally efficacious *iff (1) it addresses all target BSDs in its business objective and (2) no target BSD in its business objective can be fulfilled unless the respective conditions defined by the business objective are fulfilled.*

*A BP model is* fully efficacious *iff (1) it is formally efficacious and (2) all conditions which the model poses to target BSDs in addition to those defined by the business objective are considered as reasonable by subject matter experts.*

*A BP model is* ideally efficacious *iff (1) it is formally efficacious and (2) there are no additional conditions posed to target BSDs beyond those defined by the business objective.* ☐

**Definition 9 (BP Model Efficiency).** *A BP model is* efficient *iff all resource consumption requirements which the model poses to target BSDs are considered as reasonable by subject matter experts.*

*A BP model is* ideally efficient *iff it poses no resource consumption requirements to target BSDs.* ☐

| **Modeling Approach** | **Business Objectives Representation** | **Resource Availability and Consumption Representation** |
|---|---|---|
| **Petri Nets** [220, 221] | Petri nets address the functional perspective of process modeling only. Accordingly, business objectives and resource requirements are not considered. | |
| **Event-driven Process Chains** [222] | The Event-driven Process Chains approach provides three “pillars” to describe aspects relevant to BP management on various levels of abstraction. Business objectives are typically described on the “requirements definition” level, i.e. with the highest degree of abstraction, as part of the “functions” pillar. | Since Event-driven Process Chains are oriented at IT-supported processes, the “data” pillar addresses information resources only. The Entity Relationship diagrams used for modeling in this perspective do not restrict expressiveness in terms of model content. However, the functional perspective’s impact on the data perspective and vice versa, which encompasses resource consumption and resource availability requirements, is addressed only on an abstract level by modeling “associations”. |
| **Workflow Nets** [143] | Workflow nets extend Petri nets with additional semantics useful for BP modeling. Again, workflow nets focus on control flow aspects. Business objectives are thus not represented. | While resource consumption is not addressed, resource availability requirements may be modeled using the exclusive or split pattern, i.e. through a control flow-based workaround. This would, however, require translating all task resource availability requirements into additional split patterns. |
| **YAWL** [78] | Yet Another Workflow Language (YAWL) constitutes an extension of workflow nets to address additional workflow patterns [223]. Business objectives, however, are still not included. | Like in workflow nets, YAWL does not specify the effect of process enactment on resources (or other elements of the outer environment). Again, resource availability requirements can be modeled with a control flow-based workaround, but this is not advisable for practical settings. |
| **BPMN** [80] | Business Process Model and Notation (BPMN) is oriented at defining control flow in the sense of structuring activities. Thus, the underlying objectives are not considered, but treated as an implicit prerequisite to process modeling. | BPMN represents resources via the “data object” construct which addresses information items, and via the “swimlane” construct which can be used to model organizational responsibilities. Other types of resources are not considered. This reflects the orientation of BPMN at designing workflows supported by appropriate IT tools. Beyond the textual description of the content of tasks, effects on resources are not considered. |
| **Declare** [211] | As opposed to the approaches presented above, Declare follows a declarative instead of an imperative process modeling paradigm, i.e., the sequence of tasks is arbitrary unless respective constraints apply. However, business objectives are not considered. | Declare covers the modeling of tasks and corresponding enactment constraints. Resources are not considered. |

Table 7.1: Quality-aware Business Process Modeling: State of the Art

Definitions 8 and 9 reflect the notion of "reasonable" requirements (cf. Definition 1). Assessing efficacy and efficiency requires to analyze process models regarding the conditions they pose towards fulfilling target BSDs. To assess formal efficacy, the conditions obtained are then compared to the conditions posed by the objective model. Moreover, to assess full efficacy and efficiency, they are matched against subject matter experts' expectations. In the following chapters, characteristics to determine whether efficacy and efficiency relations can be considered as "reasonable" will be developed on the basis of the results presented in this chapter. They will be represented through the concept of *quality attributes*.

**Example 29 (Efficacy Assessment).** Consider a loan approval process. Comparing it to the business objective will clarify whether the process addresses the respective target BSD:

- Does the process include tasks where the loan decision is logged?
- Are decision criteria for loan approval such as the credit history observed?

These questions allow determining whether the process is formally efficacious. Moreover, to establish full efficacy it is necessary to consider whether an unreasonable amount of working time is required to enact the process. If this is the case, the process will not be efficacious *in practice* since the organization cannot supply sufficient resources to enact the process on a timely basis. Since this is a matter of implementation through a business process, it is not documented in the business objective, but requires further consideration of the process model, and the judgment of subject matter experts.

Note that neither *ideally efficacious* nor *ideally efficient* business processes occur in practice as each business process consumes resources not being part of the business objective (e.g., expenditure of labor during enactment or implementing information systems during design & implementation). This leads to the following definition:

**Definition 10 (Quality-aware BP Model).** *A BP model is* quality-aware *if and only if it is sufficiently expressive to assess whether it is fully efficacious and fully efficient. A BP meta-model is quality-aware if and only if it ensures that each creatable BP model is quality-aware.* □

As a stipulation to be demonstrated in the following sections, efficacy and efficiency of a BP model can be assessed if information on the relations between target BSDs and their environmental conditions is available according to the meta-model given in Figure 7.3. Again, the RML as presented in [195] is used since it provides a concise means of describing set relations.

Note that, according to Figure 7.1, resource availability conditions encompass resources to be consumed since these must be available as well. The meta-model in Figure 7.3 closely resembles the business objectives meta-model in Chapter 6, because environmental conditions have to be related to induced target BSDs in a similar manner. Note that the first precondition towards formal efficacy, i.e. that all target BSDs are addressed by a business process, is implicitly covered in Figure 7.3. As discussed above, enacting processes generally requires resources or the fulfillment of a conditional proposition. However, if a process poses a conditional proposition towards a target BSD, the target BSD is addressed by the business process. To enable subject matter experts to assess whether the relation can be considered as

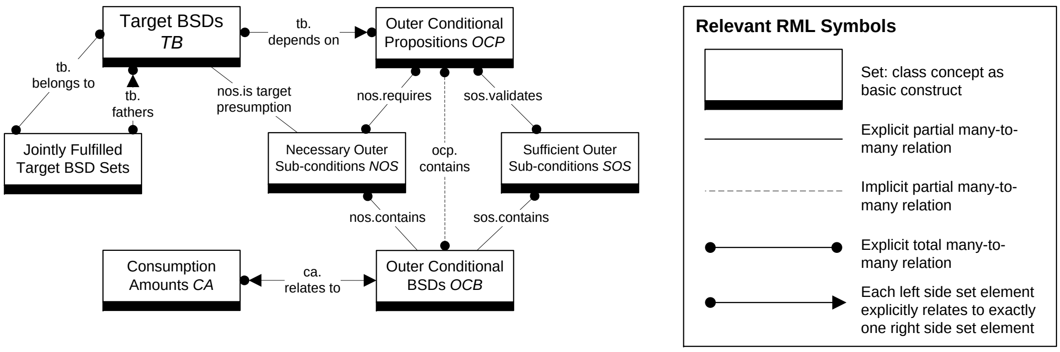



Figure 7.3: Relating Target BSDs and the Outer Conditional Environment

reasonable, necessary and sufficient sub-conditions are evaluated similarly to the definition of business objectives (cf. Chapter 6). That is, if at least one necessary outer sub-condition is not fulfilled, a target BSD cannot be fulfilled. *Sufficient outer sub-conditions* comprise sets of BSDs that suffice to determine that a process instance will fulfill a target BSD: As soon as at least one sufficient outer sub-condition is fulfilled, the target BSD will be fulfilled as well.

**Example 30 (Necessary and Sufficient Sub-conditions).** Re-consider the approval of loans. The availability of customer master data and the responsible manager both constitute necessary sub-conditions. Assuming that the customer's credit history is usually available in the data base, but may also have to be obtained manually, there are two sufficient sub-conditions: the described necessary sub-conditions plus the data base entry, and the described necessary sub-conditions plus the availability of a clerk for manual evaluation.

Moreover, to achieve formal efficacy (cf. Definition 8), the environmental conditions resulting from a BP model with respect to a target BSD must "encompass" the environmental conditions specified in the objective model. More precisely, each necessary sub-condition of the target BSD in the business objective should be a necessary sub-condition in the process model as well. Therefore, the following sections discern between the *outer conditional environment* and the *inner conditional environment* defined by process and objective model, respectively. Figure 7.4 summarizes these considerations: formal efficacy is violated if the inner conditional environment defined by the business objective is not fully comprised in the outer conditional environment defined by the business process. Full efficacy, in turn, is impeded if the outer conditional environment becomes too "large", i.e., if the business process poses too many prerequisites to achieve target BSDs. Note that in contrast to business objective models, instead of just bivalent target BSDs, each target BSD is related to an outer conditional proposition. This reflects that activities to induce BSDs generally require resources, which are part of the outer conditional environment.

The *consumption amounts* construct accommodates the characteristic of resource *consumption* conditions' being comprised in resource *availability* conditions. Accordingly, resource consumption conditions are not modeled separately, but as an attribute to the wider concept of outer conditional BSDs. Note that the attribute models a sub-condition to the outer

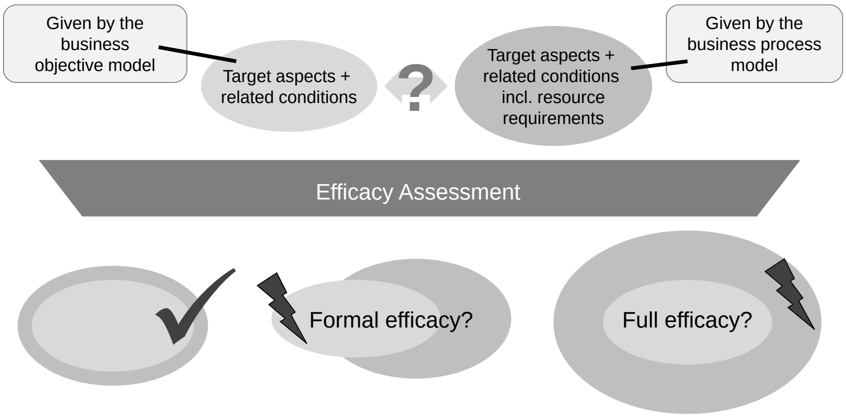



Figure 7.4: Efficacy Assessment Based on Inner and Outer Conditional Environment

conditional BSD since it is also conceivable that only part of a resource amount that has to be available is actually consumed by a business process. As an example, consider minimum storage amounts for materials kept for other processes of higher priority.

As an additional requirement towards efficiency assessment, it must be considered that resources consumed may be "shared" between target BSDs. For example, the amount of human labor expended to complete a process instance addressing multiple target BSDs generally cannot be attributed to an individual target BSD. Rather, the value of resources consumed must be matched against the target BSDs fulfilled in a consolidated manner. Note that this does not apply to efficacy assessment since business objectives demand that each target BSD is fulfilled according to present environmental conditions with as few additional prerequisites as possible. This consideration reflects the tradeoff between efficacy and efficiency that occurs in many practical cases, and is illustrated in Example 31. Accordingly, Figure 7.3 comprises a construct modeling *jointly fulfilled target BSD sets* in the sense of sets of target BSDs that could be fulfilled by a single process instance. Each set consists of at least one target BSD and exactly one "parent" target BSD. If the "parent" target BSD is fulfilled, all other BSDs in the set are fulfilled as well, i.e., the "parent" BSD would be fulfilled by the last relevant task of the process instance. Note that if the last relevant task fulfills multiple target BSDs, one may be chosen arbitrarily.

**Example 31 (Efficacy and Efficiency Tradeoff).** Consider the process of managing customer data in a mail order business. If a new order arrives, a clerk is tasked to print an address form for shipping and to earmark the customer to receive the next catalog. For both targets, it is required to check whether the address data is already available in the data base. To fulfill this requirement, it would be possible to include a single task to print the address form out of the database, which would also address the check for catalog earmarking (Option A). As an alternative, it would be possible to separate the catalog address check from address printing (Option B).

In terms of efficacy, both options are equivalent since the same prerequisites (e.g., availability of the address data base) must be given to fulfill the target BSDs. In terms of efficiency, however, Option A would be preferable as it uses the address printing task to check whether the address is available without additional effort. Nevertheless, when considering catalog earmarking as an isolated target, Option B would seem advantageous since address checking requires less effort than address printing. Accordingly, for efficiency assessment the full set of target BSDs must be matched with all resources consumed *at the same time* to enable correct and comprehensive appraisal.

The concept of *target presumptions* represents environmental conditions which are not caused by control flow, but by the state operation inducing a target BSD. For instance, this occurs if a process does not aim at creating, but at altering a target artifact. Further illustration is provided in Example 32 and in Section 7.4. On that basis, the terminology a quality-aware BP meta-model needs to cover can be derived, and the resulting constructs can be related to available BPMN terms.

**Example 32 (Target Presumptions).** Re-consider the mail order business. If a new order arrives, a clerk might be tasked to print an address form for shipping and to earmark the customer to receive the next catalog. If a customer returns an item, the purchase order must be marked accordingly. Thus, the purchase order must be available to enact the state operation *mark as returned*. Accordingly, the availability of the purchase order constitutes a *target presumption*.

To assess the efficacy of a business process, the necessary and sufficient sub-conditions given by the BP model can be compared to necessary and sufficient sub-conditions given by the objective model for each target BSD. To assess the efficiency of a business process, the necessary and sufficient sub-conditions given by the BP model in terms of consumption amounts for each set of target BSDs that can be jointly fulfilled are considered. The required steps will be described in the next section.

Typically, BPMN and comparable languages allow modeling the execution semantics of business processes in terms of a sequence of tasks refined by split and join gateways. In that respect, widely used “workflow patterns” are defined in [223]. In BPMN, the modeler is generally free with respect to the level of granularity regarding tasks and activities as atomic or aggregate constructs. In the context of this thesis, this degree of freedom is limited to obtain stricter execution semantics. Accordingly, tasks are required to be enacted atomically, i.e. either in total or not at all. Thus, there is no further execution semantics internal to tasks. Trivially, this can be achieved by sufficiently refining tasks during modeling.

BP modeling languages are mostly oriented at BP execution semantics required for computerized workflow implementation. In terms of content, this requires modeling possible task sequences. However, it does not require modeling the full impact on target BSDs (e.g., through human activities) or all preconditions to enactment (e.g., the availability of labor). This observation is reflected in the discussion of the state of the art in BP modeling (cf. Table 7.1). Thus, BPMN or similar approaches need to be extended towards a quality-aware BP meta-model. Table 7.2 summarizes the additional terminology required. The meta-model presented in Figure 7.5 shows how the necessary terms are interrelated, and how they integrate with BPMN concepts.

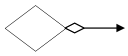
| **BPMN Terms** | **Efficacy-aware Meta-model Terms** | **Semantic Adaptations** |
|---|---|---|
| Data objects  | Environmental elements: affecting and affected elements, target and conditional elements | Environmental elements replace data objects. From the perspective of the business process, they comprise the overlapping sub-sets of affecting elements (e.g., data fields altered) and affected elements (e.g., resources spent). From the perspective of the business objective, they comprise target elements and conditional elements (cf. Chapter 6). Both perspectives overlap. For example, a target element can be an affected element and an affecting element. |
| Conditions attached to split gateways  | Branches and branch-conditional BSDs | A sequence flow following a conditional split gateway is referred to as a branch. Branch-conditional BSDs take up the concept presented in Chapter 6. They are used to describe split gateway conditions by relating affecting elements to absolute or relative conditions (e.g., "A = 5" or "A < B"). Thus, branch-conditional BSDs represent environmental conditions that co-determine which tasks are enacted. |
| *[none]* | Task-requisite BSDs | The BSD concept is also used to describe enactment preconditions attached to tasks. Semantically, it is assumed that a task that is enabled is enacted if and only if *all* task-requisite BSDs are fulfilled. Task-requisite BSDs may relate to resources that just need to be available (e.g., "information system available = true"), or to resources that are actually spent (e.g., "working time available > 1h"). Task-requisite BSDs modeling the availability of resources to be consumed by the task are linked to the respective state operation. |
| *[none]* | State operations | State operations related to tasks model effects on individual affected elements as functions (e.g., "A = A + B"). It is assumed that if a task is enacted, *all* related state operations are executed, and that state operations related to tasks are the only elements of BP models with an impact on affected elements. State operations modeling the consumption of resources are linked to a respective task-requisite BSD which models the required resource availability. |

Table 7.2: Required Terminology

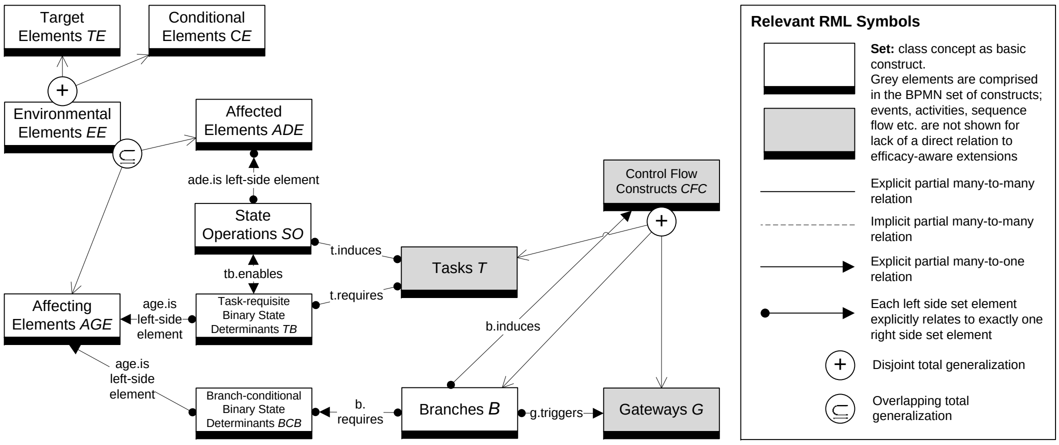



Figure 7.5: Quality-aware BP Meta-model

## 7.4 Building Quality Relations and Sample Validation

The presented approach towards a quality-aware BP meta-model extends BPMN with only a small set of elements. The *Transparency and retraceability* and *Cost effectiveness* criteria (cf. Section 3.1) may thus be assumed to be properly addressed. Accordingly, the validation of the approach can focus on the functional requirement, i.e. the ability to relate business objective achievement to resource availability and resource consumption conditions. This section describes the approach based on a running example. To enhance readability, the formalization of the concepts presented has been referred to Section 7.5.

Figure 7.3 shows what information must be available to allow describing quality relations. By means of a sample business objective (cf. Example 33 and an exemplary business process (cf. Figure 7.7), this section discusses how this information can be derived from a quality-aware BP model.[2] It thus validates the quality-aware BP meta-model in terms of the functional requirement (cf. Section 7.1).

**Example 33 (Business Objective: Year-end Receivables Processing).** Properly processing receivables constitutes a business objective during year-end closing in accounting. Figure 7.6 informally presents the respective objective model described in the following. The top four horizontal lines in the model correspond to the relevant conditional elements, and the bottom four lines correspond to target BSDs. Vertical lines and nodes are used to link conditional elements and target BSDs by way of conditional propositions. For reference, target BSDs and sub-conditions have been amended with numbers and literals, respectively. The individual target BSDs are modeled as follows:

- Target BSD *Clearing document posted* (1): If payment has been received for the receivable, it must be cleared. If not, a clearing document must not be posted. Accordingly,

[2]Note that business objective and BP models are separated following the considerations on this matter in Chapter 6 – the two types of models can be compared to requirements definition and implementation design, respectively.

*Clearing document posted* constitutes a fully determinate bivalent target BSD linked to *Payment received* (A) as a conditional BSD. Since there are no other conditions to be considered, there is no need to discern sufficient and necessary sub-conditions.

- Target BSDs *Impairment document (NOT) posted / Impairment document NOT posted*: An open receivable must be impaired (i.e. devalued) in certain cases, but it must not be impaired in others. Moreover, there may be circumstances where it does not matter whether the receivable is impaired. Accordingly, *Impairment document posted* constitutes a trivalent target BSD which is resolved into two partially determinate bivalent ones: *Impairment document posted* and *Impairment document NOT posted*.
  - Target BSD *Impairment document posted* (2): If there is no payment, but a business impairment requirement (B), or if there is no payment but a reporting impairment requirement (C), the impairment must be posted. Accordingly, the OR label associated with the target BSD indicates that there are two sufficient sub-conditions, each consisting of two conditional BSDs.
  - Target BSD *Impairment document NOT posted* (3): If no payment has been received (D), and there is neither a business nor a reporting impairment requirement (E and F), an impairment must not be posted. Thus, there are three necessary sub-conditions as indicated by the AND label going with the target BSD.
- Target BSD *Age class amended* (4): If an open receivable has not been cleared (D) and its amount is greater than the amount to be impaired (G), it must be amended with an age class for correct balance sheet reporting. If the receivable has been reduced to zero through clearing or impairment, we are indifferent whether an age class is amended. Therefore, *Age class amended* constitutes a partially determinate target BSD with two necessary sub-conditions.

Note that, for target BSDs with more than one conditional BSD, the notation allows showing either necessary or sufficient sub-conditions, depending on the modeler's choice, and that monovalent Target BSDs do not occur in this example.

The business objective described in Example 33 is addressed through a corresponding business process as presented in Example 34.

**Example 34 (Business Process: Year-end Receivables Processing).** Figure 7.7 depicts a sample process model which corresponds to the business objective model from Figure 7.6. Note that the process model has been amended with branch-conditional BSDs, task-requisite BSDs, and state operations (see Table 7.2 for explanations). Thus, it instantiates the efficacy-aware meta-model (cf. Figure 7.5). Relevant control flow elements (CFEs, i.e., tasks and branches, cf. Table 7.2) have been annotated with reference numbers 1-12. Business objective and business process relate to the management of receivables during year-end closing. Receivables are first matched against unallocated payments (1). If payment has been identified (3), the receivable is cleared (12). Otherwise (2), it is assessed in an impairment test based on management's appraisal (4) and formal criteria (5). If an impairment amount has been identified (7), the impairment is posted (8). If an open item remains (10),

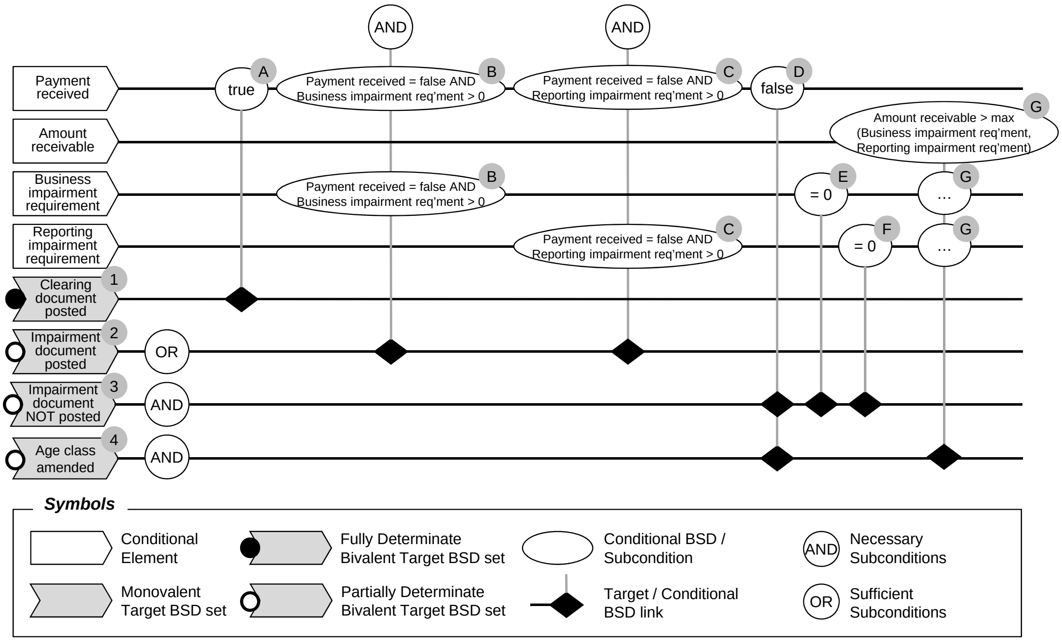



Figure 7.6: Exemplary Business Objective: Year-end Receivables Processing

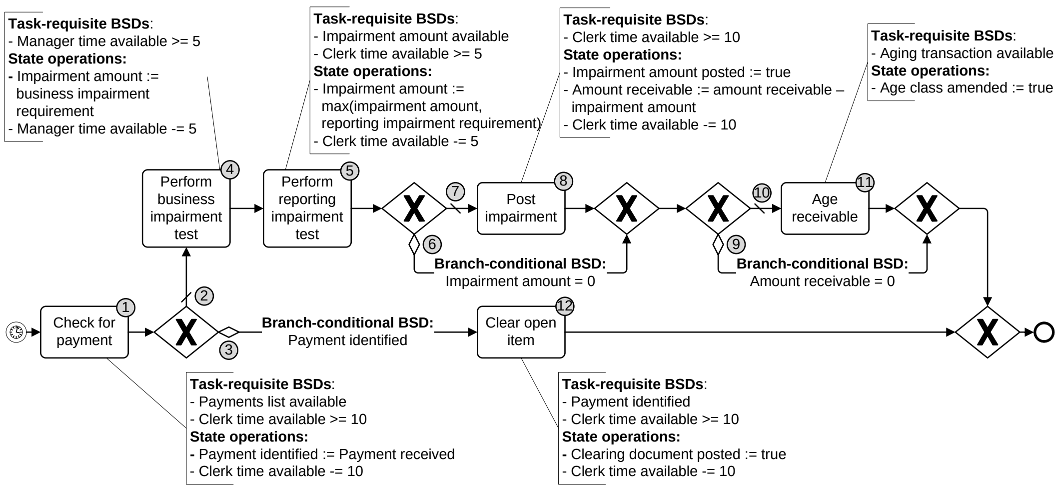



Figure 7.7: Exemplary Business Process: Year-end Receivables Processing

it is allocated to an age class (11). The latter task, for instance, can be enacted if it is enabled and the aging transaction is available (task-requisite BSD). If it is enacted, the age class is amended (state operation).

To enable determination of the quality relations for the BP model, information about the outer conditional environment of target BSDs as described in Figure 7.3 is required. This information is obtained by executing three steps presented in the following. The fourth step constitutes the evaluation of the efficacy and efficiency relations (cf. Figure 7.1). Note that the description of relevant steps is amended with formal definitions of required concepts.

### 7.4.1 Matching Target Binary State Determinants, State Operations, and Possible Enactment Paths

The first step towards building quality relations for a given process model determines *possible enactment paths* (PEPs) relevant to each target BSD.

**Step 1 (Matching Target BSDs, State Operations, and Possible Enactment Paths).** State operations describe the actions carried out on environmental elements when enacting tasks (cf. Table 7.2). In particular, they are required to fulfill target BSDs. Hence, target BSDs are matched to relevant state operations (cf. Definition 13) and PEPs to enact the state operations. Thus, it is assured that the first prerequisite towards formal efficacy is fulfilled, i.e. the business process comprehensively addresses all relevant target BSDs through corresponding state operations (cf. Definition 8). Note that in the context of BP compliance assessment, *actually enacted* control flow paths are also addressed as "traces" (e.g., [155]).

Building relevant PEPs necessitates traversing the process model. This is trivial for the simple example presented here, but may grow more complex in other cases. Respective algorithms are available (e.g., [224]). In turn, Section 7.5 presents corresponding formalisms.

To obtain *jointly fulfilled target BSD sets*, it is necessary to consider PEPs that address multiple target BSDs. Trivially, each target BSD addressed by a process is the parent of at least one set since the process might terminate after the respective task. Note that multiple jointly fulfilled target BSD sets comprising the same elements may occur. This happens if there are multiple PEP alternatives addressing the same set of target BSDs.

Example 35 illustrates the results of executing Step 1 for the sample process described in Example 34.

**Example 35 (Step 1).** Table 7.3 matches target BSDs, relevant state operations, and possible enactment paths to enact the state operations.

Note that, for the third target BSD (*Impairment document NOT posted*), the relevant state operation *must not* be executed to fulfill the target BSD. This issue generally occurs for fully determinate bivalent target BSDs and for de-composed trivalent target BSDs (cf. Chapter 6).

Table 7.4 records the jointly fulfilled target BSD sets resulting from relevant PEP alternatives.

| **Target BSD (cf. Fig. 7.6)** | **Relevant State Operation (Task No.)** | **PEP Alternatives** |
|---|---|---|
| Clearing document posted | Clearing document posted = true (12) | (1-3-12) |
| Impairment document posted | Impairment document posted = true (8) | (1-2-4-5-7-8) |
| Impairment document NOT posted | Impairment document posted = true (8) | NOT (1-2-4-5-7-8) |
| Age class amended | Age class amended = true (11) | (1-2-4-5-7-8-10-11) OR (1-2-4-5-6-10-11) |

Table 7.3: Target BSDs, State Operations, and Possible Enactment Paths

| **PEP Alternatives** | **Jointly Fulfilled Target BSD Sets** (excl. Parent BSDs) | **Parent BSDs** |
|---|---|---|
| (1-3-12) | n/a | Clearing document posted |
| (1-2-4-5-7-8) | n/a | Impairment document posted |
| NOT (1-2-4-5-7-8) | n/a | Impairment document NOT posted |
| (1-2-4-5-7-8-10-11) | Impairment document posted | Age class amended |
| (1-2-4-5-6-10-11) | Impairment document NOT posted | Age class amended |

Table 7.4: Possible Enactment Paths and Jointly Fulfilled Target BSD Sets

### 7.4.2 Consolidating Possible Enactment Paths

PEPs fail to facilitate appraising prerequisites and impacts (i.e., the outer conditional environment) of enacting them in a straightforward manner. For instance, interdependencies between CFEs addressing common environmental elements do not become transparent. The central step of deriving quality relations addresses this issue by merging of CFEs within relevant PEPs into one *virtual control flow element (VCFE)* for each relevant PEP. A VCFE summarizes the outer conditional environment of enacting a PEP through a single set of BSDs and a single set of state operations. Thus, a PEP assumes the structure of a CFE. As opposed to a PEP, this structure allows appraising the outer conditional environment "at a glance". Hence, it facilitates evaluating the efficacy and efficiency relations for a business process.

**Step 2 (Consolidating Possible Enactment Paths).** To determine the outer conditional environment required by the BP model to fulfill a target BSD, the BSDs comprised in relevant alternative PEPs have to be consolidated considering the respective state operations. In particular, whether environmental elements are affected by multiple state operations, or whether state operations affect environmental elements relevant to later CFEs must be taken into account. This can be achieved by properly *merging* subsequent CFEs of PEPs until the PEP has been consolidated into one set of environmental conditions expressed by conditional

BSDs. The required operations are sketched in this step, but formalized in Section 7.5. Two subsequent CFEs are merged as follows:

($a$) Apply the state operations of the first CFE to the BSDs of the second CFE. This is necessary to consider that state operations of the first CFE might affect BSDs of the second CFE.

($b$) Merge the resulting BSDs with the BSDs of the first CFE.

($c$) Merge the state operations of the first CFE with the state operations of the second CFE.

This results in new sets of BSDs and state operations, which jointly describe a new VCFE. To consolidate an entire PEP, the merge procedure is executed recursively. For more complex processes, it makes sense to structure the consolidation of PEPs along sub-processes that occur multiple times. This way, VCFEs can be re-used. In general, this is possible for jointly fulfilled target BSD sets comprising more than the parent BSD.

Note that the issue of parallel execution paths can be resolved by using block-structured process models [25] and recursively consolidating parallel paths into activities. As an additional consistency condition, this requires that the affected elements of neither parallel path are affecting elements of the other one (cf. Table 7.2). This topic is further discussed in Section 7.5.

Example 36 illustrates the results of executing Step 2.

**Example 36 (Step 2).** (cf. Table 7.3)

Figure 7.8 presents the results of recursively following through the VCFE merge procedure for the first relevant control flow path of the *Age class amended* target BSD. The top line depicts the relevant PEP alternative extracted from the process model (cf. Figure 7.7). In the second line, the merge operation has been executed for the first two CFEs. In this case, the respective sets of BSDs do not address common environmental elements. Accordingly, the branch-conditional BSD of (2) has simply been added to the merged set of BSDs of the new VCFE. The third line shows the results of following through the merge procedure for the entire PEP alternative, so that only one VCFE remains. This VCFE bundles all BSDs and state operations comprised in the process model as though they would be enacted in a single task.

### 7.4.3 Building the Outer Conditional Environment

The sets of VCFEs relevant to each target BSD immediately allow deducting the respective outer conditional environment.

**Step 3 (Building the Outer Conditional Environment).** To obtain outer conditional BSDs (cf. Figure 7.3) from the set of BSDs of a VCFE, the BSDs have to be amended with consumption amounts as appropriate.

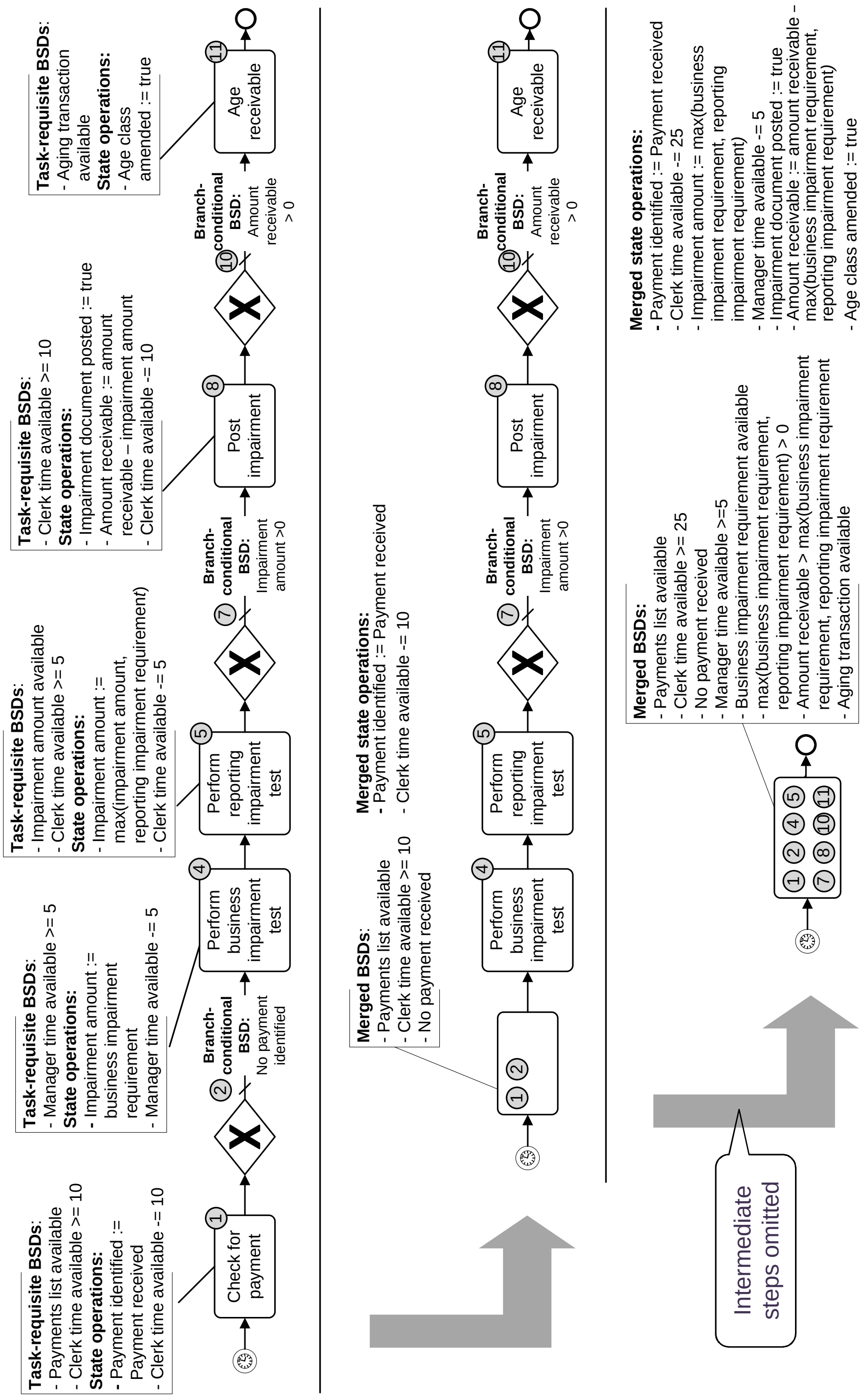



Figure 7.8: Possible Enactment Path Consolidation for *Age class amended, first control flow path alternative* (cf. Fig. 7.7)

To facilitate this operation, state operations modeling resource consumption correspond to task-requisite BSDs modeling sufficient availability of the resource in question (cf. Table 7.2). The consolidation approach lined out above ensures that both state operations and task-requisite BSDs are incremented appropriately if the resource in question is required to enact multiple tasks. It is therefore sufficient to amend outer conditional BSDs with consumption amounts once recursive consolidation of control flow elements has been completed. Thus, task-requisite BSDs can be compared to state operations: if a task-requisite BSD has only one affecting element (cf. Section 7.5), the affecting element corresponds to the affected element of a state operation, and the state operation has its affected element as its sole affecting element, the *consumption amount* is derived from the state operation and amended to the original BSD (cf. Section 7.5).

Necessary and sufficient outer sub-conditions for a target BSD as defined in Figure 7.3 can now be derived according to a simple schema:

- Each set of merged BSDs of a consolidated control flow path constitutes a sufficient outer sub-condition.
- Any merged BSD that occurs in each control flow alternative constitutes a necessary outer sub-condition. Note that, for the purpose of necessary outer sub-conditions, BSDs where the same set of affecting elements is covered in each control flow alternative are represented by their most “relaxed” form (in the case presented, this applies to the available clerk time).
- If the state operation fulfilling the target BSD has affecting elements, an additional necessary outer sub-condition is derived from the impact relation describing the content of the operation (cf. Definition 11 in the appendix): the target presumption as included in Figure 7.3. To this end, the function’s affected element in the target BSD is substituted by the impact relation. The resulting term yields the target presumption. It represents environmental conditions not caused by control flow, but by the final state operation itself.

The resulting sub-conditions are then compared to the respective necessary sub-conditions of the target BSD as per the objective model.

Example 37 describes the results of Step 3 for the sample process.

**Example 37 (Step 3).** Table 7.5 presents the results of deriving consumption amounts for the first control flow path alternative to fulfill the *Age class amended* target BSD.

Also for the *Age class amended* target BSD, Table 7.6 shows the comparison of the sub-conditions derived according to Step 3 to the necessary sub-conditions given by the business objective model (cf. Figure 7.6).

Finally, the outer conditional environment can be used to appraise efficacy and efficiency of the business process.

**Step 4 (Quality Relations: Assessing Efficacy and Efficiency).** Corresponding to the respective quality relations (cf. Figure 7.1), efficacy and efficiency can now be inspected by comparing the outer conditional environment of the process model to the conditional

| **Merged BSDs** | **Consumption Amounts** |
|---|---|
| Payment list available | n/a |
| Clerk time available $\geq 25$ | Clerk time available $\geq 25$ |
| No payment received | n/a |
| Manager time available $\geq 5$ | Manager time available $\geq 5$ |
| Business impairment requirement available | n/a |
| max(business impairment requirement, reporting impairment requirement) $> 0$ | n/a |
| Amount receivable $>$ max(business impairment requirement, reporting impairment requirement | n/a (note that there is no consumption amount for this BSD since it has multiple affecting elements) |
| Aging transaction available | n/a |

Table 7.5: Outer Conditional BSDs with Consumption Amounts for *Age class amended, first control flow path alternative*

| **Binary State Determinants** | **Outer Sub-conditions** | | | **Objective Model: Necessary Sub-conditions** |
|---|---|---|---|---|
| | **Sufficient: Path 1** | **Sufficient: Path 2** | **Necessary** | |
| Payments list available | X | X | X | |
| Clerk time available $\geq 25$ | | X | | |
| Clerk time available $\geq 15$ | X | X | X | |
| Payment received = false | X | X | X | X |
| Manager time available $\geq 5$ | X | X | X | |
| Business impairment req'ment available | X | X | X | |
| max(impairment requirements) $> 0$ | X | | | |
| max(impairment requirements) $= 0$ | | X | | |
| Amount receivable $>$ max(imp. req's) | X | X | X | |
| Aging transaction available | X | X | X | |

Table 7.6: Target BSDs and the Conditional Environment for *Age class amended*

environment of the business objective. According to Definition 8, this enables drawing conclusions as follows:

- Target BSDs included in the business objective but not addressed by state operations signify that the business process is *not formally efficacious*, because the business process alone is not sufficient to fulfill all target BSDs.
- Necessary sub-conditions of the business objective not covered by the process indicate that the business process is *not formally efficacious*, because it may induce target BSDs without considering relevant constraints.
- Necessary outer sub-conditions of the process model with regard to a Target BSD which do not correspond to necessary sub-conditions of the business objective indicate the resources required by the business process to fulfill a target BSD. It needs to be judged whether these are considered as reasonable – the process may be *not fully efficacious* even if it is *formally efficacious*.

With regard to efficiency, the consumption amounts associated with jointly fulfilled target BSD sets must be considered. Moreover, besides consumption amounts given by the process model for individual process instances, resources may be consumed during the *design & implementation* lifecycle phase. This is reflected by “availability” conditional BSDs that refer to capital goods, such as the availability of information systems. To fully appraise efficiency, these have to be considered as well.

As described above, each jointly fulfilled target BSD set corresponds to one control flow path alternative through the process model. Since there is no “formal efficiency” comparable to formal efficacy (cf. Definitions 9 and 8), necessary outer sub-conditions do not have to be compared to the business objective for efficiency assessment. Moreover, as observed in Section 7.3, ideally efficient business processes do not exist in practice. The assessment of full efficiency, however, still requires the judgment of subject matter experts. In this respect, the efficiency relation assumes a supporting role only.

Example 38 illustrates the results of executing Step 4.

**Example 38 (Step 4).** Table 7.6 documents the *efficacy relation* (cf. Figure 7.1) for *Age class amended.*

For the sample process, it can be concluded that the process is formally efficacious since all target BSDs are considered while covering all respective necessary sub-conditions of the business objective. Whether it is fully efficacious will mainly depend on whether the associated requirements regarding available labor resources are deemed as reasonable by subject matter experts.

Table 7.7 lists the associations between jointly fulfilled target BSD sets and consumption amounts that are relevant to the *Age class amended* target BSD. It thus documents the *efficiency relation* (cf. Figure 7.1) for *Age class amended.*

| Jointly Fulfilled Target BSD Sets | Consumption Amounts |
|---|---|
| Age class amended<br>Impairment document posted | Clerk time $\geq 25$<br>Manager time $\geq 5$ |
| Age class amended<br>Impairment document NOT posted | Clerk time $\geq 15$<br>Manager time $\geq 5$ |

Table 7.7: Jointly fulfilled target BSD sets and consumption amounts for *Age class amended*

## 7.5 Technical Implementation Aspects

Corresponding to the method to build quality relations (cf. Figure 7.1), which was discussed along a sample process in Section 7.4, this section presents more concise and formal definitions of the concepts required. This constitutes an important prerequisite to enable the implementation of quality-aware BP modeling tools and corresponding BPM systems (cf. Chapter 9), and reflects experience gained from the implementation of a respective prototype. Readers primarily interested in the conceptual discussion of quality-aware process modeling and its implications may skip this section.

### 7.5.1 Binary State Determinants and State Operations

Since CFEs are described through BSDs and state operations, formal representations of both are required to enable consolidating control flow paths (cf. Section 7.4).

Let $E$ be the set of environmental elements of a BP model. Further, let $e \in E$ be an individual environmental element with domain $\mathcal{D}(e)$ as the range of its possible values. On that basis, BSDs have been formally defined in Section 6.4 (cf. Definition 3). The following considerations apply to the definition of state operations: State operations model the impact of enacting tasks on elements of the affected environment. In some cases, this impact can be fully defined at design time, i.e., the impact of the state operation is fully deterministic. In other cases, this is not reasonable or even impossible, e.g., in the case of human decisions which are, to some degree, arbitrary. Thus, as illustrated in Example 39, there are *deterministic* and *non-deterministic state operations.*

**Example 39 (Deterministic and Non-deterministic State Operations).** Consider Sample Processes A and B from Figures 2.5 and 2.6, the management of incoming invoices and subsequent payment runs. When an incoming invoice is entered into an ERP system, the latter can automatically determine whether the invoice corresponds to a goods receipt. The result of this assessment can be documented through a deterministic state operation, since the impact of the state operation can be defined in the process model.

In turn, final approval of a payment run may be subject to senior management decision. In this context it may not be practical to include all factors to this decision in the process model. Hence, the state operation documenting this decision will be non-deterministic.

**Definition 11 (State Operation).** *A state operation* $\delta = \langle \theta_\delta, E_\delta, i_\delta \rangle$ *is defined by an affected element* $\theta_\delta \in E$*, a sequence of affecting elements* $E_\delta = (e_{\delta_1}, \dots, e_{\delta_n})$*,* $e_{\delta_i} \in E$*, which may be empty, and an impact relation*

$$i_\delta \subseteq \prod_{e_i \in E_\delta} \mathcal{D}(e_i) \times \mathcal{D}(\theta_\delta)$$

*with* $\mathcal{D}(\theta_\delta)$ *being its co-domain. The impact relation is fully defined, i.e.,*

$$\forall d \in \prod_{e_i \in E_\delta} \mathcal{D}(e_i) : \exists i \in \mathcal{D}(\theta_\delta) \text{ with } \langle d, i \rangle \in i_\delta$$

*A state operation is* reflective *iff its affected element is comprised in its set of affecting elements. If the sequence of affecting elements is not empty, the impact relation is uniquely defined, i.e., it constitutes a function such that*

$$\forall d \in \prod_{e_i \in E_\delta} \mathcal{D}(e_i), i \in \mathcal{D}(\theta_\delta), i' \in \mathcal{D}(\theta_\delta) : \langle d, i \rangle \in i_\delta \wedge \langle d, i' \rangle \in i_\delta \Rightarrow i = i'$$

*In this case, the state operation is* deterministic*. The same applies if the co-domain of the impact relation contains only one element, such that* $|i_\delta| = 1$*. In other cases, i.e., if the state operation comprises no affecting elements and there is more than one element in the co-domain of the impact relation, the state operation is* non-deterministic*.* □

The impact relation describes how the new state of the affected element is derived from the affecting elements of the state operation. Note that for non-deterministic state operations, it is not reasonable to include affecting elements in the process model, which is reflected in Definition 11. In addition, note that reflective state operations often occur in conjunction with consumable resources being decremented during process enactment.

Moreover, state operations enacted within the same task may affect neither one common environmental element nor each other (i.e., no affected element of a state operation may be an affecting element of another one) since their sequence is not defined. Therefore, the following stipulation applies:

**Definition 12 (Operational Consistency of Tasks).** *Let* $\Delta_t$ *be the set of state operations associated with a task* $t$*. The task is operationally consistent iff:*

$$\nexists \langle \delta_1, \delta_2 \rangle \in \Delta_t^2 : \delta_1 \neq \delta_2 \wedge (\theta_{\delta_1} = \theta_{\delta_2} \vee \theta_{\delta_1} \in E_{\delta_2})$$

□

A task potentially fulfills or *addresses* a target BSD iff the following conditions apply to one of its state operations:

*(i)* The affected element of the state operation is comprised in the sequence of affecting elements of the target BSD.

*(ii)* Considering the co-domain of the impact relation of the state operation, it is possible that the state operation induces a state of its affecting element for which the target BSD is fulfilled.

Formally, these conditions can be defined as follows:

**Definition 13 (State Operations Addressing Binary State Determinants).** *Let $\delta = \langle \theta_\delta, E_\delta, i_\delta \rangle$ be a state operation, let $\gamma = \langle E_\gamma, \Lambda_\gamma \rangle$ be a BSD, and let $e_{\gamma_i}$ be the i-th element of the sequence of affecting elements of the BSD.*

$$cond_1 := \exists\, e_{\gamma_i} \in E_\gamma \text{ with } e_{\gamma_i} = \theta_\delta \qquad (i)$$
$$cond_2 := \big(\mathcal{D}(e_{\gamma_1}) \times \ldots \times \mathcal{D}(e_{\gamma_{i-1}}) \times i_\delta \times \mathcal{D}(e_{\gamma_{i+1}}) \times \ldots \times \mathcal{D}(e_{\gamma_n})\big) \cap \Lambda_\gamma \neq \emptyset \qquad (ii)$$

$$addresses(\delta, \gamma) := \begin{cases} true & \text{if } cond_1 \wedge cond_2 \\ false & else \end{cases}$$

□

As described above, task-requisite BSDs (cf. Table 7.2) modeling the availability of resources to be consumed by a task via a state operation have exactly one affecting element: the affecting element of the state operation.

As discussed in Section 7.3, tasks and branches constitute the types of CFEs which are relevant to the fulfillment of target BSDs, or to resource availability or consumption requirements. CFEs can be defined as follows:

**Definition 14 (Control Flow Element).** *Let $\Phi_p = T_p \cup B_p$ be the set of CFEs comprised in a BP model p, and let $T_p$ and $B_p$ be the sets of tasks and branches comprised in p.*

*A CFE $\phi \in \Phi_p$ is then defined as follows:*

$$\phi = \langle \Gamma_\phi, \Delta_\phi \rangle$$

*with $\Gamma_\phi \cup \Delta_\phi \neq \emptyset$. That is, a CFE comprises a set of BSDs and a set of state operations, and at least one of these sets must not be empty. If the set of state operations of a CFE is empty, the CFE is a branch, otherwise, it is a task:*

$$\Delta_\phi = \emptyset \Rightarrow \phi \in B_p,\ \phi \in B_p \Leftrightarrow \phi \notin T_p$$

. □

Based on Definitions 13 and 14, it is possible to determine whether a control flow element addresses a (target) BSD or another control flow element. For now, this is relevant for tasks only, but the method of conditional consolidation described below will yield additional *virtual control flow elements* comprising state operations as well:

**Definition 15 (Control Flow Elements Addressing Binary State Determinants).** *Let $\phi = \langle \Gamma_\phi, \Delta_\phi \rangle$ be a control flow element, and let $\gamma = \langle E_\gamma, \Lambda_\gamma \rangle$ be a BSD. The BSD $\gamma$ is addressed by the control flow element $\phi$ iff it is addressed by at least one of the state operations of the control flow element:*

$$addresses(\phi, \gamma) := \begin{cases} true & \text{if } \exists \delta \in \Delta_\phi : addresses(\delta, \gamma) = true \\ false & else \end{cases}$$

*On that basis, a control flow element* $\phi$ *addresses a second control flow element* $\phi' = \langle \Gamma_{\phi'}, \Delta_{\phi'} \rangle$ *iff it addresses at least one of its BSDs:*

$$addresses(\phi, \phi') := \begin{cases} true & if\ \exists \gamma \in \Gamma_{\phi'} : addresses(\phi, \gamma) = true \\ false & else \end{cases}$$

*A control flow element* $\phi$ *is* reflective *iff it addresses itself (i.e., at least one of its BSDs) such that*

$$addresses(\phi, \phi) = true$$

□

### 7.5.2 Conditional Consolidation

Based on the definitions from Section 7.5.1, it is possible to define a summary merge operation for relevant control flow elements, i.e. tasks and branches (cf. Table 7.2). This merge operation is based on operations to:

- Apply state operations to BSDs
- Merge sets of BSDs
- Merge sets of state operations

These operations reflect the procedure lined out in Step 2 in Section 7.4: to merge two subsequent CFEs, the state operations of the first CFE are applied to the BSDs of the second one. Then, both sets of BSDs and both sets of state operations are merged. The resulting tuple consisting of a set of BSDs and a set of state operations describes the resulting VCFE.

When a state operation is applied to a BSD, the BSD remains unchanged iff the affected element of the state operation is not an affecting element of the BSD. Otherwise, the affecting elements of the state operation replace its affected element in the sequence of affecting elements of the BSD. To fulfill the altered BSD, its affecting elements must be in a state where the result of the impact relation of the state operation together with the state of the affecting elements that have not been replaced fulfill the "old" BSD. Note that this operation can lead to multiple occurrences of particular environmental elements in the sequence of affecting elements of a BSD. In this case, note that value tuples of a BSD $\gamma$ where contradictory values are required for the same affecting element cannot be fulfilled. Accordingly, the respective value tuples may be eliminated from $\Lambda_\gamma$.

A set of state operations (of a preceding task) is then applied to a BSD by subsequently applying each individual state operation. The order in which state operations are applied is of no concern since the operational consistency of tasks (cf. Definition 12) requires state operations within one task not to affect one another or have the same affected element.

**Definition 16 (Applying State Operations to BSDs).** *A state operation $\delta = \langle \theta_\delta, E_\delta, i_\delta \rangle$ is applied to a BSD $\gamma = \langle E_\gamma, \Lambda_\gamma \rangle$ as follows:*

$$\delta \triangleright \gamma \equiv \begin{cases} \langle (e_{\gamma_1}, \dots, e_{\gamma_{k-1}}, E_\delta, e_{\gamma_{k+1}}, \dots, e_{\gamma_n}), \Lambda_{\gamma'} \rangle & \textit{if } \theta_\delta = e_{\gamma_k} \\ \gamma & \textit{else} \end{cases}$$
$$\textit{with } \Lambda_{\gamma'} = \{\lambda \,|\lambda = \langle x_1, \dots, x_{k-1}, y_1, \dots, y_l, x_{k+1}, \dots, x_n \rangle \textit{ with } \langle x_1, \dots, x_{k-1}, i_\delta(y_1, \dots, y_l), x_{k+1}, \dots, x_n \rangle \in \Lambda_\gamma\}$$

*Further, a set of state operations $\{\delta_1, \dots, \delta_n\}$ is applied to a BSD $\gamma$ as follows:*

$$\{\delta_1, \dots, \delta_n\} \triangleright \gamma \equiv \delta_n \triangleright \Big( \dots \big(\delta_2 \triangleright (\delta_1 \triangleright \gamma)\big)\Big)$$

*A set of state operations $\{\delta_1, \dots, \delta_n\}$ is then applied to a set of BSDs $\Gamma$ as follows:*

$$\{\delta_1, \dots, \delta_n\} \triangleright \Gamma \equiv \bigcup_{\gamma_i \in \Gamma} \{\delta_1, \dots, \delta_n\} \triangleright \gamma_i$$

□

Two BSDs can be merged if their sets of affecting elements are equal. This operation is commutative.

**Definition 17 (Merging BSDs).** *Two BSDs $\gamma_1$ and $\gamma_2$ are merged as follows:*

$$\gamma_1 \bullet \gamma_2 \equiv \begin{cases} \{\langle E_{\gamma_1}, \Lambda_{\gamma_1} \cap \Lambda_{\gamma_2} \rangle\} & \textit{if } E_{\gamma_1} = E_{\gamma_2} \\ \{\gamma_1, \gamma_2\} & \textit{else} \end{cases}$$

*Two sets of BSDs $\Gamma_1$ and $\Gamma_2$ are then merged as follows:*

$$\Gamma_1 \bullet \Gamma_2 \equiv \bigcup_{\langle \gamma_{1_i}, \gamma_{2_i} \rangle \in \{\Gamma_1 \times \Gamma_2\}} (\gamma_{1_i} \bullet \gamma_{2_i})$$

□

To merge two state operations, one needs to consider whether they have the same affected element, and whether the affected element of the first state operation is part of the affecting elements of the second one:

- If the first characteristic applies, the state operations are merged.
- If the second characteristic applies, the affecting elements of the first state operation substitute its affected element in the set of affecting elements of the second operation, and the impact relation of the second state operation is replaced by a composition of both impact relations.

Note that both characteristics might apply at the same time. If neither characteristic applies, both state operations remain unchanged. In practical implementations, however, it may be sensible *not* to actually execute the merge operation for BSDs: As discussed with the definition of BSDs, BSDs can mostly be expressed as simple equations and inequalities (cf. Definition 3). Maintaining this simple representation instead of adopting a more complex merged one will be more understandable for the human user.

**Definition 18 (Merging State Operations).** *Two state operations $\delta_1$ and $\delta_2$ with $\delta_1$ preceding $\delta_2$ are merged as follows:*

$$\delta_1 \dagger \delta_2 \equiv \begin{cases} \{\delta_2\} & \text{if } \theta_{\delta_2} = \theta_{\delta_1} \land \theta_{\delta_1} \notin E_{\delta_2} \\ \{\delta_1, \delta_2\} & \text{if } \theta_{\delta_2} \neq \theta_{\delta_1} \land \theta_{\delta_1} \notin E_{\delta_2} \\ \{\langle \theta_{\delta_1}, E', i_{\delta_2} \circ i_{\delta_1} \rangle\} & \text{if } \theta_{\delta_2} = \theta_{\delta_1} \land \theta_{\delta_1} \in E_{\delta_2} \\ \{\delta_1, \langle \theta_{\delta_2}, E', i_{\delta_2} \circ i_{\delta_1} \rangle\} & \text{if } \theta_{\delta_2} \neq \theta_{\delta_1} \land \theta_{\delta_1} \in E_{\delta_2} \end{cases}$$
$$\text{with } E' = (e_{\delta_{2_1}}, \ldots, e_{\delta_{2_{k-1}}}, E_{\delta_1}, e_{\delta_{2_{k+1}}}, \ldots, e_{\delta_{2_n}}) \text{ where } \theta_{\delta_1} = e_{\delta_{2_k}}$$

*Two sets of state operations $\Delta_1$ and $\Delta_2$ with $\Delta_1$ preceding $\Delta_2$ are then merged as follows:*

$$\Delta_1 \dagger \Delta_2 \equiv \bigcup_{\langle \delta_i, \delta_j \rangle \in \{\Delta_1 \times \Delta_2\}} (\delta_i \dagger \delta_j)$$

□

Definitions 16-18 provide operations to apply state operations to BSDs, merge sets of BSDs and merge sets of state operations. On that basis, it becomes possible to define an operation to merge control flow elements:

**Definition 19 (Merging Control Flow Elements).** *Let $\phi_1$ and $\phi_2$ be two subsequent control flow elements, i.e., tasks or branches, where $\phi_2$ is the direct successor of $\phi_1$, as part of a possible enactment path enabled by a process model. Let $\Gamma_\phi$ and $\Delta_\phi$ be the sets of BSDs and state operations associated with a control flow element $\phi = \langle \Gamma_\phi, \Delta_\phi \rangle$. For branches, the set of state operations is empty. The control flow elements $\phi_1$ and $\phi_2$ are then merged as follows:*

$$\phi_1 \Diamond \phi_2 \equiv \langle \Gamma_{\phi_1} \bullet (\Delta_{\phi_1} \triangleright \Gamma_{\phi_2}), \Delta_{\phi_1} \dagger \Delta_{\phi_2} \rangle$$

□

To fully describe the outer conditional environment, consumption amounts are required as well:

**Definition 20 (Consumption Amount).** *A consumption amount $\omega = \langle \gamma_\omega, \theta_\omega, i_\omega \rangle$ is attached to a BSD $\gamma_\omega$, refers to one affected element $\theta_\omega$, and models resource consumption through its impact relation $i_\omega$. The BSD may have exactly one affecting element which is equal to the affected element of the consumption amount, such that $E_{\gamma_\omega} = \{\theta_\omega\}$. The impact relation must be a homogeneous relation on the domain of the affected element, such that $i_\omega : \mathcal{D}(\theta_\omega) \rightarrow \mathcal{D}(\theta_\omega)$.* □

Finally, consumption amounts are deducted from a task or virtual control flow element (branches do not consume resources) as follows:

**Definition 21 (Deducting Consumption Amounts).** *For a task or a virtual control flow element $t$, let $\Gamma_t$ be its set of BSDs, and let $\Delta_t$ be its set of state operations.*

*The corresponding set of consumption amounts $\Omega_t$ is then deducted as follows:*

$$\begin{aligned}\Omega_t = \{\langle \gamma, \theta_\delta, i_\delta \rangle \mid \exists \langle \gamma, \delta \rangle \in \Gamma_t \times \Delta_t : & & \textit{(i)}\\ & E_\gamma = \{\theta_\delta\} & \textit{(ii)}\\ & \wedge & \\ & E_\delta = \{\theta_\delta\}\} & \textit{(iii)}\end{aligned}$$

*The following considerations apply to the lines of the equation above:*
*(i), each consumption amount reflects a BSD and a state operation belonging to the same task,*
*(ii), the affected element of the state operation constitutes the one and only element of its set of affecting elements, and*
*(iii), the affected element of the state operation constitutes the one and only element of the set of affecting elements of the BSD.* □

### 7.5.3 Possible Enactment Paths and Virtual Control Flow Elements

The first step of deducting quality relations consists of determining the PEPs through a quality-aware BP model that enable fulfilling at least one target BSD (cf. Step 1 in Section 7.4). A PEP is a sequence of tasks and branches – not necessarily in the sense of the "sequence" workflow pattern [223] – which implies sequential semantics not in the sense of *requiring* one control flow element to be enacted after another, but in the sense of one CFE being enacted later in time than another, regardless of whether the order of individual CFEs might be exchanged without impacting the end result of the business process or not.

**Definition 22 (Possible Enactment Path).** *Let $T_p$ and $B_p$ be the sets of tasks and branches comprised in a BP model p, respectively.*

*A possible enactment path $pep = (\phi_{pep_1}, \dots, \phi_{pep_n})$, $\phi_{pep_i} \in \{T_p \cup B_p\}$ in p consists of a sequence of CFEs comprising tasks and branches which may not be empty. The set of possible enactment paths $PEP_p$ for a process model p is a sub-set of all possible sequences of CFEs.*

*A possible enactment path for a process model must reflect the underlying execution semantics of the BP model, i.e. the process model constrains the set of corresponding PEPs. In other words, the process model can be viewed as a formal language on the alphabet of CFEs, with PEPs as well-formed words. Considering loops in the process model, individual CFEs may occur multiple times so that the set of PEPs of a process model with a finite set of CFEs is, in principle, countably infinite.*

*To determine whether a sequence of CFEs is a PEP of a process model, a corresponding function can be deducted from the execution semantics of the process model:*

$$enactable_p : \{(\phi_{pep_1}, \dots, \phi_{pep_n}) \mid \phi_{pep_i} \in \{T_p \cup B_p\}\} \rightarrow \{true, false\}$$

*Trivially, the process model provides an algorithm to decide whether a given PEP constitutes a well-formed word, by simply tracing the PEP through the model.* □

Accordingly, a BP model (in the sense of a control flow model) constitutes one possible representation of the set of PEPs through the corresponding process. However, this representation is not yet sufficient to appropriately enable deducting quality relations:

- It should be possible to immediately determine (i.e., "look up" without further analysis effort) PEPs which might fulfill a target BSD. This is not possible in common control flow models: while tasks with state operations addressing target BSDs can be readily determined, rather complex backward analysis (i.e., to determine how the tasks in question can be enabled) would still be necessary to derive the set of corresponding PEPs on that basis.
- Control flow models may comprise loops. In this case, the number of PEPs to reach an individual task, and thus the number of PEPs relevant to fulfilling a target BSD, is potentially unlimited.
- Semantic interdependencies between PEPs should be considered to reduce the effort involved in subsequent conditional consolidation. In particular, this pertains to PEPs extending each other: amending a PEP with one additional CFE results in a new PEP. Obviously, it makes sense to utilize these semantic interdependencies to avoid starting conditional consolidation all over for the new PEP. To this end, conditional consolidation results for PEPs should be held persistent in an appropriate structure.

To address these requirements, control flow models can be transformed into sets of *virtual control flow elements* (VCFEs), with each VCFE representing one PEP in the "flat" structure of a CFE, i.e. as a set of BSDs and a set of state operations (cf. Definition 14).

To analyze quality relations between target BSDs to be fulfilled and the corresponding resource requirements, it is not necessary to consider *all* possible VCFEs. Rather, *conditionally equivalent sets of VCFEs*, i.e., sets of VCFEs with equal BSDs and state operations, need only be represented by one VCFE. Moreover, only those VCFEs which address at least one target BSD through a corresponding task are relevant. In addition, the part of a PEP that occurs *after* the last task in the PEP addressing a target BSD can be omitted, since, from a business objective perspective, the process instance might terminate as soon as this task has been enacted – every additional task would just induce unnecessary resource requirements.[3]

To obtain the set of relevant VCFEs for a process model, the underlying control flow semantics have to be taken into account. For the purposes of this thesis, the "basic control flow patterns" of *sequence*, *exclusive choice*, *simple merge*, *parallel split*, and *synchronization* are considered [223, 14]. This set of *fundamental workflow patterns* is sufficient for many application scenarios, and can be augmented with other patterns following the approach detailed below. Due to the universal appeal of the fundamental workflow patterns used, the approach can be adapted to common declarative process modeling paradigms (cf., e.g., Table 7.1). In the following, BPMN terminology [80] is used to maintain consistency with the process examples used in this thesis.[4]

In addition, it is assumed that control flow models fulfill the requirements of *block structuring* [79, 25]. Block structuring implies that the workflow patterns mentioned above are modeled in nested (i.e., non-overlapping) and complete blocks (e.g., for each parallel split gateway, there is a parallel join gateway etc.). While there are BP models in practice which are not block-structured, block structuring remains a valid assumption for most scenarios [29, 225]. Based on earlier work [226] as well as ideas from the field of compiler construction [227], the "refined process structure tree" approach allows obtaining the underlying block structure of

[3] This aspect also needs to be reflected in the set of quality attributes relevant to BP models (cf. Quality Attribute 17 in Section 8.3).

[4] For example, instead of the *synchronization* workflow pattern, *parallel join gateways* are referred to.

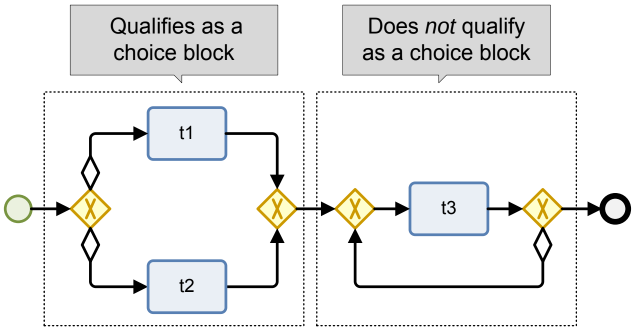



Figure 7.9: Arbitrary Cycles with Exclusive Choice and Simple Merge vs. Block Structuring

process models in linear time [228] by determining the underlying structure of "single entry, single exit" blocks. Empirical analysis of 214 process models resulted in 95% of the sample being either block-structured, or allowing transformation into a block-structured process model [67]. Due to the nesting of blocks, a block-structured process model can be viewed as a directed tree of blocks.

Beyond the "basic control flow patterns" listed above, however, the block structuring assumption requires including "structured cycles" in the set of fundamental workflow patterns covered in more detail. The reason for this is that block structuring consolidates the *exclusive choice* and *simple merge* workflow patterns into *choice blocks*, where it is assumed that the corresponding join gateway generally occurs *after* the respective split gateway. Thus, these workflow patterns cannot be used anymore to model "arbitrary cycles" [223], and another construct for loops in the sense of sets of CFEs potentially enacted multiple times is required. Figure 7.9 exemplifies this proposition in BPMN symbols [80].

Assuming that there is one root block for the process model, a block-structured process model can be defined as follows:

**Definition 23 (Block-structured BP Model).** *A block-structured process model $p = \langle BL_p, TB_p, rb_p \rangle$ is defined by a set of inner blocks*

$$BL_p = SB_p \mathbin{\dot\cup} CB_p \mathbin{\dot\cup} PB_p \mathbin{\dot\cup} LB_p \ ,$$

*a set of task blocks $TB_p$, and a root block $rb_p$. The set of inner blocks consists of non-overlapping sub-sets $SB_p$, $CB_p$, $PB_p$, and $LB_p$ which represent sequence blocks, choice blocks, parallel blocks, and loop blocks.* □

Definitions 24-28 describe block types required to cover the workflow patterns listed above, in particular with regard to relations to other blocks within the block-structured model. Sequence blocks cover the corresponding *sequence* workflow pattern:

**Definition 24 (Sequence Block).** *Let $p = \langle BL_p, TB_p, rb_p \rangle$ be a block-structured process model. A sequence block $sb$ as an inner block of $p$ is defined by a sequence of $n$ sub-blocks $(bl_{sb_1}, \dots, bl_{sb_n})$, and a parent block $pa_{sb}$ such that*

$$sb = \langle (bl_{sb_1}, \dots, bl_{sb_n}), pa_{sb} \rangle, \ sb \in SB_p \subseteq BL_p$$

*The following characteristics apply:*

$$\forall bl_{sb} \in (bl_{sb_1}, \ldots, bl_{sb_n}) : bl_{sb} \in BL_p \cup TB_p \quad \textit{(i)}$$
$$sb \notin \{bl_{sb_1}, \ldots, bl_{sb_n}\} \quad \textit{(ii)}$$
$$pa_{sb} \in BL_p \cup rb_p \quad \textit{(iii)}$$
$$pa_{sb} \notin sb \cup \{bl_{sb_1}, \ldots, bl_{sb_n}\} \quad \textit{(iv)}$$

*The sub-blocks are elements of the set of inner blocks and task blocks of the process (i), the sequence block must not be comprised in the set of its sub-blocks (ii), the parent block is an inner block or the root block of the process (iii), and the parent block must not be a sub-block of the sequence block or the sequence block itself (iv).* □

Choice blocks cover the corresponding *exclusive choice* and *simple merge* workflow patterns:

**Definition 25 (Choice Block).** *Let $p = \langle BL_p, TB_p, rb_p \rangle$ be a block-structured process model. A choice block cb as an inner block of p is defined by a set of n tuples consisting of a branch and a sub-block, respectively, $\{\langle b_{cb_1}, bl_{cb_1} \rangle, \ldots, \langle b_{cb_n}, bl_{cb_n} \rangle\}$, and a “parent” block $pa_{cb}$ such that*

$$cb = \langle \{\langle b_{cb_1}, bl_{cb_1} \rangle, \ldots, \langle b_{cb_n}, bl_{cb_n} \rangle\}, pa_{cb} \rangle,\ cb \in CB_p \subseteq BL_p$$

*The following characteristics apply:*

$$\{b_{cb_1}, \ldots, b_{cb_n}\} \subseteq B_p \quad \textit{(i)}$$
$$\nexists \langle b, b' \rangle \in \{b_{cb_1}, \ldots, b_{cb_n}\} \times \{b_{cb_1}, \ldots, b_{cb_n}\} : b = b' \quad \textit{(ii)}$$
$$\forall bl \in \{bl_{cb_1}, \ldots, bl_{cb_n}\} : bl \in BL_p \cup TB_p \cup \{\epsilon\} \quad \textit{(iii)}$$
$$cb \notin \{bl_{cb_1}, \ldots, bl_{cb_n}\} \quad \textit{(iv)}$$
$$pa_{cb} \in BL_p \cup rb_p \quad \textit{(v)}$$
$$pa_{cb} \notin cb \cup \{bl_{cb_1}, \ldots, bl_{cb_n}\} \quad \textit{(vi)}$$

*The set of branches is a sub-set of the set of branches of the process (i), no branch occurs twice (ii), the sub-blocks are either an element of the set of inner blocks and task blocks of the process or an empty element – the latter case applies if the respective branch results in the process commencing after the choice block without enacting other CFEs in between (“do nothing” branch) (iii), the choice block must not be comprised in its set of sub-blocks (iv), the parent block is an inner block or the root block of the process (v), and the parent block must not be a sub-block of the choice block or the choice block itself (vi).* □

Parallel blocks cover the corresponding *parallel split* and *synchronization* workflow patterns:

**Definition 26 (Parallel Block).** *Let $p = \langle BL_p, TB_p, rb_p \rangle$ be a block-structured process model. A parallel block pb as an inner block of p is defined by a set of sub-blocks $\{bl_{pb_1}, \ldots, bl_{pb_n}\}$, and a “parent” block $pa_{pb}$ such that*

$$pb = \langle \{bl_{pb_1}, \ldots, bl_{pb_n}\}, pa_{pb} \rangle,\ pb \in PB_p \subseteq BL_p$$

*The following characteristics apply:*

$$\begin{aligned} &\{bl_{pb_1}, \ldots, bl_{pb_n}\} \subseteq BL_p \cup TB_p, && (i) \\ &pb \notin \{bl_{pb_1}, \ldots, bl_{pb_n}\} && (ii) \\ &pa_{pb} \in BL_p \cup rb_p && (iii) \\ &pa_{pb} \notin pb \cup \{bl_{cb_1}, \ldots, bl_{cb_n}\} && (iv) \end{aligned}$$

*The sub-blocks are elements of the set of inner blocks and task blocks of the process (i), the parallel block must not be comprised in its set of sub-blocks (ii), the parent block is an inner block or the root block of the process (iii), and the parent block must not be a sub-block of the parallel block or the parallel block itself (iv).* □

Loop blocks cover the corresponding *structured cycle* workflow pattern:

**Definition 27 (Loop Block).** *Let* $p = \langle BL_p, TB_p, rb_p \rangle$ *be a block-structured process model. A loop block* $lb$ *as an inner block of* $p$ *is defined by a loop condition modeled as a branch* $b_{lb}$*, a loop body* $bl_{lb}$*, and a "parent" block* $pa_{lb}$ *such that*

$$lb = \langle b_{lb}, bl_{lb}, p_{lb} \rangle, \ lb \in LB_p \subseteq BL_p$$

*The following characteristics apply:*

$$\begin{aligned} &b_{lb} \in B_p, && (i) \\ &bl_{lb} \in BL_p \cup TB_p \setminus lb && (ii) \\ &pa_{lb} \in BL_p \cup rb_p && (iii) \\ &pa_{lb} \notin lb \cup bl_{lb} && (iv) \end{aligned}$$

*The loop condition is an element of the set of branches of the process (i), the loop body is an element of the set of inner blocks and task blocks of the process, but not the loop block itself (ii), the parent block may be an inner block or the root block of the process (iii), the parent block must not be a sub-block of the parallel block or the parallel block itself (iv).* □

**Definition 28 (Task Block).** *Let* $p = \langle BL_p, TB_p, rb_p \rangle$ *be a block-structured process model. A task block* $tb$ *as an inner block of* $p$ *is defined by a task* $t_{tb}$ *and a "parent" block* $pa_{lb}$ *so that:*

$$tb = \langle t_{tb}, pa_{tb} \rangle, \ tb \in TB_p$$

*The following characteristics apply:*

$$\begin{aligned} &t_{tb} \in T_p, && (i) \\ &pa_{tb} \in BL_p \cup rb_p && (ii) \\ &pa_{tb} \neq tb && (iii) \end{aligned}$$

*The task is an element of the set of tasks of the process (i), the parent block may be an element of the set of inner blocks and the root block of the process (ii), and the parent block must not be the task block itself.* □

In addition, the following consistency requirements apply with regard to tasks and branches:

**Definition 29 (Consistency of Tasks and Branches within Block-structured Process Models).** *Let $p = \langle BL_p, TB_p, rb_p \rangle$ be a block-structured process model. Let $T_p$ and $B_p$ be the sets of tasks and branches comprised in the process model p such that*

$$T_p = \bigcup_{tb \in TB_p} t_{tb}$$

*That is, the set of tasks comprised in the process model is equal to the merged set of tasks of its task blocks, and*

$$B_p = \left( \bigcup_{cb \in CB_p} b_{cb} \right) \cup \left( \bigcup_{lb \in LB_p} b_{lb} \right)$$

*That is, the set of branches comprised in the process model is equal to the merged set of branches of its choice and loop blocks.*

*The following consistency conditions apply to process models:*

- *A task occurs only in one task block of the process model, i.e.,*

$$\nexists \langle tb, tb' \rangle \in TB_p \times TB_p : tb \neq tb' \wedge t_{tb} = t_{tb'}$$

- *A branch occurs only in one choice block or loop block of the process model, i.e.,*

$$\nexists \langle cb, cb' \rangle \in CB_p \times CB_p : cb \neq cb' \wedge \{b_{cb_1} \dots b_{cb_n}\} \cap \{b_{cb'_1} \dots b_{cb'_m}\} \neq \emptyset$$

$$\nexists \langle lb, lb' \rangle \in LB_p \times LB_p : lb \neq lb' \wedge b_{lb} = b_{lb'}$$

$$\nexists \langle cb, lb \rangle \in CB_p \times LB_p : b_{lb} \in \{b_{cb_1} \dots b_{cb_n}\}$$

- *The root block occurs as the parent of one and only one inner block, i.e.,*

$$\exists bl \in BL_p : pa_{bl} = rb_p$$

$$\nexists \langle bl, bl' \rangle \in BL_p \times BL_p : pa_{bl} = pa_{bl'} = rb_p$$

- *Each inner block must be the parent block of at least one other block, i.e.,*

$$\forall bl \in BL_p : \exists bl' \in BL_p \cup TB_p \mid pa_{bl'} = bl$$

*Note that both inner blocks and task blocks refer to a parent block which may be an inner block or the root block of the block-structured process model. Thus, the leaves of the directed tree consist of task blocks only.* □

Figure 7.10 presents an example for a block-structured process model in BPMN (task blocks and the root block are not explicitly represented). Figure 7.11 depicts the corresponding directed parse tree of blocks [228].

From a block-structured process model, the set of relevant VCFEs for the entire process model can be obtained by recursively merging VCFEs representing each block considering the semantics underlying each block type. In the following, important aspects in this regard

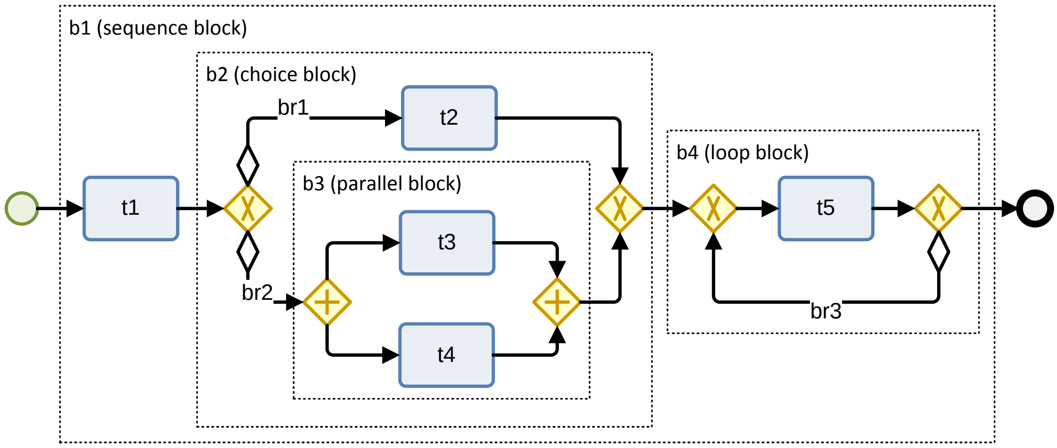



Figure 7.10: Example of a Block-structured Business Process Model

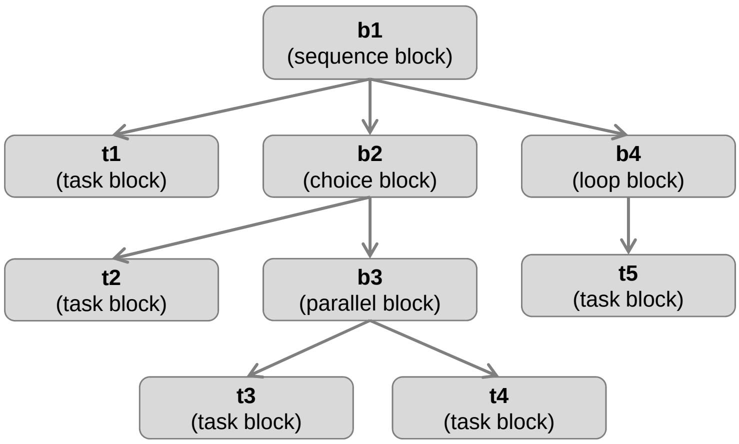



Figure 7.11: Example of a Block-structured Business Process Model: Parse Tree

are discussed on the basis of a corresponding prototypical implementation. In particular, the requirements lined out above have been considered. Algorithms to deduct the set of VCFEs for relevant block types are included in Appendix B.

Figure 7.12 presents an UML class diagram [229] implemented by the prototype. Note that the individual classes represent the constituents of a block-structured process model (cf. Definition 23), and the relevant types of CFEs.

Each class implementing the *block* interface provides a method *getVirtualControlFlowElements* delivering the set of VCFEs for the corresponding block. For each block, the set of VCFEs needs to be computed only once by executing the *createVirtualControlFlowElements* method provided by the corresponding block class. Since sub-blocks need to be considered when creating VCFEs for a block, this method will recursively call the *getVirtu-*

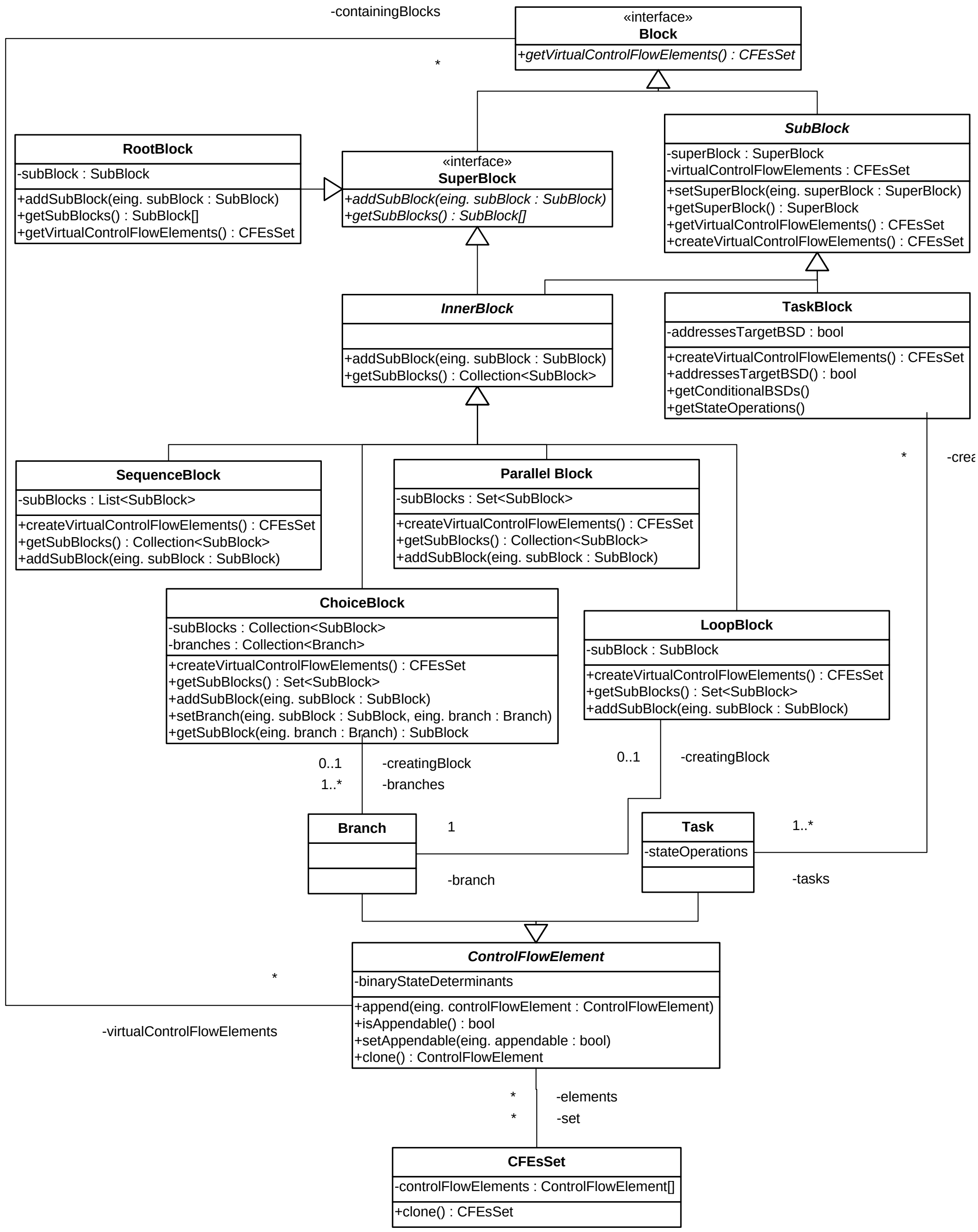



Figure 7.12: Class Diagram: Block-structured Process Model

*alControlFlowElements* methods (and thus the *createVirtualControlFlowElements* methods) of sub-blocks. Thereafter, the VCFEs can be kept persistent to avoid computational effort for subsequent *createVirtualControlFlowElements* calls. This is achieved by implementing *getVirtualControlFlowElements* as a final method of the abstract *SubBlock* class, which is extended by each block class except *RootBlock*.[5] The computation of VCFEs according to the semantics of each block type lie at the core of the approach proposed. It is important to keep in mind that the VCFEs associated with each block represent *alternative* paths through the block.

With regard to CFEs, note the *append* method and the *appendable* attribute. If the *append* method is called and the CFE is *appendable*, the conditional BSDs and state operations of the CFE are adapted to represent the merged CFE resulting from merging the original CFE with the parameter CFE. The distinction of *appendable* and *non-appendable* CFEs helps to restrict the computation of VCFEs to relevant VCFEs only:

- As discussed, to analyze quality relations, only the PEPs that terminate with a task addressing a target BSD need to be considered. Therefore, as soon as there is a task addressing a target BSD, the PEP terminating in this task is retained as part of the set of relevant PEPs. No further CFEs should be appended to this PEP since this would add resource requirements to the corresponding VCFE which are not necessary to address the target BSD in question.
- Besides the non-appendable PEPs, it is necessary to determine the PEPs through a block that allow process enactment to continue after the block. The resulting VCFEs represent resource requirements that need to be fulfilled to enable the enactment of subsequent blocks [22].

**Example 40 (Appendable and Non-appendable Control Flow Elements).** Consider a parallel block of two tasks where one task addresses a target BSD (cf. Definition 15). For the analysis of quality relations, a PEP terminating in the task addressing the target BSD is relevant. From this perspective, however, it is irrelevant whether the other task is enacted as well. The corresponding resource requirements must not be considered as prerequisites to address the target BSD. However, if there is a third task in the process model which addresses a second target BSD directly after the parallel join, the PEP *through* the parallel block needs to be considered as well, since it provides relevant resource requirements for the second target BSD. The corresponding VCFE needs to be *appendable* because the third task must be merged to it.

As discussed above, *task blocks* constitute the leaves of the tree structure representing a block-structured process model. Accordingly, when recursively creating VCFEs from a block-structured process model, VCFEs will be initiated by task blocks. As presented in Algorithm B.1, the *createVirtualControlFlowElements* method of a task block will provide either one or two VCFEs: an appendable one in any case, and an additional non-appendable VCFE if the task block addresses a target BSD.

For *sequence blocks*, the set of VCFEs can be determined as defined in Algorithm B.2 (cf. Appendix B). In a sequence block, one of the PEPs of the first sub-block must be completed

[5] A final method cannot be overrun by classes extending the class defining the abstract method. An abstract class cannot be instantiated. It only serves as a basis to define extending classes.

before one of the PEPs of the second sub-block can be enacted, and so on. Accordingly, the number of VCFEs of a sequence block can be determined as follows:

**Corollary 1 (Number of Virtual Control Flow Elements of a Sequence Block).** *Let sb be a sequence block with* $(bl_{sb_1}, \ldots, bl_{sb_n})$ *as its sequence of sub-blocks. Let* $n$ *be the number of sub-blocks of the sequence block, let* $n_i$ *be the number of appendable VCFEs of the i-th sub-block, and let* $o_i$ *be the number of non-appendable VCFEs of the i-th sub-block. The number of relevant VCFEs of the sequence block sb is then derived as follows:*

$$| VCFE_{sb} |= \prod_{i=1..n} n_i + o_1 + \sum_{i=2..n} \left( \left( \prod_{j=1..(i-1)} n_j \right) \cdot o_i \right)$$

*All appendable VCFEs through the sub-blocks are combined to obtain the possible paths* through *the sequence block. Beyond that, for each non-appendable VCFE of each sub-block, another non-appendable VCFE for the super block is created for each path* to *the sub-block.* □

Note that, unlike the other inner block types, sequence blocks obviously require an ordering of their sub-blocks.

For *choice blocks*, the set of VCFEs can be determined as defined in Algorithm B.3 (cf. Appendix B). In this case, it is first necessary to create an appendable CFE for each branch of the choice block representing the conditional BSDs to enter the respective sub-block. For each sub-block, the respective branch is cloned for each VCFE, and the VCFEs are appended to the clones. Note that "do nothing" branches can be addressed by inserting a VCFE representing an "empty" task, i.e., a task without conditional BSDs or state operations. Accordingly, the number of VCFEs for the choice block is simply determined as follows:

**Corollary 2 (Number of Virtual Control Flow Elements of a Choice Block).** *Let cb be a choice block with* $\{b_{cb_1}, \ldots, b_{cb_n}\}$ *and* $\{bl_{cb_1}, \ldots, bl_{cb_o}\}$ *as its sets of branches and sub-blocks, respectively. Note that* $n \geq o$*, i.e. there are at least as many branches as there are sub-blocks. This characteristic reflects potential "do nothing" branches. Let* $o_i$ *be the number of VCFEs of the i-th sub-block. The number of relevant VCFEs of the choice block cb is then derived as follows:*

$$| VCFE_{cb} |= \sum_{i=1..o} o_i + (n - o)$$

*The number of VCFEs for the choice blocks is the total of the number of VCFEs for each sub-block plus one additional VCFE for each branch without a sub-block.* □

For *parallel blocks*, the set of VCFEs can be determined as presented in Algorithm B.4 (cf. Appendix B). The semantics of this workflow pattern result in one appendable VCFE representing the case that all sub-blocks of the parallel blocks are completely (i.e., appendably) enacted – only then will a BP instance continue after the parallel block. Moreover, it is necessary to consider all relevant combinations of partial enactment of sub-blocks addressing target BSDs. The corresponding VCFEs represent process instances that address one or more target BSDs and terminate *within* the parallel block. They are obtained by combining non-appendable VCFEs of sub-blocks.

To this end, it is necessary to establish the principle that parallel sub-blocks must be *conditionally independent* from each other, which follows from the assumption that the sequence in which parallel sub-blocks are enacted is not determined. Using quality-aware BP modeling constructs, this characteristic can be defined more closely:

**Definition 30 (Mutual Conditional Independence of Parallel Sub-blocks).** *Let $pb = \langle\{bl_{pb_1}, \ldots, bl_{pb_i}\}, pa_{pb}\rangle$ be a parallel block. Let $\Phi_j = \{\phi_{j_1}, \ldots, \phi_{j_k}\}$ be the set of CFEs (i.e., tasks and branches) comprised in the task, choice, and loop block children of the j-th parallel sub-block $bl_{pb_j}$. Let $\Delta_\phi$ and $\Gamma_\phi$ be the sets of BSDs and state operations comprised in a CFE $\phi$ (cf. Definition 14). Mutual conditional independence of the parallel sub-blocks is given iff there are no environmental elements which are affected in one parallel path, but constitute affecting elements of another one. Thus, the following condition applies to maintain mutual conditional independence of parallel sub-blocks as a prerequisite of the overall semantic consistency of the process model:*

$$\forall\langle bl_{pb_l}, bl_{pb_m}\rangle : \left(\bigcup_{\phi\in\Phi_l}\left(\bigcup_{\delta\in\Delta_\phi}\theta_\delta\right)\right)\bigcap\left(\bigcup_{\phi\in\Phi_m}\left(\bigcup_{\gamma\in\Gamma_\phi}E_\gamma\right)\right) = \emptyset$$

□

On that basis, the number of VCFEs for a parallel block can be computed as follows:

**Corollary 3 (Number of Virtual Control Flow Elements of a Parallel Block).** *Let pb be a parallel block with $\{bl_{pb_1}, \ldots, bl_{pb_n}\}$ as its set of sub-blocks. Let $n_i$ and $o_i$ be the number of appendable and non-appendable VCFEs, respectively, of the i-th sub-block. The number of relevant VCFEs of the parallel block pb is then derived as follows:*

$$\mid VCFE_{pb} \mid = \prod_{i=1..n} n_i + \prod_{i=1..n} (o_i + 1) - 1$$

*All appendable VCFEs through the sub-blocks are combined to obtain the possible paths* through *the parallel block. In addition, all possible combinations of VCFEs where each sub-block contributes either one or no non-appendable VCFE (minus one, for the combination of no VCFE for any sub-block) represent the possible combinations of target aspects addressed without necessarily completing the parallel block.* □

*Loop blocks* constitute a particular challenge since, at first glance, they extend the number of VCFEs for a process model to infinity. The loop block consists of a loop condition[6] modeled as a branch and a loop body modeled as a sub-block. On that basis, the sequence of VCFEs of the loop block presents itself as follows: Let $lb = \langle b_{lb}, bl_{lb}, pa_{lb}\rangle$ be a loop block with $b_{lb}$ as its loop condition branch and $bl_{lb}$ as its loop body sub-block. Let $VCFE_{bl_{lb}}$ be the set of VCFEs of the loop body. The set of VCFEs representing PEPs through the loop block is then defined as follows:

$$VCFE_{lb} = \{(v_1, b_{lb}, v_2, b_{lb}, \ldots) \mid v_i \in VCFE_{bl_{lb}}, i \in \mathbb{N}\}$$

[6] Note that this implies *do..while* semantics for loop blocks. Other relevant loop semantics (namely, *while..do* loops) can be converted trivially.

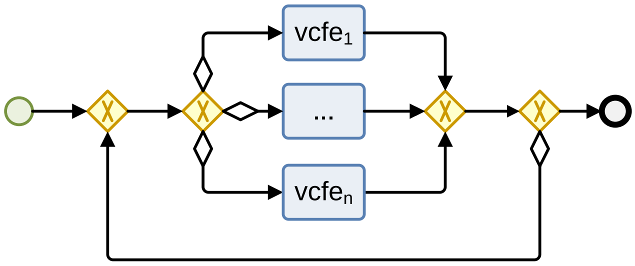



Figure 7.13: General Form of Loops

That is, the VCFEs of the loop block consist of alternating sequences of any VCFE of the loop body and the loop condition of any conceivable length. Considering conditional consolidation of the loop block, loops thus generally assume the form given in Figure 7.13. Note that, to avoid infinite loops, the loop condition must be *addressed* (cf. Definition 15) by at least one VCFE of the loop body.

To further assess loops, several *loop types* can be discerned based on two characteristics, as depicted in Figure 7.14:

- There are *first order* and *n-th order loops*. First order loops represent a particular type of loops where the loop body can be consolidated into only one VCFE. This is the case if the loop body solely consists of sequence and task blocks. As opposed to a first order loop, an n-th order loop has a loop body with *n* VCFEs. Whether a loop is a first or n-th order loop can be determined by inspecting the loop body.

- There are *deterministic* and *non-deterministic loops*. Deterministic loops are characterized by the existence of a state operation within a VCFE of the loop body which addresses the loop condition and whose impact relation is deterministic, i.e., fully determined by elements of the outer conditional environment. In addition, the choice of the loop body VCFE enacted in each iteration must be determined by task-requisite BSDs addressed only by deterministic state operations within the loop body. In other words, for deterministic loops, the number of iterations through the loop is pre-determined by the state of the outer conditional environment when the loop is entered. Whether a loop is a deterministic or non-deterministic loop can be determined by inspecting the loop condition *and* the loop body. In typical application scenarios, non-deterministic loops model attempts to achieve a human decision (e.g., finding the right contact partner for an inquiry). Deterministic loops mostly represent the step-by-step accumulation of resources required for subsequent activities, or deal with multiple uniform environmental elements where the total number of elements is not known at design time (e.g., checking line items of a supplier invoice).

Deducting quality relations aims at enabling assessment of the resource availability and consumption requirements posed by a BP model towards achieving aspects of a business objective. In this respect, the semantics underlying loops in process models must be taken into account to effectively delineate their impact on BP quality. In other words, effective appraisal of the quality impact of loops will ask what causes parts of the BP model to be en-

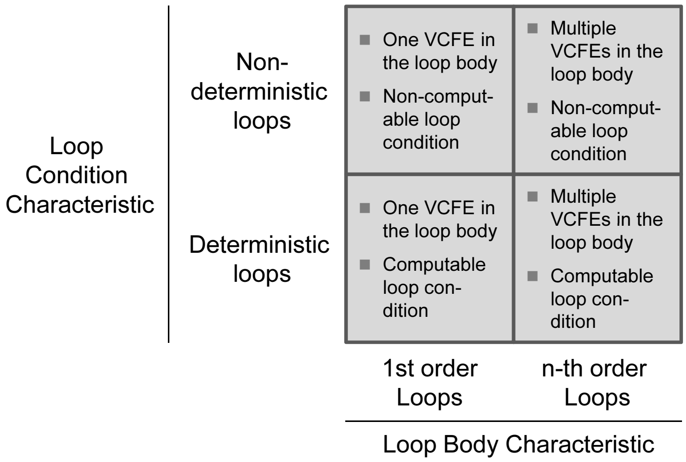



Figure 7.14: Loop Types

acted repeatedly, and how this is to be appraised in terms of resulting resource requirements. To facilitate this discussion, several *loop patterns* can typically be distinguished.[7]

- *Loops as sub-processes to manage sets of target elements:* This loop pattern is typically used to deal with aspects of business objectives where the cardinality of an underlying set of target elements is not known at design time. For example, consider checking line items in a supplier invoice or similar processes to manage list constructs. During business objective modeling and process design, it is customary to use a construct like, e.g., "list completed" as loop condition. To appraise resource requirements for this type of loop, one will be interested in the quality relations associated with a single list item, since the total number of list items is not determined by the business process. Accordingly, it makes sense to deduct separate quality relations for dealing with one *single element* of the target element set underlying the loop, and appraise this in addition to the quality relations of the surrounding business process.
- *Loops as sub-processes to ensure resource availability:* This type of loops is typically used to incrementally approach a resource availability level required for subsequent activities. For example, consider the collection of a set of measurement readings to enable medical diagnosis. In many cases, the required number of iterations through the loop will depend on an initial level of resource availability, e.g. when replenishing shelf slots in retailing. Similar to *loops as sub-processes to manage sets of target elements*, one will be interested in resource requirements associated with obtaining a *single unit* of the resource targeted by the loop. Accordingly, it is again advisable to deduct separate quality relations for the loop.
- *Loops as sub-processes to manage non-deterministic state operations:* In some application scenarios, the progress of a business case depends on achieving a particular result out of a non-deterministic state operation (cf. Definition 11). For example, consider

[7]Loops which do not fall into one of these categories would violate Quality Attribute 2, *Effective Tasks*, as will be discussed in Section 8.3.

approval processes where a process instance may be handed over from one contact partner to another until the responsible person has been found. In this case, the number of iterations required is not determined by the process model, but by human involvement in the business process, with qualified human labor as a resource. Again, it thus makes sense to consider resource requirements caused by a *single iteration* through the loop according to the process model, and in addition capture needless iterations caused by human involvement in the business process as a matter of *enactment* quality.[8]

The above considerations on effective appraisal of the contribution of loop patterns to resource requirements lead to the conclusion that loops should, in terms of quality appraisals, be treated as sub-processes with corresponding *sub-business objectives* expressing what the loop should achieve. This entails a number of challenges to be resolved in the following: *Firstly,* how to deduct the sub-business objective of a loop block? *Secondly,* how to deal with n-th order loops? *Thirdly,* how to deal with nested loops?

To deduct the sub-business objective of a loop pattern, it is necessary to inspect the loop condition and the loop body VCFEs addressing the loop condition. The sub-business objective is then defined by a set of BSDs representing all state operations addressing the loop condition within the loop body:

**Definition 31 (Sub-Business Objective).** *For a loop block $lb = \langle b_{lb}, bl_{lb}, pa_{lb} \rangle$ with $b_{lb}$ as its loop condition branch and $bl_{lb}$ as its loop body sub-block, let $VCFE_{bl_{lb}}$ be the set of VCFEs of the loop body. The sub-business objective $sbo_{lb}$ for the loop is then defined as follows:*

$$\begin{aligned} sbo_{lb} = &\{\langle \{e\}, \Lambda \rangle \mid && \textit{(i)} \\ & e \in E_{b_{lb}}, && \textit{(ii)} \\ & \exists \delta \in \bigcup_{vcfe \in VCFE_{bl_{lb}}} \Delta_{vcfe} : \theta_\delta = e, \Lambda = i_\delta \} && \textit{(iii)} \end{aligned}$$

*That is, since the loop condition must be fulfilled unconditionally, the target BSDs of the sub-business objective are not linked to conditional propositions, and refer just one affecting element each (i). The affecting element of each target BSD corresponds to an affecting BSD of the loop condition (ii). For each target BSD, there is a state operation within the loop body addressing the target BSD and hence the loop condition. The set of states for which the target BSD is fulfilled corresponds to the image functions of the state operation (iii). Accordingly, a loop iteration through a loop VCFE addressing a target BSD would fulfill a target BSD.*

Note that the above definition results in the sub-business objective of a loop being defined to match what can be achieved within one loop iteration. This characteristic is useful for practical appraisal of loop quality relations since it corresponds to the usual human thought pattern of comparing the results of one loop iteration to corresponding resource requirements.

n-th order loops entail the possibility of loop body VCFEs which do not address the loop condition. Semantic consistency of loops, however, requires that VCFEs not addressing the loop condition enable enacting another loop body VCFE in a subsequent loop iteration –

[8] Cf. Quality Attribute 14 as presented in Section 8.3.

otherwise, the loop might not be completed. Note that this also means that loop body VCFEs which do not address the loop condition are reflective (cf. Definition 15), since only one loop body VCFE can be enacted per loop iteration. In the following, these loop body VCFEs are referred to as *indirect loop body VCFEs*. In terms of semantics, indirect loop body VCFEs are generally used to provide required resources for VCFEs addressing the loop condition. Accordingly, they can also be viewed as nested loops with a loop condition derived from task-requisite BSDs of subsequent VCFEs. Example 41 illustrates these considerations.

**Example 41 (Indirect Virtual Control Flow Elements and Nested Loops).** Consider Sample Process B from Figure 2.6. In the course of managing payment lists, it is conceivable that payment list line items above a given threshold value require final approval by a responsible senior manager. To this end, a loop block with a loop condition of having all respective line items approved or dis-approved might be introduced. In this context, relevant line items could be sent to one senior manager after another to find the relevant contact partner.

Figure 7.15 describes two possible modeling options for this approach. The upper model fragment shows how the approach might be modeled as a second order loop: The line item is first sent to a senior manager. If the manager has been the right contact partner, she approves or dis-approves the line item. If the final item has been approved or dis-approved, the loop terminates, otherwise, another iteration commences. The lower model fragment shows how the approach might be modeled with nested loops: The inner loop finds the right senior manager as contact partner, before the line item is approved or dis-approved in the remaining tasks of the outer loop.

With regard to resource requirements, the right contact partner to check the line item constitutes a task-requisite BSD to the respective task. Providing this resource can either be modeled as a VCFE not addressing the loop condition, or as a nested loop.

To convert an indirect VCFE to an inner nested loop, it is necessary to build a new loop condition for the inner nested loop. The new loop condition can be derived from the VCFEs addressed by the indirect VCFEs. To this end, the *addresses* interrelations (cf. Definition 15) between indirect VCFEs and other VCFEs are assessed. The following consistency requirements apply:

- The loop condition must be addressed by at least one loop body VCFE to avoid infinite loops.
- Since VCFEs through one block are defined as mutually exclusive for a particular state of the relevant outer environment, each loop body VCFE must address at least one other loop body VCFE or the loop condition to avoid infinite loops.

For very complex loops (which are not probable to occur in practical situations), this may result in a tree structure, corresponding to multiple nesting levels of inner loops. For each indirect VCFE, the relevant loop condition corresponds to the union set of the task-requisite BSDs of VCFEs addressed. Accordingly, a sub-business objective can be derived as described in Definition 31.

On the basis of the above discussion, nested loops result in a tree structure of sub-business objectives according to their nesting. For each sub-business objective, quality relations can

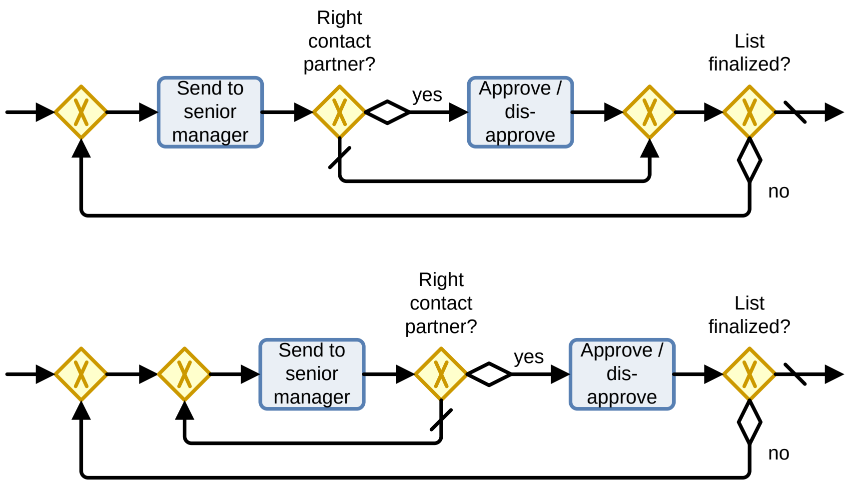



Figure 7.15: Example: Indirect Virtual Control Flow Elements and Nested Loops

be readily deducted from the corresponding VCFEs. This approach aligns well with the human thought pattern of appraising results and respective resource requirements on the basis of possible single loop iterations. This also implies that loop blocks are not further considered in the conditional consolidation of their parent blocks by way of *appendable VCFEs*. This way, resource requirements that refer to resources made available through loops are not eliminated, but can be compared to resource requirements associated with corresponding loop sub-business objectives.

In the case of target BSDs directly addressed in the loop body, the corresponding loop body VCFEs are considered as non-appendable VCFEs in the course of conditional consolidation. Again, if the corresponding loop body VCFEs depend on other loop body VCFEs to be enabled (due to nested inner loops or n-th order loops), the respective resource requirements are made transparent through the relevant sub-business objectives.

## 7.6 Conclusion

In this chapter, an approach towards quality-aware BP modeling has been built based on functional and practical applicability requirements, insights on related work, and available approaches towards business objectives modeling (cf. Chapter 6). Required terminology has been derived from functional requirements, and results have been integrated into a meta-model extending BPMN. A method to assess the efficacy and efficiency of a corresponding BP model has been described. Moreover, the method has been applied to a sample case to initially demonstrate its validity. Relevant concepts, including state operations, conditional consolidation, and possible enactment paths, have been described technically to enable the implementation of corresponding tools. A corresponding prototype implementation has been used to further ensure the validity of relevant technical propositions.

Application to a sample case has shown that the quality-aware BP meta-model (cf. Figure 7.5) fulfills the functional requirement of enabling the derivation of quality relations (cf. Figure 7.1). However, the assessment of *full efficacy* and *full efficiency* still remains an issue where subject matter experts' judgment is required. Nevertheless, quality-aware BP modeling provides a basis to further elaborate assessment criteria in this respect, since it addresses the up to now poorly understood "interrelation between control flow and quality" [123]. Moreover, the concise definition and description of quality-aware concepts in relation to BP models enables supporting subject matter experts' appraisal of quality attributes by implementing corresponding tools, or integrating relevant functionality into available BPM systems. The issue of more detailed assessment criteria will be further pursued in Chapter 8.

Comparable to other modeling tasks in the field of BPM and beyond, compiling quality-aware BP models may entail substantial effort. Future work will therefore look into possibilities to support quality-aware BP modeling as far as possible. Notable approaches in this respect might include the use of process mining technology [28], e.g., to identify similar tasks that might share task-requisite BSDs and state operations, or repositories of process "snippets" in the sense of re-usable fragments. Note that these considerations also apply to the modeling of business objectives as discussed in Chapter 6.

# 8 Business Process Quality Model

An effective BP quality model constitutes one of the central deliverables required to attain the research objectives of this thesis (cf. Section 1.5). As lined out in Chapter 7, its fundamental purpose lies in facilitating appropriate appraisal of the *efficacy* and the *efficiency relation* as constituents of overall BP quality. In terms of design science, it constitutes a *model* since it interrelates relevant concepts [77]. Figure 8.1 puts the BP quality model into the context of the preceding chapters: The BP quality definition of Chapter 5 enables a common understanding of what constitutes a good business process, and Chapters 6 and 7 provide the means to derive the efficacy and efficiency relations. On that basis, this chapter finally develops an approach to enable quality appraisal. In particular, this pertains to *quality attributes* that determine whether quality relations can be considered as “reasonable” in the sense of Definition 1.

Note that, in the sense of this thesis, quality attributes can also be viewed as the factors that drive BP efficacy and efficiency. Similar to this notion, the EcoPOST approach developed by Mutschler et al. analyzes factors that drive the implementation cost of PAIS [230, 231]. In comparison to the proposition presented here, EcoPOST addresses a narrower field.[1] This allows considering *cost and impact factors* in much more detail. For example, it is possible to model multiple layers of interrelated factors. However, this means that there is no *generic* set of cost and impact factors comparable to the quality attributes discussed here. Rather, a cost and impact factor model must be developed or adapted for each individual application case, e.g., on the basis of *evaluation patterns* as proposed in [232].

Section 7.1 presents the components of an effective BP quality model. The challenge of *comprehensive coverage* is of particular relevance in this regard (cf. Chapter 10), and will be

[1]PAIS implementation cost can be viewed as a particular aspect relevant to efficiency during the *design & implementation* lifecycle phase (cf. Section 8.4.1).

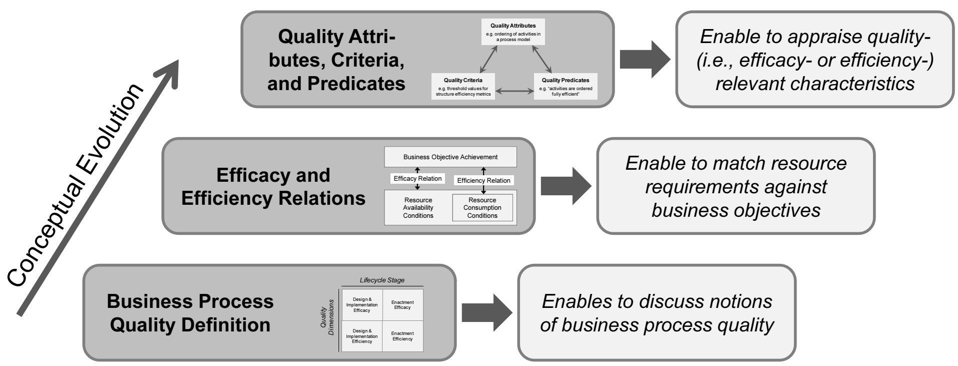



Figure 8.1: Conceptual Evolution towards the Business Process Quality Model

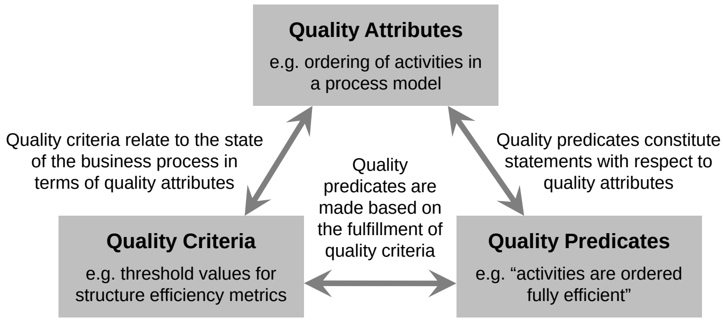



Figure 8.2: Business Process Quality Model Components

addressed through a rigorous deductive approach (cf. Section 3.2). This can be achieved by breaking down the complex *knowledge problem* [104] of finding a comprehensive set of *quality attributes* (cf. Section 8.1) into smaller problems related to particular sub-sets of quality attributes. Sub-sets of quality attributes are defined by appropriate classification. Thus, the corresponding *quality attribute types* are discussed in this chapter. On that basis, the individual quality attributes are defined. To reflect the *Transparency and retraceability* as well as the *Cost effectiveness* criteria (cf. Section 3.1), this chapter discusses which quality attribute aspects can be assessed with the support of automated tools. In Chapter 11, the resulting set of quality attributes is then matched against comparable results from related literature to assure its completeness, and to discuss the additional contribution of this thesis in that respect.

## 8.1 Quality Model Components

Quality assessment amounts to appraising whether the efficacy and efficiency relations of a business process can be considered as *reasonable*. Without further elaborating on what can be considered as *reasonable*, this approach of arbitrary judgment, however, fails to address the effectiveness criteria defined for this thesis (cf. Section 3.1):

- In terms of *Congruence to organizational targets*, it is not possible to ensure that relevant aspects have been considered *comprehensively*, but *exclusively*.
- In terms of *Transparency and retraceability*, it is not possible to track how assessment results have been achieved.
- In terms of *Cost effectiveness*, an efficient approach cannot be ensured for lack of structured procedures, for instance regarding the use of assessment automation potentials.

To address these issues, the granularity of the *efficacy* and *efficiency* concepts must be raised substantially by providing organizations with a well-structured *quality model* to support assessment procedures. To this end and according to the research methodology applied in this thesis (cf. Chapter 3), three interrelated sets of elements as illustrated in Figure 8.2 are required.

**Definition 32 (Quality Attributes).** Quality attributes *describe properties of business processes or their outer environment which are relevant to BP quality. This means that changes to quality attributes may change the perception of the overall quality of a business process.* □

Quality attributes serve as the basis for BP quality assessment and management, and enable the definition of quality criteria (see below). For an effective quality model, they must cover the quality definition framework *exclusively* and *comprehensively.* On the one hand, this means that no attributes may be comprised that do not relate to the quality definition. On the other hand, the attributes must suffice to cover all aspects of the quality definition.[2] Quality attributes are independent from a concrete business process or application domain. As a simple example, consider *task automation*, i.e., the question whether tasks within the business process are enacted manually or automated through information systems or machinery.

**Definition 33 (Quality Criteria).** Quality criteria *amend quality attributes with target or threshold values which are at least ordinally scaled. They describe aspirations for a related quality attribute. That is, quality criteria capture desirable expressions of quality attributes.* □

Quality criteria serve as the link between quality attributes and quality predicates (see below), and may or may not be specific to a concrete business process or application domain. As examples of quality criteria referring to *task automation*, consider the following possibilities:

- All tasks within the business process may be automated as far as possible.
- All tasks may be automated as far as reasonable. That is, the required capital investments of additional automation would exceed the potential benefits.
- There may be tasks where additional automation would be reasonable.

**Definition 34 (Quality Predicates).** Quality predicates *are linked to quality criteria and comprise statements on quality that are valid iff the respective quality criteria are fulfilled. That is, quality predicates constitute statements describing a business process that are valid if conditions expressed as quality criteria are fulfilled.* □

Quality predicates should be defined in a way to enable active quality management, i.e., improvement potentials should be pointed out as appropriate. A rigorous deductive approach to quality modeling ensures that subsequent quality management actions are aligned to the quality definition framework, and thus to organizational targets. Quality predicates may or may not be specific to a concrete business process or application domain. As examples of quality predicates referring to *task automation* and the quality criteria above, consider the following statements:

- There are no further task automation potentials.
- Existing task automation potentials *should not* be further pursued.

[2] Cf. the discussion of "satisficing" design artifacts in [103]. Again, this clarifies the need for a concise quality definition construct. The quality definition of this thesis enables discussing the underlying notions on an aggregated level as well as evaluating subsequent design artifacts like a quality model.

- Existing task automation potentials *should* be further pursued.

Note that these types of model components exhibit the common "nested" character of design science research: the consecutive resulting research questions alternatingly address both *practical problems* and *knowledge problems* [104]. The construction of a short and concise definition framework for BP quality in Chapter 5 corresponds to the practical problem of finding a definition that is useful for management purposes. In this chapter, relevant quality attributes that influence BP quality are evaluated. In the sense of a theory on BP quality levers, this constitutes a knowledge problem. In a second step, the resulting quality attributes are augmented with criteria and predicates based on what is efficient to enable BP management and drive BP optimization. This, in turn, amounts to a practical problem.

## 8.2 Quality Attribute Types

In line with other BPM concepts, a definition for BP quality constitutes a goal-bound design artifact in the sense of [103]. Its purpose is to contribute to a quality model which is, in turn, oriented at the effectiveness criteria described in Section 3.1. The individual quality attributes, and hence the content of the quality model, in contrast, are based on a natural science paradigm. To define them, one will not ask the practical question of what would be useful, but the knowledge question of how to actually determine the impact of the business process on its outer environment with regard to organizational targets.

Regarding the aspects of Effectiveness Criterion 1, *Congruence to organizational targets*, *exclusive coverage* can be assured in a fairly straightforward manner, i.e. by determining whether each quality attribute relates to an organizational target, and is in the scope of influence of the business process. *Comprehensive coverage*, however, constitutes a major challenge in this respect. In this thesis, the challenge of *comprehensive coverage* is addressed by following through a deductive approach (cf. Section 3.2). Quality attributes are derived from well-understood concepts by inspecting all relevant characteristics and interrelations. To support this endeavor, this section discusses various types of quality attributes which help to determine relevant aspects of the concepts to be considered. In other words, the various classifications of quality attributes can be understood as mental techniques to comprehensively identify quality attributes.

### 8.2.1 Inductive vs. Deductive Quality Attributes

To assess the impact of a business process on its outer environment, the functional interrelation between the affecting environment, the business process, and the affected environment has to be considered. The business process *and* the affecting environment interact to impact the affected environment. To obtain a meaningful assessment of the quality of the business process, it is necessary to delineate accordingly. It is thus not adequate to employ elements of the affected environment on their own as quality attributes for a business process. For example, it is not sufficient to determine whether target aspects are fulfilled by process instances to determine BP quality – one needs to consider that failure to fulfill target aspects may be caused, for instance, by a lack of resources. In that case, one might observe poor resource management, but not necessarily poor process quality. Rather, the quality

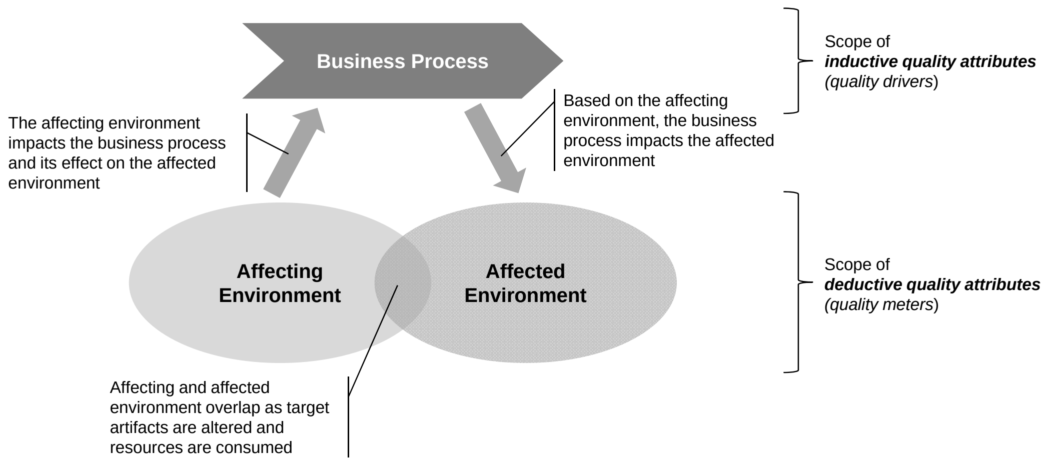



Figure 8.3: Quality Attributes Context

of a business process in the sense of this thesis can be determined by either assessing its characteristics directly, or by indirectly analyzing its affected *and* affecting environment. According to Definition 1, the affected environment – as far as it is relevant to BP quality – consists of target artifacts and resources. Thus, there are two types of quality attributes in terms of the appropriate assessment approach.

**Definition 35 (BP Quality Attribute Types: Assessment Approach).** Inductive quality attributes *or* quality drivers *are characteristics of a business process apt to* a priori determine *its impact on its outer environment with respect to efficacy and efficiency. Quality drivers can be assessed by inspecting a BP model including interactions with the outer environment. They do not require analyzing an actual outer environment related to a set of BP instances. Quality drivers relate to the theoretical achievement of the business objective, or to the theoretical consumption of resources, or to the relation between these two aspects. Accordingly, quality drivers pertain to the design & implementation lifecycle stage.*

Deductive quality attributes *or* quality meters *are relations between the affecting and the affected environment of a business process apt to* ex post measure *its impact on its outer environment with respect to efficacy and efficiency. Quality meters can be assessed by analyzing the outer environment of a set of actual BP instances or of an actual BP implementation. They do not necessarily require knowledge of the underlying BP model. A quality meter either relates the achievement of the business objective, or the consumption of resources, or both aspects to the affecting environment. Quality meters may pertain to either the design & implementation or to the enactment lifecycle stages.* □

Figure 8.3 summarizes the interrelations between the business process, its affecting and affected environment, and the two types of quality attributes. The latter are further illustrated in Example 42.

**Example 42 (Inductive and Deductive Quality Attributes).** Consider Sample Process A from Figure 2.5. The process comprises activities to check whether senior management approval is required for the invoice. Whether these activities are sufficient constitutes an *inductive quality attribute* or *quality meter*. It directly considers a property of the business process with relevance towards efficacy. No elements of the *actual* outer environment of the business process need to be taken into account.

The average processing cost for invoices considering whether purchase order data are properly maintained constitute an example for a *deductive quality attribute* or *quality meter*. The invoice is part of the affected environment, and the purchase order is part of the affecting environment. Other parts of the outer environment that, however, do not need to be considered for this quality attribute comprise the working time of staff employed for the affected environment, and the IT system in use for the affecting environment.

The differing quality requirements in terms of BP efficacy and efficiency in the course of the BP lifecycle are not equally suited for *a priori* determination and *ex post* measurement. The following issues have to be taken into account:

- Only actually affected parts of the environment can be measured. Note that during BP design & implementation, parts of the affected environment do not relate to the *actual* impact of the business process, but to its *expected* impact during enactment. This characteristic applies to target artifacts as well as all resources except capital expenditures (i.e., investments, e.g., in new IT systems).
- Common BPM and enterprise resource management information systems limit the scope of what can be measured in the enactment phase with reasonable effort. The success of BP instances with respect to the corresponding impact on target artifacts can often be determined via execution logs or similar means. However, the actual impact of a business process on resources is mostly recorded only in terms of what is required for reporting and accounting purposes (e.g., [233] defines requirements for inventory valuation).

According to these considerations, deductive quality attributes for the *design & implementation* lifecycle phase are only suitable in regard to resource requirements related to capital expenditures, and hence to particular aspects of efficiency. In the *enactment* lifecycle phase, practical applicability of deductive quality attributes is restricted by the capacity to track actual process enactment through workflow management systems (WfMSs) or similar means [14]. Even if WfMSs are able to track actual process enactment for each process instance, the consumption and required availability of resources (e.g., clerks' working time) is typically not measured. This circumstance severely restricts the situations where deductive quality attributes can actually be employed. Figure 8.4 consolidates the resulting strategies, i.e., the applicability of types of quality attributes according to Definition 35, with respect to quality requirements and lifecycle stages.[3]

According to Figure 8.4, the applicability of quality meters is limited in practice. During process *enactment*, actual resource consumption cannot be measured in most cases. During process *design & implementation*, resource consumption pertains to capital goods only. With regard to the achievement of business objectives, measurement will, in practice, be

[3] Comparably to the considerations on quality requirements and lifecycle stages in Figure 8.4, Hammer noted that processes may "fail to meet performance requirements either because of faulty *design* or faulty *execution*" [234].

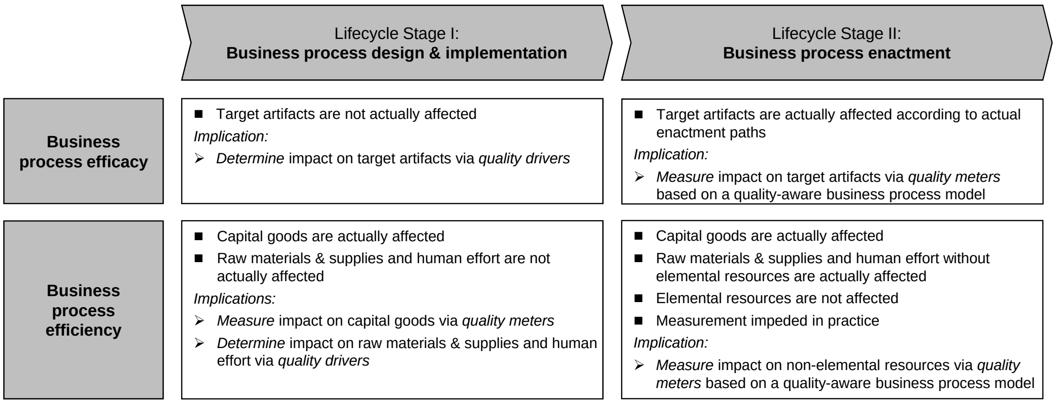



Figure 8.4: Strategies to Cover Quality Aspects

restricted to tracing actual enactment paths.[4] Inspecting target artifacts will be possible in special cases only, e.g., when samples are tested. Accordingly, design quality is generally not measurable ex-post, but needs to be determined based on a-priori inspection. Due to the limited practical measurability of the state of the affecting and affected environment during the *enactment* lifecycle phase, quality meters are restricted to particular aspects, and are most effective in conjunction with a comprehensive set of quality drivers (cf. Section 8.4). Thus, the considerations in the following subsections generally pertain to particular types of *quality drivers*. The special cases where quality meters are appropriate will be subject to separate discussion.

### 8.2.2 Absolute vs. Relative Quality Drivers

The objective of this thesis is to develop a *generic* approach to BP quality. This means that the quality model and, in turn, the set of quality attributes should be valid for business processes in general. This requirement, however, does not necessarily pertain to the interpretation of the respective quality criteria. Thus, quality predicates related to a quality driver may or may not be subject to interpretation by a process expert. In other words, quality predicates may be either *absolute* or *relative* to the domain of application. For relative quality drivers, the respective domain must be considered when assessing quality criteria. Each quality driver belongs to one of these two types.

**Definition 36 (BP Quality Driver Types: Domain Relevance).** Absolute quality drivers *are characteristics of a business process or relations between its affecting and affected environment where assessment according to the related quality criteria* does not *require expertise specific to a concrete domain of application.*

Relative quality drivers *are characteristics of a business process or relations between its affecting and affected environment where assessment according to the related quality criteria* does *require expertise specific to the concrete domain of the business process under assessment.* □

[4]For example in the context of compliance management, actual enactment paths are referred to as "traces" [155].

Example 43 illustrates the two types of quality drivers presented in this section.

**Example 43 (Absolute and Relative Quality Drivers).** Consider Sample Process C as presented in Figure 2.7. Assuming that examination C is part of a corresponding business objective model, whether it is enacted as part of the process model can be assessed without additional domain-specific knowledge. Accordingly, the completeness of tasks with respect to the business objective model constitutes an *absolute quality driver.*

On the other hand, whether examination C is executed on the basis of an efficient diagnostic method, i.e. if the task in itself can be considered as reasonably efficient, cannot be determined without domain-specific knowledge. Accordingly, the efficiency of individual tasks constitutes a *relative quality driver.*

### 8.2.3 Formal Efficacy vs. Resource-related Quality Drivers

Quality drivers relate to the quality requirement dimensions presented in Chapter 5, i.e. *efficacy* and *efficiency.* The discussion of quality-aware BP modeling in Chapter 7 has shown that resource consumption requirements, as constituents of the efficiency relation, are comprised in the resource availability requirements that make up the efficacy relation. Therefore, there will be no characteristics of business processes or their environment affecting efficiency, but not efficacy.

However, when comparing Definitions 8 and 9, it is notable that the concept of *formal efficacy* has no parallel in terms of efficiency. Since *formal efficacy* does, as opposed to all other quality aspects, not refer to the availability or consumption of resources, quality drivers in this respect can be separated from resource-related quality drivers. The formal modeling of the business objective in conjunction with quality-aware BP modeling allows assessing formal efficacy *without* domain-specific knowledge. Hence, quality drivers related to formal efficacy are always absolute quality drivers (cf. the previous section).

**Definition 37 (Quality Driver Types: Resource Relation).** Quality drivers related to formal efficacy *are characteristics of a business process impacting whether the associated business objective can be achieved in principle without considering resource requirements.*

Quality drivers related to resource requirements *are characteristics of a business process impacting resources required to be available or consumed to achieve the associated business objective. These types of resource requirements correspond to the efficacy and efficiency relation, respectively (cf. Chapter 7).* □

Example 44 illustrates the types of quality drivers presented in this section.

**Example 44 (Formal Efficacy-related and Resource-related Quality Attributes).** Consider Sample Process A from Figure 2.5. Whether the business process appropriately considers senior management approval as a requirement defined by the business objective constitutes a *quality driver related to formal efficacy.*

However, whether additional checks *not* required by the business objective are included in the process to approve an invoice constitutes a *quality driver related to resource requirements,* since the additional resource requirements posed pertain to both the availability and the consumption of resources.

### 8.2.4 Presence-based vs. Absence-based Quality Drivers

Quality drivers can also be classified according to whether the related quality criteria require a certain characteristic to be present or to be absent. In other words, the related quality criteria can be phrased positively or negatively. It is generally possible to convert from one type to the other (e.g., “all tasks must be...” vs. “no task may be...”). The categorization of drivers along the presence of characteristics is thus subject to individual preference.

It is also possible that a particular BP characteristic occurs in both presence-based and absence-based drivers, e.g. when considering formal efficacy- and resource-related attributes. However, to maintain formal efficacy- and resource-related drivers as a disjoint classification, presence-based and absence-based drivers shall be modeled separately in this case.

**Definition 38 (Quality Driver Types: Presence of Characteristics). Presence-based quality drivers** *are characteristics of a business process where the related quality criteria require the respective characteristic to be present.*

**Absence-based quality drivers** *are characteristics of a business process where the related quality criteria require the respective characteristic to be not present.* □

Example 45 illustrates the types of quality drivers presented in this section.

**Example 45 (Presence-based and Absence-based Quality Drivers).** Consider Sample Process B as presented in Figure 2.6. Whether the process model comprises all tasks required by the business objective, such as having the proposed payment list approved, constitutes a *presence-based quality driver* from a formal efficacy perspective.

Whether the process model comprises tasks that are not demanded by the business objective (and not required to execute other required tasks) such as advising the procurement department of payments made to suppliers, constitutes an *absence-based quality driver* from a resource-related perspective – if such tasks are present, resources are required to be available and consumed unnecessarily.

Whether all tasks in the process model are appropriately automated or supported by information systems, like the checking of master data in the sample process, constitutes a quality driver where, on the one hand, appropriate presence of information systems is required to achieve efficient process enactment. On the other hand, during design & implementation, efficiency is impeded by implementing unnecessary information systems. According to the principle of disjoint presence- and absence-based quality drivers, the respective underlying characteristic of the business process pertains to (at least) two quality drivers.

### 8.2.5 Quality Driver Levels

Chapter 7 presented how the efficacy and efficiency relations can be deducted from a quality-aware BP model. Retracing the steps discussed therefore provides valuable insights on how efficacy and efficiency are affected, which can be transferred into additional quality driver types.

Considering Step 1 in Section 7.4, a quality driver might refer to control flow or the possible enactment paths provided by a BP model. That is, the sequence of tasks and decision gateways in a model may impact formal efficacy as well as resource requirements, for example with regard to the demand to approve or disapprove conditional propositions as early as possible (cf. Section 6.5). Moreover, it is possible that a quality driver refers to an individual control flow element instead of possible enactment paths or sequences of control flow elements. On the level of individual control flow elements, resource availability may be required by tasks or by decision gateways, but resource consumption may be required by tasks only. Since the effect of decision gateways is reflected on the control flow path level, and decision gateways do not "waste" additional resources, only tasks are relevant to quality on the level of individual control flow elements. Poor quality in this sense means that the desired outcome of the task could be achieved with reduced resource requirements. Resource requirements are consolidated and assessed in Steps 2 and 3 in Section 7.4.

Finally, it is also possible that the fundamental idea behind the business process needs to be questioned. This aspect is typically addressed in *BP reengineering*, which aims not at incremental optimization of control flow or individual tasks, but at "radically" improving processes "in the large" [38, 37]. Typically, reengineering addresses one of the following aspects:

- Reengineering might pertain to the business objective instead of the business process. That is, reengineering might ask the question whether the result an organization aims to achieve through a business process is really required. However, as discussed in Chapter 2, this is not in the scope for BP quality management since it cannot be addressed in the course of BPM responsibilities. Accordingly, this aspect of reengineering is not considered further in this thesis.

- Reengineering might pertain to the use of (information) technology to drive process automation. Comparable to qualitative benchmarking [45], this aspect is addressed by appraising the applicability of domain-specific practices such as the use of particular software tools available in the market. Part of the appeal of this approach lies in the fact that many organization assume that it reduces the complexity of properly assessing technology investments. In other words, orientation at "industry best practice" is thought to minimize investment risk. This aspect is relevant to BP quality. In the following, it will be addressed by *conceptual level* quality drivers.

According to the considerations above, conceptual level quality drivers will always be industry-specific. Thus, they constitute *relative quality drivers.*

In summary, three *design levels* of quality drivers are to be discerned:

**Definition 39 (Quality Driver Types: Design Level).** Task level drivers *pertain to the efficacy and efficiency of individual tasks.*

Control flow level drivers *pertain to the efficacy and efficiency of possible enactment paths in the sense of sequences of control flow elements enabled by a BP model.*

Conceptual level drivers *pertain to the general approach towards achieving the business objective pursued by a business process.* □

Example 46 illustrates the types of quality drivers presented in this section.

**Example 46 (Quality Drivers and Levels of Process Optimization).** Consider Sample Process A as presented in Figure 2.5. If this process comprised a manual match of the purchase order instead of using an available data base, this would be a *task level* issue. If the process would require senior management approval before matching the goods receipt, thus incurring the risk of approving an invoice although the goods receipt is missing, this would constitute a *control flow level* issue. If, however, the process should be replaced by a credit notes procedure[5] to reflect industry standard, thus "sourcing" out invoice checking effort to the supplier, this would be a *conceptual level* issue.

### 8.2.6 Summary

The following propositions summarize the considerations on quality attributes made above, and provide guidance to further discussion:

1. A *natural science paradigm* applies to quality attributes. Accordingly, quality attributes can be elucidated by considering which characteristics of business processes *do* impact efficacy and efficiency, but not by considering which characteristics *should* impact efficacy and efficiency.
2. *Completeness of coverage* poses a major challenge to be addressed via a rigorous deductive methodology. Various types of quality attributes will support comprehensively identifying relevant aspects.
3. There are inductive *a priori quality drivers*, and deductive *ex post quality meters*. In this respect, quality meters are applicable to special circumstances only. In general, effective quality assessment will need to rely on quality drivers.
4. There are *absolute* and *relative* quality drivers, depending on whether domain expertise is required for assessment.
5. There are *formal efficacy* and *resource-related* quality drivers, depending on which quality requirement dimension is addressed by the quality driver. Formal efficacy quality drivers are always absolute quality drivers.
6. There are *presence-based* and *absence-based* quality attributes, depending on whether the related characteristics should be present or not present. Presence-based drivers can be converted into absence-based drivers, and vice versa.
7. There are quality drivers on the *conceptual level*, on the *control flow level*, and on the *task level*, depending on the scope addressed by the quality driver. Conceptual level quality drivers are always relative quality drivers.

Considering the mutual exclusivity of formal efficacy and relative quality drivers on the one hand, and conceptual level and absolute quality drivers on the other hand, reengineering

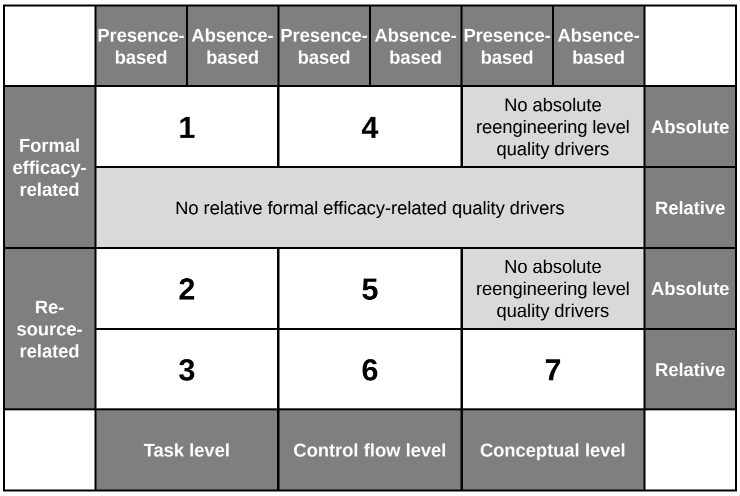



Figure 8.5: Relevant Types of Quality Drivers

level quality drivers will always be resource-related. This corresponds to the stipulation that "objectives" reengineering is not addressed in the scope of this thesis.

Figure 8.5 summarizes the resulting classification of quality driver types. Each relevant combination of types can be expressed in the form of a corresponding *guiding question* identified by the reference numbers in the figure. Since positive and negative quality drivers are convertible into one another (cf. Section 8.2.4), they are addressed through common guiding questions. This mental technique is used to support the identification of quality drivers, in particular with regard to obtaining a comprehensive set.

In the following section, quality drivers (i.e., inductive quality attributes) are discussed along the classification presented above. Subsequently, the special cases where quality meters (i.e., deductive quality attributes) are appropriate are revisited.

## 8.3 Quality Drivers

This section traverses the classification depicted in Figure 8.5 to identify individual quality drivers. Accordingly, the discussion of quality drivers is structured primarily along the design level of quality drivers, secondly along the relation to either formal efficacy or resource requirements, and thirdly along absolute and relative characteristics. The classification applied is reflected in the respective *guiding questions* which provide a mental technique to comprehensively identify relevant quality drivers.

For each quality driver, Appendix C presents the respective content, assessment methods, quality criteria, quality predicates, and an example. Figure 8.6 summarizes the quality drivers shortly discussed in the following sections according to the typing described above.

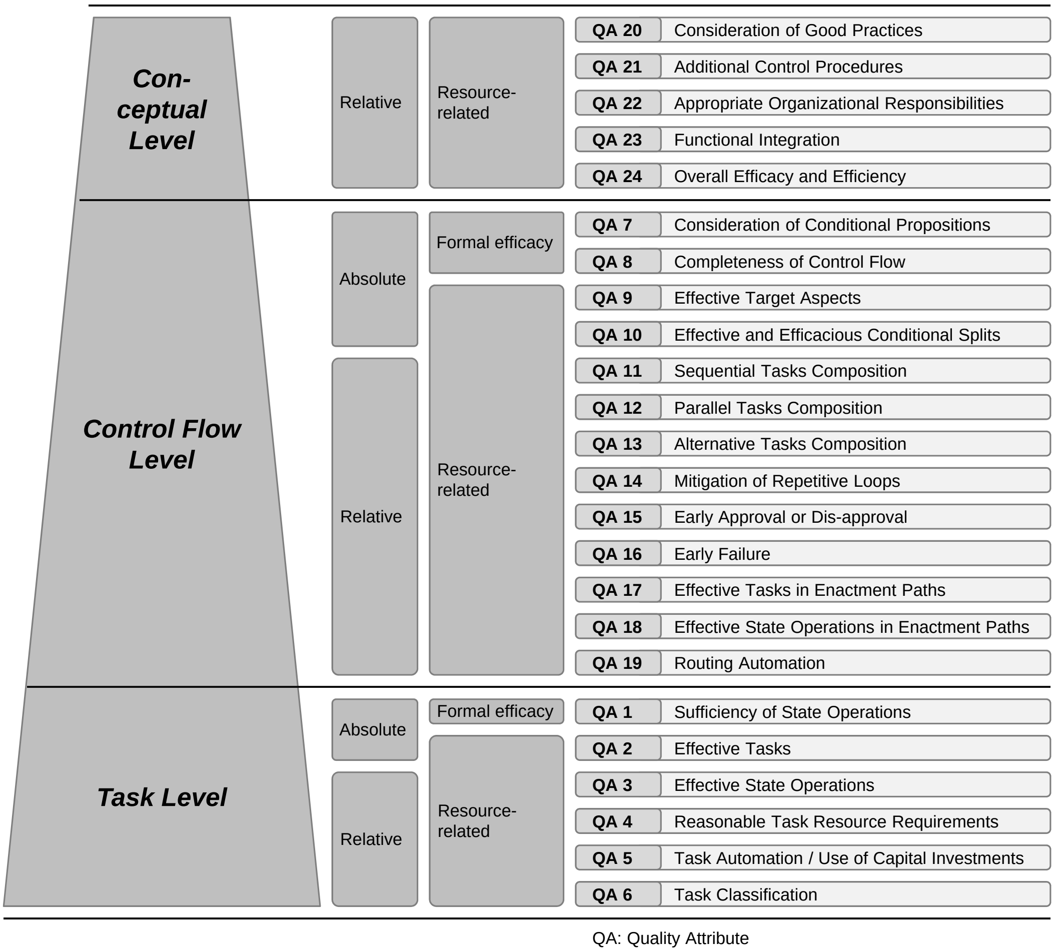



Figure 8.6: Summary: Quality Drivers

### 8.3.1 Task Level Quality Drivers

This section describes quality drivers which can be assessed by considering the set of tasks comprised in the business process without taking into account control flow.

**Guiding Question 1.** *On the level of individual tasks, which characteristics are relevant regardless of the application domain to achieve formal efficacy?*


As a requirement towards formal efficacy, the process must comprise a set of state operations sufficient to address all target BSDs given by the business objective model. Figure 8.7

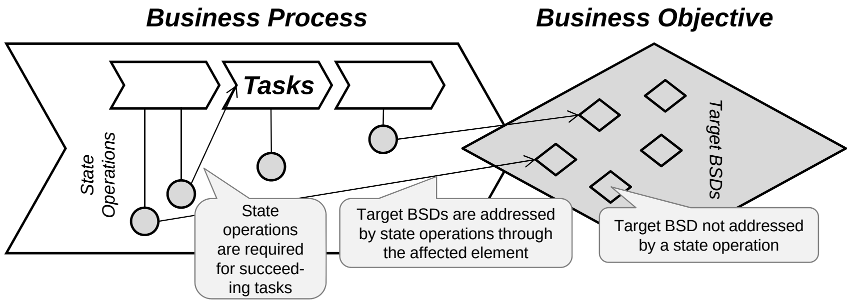



Figure 8.7: Sufficiency of State Operations

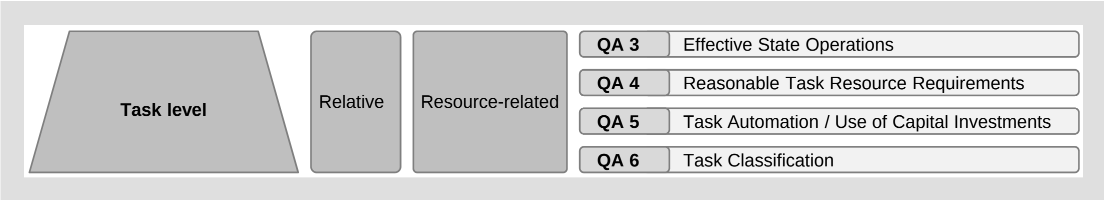
sketches a process and objective pairing where this is not the case: the business objective comprises BSDs which are not addressed by any state operation. The business process is thus not sufficient to fulfill its business objective on its own. Since both state operations and target BSDs can be formally specified (cf. Chapters 6 and 7, and Definition 13), assessment of the resulting ***Quality Attribute (QA) 1: Sufficiency of state operations*** may be fully automated.

Beyond that scope, additional quality drivers on the level of individual tasks require subject matter expert knowledge and must thus be assessed considering the application domain.

**Guiding Question 2.** *On the level of individual tasks, which characteristics are relevant regardless of the application domain to limit resource requirements?*


Each task comprised in a business process requires resources. Even if the task is fully automated, it is necessary to provide the corresponding capital investments by implementing, for example, a respective PAIS. Accordingly, there should be no tasks that are not required to further pursue control flow or to fulfill a target BSD. In other words, each task should comprise at least one state operation with an *affected* element that is an *affecting* element of the task-requisite BSDs of another task, or of the target BSDs of the business objective. Note that the resulting ***QA 2: Effective tasks*** does not yet consider possible sequences of tasks, which will be subject to control flow level quality drivers. In Figure 8.7, the quality driver is not fulfilled since the second task's state operation refers to neither a target BSD nor to another control flow element.

**Guiding Question 3.** *On the level of individual tasks, which characteristics are relevant considering the application domain to limit resource requirements?*

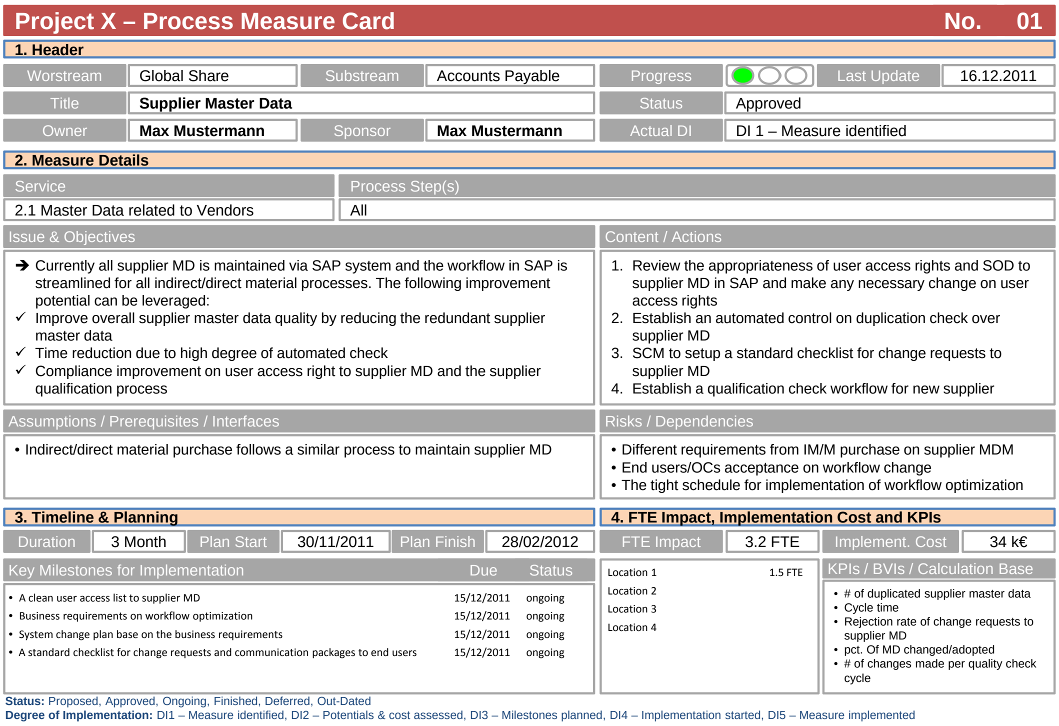



*QA 2: Effective tasks* demands that each task comprises at least one state operation required in the course of control flow or affecting a target BSD. Obviously, it is also desirable that each individual state operation not modeling resource consumption instead of just at least one per task fulfills this characteristic. However, since it may be assumed that each task consumes resources while there is no resource consumption assigned to individual state operations, subject matter experts' appraisal is required to determine if "superfluous" state operations raise resource requirements, and, accordingly, constitute a quality issue or just a technical matter. Therefore, ***QA 3: Effective state operations*** is included as a separate relative quality driver.

In addition, resource requirements associated with tasks can be appraised considering the respective desired outcome of each task. ***QA 4: Reasonable task resource requirements*** is fulfilled if the resources necessary to enact a task are deemed as appropriate by subject matter experts. Note that, by taking into account the desired results of tasks, this quality driver conflicts with the principle of not considering business objectives as a factor of BP quality. However, for reasons of practical relevance, this deviation is considered as acceptable.

On the level of individual tasks and considering subject matter expertise, the question arises whether potentials to automate processes have been utilized appropriately. ***QA 5: Task automation / use of capital investments*** refers to investing in technology to replace manual effort, which needs to take into account labor costs as well as the evolution of the IT system landscape. Thus, one-off capital expenditures to acquire hardware and software as well as implementation effort must be justified by recurring savings. In this case, recurring savings refer to eliminated manual effort valued on the basis of associated factor costs and diminished by ongoing maintenance for the automated solution. If this information is available, financial indicators such as the *net present value* of the measure can be calculated [235]. In practice, many enterprises demand that one-off implementation costs may not exceed three times annual savings. Implementation measures with regard to this quality driver are typically planned and tracked through so-called *measure cards*. Figure 8.8 depicts a practical example from a client project.

Resource requirements associated with a task are generally defined to satisfy all conceivable case variants that might occur for a task. This means that resource requirements are oriented at the most complex case variants. Accordingly, if case variants vary widely in terms of actual resource requirements (e.g., the level of qualification of involved staff), it may be appropriate to model separate tasks for case variants. Note that ***QA 6: Task classification*** requires to check for the actual case variant before selecting the appropriate task. The additional effort thus incurred must be justified according to subject matter experts' judgment. In addition, it needs to be taken into account that, in practical settings, the corresponding resources must possibly be kept available regardless of whether they are utilized or not, which might lead to reduced overall (i.e., cross-process) resource utilization.



**Project X – Process Measure Card** No. 01

**1. Header**

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| Worstream | Global Share | Substream | Accounts Payable | Progress | | Last Update | 16.12.2011 |
| Title | **Supplier Master Data** | | | Status | Approved | | |
| Owner | **Max Mustermann** | Sponsor | **Max Mustermann** | Actual DI | DI 1 – Measure identified | | |

**2. Measure Details**

| Service | Process Step(s) |
|---|---|
| 2.1 Master Data related to Vendors | All |

Issue & Objectives

➔ Currently all supplier MD is maintained via SAP system and the workflow in SAP is streamlined for all indirect/direct material processes. The following improvement potential can be leveraged:
✓ Improve overall supplier master data quality by reducing the redundant supplier master data
✓ Time reduction due to high degree of automated check
✓ Compliance improvement on user access right to supplier MD and the supplier qualification process

Content / Actions

1. Review the appropriateness of user access rights and SOD to supplier MD in SAP and make any necessary change on user access rights
2. Establish an automated control on duplication check over supplier MD
3. SCM to setup a standard checklist for change requests to supplier MD
4. Establish a qualification check workflow for new supplier

Assumptions / Prerequisites / Interfaces

- Indirect/direct material purchase follows a similar process to maintain supplier MD

Risks / Dependencies

- Different requirements from IM/M purchase on supplier MDM
- End users/OCs acceptance on workflow change
- The tight schedule for implementation of workflow optimization

**3. Timeline & Planning**

| Duration | 3 Month | Plan Start | 30/11/2011 | Plan Finish | 28/02/2012 |
|---|---|---|---|---|---|

| Key Milestones for Implementation | Due | Status |
|---|---|---|
| A clean user access list to supplier MD | 15/12/2011 | ongoing |
| Business requirements on workflow optimization | 15/12/2011 | ongoing |
| System change plan base on the business requirements | 15/12/2011 | ongoing |
| A standard checklist for change requests and communication packages to end users | 15/12/2011 | ongoing |

**4. FTE Impact, Implementation Cost and KPIs**

| FTE Impact | 3.2 FTE | Implement. Cost | 34 k€ |
|---|---|---|---|

| Location | FTE |
|---|---|
| Location 1 | 1.5 FTE |
| Location 2 | |
| Location 3 | |
| Location 4 | |

KPIs / BVIs / Calculation Base

- # of duplicated supplier master data
- Cycle time
- Rejection rate of change requests to supplier MD
- pct. Of MD changed/adopted
- # of changes made per quality check cycle

**Status:** Proposed, Approved, Ongoing, Finished, Deferred, Out-Dated
**Degree of Implementation:** DI1 – Measure identified, DI2 – Potentials & cost assessed, DI3 – Milestones planned, DI4 – Implementation started, DI5 – Measure implemented

Figure 8.8: Measure Card Example

### 8.3.2 Control Flow Level Quality Drivers

This section describes quality drivers which can be addressed by considering control flow spanning multiple tasks or gateways. In general, these quality drivers can be managed by re-arranging control flow.

**Guiding Question 4.** *On the level of control flow, which characteristics are relevant regardless of the application domain to achieve formal efficacy?*


***QA 7: Consideration of conditional propositions*** refers to the requirement that, to achieve formal efficacy, conditional propositions given by the business objective must be considered in corresponding control flow models. This means that state operations to fulfill fully determinate bivalent target BSDs may only occur when the preceding control flow has established, by way of task-requisite or branch-conditional BSDs, that the respective conditional BSDs are fulfilled. This characteristic can be formally evaluated (cf. Chapter 7), and is therefore not subject to the judgment of subject matter experts.

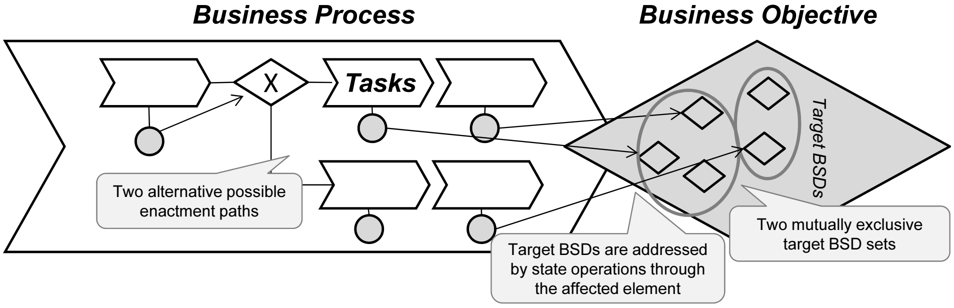
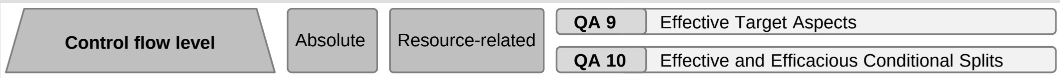



Figure 8.9: Completeness of Control Flow

An efficacious business process will ensure that all relevant target BSDs (according to the business objective) can be addressed. To reflect this issue, ***QA 8: Completeness of control flow*** pertains to the following requirement: For each set of target BSDs which are not mutually exclusive as defined by their respective conditional propositions, there must be at least one possible enactment path through the process model where all elements of the set are addressed. Figure 8.9 depicts a case where this characteristic is not given: in each mutually exclusive group of target BSDs, there is one BSD not addressed by the corresponding possible enactment path through the process model.

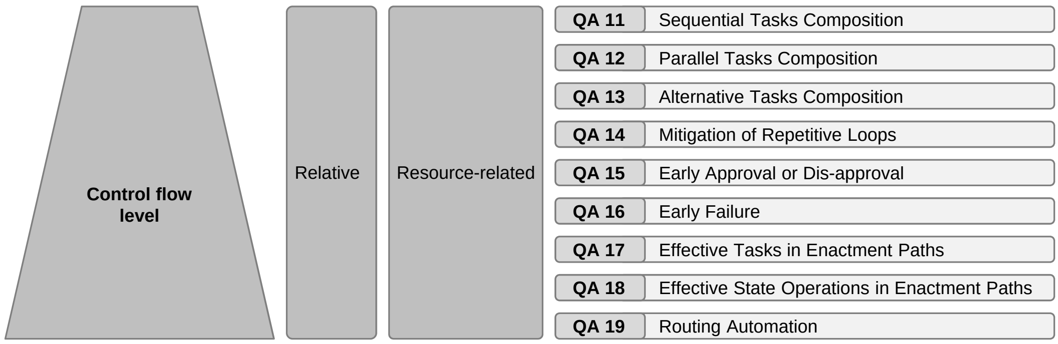
**Guiding Question 5.** *On the level of control flow, which characteristics are relevant regardless of the application domain to limit resource requirements?*




Partially determinate bivalent target BSDs and trivalent target BSDs reflect aspects of the business objective which *must* be fulfilled for particular conditional settings, and *may* be fulfilled for others. However, fulfilling target BSDs that are not strictly necessary to maintain efficacy will, in general, cause unnecessary resource requirements through the respective tasks. Accordingly, ***QA 9: Effective target aspects*** stipulates that control flow should ensure that target aspects are fulfilled only if required. Whether it is required to fulfill a target aspect, in turn, is defined by the business objective. However, note that if the corresponding task also addresses additional requirements of succeeding control flow elements, a respective target aspect cannot be considered as ineffective.

Conditional splits (e.g., OR or XOR split gateways [223]) generally require to check the associated conditions. According to the quality-aware process modeling approach (cf. Chapter 7), this checking action must be modeled as a "checking task" to capture the associated resource requirements. Thus, to properly limit resource requirements, only ***QA 10: Effective and efficacious conditional splits*** should occur. This is the case if each conditional split is required to properly reflect conditional propositions given by the business objective, or to check whether resources required in further control flow are available. In the latter case, the conditional split will initiate actions to mitigate resource shortages or “classified” tasks (cf. *QA 6: Task classification*).

**Guiding Question 6.** *On the level of control flow, which characteristics are relevant considering the application domain to limit resource requirements?*




Task-requisite BSDs model the technical resources necessary to enact a task. According to the quality-aware process modeling approach (cf. Chapter 7), tasks are enacted only if all task-requisite BSDs are fulfilled. This characteristic can be leveraged to avoid unnecessary resource consumption caused by process instances aborting due to unfulfilled task-requisite BSDs. Each pair of tasks joined by a sequence gateway [223] entails the risk that the first task can be completed under consumption of resources but without fulfilling a target aspect, while the second task aborts because task-requisite BSDs are not fulfilled. To mitigate this risk, ***QA 11: Sequential tasks composition*** demands that sequential pairings of tasks occur only if there is a reason. Examples for valid reasons include state operations of the preceding task that impact affecting elements of the subsequent task (this circumstance can be assessed regardless of the application domain) or differing responsible stakeholders (consideration of the application domain is required).

Parallelization is often used in process models to shorten cycle times if there is no requirement to enact tasks and gateways in a particular sequence (which is visible in quality-aware models through shared affecting and affected elements). However, similar to sequential tasks, parallelization bears the risk of resource waste if a parallel branch cannot be fully enacted due to unfulfilled resource requirements. In this case, the entire process instance cannot be completed beyond the parallel join gateway, which may render resources consumed in the other parallel branches irrelevant. Thus, ***QA 12: Parallel tasks composition*** should be employed in scenarios where, according to subject matter experts, parallelization is not required to fulfill cycle time restrictions.

As discussed with regard to *QA 10: Effective and efficacious conditional splits*, decisions modeled via split gateways in process models may be required to properly address target BSDs, i.e., to consider conditional propositions, or to selectively fulfill resource requirements comprised in the further course of process enactment. In the latter case, however, it is possible that the “checking action” associated with a decision causes resource requirements beyond the difference in resource requirements between the alternative paths. In that case, the decision in question should be replaced by an inclusive activity which is able to address all case variants. ***QA 13: Alternative activities composition*** thus requires subject matter experts to appraise the merits of decisions in the process model which are aimed at selectively fulfilling resource requirements.

Loops in process models (i.e., recurring activities in the sense of sub-processes comprising at least one task, but potentially multiple control flow elements) generally occur for three reasons. First, loops occur if the process must handle multiple uniform artifacts (e.g., line items in a purchasing request). In BPMN, this case is often modeled with the sub-process construct. Second, loops occur if the process incrementally creates resources required later on, or incrementally alters a singular target element. Third, loops occur if the process is modeled using a "trial-and-error" paradigm (e.g., forwarding issues to one possible contact partner after another until the responsible person is found). Clearly, the third case bears the risk of inducing unnecessary resources requirements. Thus, ***QA 14: Mitigation of repetitive loops*** addresses the question whether the latter kind of loop occurs, and whether sufficient mitigatory action has been considered to avoid resource waste as far as possible.

In practice, many processes expend significant effort to determine whether particular target aspects (i.e., bivalent target BSDs) must be fulfilled for a certain instance or not. For these "knock-out processes" [210], a strategy of ***QA 15: Early approval or dis-approval*** should be pursued. To achieve this, the respective checking actions (cf. *QA 10: Effective and efficacious conditional splits*) should be arranged in a way to reach decisions regarding individual target aspects with resource requirements as limited as possible. To this end, the relative probability of individual checking action outcomes as well as the respective resource requirements must be considered by subject matter experts on the basis of the business objective model (cf. Chapter 6). Proper consideration of this quality driver will, for example, result in an ordering of tasks that checks basic necessary conditions to approve a document as early as possible before entering into more elaborate (and resource-intensive) appraisal.

Similar to *QA 15: Early approval or dis-approval*, processes that bear the risk to fail before target BSDs can be fulfilled should be arranged in a way to promote ***QA 16: Early failure***. This quality driver will contain resource requirements incurred for incomplete process instances as much as possible. Accordingly, activities with a high probability of failure (e.g., because resource requirements cannot be fulfilled) and with a low amount of resource requirements should occur first in a "risky" process model. Figure 8.10 exemplifies a process with improvement potentials regarding this quality driver: The one "risky" task occurs straight before the final task addresses target aspects. Assuming that the sequence of the first three tasks is interchangeable, this would constitute a quality issue.

*QA 2: Effective tasks* demands that all tasks comprise at least one affected element which is an affecting element of another control flow element, or of a target BSD. By taking into account control flow, this requirement can be further refined: within possible enactment paths, each task should affect a *subsequent* control flow element or a target BSD. In addition, each possible enactment path should terminate with a task potentially fulfilling a target BSD. For example, resource requirements that are relevant only in a particular branch after a conditional split gateway should be fulfilled as part of the branch, but not before. Note that the resulting ***QA 17: Effective tasks in enactment paths*** pertains to the quality of control flow in the sense of an arrangement of tasks, but not the quality of individual tasks (the latter issue has been addressed in *QA 2: Effective tasks* already).

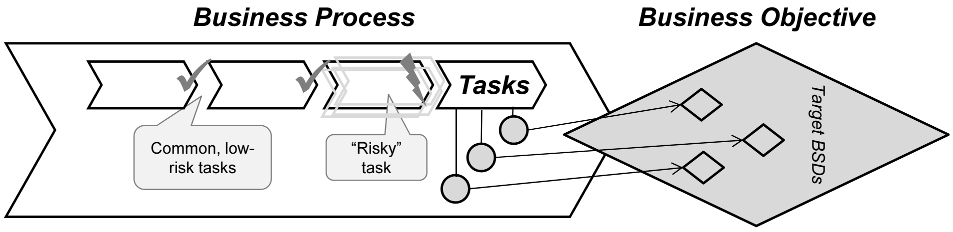



Figure 8.10: Early Failure

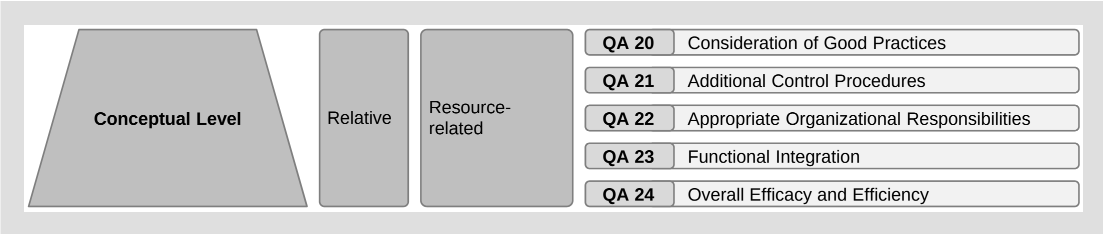
Similar to *QA 2: Effective tasks* in comparison to *QA 3: Effective state operations*, it is possible to further tighten the stipulation of *QA 17: Effective tasks in enactment paths*. To maintain the *Cost Effectiveness* criterion (cf. Section 3.1), assessment of the ***QA 18: Effective state operations in enactment paths*** quality driver should discard topics already addressed in preceding quality attributes. Note that this also excludes reflective state operations modeling, e.g., resource consumption as defined in Definition 11. Accordingly, *QA 18: Effective state operations in enactment paths* only considers *contradictory or redundant* state operations. This may be the case if, in a possible enactment path, an affected element (which is not modeling resource consumption) occurs multiple times. Judgment by subject matter experts is required to exclude technical reasons.

To avoid unnecessary resource requirements, one must also consider the routing capabilities provided by today's WfMSs. This means that ***QA 19: Routing automation*** should be employed to automatically evaluate conditions attached to split gateways, or to pass information elements to staff involved in process enactment. Subject matter expert appraisal is required to evaluate whether additional implementation costs are justified considering transactional volume and possible reductions in manual effort per process instance.

### 8.3.3 Conceptual Level Quality Drivers

This section describes quality drivers that need to be assessed on the basis of the business process as a whole. In other words, it discusses conceptual characteristics which cannot be improved by, for example, altering individual tasks or re-arranging control flow. Rather, it addresses requirements towards fundamental changes to the underlying idea of a business process. Accordingly, *conceptual level quality drivers* require close examination of the respective business objective, fundamentally inquiring whether there might be an overall process design more apt to achieve the objective at hand.

**Guiding Question 7.** *On the conceptual level, which characteristics are relevant considering the application domain to limit resource requirements?*




For many application domains, collections of good practices or process patterns are available representing the "state of the art" for the respective field. These might address a business objective as a whole or just individual processing aspects, and good practices may be documented explicitly (e.g., the IT Infrastructure Library in the information management domain [118]) or available as organizational knowledge of subject matter experts [236]. ***QA 20: Consideration of good practices*** available for the application domain of the process in question constitutes a quality driver to be assessed by knowledgeable subject matter experts.

On the control flow level, *QA 15: Early approval or dis-approval* and *QA 16: Early failure* pertain to the arrangement of activities to avoid resource waste caused by enacting activities that prove as irrelevant later on. On the conceptual level, ***QA 21: Additional control procedures*** should be addressed as well. When considering an overall process, it may also be reasonable to not only rearrange activities, but to introduce entirely new "early controls" to prevent undesired evolution in a process instance. In this respect and in many practical examples, a late "overall check" can be amended by "early controls" appraising individual aspects. For example, if final approval of payment runs (cf. Sample Process C) often fails due to insufficient bank account balances, it may make sense to include an additional control covering this aspect early in the process.

When defining business processes, designers must balance between the "case handling" approach [129] which tries to limit interface issues by having entire process instances handled by one responsible person, the need to employ specialists for particular tasks, and the desire to limit factor cost by aligning the qualification level of personnel to the task at hand (cf. *QA 6: Task classification*). Note that the results of this balancing are often referred to as "economies of scope" by practitioners. Whether the resulting ***QA 22: Appropriate organizational responsibilities*** has been considered must be evaluated by subject matter experts.

In particular when transgressing the consideration of particular process instances (as reflected in common process modeling) or even business processes, the issue of realizing scale effects in the enactment of tasks must be taken into account. ***QA 23: Functional integration*** pertains to bundling or stacking uniform activities (or individual tasks), thus following a principle of functional integration instead of process integration, e.g., to minimize changeover cost [237].[6] Implementing bundling of uniform activities in process models can be achieved by integrating appropriate triggering events as the precedents of respective activities in process models [80]. While this paradigm lies in stark contrast to the BPM paradigm of process integration, it nevertheless needs to be applied to process models by subject matter experts to identify possible improvement potentials.

[6]The underlying fundamental principle of differentiation of labor has been described by Adam Smith in his famous pin factory example [75].

Even if all quality drivers have been applied, it is still possible that a business process as a whole appears as unreasonable or economically not viable to subject matter experts. This reflects the consideration that, in practical application scenarios, not all aspects relevant to BP quality in a certain domain can be generalized (cf. the concept of “bounded rationality” [115]). Therefore, the final quality driver, ***QA 24: Overall efficacy and efficiency***, pertains to the ultimate comparison of resource requirements to the desired outcome of a business process. It constitutes an additional control which is enabled through quality-aware BP modeling and the derivation of quality relations (cf. Chapter 7 and Figure 7.1). In practice, the results of quality assessment will be presented to responsible stakeholders together with consolidated quality relations, and will be subject to final discussion.

## 8.4 Quality Meters

Quality meters are characteristics relating the affecting and the affected environment of a business process which allow drawing conclusions regarding its quality. According to Figure 8.4, they are particularly relevant to assess BP quality in the enactment lifecycle phase or *enactment performance*. Quality drivers, on the other hand, are based on inspecting BP models, which naturally assigns them to the design & implementation lifecycle phase. Besides this role, quality meters can also be used to capture parts of the quality-relevant impact of a business process on its outer environment which already occur during design & implementation, namely with regard to capital investments.

### 8.4.1 Appropriate Capital Investments as a Design & Implementation Quality Meter

According to the considerations lined out in Figure 8.4, capital investments during the design & implementation lifecycle phase are an issue to be addressed through quality meters.

Figure 8.11 depicts relations between capital investments, manual enactment effort, transactional volume, cost per instance, and total cost of a business process. Analyzing these relations for a business process will result in an optimum (“balanced”) level of capital investment depending on the transactional volume in the sense of the number of instances occurring in a given timeframe as well as investment potentials for the process in question. As a general rule, capital investments are used to reduce the amount of manual work required to enact a business process by either automating tasks or control flow entirely, or at least supporting their manual enactment. Thus, for a given transactional volume, capital expenditure (i.e., the cost of capital investments) per process instance will grow in a linear manner while the cost for manual effort incurred will decrease ever more slowly. This reflects the characteristic that possible automation measures differ in terms of their effectiveness regarding the manual effort reductions achieved, and organizations will implement the most effective measures first. Moreover, for a given amount of investment, manual effort incurred per process instance will slightly decrease with growing transactional volume since personnel resources are better utilized. Accordingly, the total cost of manual effort in process enactment will, by a small measure, not grow proportionally to transactional volume. This phenomenon is commonly referred to as “economies of scale” [238]. Accordingly, the amount

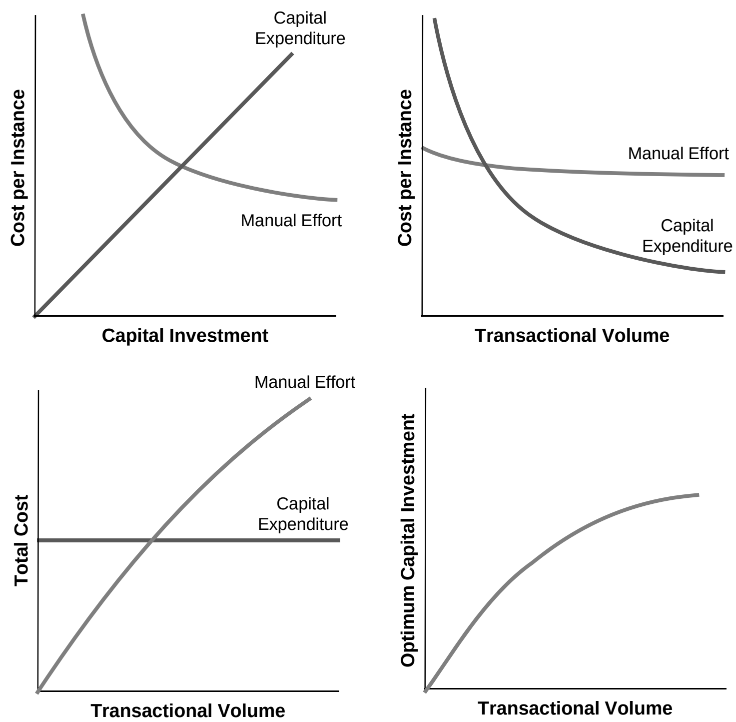



Figure 8.11: Considerations on Capital Investments

of ***QA 25: Appropriate capital investments*** in implementing a business process will grow with transactional volume, but will be subject to a boundary.

## 8.4.2 Enactment Quality Meters

According to Figure 8.4, quality meters are particularly relevant with regard to *BP enactment quality* or *process performance*. As discussed in Chapter 5, enactment performance takes into account not only the process model and its implementation *per se* – which is addressed as design quality –, but also the quality of human effort involved in the process. In principle, this is a deviation from the paradigm to delineate process quality from the quality of the affected environment, since human effort involved might be perceived as a resource to the process. However, fully applying this paradigm would entail treating BP quality as a matter of design & implementation only. This approach, in turn, would violate the underlying purpose of the BP quality concept as a means of managerial analysis and control, in particular with respect to Effectiveness Criteria 1 and 2, *Congruence to organizational targets* and *Transparency and retraceability*. To manage these topics, the appraisal of the performance of process operations, i.e., of the effort of involved staff and managers, needs to be considered as well. Due to that reason, the quality model proposed in this thesis includes quality meters with regard to the enactment lifecycle stage besides quality drivers considering the design & implementation lifecycle stage.

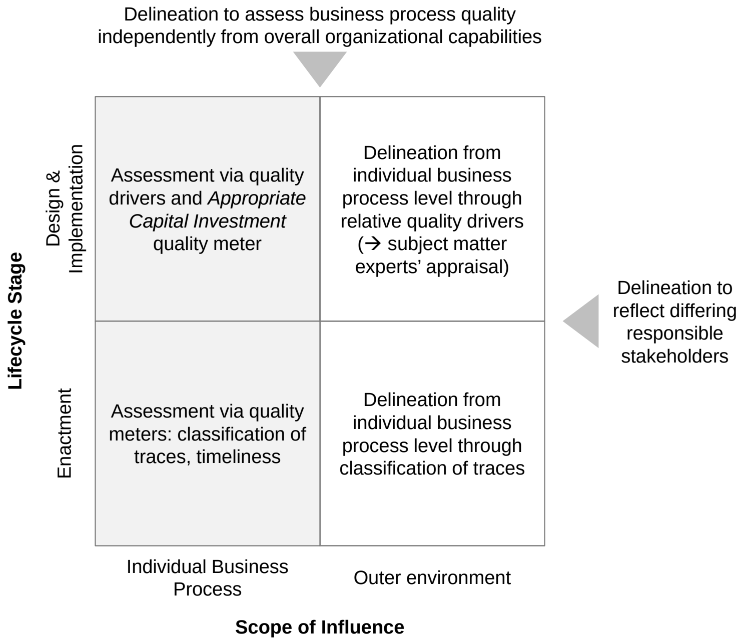



Figure 8.12: BP Quality Delineation Requirements

As discussed in Section 5.2, it is instrumental to delineate enactment performance from design quality. In addition, it is necessary to delineate quality assessment according to the scope of influence within the organization: the impact of design & implementation as well as enactment of the particular business process – and the stakeholders involved – is to be delimited against the impact of the outer environment of the business process which reflects wider organizational capabilities, such as the availability of resources. During the enactment lifecycle phase, this means that the performance of the business process in question must be delineated from from the performance of other processes within a process chain. Figure 8.12 summarizes the requirements in this respect, and provides an outlook on corresponding strategies further detailed in the quality meters described in this section. In other words, *individual process performance* must be delineated from *end-to-end process performance*. In particular, this applies to preceding processes where process output (i.e., target elements) is used as process input (i.e., affecting elements) for the process to be assessed. If these aspects were not considered properly, *Transparency and retraceability* would be severely impaired: quality attributes would not reflect the scope of influence of individual managers and staff, and could therefore not be used to address individual performance. This would impede the utility of the quality model as a means of organizational (i.e., behavioral) control.

Considering the focus of this thesis on transactional G&A processes, and in addition to the considerations above, the practical measurability of factors relevant to BP quality during the enactment phase is limited. In practice, it is generally not possible to fully appraise the actual impact of BP enactment on target artifacts and resources consumed. There might even be even legal limitations to what can be measured in typical BP settings.[7] Rather, performance appraisal must rely on "stand-in" measures based on available measures in typical WfMSs or enterprise resource planning (ERP) systems , and on sampling methods. The first two quality meters presented in this section address the former option, and the third meter addresses the latter topic.

In the context of WfMS or ERP system based quality meters, the tracking of the possible enactment paths (cf. Section 7.5) which have actually occurred during process enactment is of particular relevance. In the following, these are denoted as *traces* [155]. Traces allow delineating process performance from process design quality:

- The set of *possible enactment paths* including the respective impact on efficacy and efficiency considerations constitute a matter of process design quality. Hence, an effective appraisal of enactment performance will not be based on measuring the actual impact on target artifacts and resources. However, each possible enactment path can be appraised to determine whether it is *desirable* considering the respective fulfillment of target BSDs, and the related consumption of resources – in other words, considering efficacy and efficiency aspects.

- The set of actually occurring *traces* comprises one element of the set of possible enactment paths for each process instance. The "selection" of elements reflects the availability and the state of resources as well as the effects of human effort incurred during enactment of the instance in question. It thus reflects end-to-end process performance. High and poor process performance will entail desirable and undesirable traces, respectively. Accordingly, this aspect provides a basis to evaluate enactment performance which abstracts from the overall impact on target artifacts and resources. Hence, delineation between design quality and enactment performance is achieved.

Figure 8.13 summarizes the resulting two stages required to appraise ***QA 26: Efficacious and efficient enactment performance***. In a practical setting, process mining tools [28] provide a means to consolidate log data gained, e.g., from ERP systems [240] or dedicated WfMSs into information on the frequency of traces. This information is commonly enriched with data on cycle times. Figure 8.14 exhibits an example of an analysis on the frequency of traces taken from an industry project. A more detailed example of analyzing actual process enactment data is provided in the experience report described in Chapter 12.

The timeliness of fulfilled target BSDs, i.e., the timely availability of process results, constitutes an important matter in practice. In this context, it is important to understand that in transactional G&A processes, which constitute the major focus of this thesis (cf. Section 1.3), it is generally not an objective to enact process instances as fast as possible. Minimizing cycle times would conflict with the strategic goal to utilize available capacities (e.g., employees or tools and machinery) evenly, since utilization peaks to be accommodated entail corresponding "surplus" capacities to be held available. Rather, timeliness in terms of process results generally does not refer to having target BSDs fulfilled as fast as possible, but within a certain timeframe, as illustrated in Example 47.

[7] In Germany, for instance, automated evaluation of the individual work performance of employees is subject to co-determination with the workers' council [239].

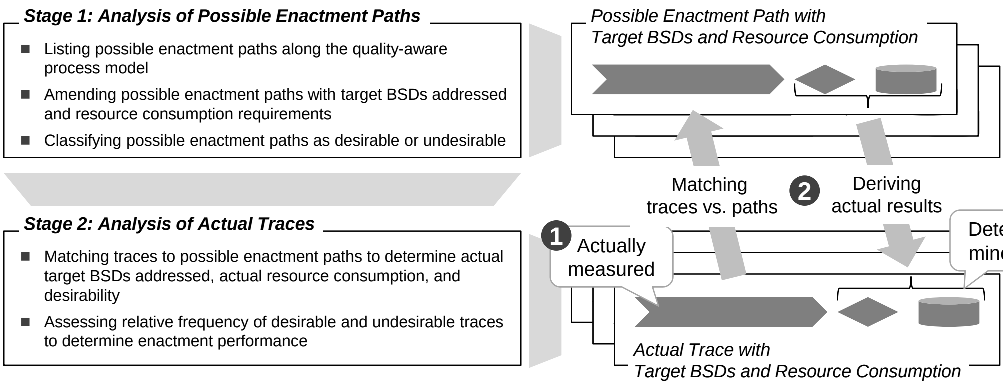



Figure 8.13: Efficacious and Efficient Enactment Performance: Assessment Stages

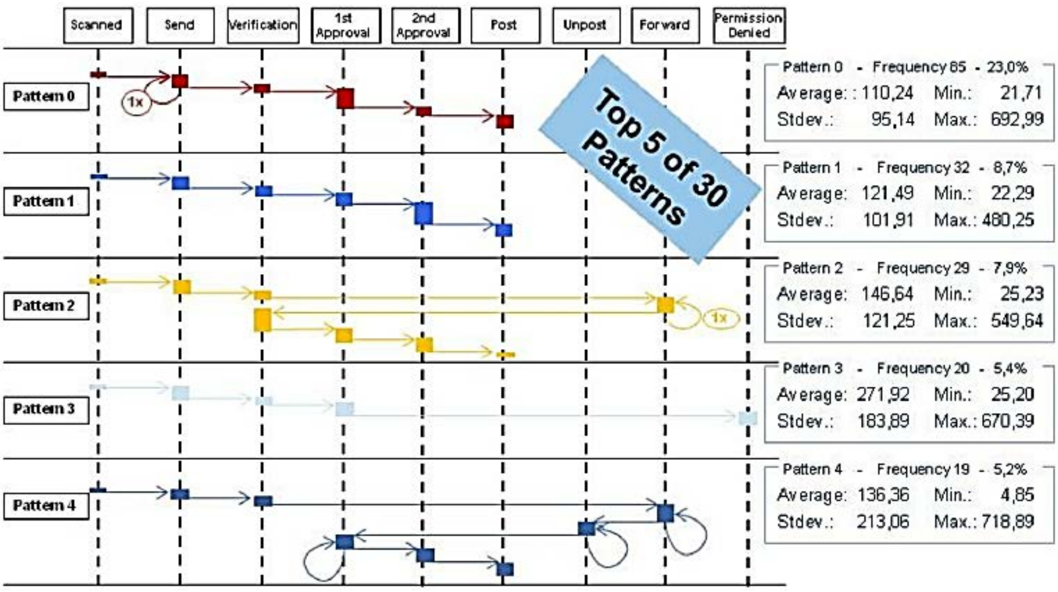



Figure 8.14: Industry Example: Frequency of Traces

**Example 47 (Timely Availability of Process Results).** Consider Sample Process A from Figure 2.5. The process relates to accepting or declining supplier invoices. In this context, an organization might aim at handling invoices within a timeframe that might be given by firm policy or payment terms. Compliance with the timeframe is important since cash discounts can be realized if payments are on time. However, approving or declining invoices earlier than required by the given timeframe will not provide additional benefits.

Similarly, in the subsequent business process of issuing payments (cf. Figure 2.6), the payment list needs to be approved in time for the payment run. Earlier approval, however, will not provide additional benefits to the organization.

Despite its apparent relevance, the issue of timeliness has been considered neither in business objective modeling (cf. Chapter 6) or formal efficacy considerations, nor in quality drivers related to design quality. The reason behind this is that defects with regard to the timely fulfillment of target BSDs are usually not an issue of actual process design, but caused by faulty process enactment, either in upstream processes or with respect to the process at hand. This becomes clearer when considering how actual enactment times, or cycle times, of process instances are determined:

- On the task level (cf. Section 8.2.5), the cycle time between a control flow element being enabled and terminated [22]) is subject to a lower boundary reflecting the required amount of manual effort. This is measured in time units, assuming that a single control flow element requiring manual effort is enacted by one person at a time. In principle, the control flow element can be completed as soon as the required resources including, if necessary, the person to enact the task are available. Cycle time is thus determined by the availability of resources.
- On the control flow level, the cycle time of a trace is determined by the cycle time of all control flow elements comprised. In case of parallel branches, the maximum cycle time of all branches is relevant. Process design can thus influence cycle times by altering possible enactment paths which are reflected in the corresponding traces during process enactment. Reducing cycle times through process design is usually achieved by automating tasks (cf. the quality driver *QA 5: Task automation / use of capital investments*), by automating control flow (cf. the quality driver *QA 19: Routing automation*), or by parallelizing activities. In this regard, individual tasks might even be split to allow multiple employees to work on the respective topic in parallel.
- During process enactment, the selection of a particular possible enactment path resulting in the actual trace is a determinant of cycle time. As discussed above (cf. Figure 8.13), this reflects the availability of resources as well as the quality of manual effort. In practical settings, repetitive loops are of particular relevance in this respect (cf. the *QA 14: Mitigation of repetitive loops* quality driver).

In practical settings, end-to-end process chains as well as external restrictions such as payment terms (cf. Example 47) are designed to accommodate cycle times enabled by the design of particular processes. Thus, defective cycle times are usually caused by defects in resource availability, which may be caused by upstream processes, or in manual effort during process enactment, but do not reflect a general issue of faulty process design. In this context, empirical studies have found that there is no general optimum strategy to minimize cycle times as soon as huge transactional volumes with a corresponding workload on resources employed are considered [241]. This "root cause analysis" is to be taken into account when assessing ***QA 27: Timely process enactment*** as a quality meter.

In terms of quality attributes reflecting enactment performance, both *QA 26: Efficacious and efficient enactment performance* and *QA 27: Timely process enactment* relate to topics which can be addressed by analyzing available enactment log data with the support of process mining [28] or process intelligence [242] tools. This approach corresponds to the observation made with regard to possible strategies to cover BP quality aspects (cf. Figure 8.4): Quality meters should be defined keeping in mind the restrictions of practically available means of enactment-related indicators.

Automated means of capturing enactment performance data, however, are restricted to the occurrence of control flow elements and to cycle times. This information is usually represented by log entries and the corresponding timestamps in enactment logs provided by WfMSs and ERP systems. This restricted source of information will not enable capturing the quality of BP output in the sense of the efficacy of process enactment:

- Are target BSDs actually fulfilled if the respective tasks have been carried out?
- Are conditional propositions related to target BSDs actually considered if the respective possible enactment paths have been selected as actual traces?

The former issue is relevant only in case of target BSDs not being fulfilled by entering information into an IT system. In this case, the fulfillment state of target BSDs cannot be traced automatically, but may still be confirmed by using appropriate statistical sampling methods and manually examining process instances and target artifacts [243]. The latter topic is further illustrated in Example 48.

**Example 48 (Target BSD Error Types).** Consider Sample Process C from Figure 2.7. In the related field of medical treatments, the impact of errors surely is of grave importance. On the other hand, the risk of clinical procedures failing is subject to rigorous management. For instance, in the sample process, drugs are administered based on the results of conditions determined on the basis of laboratory examinations.

In comparison, Sample Process A (cf. Figure 2.5) exhibits less serious effects in case of errors in the fulfillment of target BSDs: falsely declined invoices will lead to complaints from the supplier, and falsely approved invoices will lead to overpayments which can, however, be settled with future deliveries. On the other hand, the sample process is based on manual approval or disapproval of documents which might lead to a comparatively high risk of errors.

To address these highly relevant topics while maintaining manageable requirements with regard to the analysis effort required (cf. Effectiveness Criterion 3: *Cost Effectiveness*), the resulting ***QA 28: Trace deviation errors*** should be employed considering the actual level of risk and the economic impact of errors in process enactment.[8] Accordingly, process

[8]In the field of financial auditing, similar considerations have led to the development of the risk-oriented audit approach [244].

managers should evaluate whether the risk, the respective impact, and the possibility of mitigatory measures call for sample assessment of process enactment results.

In summary, the three quality meters presented in this section allow appraising enactment quality with proper delineation from both design quality and the quality of upstream processes responsible to provide required resources. Available means in typical WfMSs and ERP systems have been taken into account through the concept of analyzing traces in comparison to possible enactment paths while utilizing timestamp data. Appropriate management of manual effort in the sense of Effectiveness Criterion 3, *Cost effectiveness*, can be achieved by assessing *QA 28: Trace deviation errors* on the basis of error risks and error impact.

## 8.5 Conclusion

The quality attributes presented in this chapter reflect the quality of an explicit or implicit process model and its implementation in terms of PAISs or other capital goods and organizational capabilities (i.e., *design quality*), as well as the quality of its execution in terms of the resulting process instances (i.e., *enactment performance*). As lined out in Chapter 3, the approach to identify valid quality attributes pursued in this thesis is based on *deriving* quality attributes from a general definition of BP quality, the scope of influence a business process governs, a quality aware BP meta-model and, ultimately, the support of business processes towards organizational targets. This approach helps to achieve a reasonable level of assurance with respect to comprehensive coverage of the quality model as demanded in Effectiveness Criterion 1, *Congruence to organizational targets*. Exclusive coverage as the second aspect of Effectiveness Criterion 1 is addressed through the delineation between design quality and enactment performance. This means that for each role in BPM, whether in process design & implementation or in process enactment, only factors that can be influenced are considered in quality assessment.

The secondary Effectiveness Criteria 2 and 3, *Transparency and retraceability* and *Cost effectiveness*, pertain to the ability to formalize elements of the quality model. In conjunction with the concept of quality-aware BP modeling (cf. Chapter 7), the model comprises quality attributes which can be fully formalized as well as quality attributes where formalizable aspects enable parts of the assessment process. For each quality attribute, the respective appraisal method is discussed in Appendix C.

# 9 Quality-aware Business Process Management: Procedures and System Landscapes

This chapter deals with management challenges in conjunction with implementing and utilizing design quality and enactment performance management concepts in organizations. To this end, Section 9.1 discusses relevant preliminary considerations. These pertain to the organizational environment of business processes typically in scope of BPM. These considerations result in *governance challenges* to be addressed by BPM in organizations. Section 9.2 provides an overview on corresponding approaches generally found in organizations today. In contrast to Chapter 4, its focus lies on methods and tools encountered in practice instead of related scientific research. Section 9.3 then discusses an extended quality-aware BPM lifecycle considering the common organizational environment as well as resulting challenges for the scope of this thesis. Beyond the organizational implementation of BP quality management, Section 9.4 addresses requirements towards a quality-aware BPM system landscape. It discusses relevant components including their interaction as well as functional demands posed by the concepts developed in this thesis. Section 9.5 concludes the chapter with a discussion of results achieved.

## 9.1 Organizational Environment

G&A processes constitute the main scope of this thesis (cf. Section 2.1). Therefore, the respective organizational environment is particularly relevant. It is common practice in organizations today to host G&A processes in dedicated *service functions* which may be implemented as a *shared services organization (SSO)* or even outsourced to a third party provider [245]. Note that the term "service function" in this context does not pertain to the provision of services to external customers [246]. Instead, it refers to G&A service processes internal to the organization as part of their value chain (cf. Figure 2.2) [57]. In this context, it is instrumental to keep in mind that the process scope commonly addressed in BPM corresponds well to the process scope typically implemented in today's SSOs. SSOs are mostly focused on executing standardized business processes based on formal interaction with service clients (SCs) . Figure 9.1 illustrates the underlying "activity split" principles used in SSO implementation projects. Note that the activity split is based on two dimensions of process characteristics. Process complexity reflects the following issues:

- *Transactional processes* entail low complexity. In practice, the following characteristics are considered as indicators for transactional processes:[1]
  - *Formalized* processes are explicitly defined, e.g. through a corresponding BP model or through explicit guidelines for involved personnel.

[1]Note that, in this context, the term "transactional" is not used in the sense usually encountered in computer science [247].

  - *Repetitive* processes entail a substantial volume of individual process instances within a certain timeframe.
  - *Uniform* processes exhibit consistent process instances. This relates to the types of environmental objects addressed, procedure or workflow, and involved parties.
  - *Atomic* processes exhibit no interference between individual process instances except the use of shared resources such as personnel or PAISs.[2]

- A high degree of *process structuring* (e.g., automated processes) entails low complexity.
- *Effectiveness-driven processes*, i.e., processes where the organization considers the quality of process results as more important than process cost (e.g., legal proceedings or negotiations with suppliers), entail high complexity.
- *Extensive communication requirements* and informal interfaces to other organizational functions entail high complexity.
- A high degree of *required staff qualification* implies high complexity.

The degree of process harmonization reflects the following characteristics to be compared between, e.g., business units, plants or legal entities within the group:

- *Common interfaces* to operations processes (e.g., procurement, logistics) entail a high degree of process harmonization.
- *Common process results or output* (considering statutory requirements) imply a high degree of process harmonization.
- *Process standardization*, e.g. via a common ERP system, implies a high degree of process harmonization.
- *Central process control and governance*, e.g. via an end-to-end process ownership concept [249], implies a high degree of process harmonization.

Note that the term "CxO organizations" refers to group, regional, country, or business unit (BU) headquarter departments subordinate to the group chief financial officer (CFO) , chief information officer (CIO) etc. As presented in Figure 9.1, the term "shared services" mostly implies a focus on reducing factor costs[3], for example by relocating processes to offshore sites. To the considerations of this thesis, however, the challenge of reducing factor costs is of minor relevance. Rather, BP quality in the sense of efficacy and efficiency relates to reducing factor *needs*. Therefore, the following sections refer to *service organizations (SOs)* instead of SSOs, assuming SOs as functions within an organization dedicated to enact G&A processes which are suitable for BPM implementation as discussed above.

To analyze the management challenges that arise in this context, the initial BP lifecycle model presented in Figure 1.1 can be extended to range from *process design & implementation* over *process enactment* to *process performance measurement*, *services charging*, and *process control* as summarized in Figure 9.2. In this context, organizational responsibilities for process design & implementation on the one hand, and process enactment on the other hand alternate between SOs and SCs.

[2] Again, note that the term "atomicity" is not used in the sense of "it either happens or it does not" [248] commonly applied in computer science.

[3] Factor costs are the costs of providing "production factors" required for manufacturing or service delivery, i.e., labor, capital goods, and supplies.

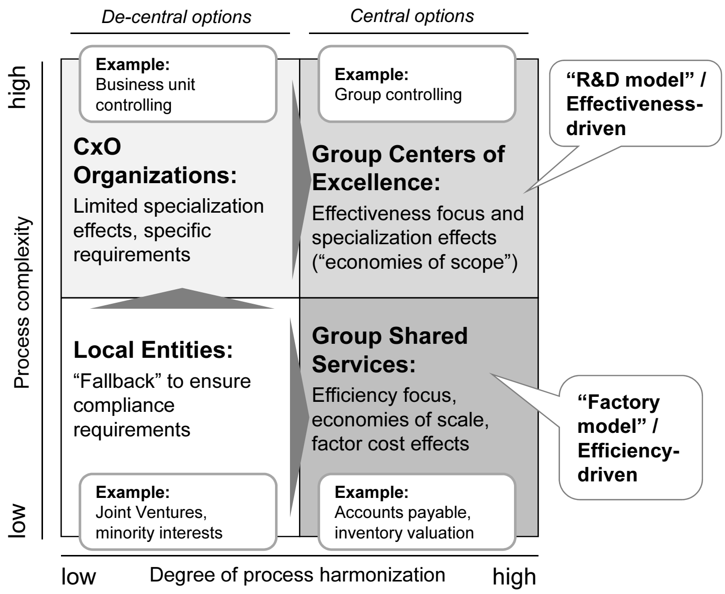



Figure 9.1: Activity Split Example

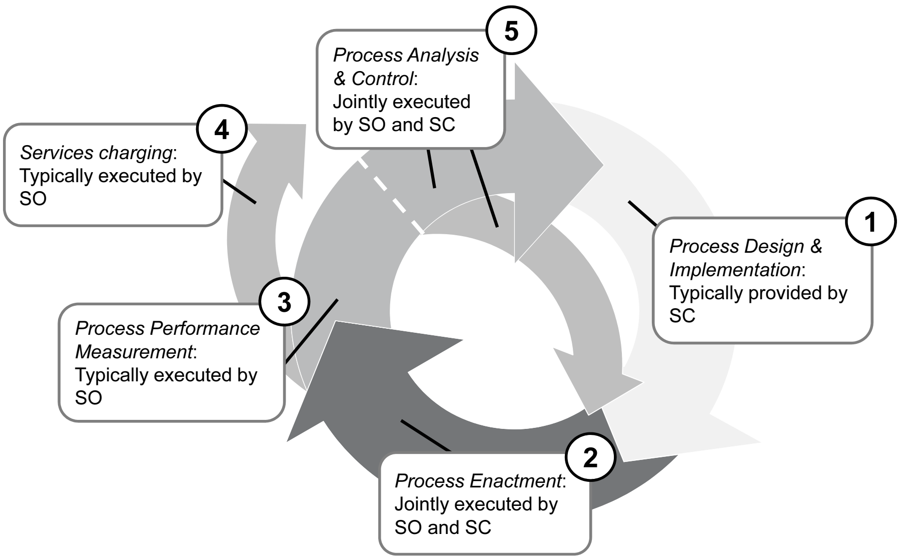



Figure 9.2: Extended BP Lifecycle for SOs and SCs

Accordingly, SOs assume joint responsibility for the BP lifecycle together with their customers. Along the BP lifecycle stages, these can be described as follows:

- *PAISs or other assets* reflecting a formal or informal underlying process model are typically provided by SCs. Accordingly, responsibility[4] for *process design & implementation* mostly lies with the SC.
- *Process enactment* is generally executed jointly by the SO and its SCs in the sense of an end-to-end process. Typically, the SO relies on its SCs regarding required process input such as making data available, taking decisions, or approving results.
- *Process performance measurement* is mostly executed by the SO by assessing performance indicators such as cycle times. Results are, however, provided to SCs as well, for instance, in a so-called *dashboard* or *management cockpit* tool [250].
- *Service charging*, i.e., the allocation of service costs in management accounting, is typically executed by the SO in reconciliation with SCs. Charging may be based on budgetary indicators (e.g., cost plus 6%[5], allocated by revenue), or on transactional volumes. In many cases, budgets or volume prices are re-negotiated on an annual basis.
- *Process analysis & control* is executed jointly by the SO and its SCs, generally on the basis of process performance measurement results: As described above, feedback into process design and implementation should be executed by SCs since they govern, for instance, the respective IT systems. Feedback into process enactment must be executed by both parties for the respective work share. Note that constructs to measure process quality and performance constitute tools to be applied in this lifecycle stage.

Example 49 illustrates SO and SC work shares for a sample process.

**Example 49 (SO and SC Work Shares).** Consider Sample Process A, the management of incoming invoices as described in Figure 2.5. Scanning invoices, matching them against purchase orders and goods receipts, obtaining invoice approvals and managing escalation in case of differences constitute typical tasks for shared services.

Regarding process design & implementation, the execution of the related activities relies on the BP design implemented in SCs' logistics and accounting systems. Applying a shared service centers' own workflow system is still an exception in this standard situation. Process enactment is executed jointly: the SO relies on procurement and materials management data provided by its customers, who are also responsible to approve invoices if required. Accordingly, performance measurement results, e.g., the number of invoices managed in comparison to personnel resources available, will reflect the performance of both parties involved. Obtaining SO performance measures not biased by client performance is particularly difficult in this fairly typical setting.

On that basis, it is difficult to consider, for instance, the proper availability of supplier master data for payment terms in service charging, although this will be a major determinant of

[4]Note that responsibility for design and implementation of information systems generally lies with the IT function, while accountability in terms of requirements definition, testing and final approval lies with technical functions (e.g., human resources management). For reasons of simplicity, only the term "responsibility" will be referred to in the context of this chapter.

[5]In an international environment, a profit markup is required to comply with tax regulations.

effort incurred. Likewise, the shared services options to execute process analysis & control to foster improvement of master data availability are limited.

The shared responsibilities described pose particular *governance challenges* which apply to the entire BP lifecycle.

*Governance challenge I* pertains to the SO's responsibility to enact activities within process without being responsible for process design and implementation, which is typically governed by SCs. This may lead to issues with regard to the optimization of process design and implementation, services charging, and the execution of process control.

*Governance challenge II* pertains to the lack of end-to-end process responsibility. SSOs generally rely on process input provided by their SCs. End-to-end optimization will require driving each activity in an end-to-end process towards overall optimization. This is difficult without an overarching governance role. Moreover, proper process performance measurement, services charging, and process control may be impeded.

Figure 9.3 summarizes the governance challenges while Table 9.1 discusses relevant issues per lifecycle phase. Note that, in principle, both governance challenges exist not only in SO / SC environments, but also in conventional organizational structures. In this case, however, it is often possible to manage emerging governance issues by referring to common leadership. For instance, an accounting department within a BU might refer to BU management to resolve issues with the BU procurement department. Shared services or general SO / SC structures, however, aggravate the issues at hand, because a "market" situation resembling dealings between independent parties is created *deliberately* — it is in the best interest of each party to "sub-optimize" within its own domain. The provision of appropriate master data by a shared services client constitutes an illustrative example in this regard.

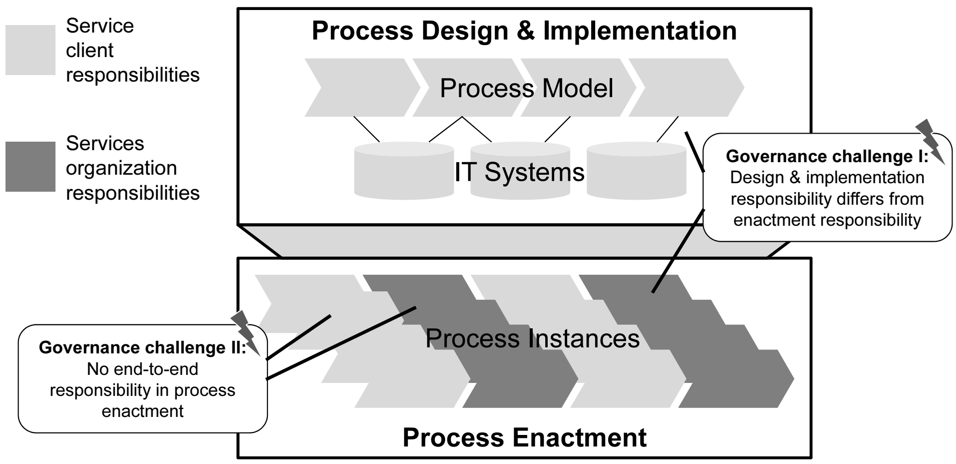



Figure 9.3: Governance Challenges in the SO / SC Lifecycle

[6]The costs-by-cause paradigm refers to the appropriate allocation of primary and secondary costs to cost objects [251] to obtain a "true and fair view" [112].

| **BP Lifecycle Stages** | **Governance Challenge I:** Differing design & implementation vs. enactment responsibilities | **Governance Challenge II:** No end-to-end responsibility in process enactment |
|---|---|---|
| *Process design & implementation* | SO requirements may not be captured, leading to sub-optimal actual process designs, in particular with regard to efficiency. | n/a |
| *Process enactment* | SO activities may not be optimally supported by the actual process design and implementation. | Activities' results may not be optimally suitable for use in subsequent steps if these are executed by a different party. |
| *Process performance measurement* | Data access to information systems operated by the SC may be limited. | SO / SC performance may be difficult to delineate since only end-to-end performance can be measured in many cases. |
| *Service charging* | The impact of differing actual process designs and implementations on SO enactment effort may be difficult to delineate, which may lead to violations of the costs-by-cause paradigm.[6] | The impact of SC activities on SO enactment effort may be difficult to delineate, which may lead to violations of the costs-by-cause paradigm. |
| *Process analysis & control* | SOs' feedback into design & implementation may be impeded by differing organizational responsibilities. | A lack of end-to-end process control due to differing organizational responsibilities may lead to ongoing "sub-optimization" on the activity level instead of overall optimization on the process level. |

Table 9.1: Governance Challenges in the SO / SC Lifecycle

## 9.2 State of the Art

To address the governance challenges described in Section 9.1, SO / SC organizations mostly rely on service level agreements (SLAs) [252, 253]. SLAs constitute formalized agreements between or within organizations to govern mutual obligations regarding the provision of services. They have been pioneered in the field of information management and comprise issues such as quality of services (QoS), collaboration duties on the customer side, or services charging [254, 255]. Figure 9.4 summarizes typical SLA content.

SLAs comprising performance indicators constitute the state of the art encountered in practice with respect to the *content* of process quality and performance management between SOs and SCs. However, SLAs are aimed at backwards-oriented control with regard to minimum requirements for both parties. They are thus limited in their ability to drive

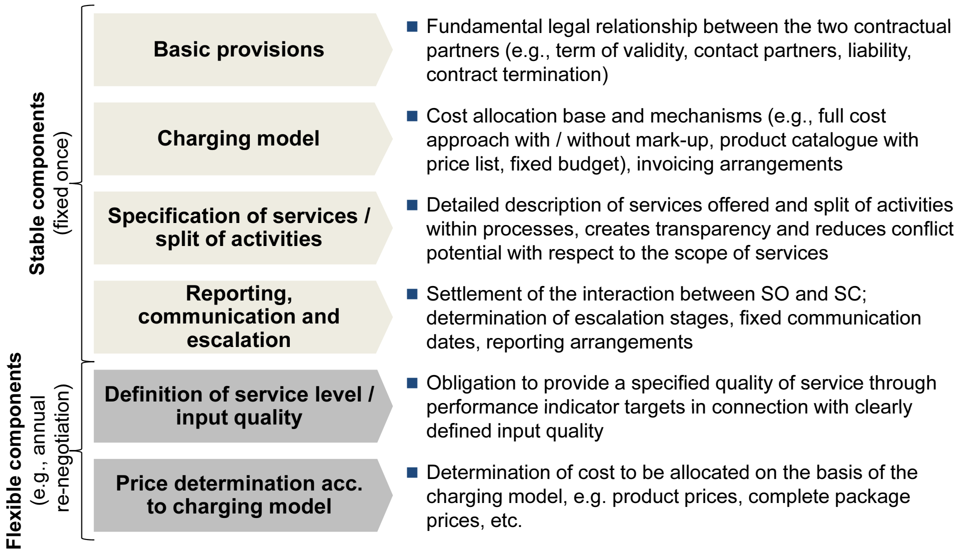



Figure 9.4: Typical Content of Service Level Agreements

future-oriented continuous optimization of business processes as required in today's competitive environment. Moreover, SLAs typically include end-to-end performance indicators which are, for instance, derived from common external benchmarking metrics. However, these indicators fail to delineate the impact of SO and SC responsibilities (cf. Example 50). They are thus not suitable to effectively control *future* behavior of involved parties. Beyond common management based on SLAs, this chapter looks into techniques to incorporate the concepts developed in this thesis into a BP lifecycle aimed at SO / SC organizations.

**Example 50 (Performance Indicators in Service Level Agreements).** Cycle times are a typical example of performance indicators included in SLAs. Since they are defined in an end-to-end manner without considering lead times in differing parties' contributions to individual tasks, they are, however, not suitable for to control individual behavior. For a typical example of a corresponding conflict case, consider increased cycle times caused by the SO waiting for master data entry by the SC. In this case, the SO may fail to fulfill its SLA without bearing responsibility for the underlying defect.

As SLAs constitute the (virtually ubiquitous) organizational approach towards process quality and performance management for SOs, today's IT environments allow supporting the respective measurement of performance indicators in various ways. The remainder of section shortly summarizes available options including application examples.

As a prerequisite for most performance indicator types, it is necessary to track process enactment on a case-by-case basis by logging appropriate events, e.g., the completion of tasks. This can be achieved by using *WfMS* which generally provide logging and analysis facilities [14]. Note that comparable facilities are also provided by *ERP packages and middleware* *tools*.[7] In certain cases, ERP packages provide the additional advantage of directly linking into resource requirements, e.g. through activity-based costing modules [256, 251]. Note that the provision of a tracking facility is one of the major reasons to implement a WfMS or an ERP system, since this capability can also help to address legal compliance issues (in Germany, e.g., the *Grundsätze ordnungsmäßiger DV-gestützter Buchführungssysteme* (principles of orderly computerized accounting systems) [257]).

*Dedicated tools for BP performance management* are increasingly offered by vendors (e.g., [46]). While they may follow an approach to provide pre-defined standard performance indicators for certain application areas, they still require data extraction, staging and integration methods. This is not the case when using "native" WfMS or ERP systems. Note that similar techniques have been developed under the notion of "process intelligence" [35, 258].

*Process mining tools* do not only enable deducting process models from enactment logs, but typically enrich extracted process models with additional information like, for instance, cycle times, actors or the relative prevalence of patterns [28]. Thus, process mining tools may allow implementing process performance indicators with the additional advantage of directly matching indicators against actual process models. In practical settings, however, one needs to keep in mind that effectively using process mining tools for process model discovery presumes detailed knowledge on the data that needs to be extracted, e.g., from an ERP system. This issue constitutes a major challenge in practical application scenarios [5].

*Business intelligence (BI) tools* are aimed at managing and presenting information extracted from transactional systems such as ERP systems [258]. The techniques employed range from data extraction and cleansing to analysis and visualization tools and are well-suited to be used in a process performance measurement context. As opposed to dedicated process performance management tools, BI tools do not provide pre-defined content in terms of indicators etc. However, this disadvantage may be more than compensated by advanced data management and visualization facilities, which constitute major challenges in typical process performance management projects, and possibly by the fact that BI tools are already in use in many organizations.

## 9.3 A Quality-aware Business Process Lifecycle

To address the issue of design quality (cf. Definition 1), Chapter 8 introduced *quality drivers* which constitute characteristics of the actual process model that are relevant to formal efficacy and resource requirements. Moreover, *quality meters* were introduced to leverage empirical data on process enactment for the purpose of quality management. The issue of appropriately delineating design quality and process performance reflects Effectiveness Criteria 1 and 2, *Congruence to organizational targets* and *Transparency and retraceability*. Extending the SO / SC BP lifecycle from Figure 9.2, Figure 9.5 summarizes issues to be considered in a *quality-aware BP lifecycle*.

- *Consideration of quality drivers* during *process design & implementation* pertains to the leverage of quality predicates and criteria associated with quality drivers to appraise and improve process designs. This approach is possible in an iterative process

[7] Middleware tools are used to integrate diverse application landscapes by providing standard interfaces to other software packages in the sense of a data broker.

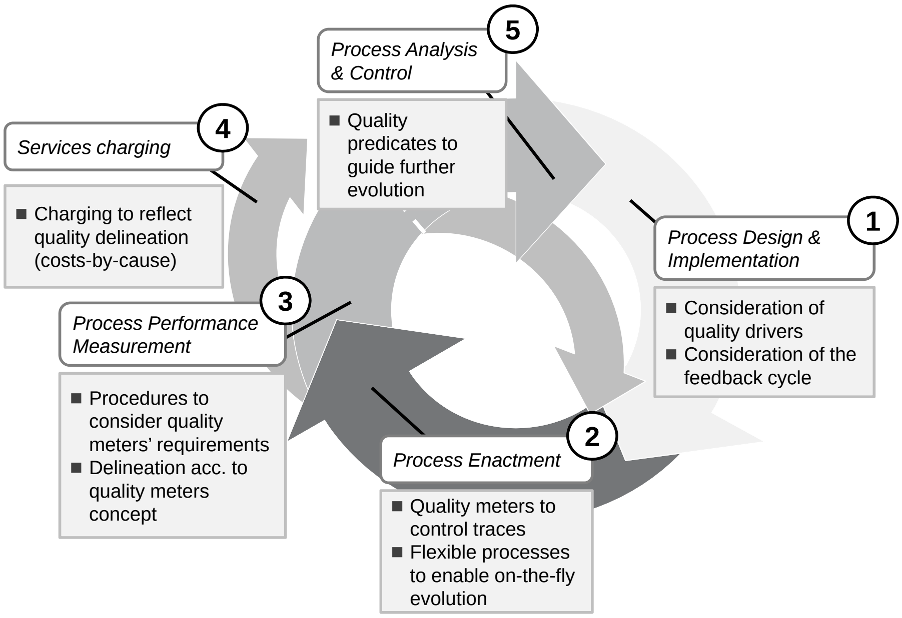



Figure 9.5: Quality-aware Business Process Lifecycle

development approach comparable to the common spiral model in software engineering [259]. In that case, process design results are matched against quality drivers to derive improvement potentials for the next iteration. That is, quality drivers assume the role of customer requirements in the spiral model. Moreover, quality drivers can be used to guide the inception of new business processes from the start on, since they highlight issues which are often not considered in practice.

- *Consideration of the feedback cycle* in *process design & implementation* addresses the use of *process analysis & control* results to improve an actual process design. Common performance indicators monitoring (e.g., as defined in a SLA, cf. Figure 9.4) can be used to alert process managers and designers to existing improvement potentials. In this context, common performance indicators usually cannot disclose root causes for improvable performance, but they can act as a "call for action" regarding more detailed analyses of quality drivers. Moreover, the *Appropriate Capital Investments* quality meter (cf. Section 8.4.1) needs to be assessed considering empirical process analysis results, e.g., with regard to actual transactional volumes.

- *Quality meters to control traces* during *process enactment* pertain to the utilization of quality meters, as defined in Section 8.4.2, in order to guide the ongoing management of process enactment. To appropriately address this issue, the actual frequency of possible enactment paths, which have been classified according to whether they are desirable or not, should be traced and made available to managers. To this end, available tracking and reporting functionality of WfMSs can be used. This facility is particularly effective if corresponding services charging or escalation procedures are in place.

- *Flexible processes to enable on-the-fly evolution* deal with the question whether processes can be altered in ongoing operations, i.e., during the *process enactment* lifecycle stage, instead of having to traverse the entire lifecycle back to *process design & implementation* [25]. This characteristic is supported only by few WfMSs since it entails managing the consistency of ongoing process instances with respect to process model alterations. The ADEPT process management suite constitutes a notable exception in this regard [260]. Nevertheless, flexibility is an useful feature of a quality-aware BP lifecycle because it allows bringing to bear quality control results independently of common release cycles (e.g. when operating ERP systems), which may delay the effective date of changes substantially.
- *Procedures to consider quality meters' requirements* during *process performance measurement* reflect the assessment procedures described in Section C.2. It is instrumental to track actual traces and cycle times for individual process instances, and to consolidate results to meaningful management reporting. In this context, legislation regarding the tracking of individual performance of staff is to be considered (cf. Section 12.4.2).
- *Delineation according to quality meters concept* during *process performance measurement* pertains to the governance challenges lined out in Section 9.1. Accordingly, process performance measurement should be able to delineate the performance of operational functions (e.g., SOs) from the impact of upstream processes (e.g., SCs) and actual process design, and vice versa. Note that this topic is considered in the definition of quality meters (cf. Section 8.4).
- *Charging to reflect quality delineation (costs-by-cause)* in the *services charging* lifecycle phase will support the effectiveness of quality appraisal with respect to ongoing process improvement, in terms of both design & implementation and enactment. This is particularly important in the context of SO / SC organizational structures since it provides an incentive towards end-to-end optimization. Accordingly, charging mechanisms should reflect the responsibility of SOs and SCs regarding aspects of design quality and enactment performance (cf. Example 51). Note that this usually cannot be achieved with common cost-plus charging mechanisms which are based on high-level statistical indicators such as turnover. Rather, a means of charging per transaction is required to statistically record actual traces (or, at least, particular trace characteristics) per SC.
- *Quality predicates to guide further evolution* during *process analysis & control* reflect the stipulation that quality predicates should be defined in a way to provide effective guidance towards quality improvement (cf. Section 8.1). Accordingly, a quality-aware BP lifecycle will monitor quality predicates achieved, and deduct appropriate measures to be implemented in *process design & implementation* and, in case of flexible BPM [25], *process enactment*.

**Example 51 (Costs-by-cause Services Charging).** Consider Sample Process A from Figure 2.5. The effort involved in managing incoming supplier invoices differs substantially with regard to design quality and enactment performance:

- If the actual process design enables early scanning, EDI transfer of documents, or a credit note procedure (cf. Section 8.3), processing effort can be lowered significantly.

Moreover, processing effort is impacted by properties of the underlying ERP system such as availability, usability etc. Commonly, these issues are controlled by the SC.

- Poor data quality from upstream processes, in particular with regard to the availability of supplier master data or purchase order transactional data, will raise processing effort significantly. This also pertains to multiple feedback loops (cf. *Mitigation of repetitive loops* as a quality driver described in Section 8.3) which may occur if invoices are approved manually by the respective operational department. Again, these issues are usually controlled by the SC.

In services charging, these issues can be considered by assigning service charges per type of transaction (i.e., per type of process instance). For example, there may be a basic charge per type of transaction per ERP system addressed. These basic charges can then be amended by penalties for defective process instances (multiple loops or missing input data). In practice, these penalties are often used only in critical cases, e.g., when there are significant differences between SCs.

In summary, organizations need to be aware of *technological requirements* associated with a quality-aware BP lifecycle. These mainly relate to the need to create transparency over both actual process designs and enactment performance, and to flexibly execute the feedback cycle of iteratively improving processes. To this end, appropriate WfMSs or PAISs, and corresponding support functions provided by BI or dedicated process intelligence systems need to be in place.

## 9.4 Quality-aware BPM System Landscapes

The previous section shortly touched upon some common ground between a quality-aware BP lifecycle on the one hand, and PAISs, WfMSs, and process intelligence systems on the other hand. This section further refines these considerations, thus illustrating a more comprehensive set of requirements towards quality-aware BPM system landscapes.

In the field of BPM, the WfMC has proposed a reference model for WfMSs and their interaction with their environment (cf. Figure 9.6) which can be used as a starting point for more detailed discussion since it represents the view of the WfMC as an industry association [62].

The WfMC reference model structures basic components required to design, implement, and enact workflows in the sense of business processes with IT-supported control flow [14]. Accordingly, it corresponds well to the quality-aware BP lifecycle (cf. Figure 9.5). However, fully leveraging the progress made in terms of IT support provided to BPM for the purpose of quality management requires consideration of additional components. Figure 9.7 therefore provides an extended system landscape model covering, in addition to Figure 9.6, the full analysis and feedback cycle which is today's standard in both BPM and quality management [13, 47], as well as the increased pervasiveness of BPM concepts in today's enterprise IT landscapes. Note that solid arrows in the model indicate the flow of transactional data, dotted arrows indicate the flow of control data, and dashed arrows indicate the implicit flow of knowledge (in this case, of implicit process models and fragments thereof).

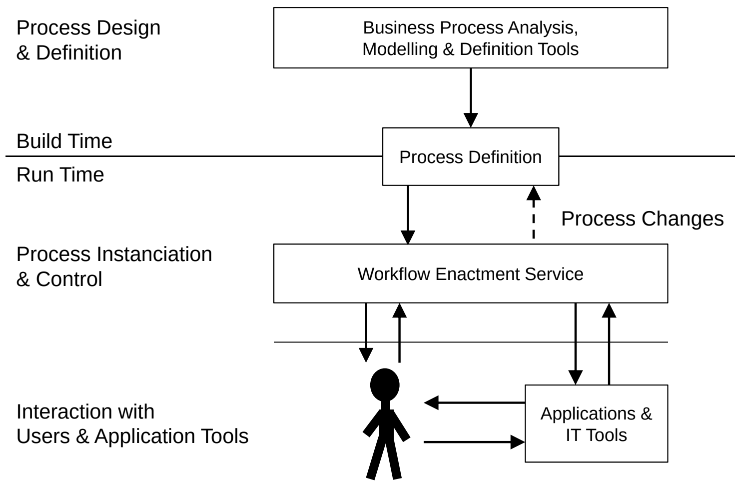



Figure 9.6: Workflow Management Coalition Reference Model [62]

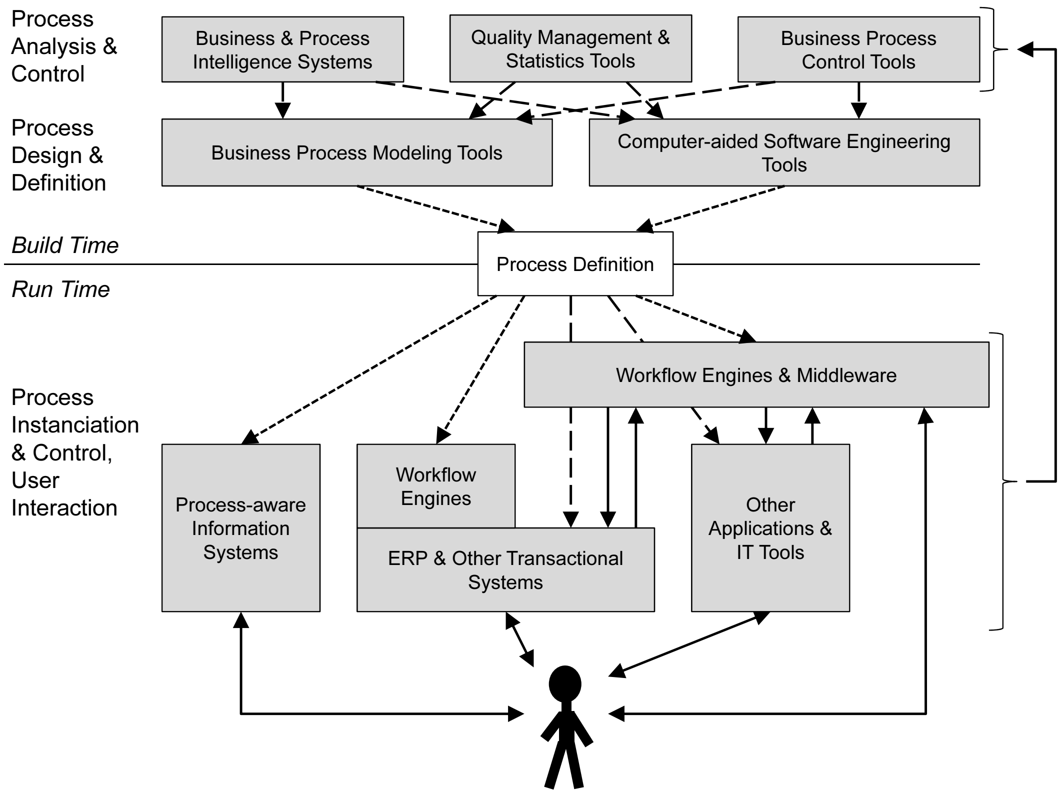



Figure 9.7: Extended BPM System Landscape

The following paragraphs discuss requirements for BPM systems landscapes which arise from BP quality management concepts developed in this thesis.

**Process-aware Information Systems (PAISs)** PAISs separate explicit process models from program code, thus adding an additional layer of abstraction to traditional software engineering approaches [16]. The major goal behind this is to increase software flexibility, maintainability, and, ultimately, overall quality [261]. As run-time systems, PAISs directly interact with users as operators for tasks, and implement explicit process definitions. To enable PAISs for process quality management, it needs to be ensured that the assessment of quality meters is facilitated by appropriate logging of process enactment. Similar to process mining, the following information is required for each each enactment event (i.e., the enactment of a control flow element): an unique identifier of the corresponding process instance, an identifier for the event class (e.g., a task within the underlying process model), and a timestamp [5]. Additional information, such as the operator of a task, may be helpful, but is not strictly required.

**ERP & Other Transactional Systems** ERP and other systems aimed at supporting enterprise transactions mostly provide facilities to log events in the sense of transactional changes to the underlying enterprise data base. These can also be used for the purposes of process quality management. In addition, contemporary ERP systems (e.g., mySAP ERP) already include support of process-oriented work through built-in workflow engines. However, it needs to be kept in mind that the built-in workflow engines of ERP systems are focused on processes handled within the respective system. Thus, they may be not optimally suited to integrate other applications. This characteristic reflects the original intention of the ERP paradigm to integrate organizational functions as much as possible into one underlying data base.

**Other Applications & IT Tools** Applications and IT tools which are used to enact business processes, but do not fall into the categories of PAISs or ERP systems can be addressed through workflow engines or manually. In the former case, the logging facilities required to assess quality meters are usually provided by the workflow engine. In the latter case, mitigation measures are required since this category of IT systems (e.g., text editors) usually does not provide logging capabilities.

**Workflow Engines & Middleware** Workflow engines[8] manage control flow by invoking ERP systems or other applications for the enactment of tasks. Besides dedicated workflow engines, other types of middleware solutions may provide comparable functionality. In this context, middleware refers to tools supporting the integration of various enterprise applications, e.g. by providing data exchange facilities. As an example, consider IBM's *WebSphere MQ* product [262]. In addition, workflow engines may also provide on-screen forms to capture data entered by operators. Thus, they assume a central role in the enactment of business processes, and mostly provide the required logging facilities to enable the assessment of quality meters. In this case, it is generally preferable to utilize workflow engines' or middleware tools' logging functionality instead of logs provided by the applications invoked, because

[8]The WfMC reference model refers to *Workflow Enactment Service* instead [14]. In the extended BPM system landscape, this term is replaced by *workflow engine*, which is more common today.

only one interface needs to be implemented, and – even more important – a process instance identifier is readily available for all enactment events. Beyond logging, workflow engines and middleware solutions also facilitate the implementation of technical measures linked to quality drivers such as *QA 19: Routing automation*. In many cases, workflow engines, either as standalone systems or as a component of an ERP solution, provide organizations with the capabilities to design and implement processes as demanded by quality drivers. The relevant functionalities available within an enterprise IT landscape thus need to be taken into account when assessing quality drivers.

**Process Definition** With regard to process definitions, it is important to recognize that these may be available as an explicit process model, but also as implicit knowledge incorporated into IT systems, the behavior of employees, or even organizational culture [236]. In the latter case, however, process quality management is severely impeded, since quality drivers in the sense of quality attributes to be assessed by inspecting the process model can be applied only rudimentarily. In particular, this pertains to formal quality drivers or other quality drivers where appraisal can be supported by utilizing quality-aware BP models (cf. Chapters 7 and 8).

**Business Process Modeling Tools** BP modeling tools are a major component of the build time BPM environment, and deliver explicit process definitions. Since quality drivers as the majority of quality attributes pertain to process models, they are of particular importance for BP quality management. The following requirements result from the concepts developed in this thesis, and are therefore not available in today's BPM tools: To fully support BP quality management, process modeling tools should implement facilities to model business objectives as described in Chapter 6, and extend process modeling functionality to address quality-aware BP models as described in Chapter 7. In addition, assessment of many quality drivers can be fully automated (formal efficacy-related quality drivers), or at least partially supported through automated tools (cf. Appendix C). Advanced quality-aware BP modeling tools will provide corresponding facilities.

**Computer-aided Software Engineering Tools** Progress in software engineering has led to increasing convergence between software engineering and BPM concepts. The use of sequence diagrams in the Unified Modeling Language (UML) as a widely spread software engineering meta-model exemplifies this consideration [229]. Thus, enterprise software developed with computer-aided software engineering (CASE) tools often incorporates implicit business processes even if there is no outright workflow management. Accordingly, CASE tools must be considered as part of the build-time BPM system landscape as well. However, one needs to be aware that CASE tools do not provide support to explicit modeling of business objectives or quality-aware processes, and do not lend themselves to be extended in that direction. This characteristic impedes the assessment of quality drivers.

**Business & Process Intelligence Systems** In contrast to process intelligence systems catering to BPM requirements, business intelligence systems aim at general applications in reporting and controlling. Both constitute important tools to assess quality meters as quality attributes based on process enactment records (cf. Section 8.4) [263, 35, 36]. In this regard, process mining tools assume a special role since they combine process intelligence functionality (e.g., measuring performance indicators such as cycle times) with the capability to deduct process models from process enactment logs [28]. However, these process models are limited to the information available in enactment logs, and do not capture the full semantics of control flow. For instance, semantically annotated split and join gateways are typically missing. The detailed descriptions of quality meters in Section C.2 comprise assessment procedures that can be addressed through business intelligence, process intelligence, or process mining tools. Additional requirements and open issues from a management and control perspective are discussed in [5].

**Quality Management & Statistics Tools** Quality management approaches are often based on statistical analysis (e.g., statistical process control (SPC) or Six Sigma [47]). To support respective procedures, general purpose statistics tools (e.g., SPSS) are available as well as specialized quality management applications (e.g., Minitab). Beyond the assessment of quality meters in cases not covered by business or process intelligence tools, these may be used to further assess proposed process improvement measures resulting from quality assessment. Corresponding detailed examples are provided in Section 12.4.

**Business Process Control Tools** BP control tools support the feedback cycle from quality assessment results in particular into ongoing operational process management. Procedures in this respect include the presentation of information on enactment quality in management cockpits [250], the management of service charges [255], and the management of interfacing issues in end-to-end process chains, as illustrated in Example 52.

**Example 52 (Intercompany Escalation Management).** At a large international engineering group, a process control tool has been implemented to speed up the monthly financial closing process. In this case, the issue at hand pertains to the management of intercompany invoices, i.e., invoices issued from one group company to another. To ensure that these invoices are booked timely on the receiver side, the process control tool automatically triggers posting in the respective ERP system, and provides additional functionality to manage emerging disputes. This way, a critical enactment interface resulting from the potentially differing ERP systems on the supplier and customer sides could be improved substantially.

## 9.5 Conclusion

This chapter discussed the organizational environment relevant to the effectiveness of BPM for processes in the scope of this thesis. In this context, it assessed the particular governance challenges that arise from the split of responsibilities between services organizations and service clients, and highlighted how these are commonly addressed today. On that basis, it lined out how BP quality management concepts can contribute to alleviating common governance challenges beyond the methods and tools used today:

*Governance challenge I*, the gap between governance over actual process design (residing with SCs) and responsibility for process enactment (partially residing with SOs), is addressed by making design quality transparent through the application of appropriate quality drivers, and by aligning charging mechanisms to differing enactment efforts depending on actual process design. Moreover, flexible processes enable on-the-fly, iterative process evolution independent of common release cycles as performance analysis results are fed back into process design. This characteristic makes it easier for SOs to assert their demands towards process design & implementation.

*Governance challenge II*, the lack of end-to-end process governance, is addressed by applying appropriate quality meters, and by reflecting differing enactment efforts depending on actual traces in charging mechanisms. Appropriately considering the requirements posed by quality meters in procedures and tools for process performance measurement will facilitate effectiveness in this respect.

For both governance challenges, the respective quality predicates associated with quality attributes and criteria simplify the derivation of actual, hands-on improvement measures.

In addition, this chapter discussed the integration of BP quality aspects into BPM systems landscapes. Based on the WfMC reference model for BPM systems landscapes, run-time components mainly need to consider logging data required for the analysis of quality meters as well as the implementation of "technical" quality drivers such as the use of automation potentials. Build-time components comprise both process design systems which should reflect the requirements of business objectives modeling and quality-aware BP model, and analysis and control systems helping to feed back quality assessment results into process enactment. In summary, these considerations provide an initial requirements definition to implementing, adapting, or rolling out corresponding tools. Note that, in this respect, the use of well-established BPM concepts to develop the business objective and quality-aware BP modeling approaches (cf. Chapters 6 and 7) supports the integration of quality concepts into an existing BPM tools landscape.

Accordingly, this chapter provided an initial discussion on the practical applicability and utility of BP management concepts in the light of contemporary BPM. It is thus well-suited to bridge into the final part of this thesis dealing with the validation and discussion of results.

# Part III

# Validation and Conclusion

# 10 Field Evaluation: Preliminary Quality Model

This chapter presents an initial application of the BP quality definition framework as presented in Chapter 5 to a practical case. This application was conducted to obtain an initial measure of validation for concepts used and developed, and to guide the further progress of this thesis. It follows the requirement for information systems research to assess practical relevance [15]. Based on a preliminary quality model, a real-world business process has been used to match the definition framework against effectiveness criteria (cf. Section 3.1) and managers' expectations, and to derive guidance used to further refine quality modeling.

According to the deductive approach underlying this thesis, it is generally desirable to rigorously derive quality attributes by applying Definition 1 to formal definitions of business processes, target artifacts, resources, business objectives and their interrelations. The research described in this chapter deviates from the general principle. It builds and evaluates a preliminary quality model which is not based on rigorous deduction, but on an informal collection of possible attributes loosely structured along quality dimensions and lifecycle phases as given in Definition 1.

There are three reasons to pursue this deviation: *First,* the quality model constitutes a central but complex deliverable of this thesis. Since an additional iteration in its development provides further guidance to the *build* procedure, the effort incurred appears as warranted. *Second,* the preliminary iteration is similar to related work based on listing possible quality attributes without rigorous derivation [123, 125]. Applying this methodology to the BP quality definition framework developed in this thesis may therefore substitute the claim that a rigorous deductive approach will lead to more effective results with respect to the criteria described in Section 3.1. *Third,* the definition framework for BP quality (cf. Definition 1) constitutes the *construct* that subsequent design artifacts *built* in this thesis are based on. It is therefore sensible to assert its effectiveness by preliminary applying it to its purpose, in this case the development of a quality model.

## 10.1 Field Evaluation Methodology

Qualitative empirical research in information systems comprises action research, case study research, ethnography, and grounded theory [264]. Due to its limited scope and purpose, this field evaluation cannot qualify as a self-contained research effort of one of the stated categories. In terms of experimental models as defined by Zelkowitz and Wallace [265], it constitutes an *assertion.* However, to still ensure sufficient rigor, the following paragraphs

This chapter is based on the following referred papers:

Lohrmann, M., Reichert, M.: Understanding Business Process Quality. In: Business Process Management: Theory and Applications. Volume 444 of Studies in Computational Intelligence, Springer (2013) 41–73

Lohrmann, M., Reichert, M.: Basic considerations on business process quality. Technical Report UIB-2010-04, Ulm University, Germany (2010)

shortly discuss the design of the field evaluation along the requirements posed by Wieringa et al. for empirical research in the field of information systems [266]. The cited contribution requires to describe *problem statement* and *research design*, and to discuss *validity* and *research execution*.

**Problem Statement** To sufficiently describe the *problem statement*, [266] demands information on the *unit of study*, the *research question*, available *relevant concepts and theory*, and the *research goal*.

The *unit of study* is the BP quality framework described in Chapter 5. The *research question* addresses the issue whether the definition framework, if put to application, meets practitioners' expectations in terms of effectiveness criteria as described in Section 3.1. The *relevant concepts and theory* have also been described in Chapter 5. The *research goal* is to answer the research question and to obtain insights that may be used to refine additional results.

**Research Design** In terms of *research design*, [266] requires contributions to discuss the *unit and environment of data collection*, *measurement instruments and procedures* as well as *data analysis procedures*.

The *unit of data collection* consists of a preliminary business quality model according to the definition framework. It is described in more detail in the remainder of this chapter. The *environment of data collection* consisted partly of telephone conferences, and partly of face-to-face interviews. As *measurement instrument*, the preliminary quality model was amended with results for a sample process (cf. Section 10.3) and converted to a guideline for semi-structured interviews [267] which made up the *measurement procedure*. *Data analysis procedures* were not employed due to the qualitative nature of the approach (however, basic statistical methods were employed to apply the preliminary quality model to the sample process, cf. Section 10.3).

**Validity** The validity of the field evaluation is limited by the use of just one business process as the sample. However, with regard to the depth of insights to be gained for further refinement of design artifacts, this approach was preferred over using a statistically valid sample of less complex exemplary processes. In this respect, the field evaluation followed the principles underlying the case study paradigm [268]. Note that, in this respect, the approach reflects the purpose of contributing to resolve a *design problem* instead of a *knowledge problem*.

**Research Execution** Research was executed by extending the unit of study to a sample application, i.e., a preliminary quality model. The preliminary quality model was then applied to a real-world business process. The corresponding preliminary quality assessment results were discussed with responsible stakeholders to determine whether effectiveness criteria had been fulfilled. Interview partners comprised the process manager, the implementation project manager, and the process designer.

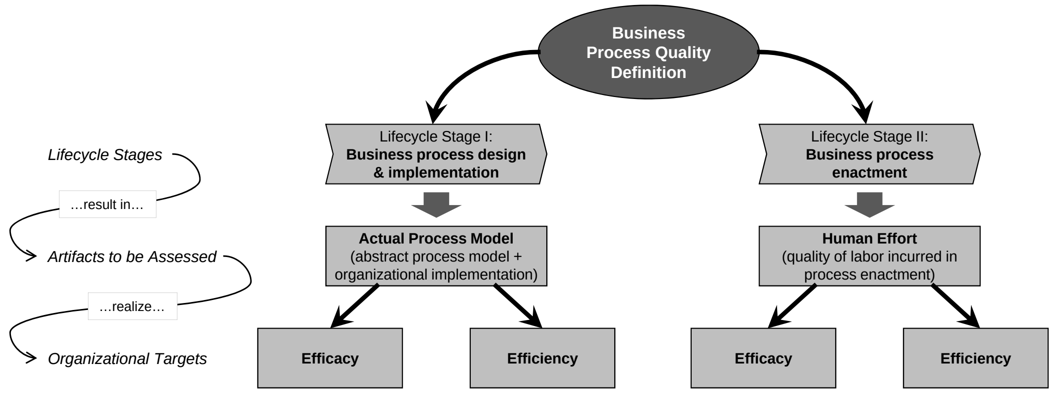



Figure 10.1: Preliminary Quality Model Deduction

## 10.2 Preliminary Quality Model

This section describes a preliminary quality model informally deducted from the BP quality definition framework of Chapter 5. It is structured into quality attributes, criteria and predicates, following the considerations made in Section 8.1.

As a mental technique to develop a preliminary quality model, this chapter considers possible *quality deficiencies* that might occur. Figure 10.1 summarizes the basic approach applied to deduct an initial, non-formalized and simplified quality model. Thus, lifecycle stages and the resulting artifacts, which are subject to quality assessment are considered first. Accordingly, assessing BP quality in Lifecycle Stage I amounts to assessing the quality of the actual process model, and assessing BP quality in Lifecycle Stage II amounts to assessing the quality of human effort during enactment. Both artifacts are then appraised with respect to their impact on the organizational targets of efficacy and efficiency. Additional guidance is provided by the overview on the resources part of the affected environment in Figure 5.3.

Table 10.1 lists quality attributes, criteria and predicates included in the preliminary quality model. Because the entire quality model is not rigorously deducted at this stage, its completeness is not yet warranted, and concisely measurable quality criteria cannot be given yet. However, the structure along the considerations made in Chapter 5 still allows for a measure of control in this respect, e.g. by considering the system of affected resources in Figure 5.3.

| Ref. | Quality Attributes | Quality Criteria | Quality Predicates |
|---|---|---|---|
| *Business Process Design & Implementation Efficacy* | | | |
| *A1* | Formal or informal documentation of the business objective | Business objective explicitly modeled or documented as prerequisite to manage efficacy | Transparent and controlled business objective |
| *A2* | Expectations and requirements regarding the actual affecting environment | Expectations regarding the actual affecting environment have been reasonably derived and documented / communicated | Managed affecting environment |
| *A3* | Relation between designated termination states and the business objective | Control flow model conforms to the business objective (e.g., by formal derivation from the business objective) | Efficacious control flow design |
| *A4* | Consideration of procedures to manage deficiencies during BP enactment | Relevant cases covered acc. to affecting environment expectations, procedures comprised in actual process design | Efficacious exception handling |
| *A5* | Relation between capital goods and BP model requirements | Capital goods available according to BP model as far as organizational resources have been available | Efficacious capital expenditures |
| *A6* | Relation between staff capacity and BP model requirements | Staff and procedures available according to BP model as far as organizational resources have been available | Efficacious organizational implementation |
| *Business Process Design & Implementation Efficiency* | | | |
| *B1* | Occurrence of non-value-adding activities and execution paths | Control flow explicitly designed to avoid non-value-adding activities and enactment paths | Controlled non-value-adding activities and enactment paths |
| *B2* | Occurrence of resource waste in activities | Activities designed to avoid materials waste (e.g. clippings) and capacity waste (e.g. through idle time for staff or capital goods) | Controlled resource consumption in activities |
| *B3* | Modeled sequence of activities: control flow designed to enable early break conditions towards termination states | Avoidance of non-value-adding activities in possible enactment paths regarding termination states, early enactment of automated checks | Efficient break conditions |

*Continued on next page*

| Ref. | Quality Attributes | Quality Criteria | Quality Predicates |
|---|---|---|---|
| *B4* | Design decisions: employment of capital goods vs. labor to implement automated vs. manual activities | Design decisions taken based on explicit business case considerations | Controlled capital goods vs. labor trade-off |
| *B5* | Skill requirements: employee skill levels required in manual activities | Design decisions taken based on explicit business case considerations, activities and procedures are properly documented and trained | Controlled skill employment |
| *Business Process Enactment Efficacy* | | | |
| *C1* | Occurrence of deviations from the BP model in manual decisions altering the actual control flow path | Prevalence reasonable with respect to the criticality of the business objective | Efficacious manual decisions in the control flow path |
| *C2* | Occurrence of deviations from the BP model in manual manipulations of target artifacts or resources relevant to the control flow in the course of activity enactment | Prevalence of deviations reasonable with respect to the criticality of the business objective | Efficacious enactment of manual activities |
| *C3* | Occurrence of time delays in manual enactment of activities | Prevalence and severity of time delays reasonable with respect to the criticality of the business objective | Timely enactment of manual activities |
| *C4* | Occurrence of manual alterations to the actual process model (e.g. overriding of IS customization) in the course of the enactment of individual process instances | Prevalence of manual alterations reasonable with respect to the criticality of the business objective | Conformance to the actual process model |
| *Business Process Enactment Efficiency* | | | |
| *D1* | Occurrence of deviations from the BP model leading to redundant activities caused by manual control flow decisions | Prevalence of redundant activities reasonable with respect to complexity of control flow decisions and additional effort incurred | Efficient enactment regarding redundant activities |
| *D2* | Occurrence of repetitive enactment of process instances or activities due to activity enactment deficiencies | Prevalence of repetitive enactment reasonable with respect to complexity of respective tasks and additional effort incurred | Efficient enactment regarding repeated process instances |

*Continued on next page*

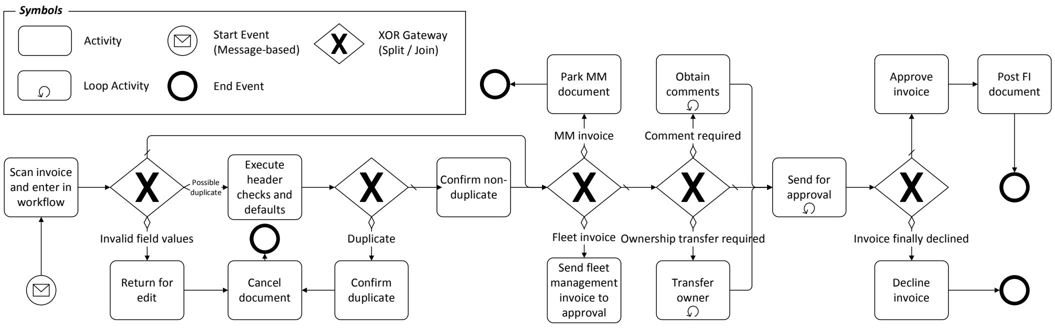



Figure 10.2: Sample Process: Invoice Handling

| Ref. | Quality Attributes | Quality Criteria | Quality Predicates |
|---|---|---|---|
| *D3* | Occurrence of additional corrective activities due to manually caused deviations or deficiencies | Prevalence of corrective activities reasonable with respect to complexity of respective tasks and additional effort incurred | Efficient enactment regarding corrective activities |
| *D4* | Occurrence of manual re-allocation of enactment responsibility for activities | Prevalence of re-allocated activities reasonable with respect to source (manual vs. automated) and validity of original allocation and additional effort incurred | Efficient enactment regarding re-allocated activities |

Table 10.1: Simplified Quality Model

## 10.3 Illustrative Case

To illustrate the results achieved and obtain insights for further discussion, the preliminary quality model has been applied to a real-world business process. Information available on the process considered comprised its actual process model and an enactment log.

In terms of content, the sample process corresponds to the examples given in Chapter 5. Its business objective is to approve or disapprove incoming supplier invoices correctly and timely. In particular, it implements the early scanning design option already mentioned in Example 9. The enactment sample covers a total of 1,130 process instances (one instance corresponds to one supplier invoice) started over the period of one week. In the enactment log data sample, the process instances have been tracked over the period of 15 weeks. Process instances not concluded within this timeframe are not considered further. Figure 10.2 presents a BPMN flow chart of the BP model [80]. In addition, evaluation is based on a central document describing the business process and its technical implementation (the so-called “blueprint”).

In Table 10.2, the quality criteria set out in the preliminary quality model of the previous section are applied to the sample process. Quality predicates are assigned to the sample process accordingly.

| Ref. | Quality Assessment | Quality Predicates |
|---|---|---|
| *Business Process Design & Implementation Efficacy* | | |
| *A1* | The business objective has not been formalized or documented in the blueprint, which governs process implementation and enactment | |
| *A2* | The expected affecting environment has not been included in the blueprint, but considered informally in actual process design; an evaluation on the expected transactional volume has been conducted | Managed affecting environment |
| *A3* | While there is no formal documentation of the business objective, use cases have been described in detail in the blueprint. As use cases have been deducted from available transactional data (cf. A2), efficacious implementation may therefore be assumed | Efficacious control flow design |
| *A4* | Exception handling routines have not been included in the actual process design | |
| *A5* | Actual process enactment as per the log sample implies appropriate capital investments according to the process design | Efficacious capital expenditures |
| *A6* | Actual process enactment as per the log sample implies issues in organizational implementation (cf. C3, D2, D3, D4) due to limited governance of process management over process participants | |
| *Business Process Design & Implementation Efficiency* | | |
| *B1* | Non-value adding activities occur in the enactment path (manual re-allocation of responsibilities), "looping" of check activities is possible | |
| *B2* | Capacity waste is avoided through the use of work item lists for all user groups | Controlled resource consumption in activities |
| *B3* | All automated checks are designed to occur at the beginning of the control flow sequence | Efficient break conditions |
| *B4* | Design option decision (early scanning plus workflow) for the business process is based on an explicit business case consideration | Controlled capital goods vs. labor trade-off |
| *B5* | Actual skill employment is based on available resources in the organization instead of documented requirements | |

*Continued on next page*

| Ref. | Quality Assessment | Quality Predicates |
|---|---|---|
| *Business Process Enactment Efficacy* | | |
| *C1* | Deviations from the BP model do not occur (process enactment is fully controlled by the WfMS) | Efficacious manual decisions in the control flow path |
| *C2* | Correct handling of invoice approval is subject to both internal and external audit procedures (risk-based audit approach) | Efficacious enactment of manual activities |
| *C3* | Total processing time exceeds two weeks in 10% of cases, mainly due to delays in the approval procedure | |
| *C4* | Manual alterations to the actual process model do not occur | Conformance to the actual process model |
| *Business Process Enactment Efficiency* | | |
| *D1* | Attribute not assessable: redundant activities may occur where approval actions beyond the requirements based on the invoice value are conducted. Due to data protection concerns, this data is not analyzed further | n/a |
| *D2* | "Return for edit" occurs in 10% of cases, leading to repeated manual check activities | |
| *D3* | "Return for edit" occurs in 10% of cases, leading to corrective activities in document capturing | |
| *D4* | Manual case ownership transfers occur in 34% of cases | |

Table 10.2: Simplified Quality Model: Sample Application

In summary, the implications from the case example presented above are twofold: First, the assessment with respect to the quality of the sample business process can be summarized. Second, and more important, assessment of initial design results with respect to the effectiveness criteria set out in Table 3.1 is enabled.

With respect to the sample process, quality predicates assigned imply that the quality of the process largely reflects the chosen design option as a contemporary "good practice". Most issues incurred relate to topics where respective approaches have not yet reached practical acceptance (e.g. A1), or to governance issues during the enactment lifecycle phase. This may be due to the fact that, in this case, process management only partially controls process participants as invoice approval is "spread" throughout the organization.

When discussing this result with the responsible process manager, it was found that conclusions closely reflect her own appraisal of the situation. The same, albeit with a more limited scope of judgment, applied to the responsible project manager who had led the design and implementation of the underlying information system, and the process designer who had been responsible for technical blueprinting.

## 10.4 Conclusion

To conclude the preliminary field evaluation, the effectiveness criteria described in Section 3.1 are applied according to the design science paradigm:

- ***Effectiveness Criterion 1:*** *Congruence to organizational targets.* Implications in respect to Effectiveness Criterion 1 are twofold. On the one hand, one can directly "drill down" from organizational targets to each quality attribute considered. Accordingly, there are no issues with respect to *exclusive coverage.* On the other hand, it is not possible to ensure *comprehensive coverage* of the quality model. According to the considerations made in Section 3.2, this restriction could have been avoided by following a rigid deductive approach when drafting the quality model. In the case of the simplified preliminary model at hand, the deviation from this principle was deliberately incurred to allow for a quick initial appraisal of the quality definition framework.
- ***Effectiveness Criterion 2:*** *Transparency and retraceability.* This chapter's quality model reflects basic organizational governance by adhering to fundamental BP lifecycle phases. However, as mentioned above with respect to the enactment lifecycle phase, a more fine-grained approach is required for the practical example. Moreover, the "binary" allocation of quality predicates may omit important graduations. While assessment results still point to issues to be addressed to improve quality, organizational acceptance might still be impeded by these issues.
- ***Effectiveness Criterion 3:*** *Cost effectiveness.* The illustrative case has shown that the simplified quality model can be applied with small effort, provided that basic information such as an implementation blueprint and an expressive enactment log sample are available. This aspect, however, needs to be tracked when moving into more detailed quality models to further accommodate Effectiveness Criteria 1 and 2.

Evaluation of the preliminary quality model against effectiveness criteria thus provides guidance to be leveraged in the subsequent refinement of the BP quality approach. A major issue to be addressed relates to *comprehensive coverage* as an aspect of *Congruence to organizational targets.* It is easily possible to achieve *exclusive coverage*, i.e. that no quality attributes are comprised in the quality model which do not positively impact organizational targets, by sensibly allocating attributes to quality requirements as comprised in the quality framework. *Comprehensive coverage*, however, cannot be guaranteed. As formal deduction of quality attributes was abandoned by employing a less rigorous collection approach, there is no way to determine whether the resulting quality attributes comprehensively cover organizational targets. This topic also impacts *Transparency and retraceability*: as long as the validity of the model with respect to Effectiveness Criterion 1 cannot be demonstrated, assessment results are difficult to uphold. Therefore, comprehensive coverage is to be addressed through a rigorous deductive methodology to derive quality attributes which are then amended with corresponding quality criteria and predicates. This approach has been pursued in Chapters 7 and 8.

# 11 Beyond Reengineering and Bottom-up Quality Management

As a means to validate a central contribution of this thesis, this chapter matches the quality model developed in Chapter 8 against results from related work. As lined out in Chapter 3, the approach to identify valid quality attributes pursued in this thesis is based on *deriving* quality attributes from a general definition of BP quality, the scope of influence a business process governs, a quality aware BP meta-model and, ultimately, business processes' support of organizational targets. According to the analysis of the state of the art regarding BP quality carried out in Chapter 4, this approach notably differs from the common methodology of identifying issues relevant to quality by leveraging results from other fields or similar "bottom-up" methods.

Moreover, it is common in the field to use "good practices", i.e., applicable experience made available by subject matter experts, to deduct process improvement potentials. This corresponds to the aims of quality predicates in the sense of this thesis. It is therefore instrumental for the appraisal of the contribution of this thesis to match its results against both related fields:

- To compare results to common "good practices", this chapter refers to the summary analysis of *reengineering best practices* compiled by Reijers and Limam Mansar [129, 130]. This analysis appears as particularly well-suited since it empirically established the prevalence of practices *in the field*.
- To compare results to related scientific work on BP quality, this chapter refers to the approaches developed by Heravizadeh et al. [269, 123, 124] and by Heinrich and Paech [125]. Both propositions aim at developing an integrated definition and model of BP quality which is not restricted to the quality of BP *models* or other individual aspects.

This comparison will enable discussing whether the top-down methodology followed in this thesis provides additional benefits with regard to comprehensive, but exclusive coverage of issues relevant to BP quality (cf. Chapter 3).

## 11.1 Quality Attributes vs. Reengineering Best Practices

Quality attributes as discussed in this section pertain to the quality of an actual process design, i.e. of an explicit or implicit process model and its implementation. Recommended process design patterns or other characteristics have been proposed under the notion of "good" or even "best practices" ever since the concept of BP reengineering [18, 17] became popular. In the works of Reijers and Limam Mansar on reengineering best practices [129, 130], a collection of corresponding practices has been established and empirically validated.

This section uses this collection for a comparison with the quality attributes derived in this thesis.

Best practices in process design and process quality attributes both aim at improving business processes and fostering the achievement of organizational targets. However, while BP quality management is focused on the design & implementation and enactment of processes, reengineering takes a wider stance and addresses the quality of business objectives and the quality of the organizational environment as well. This circumstance is illustrated in Example 53.

**Example 53 (Quality of Business Objectives).** Consider Sample Process A as depicted in Figure 2.5. This business process is enacted to manage supplier invoices. In a classical example provided by both Hammer and Champy as well as Davenport, this process is replaced by another process to issue credit notes to suppliers [18, 17]. In this case, it is not necessary for the supplier to issue invoices anymore. This is a classic example where reengineering goes beyond the individual business process: not only the business process is altered, but the underlying business objective is replaced.

As a prerequisite to the comparison conducted in this section, it is therefore instructive to classify the reengineering best practices presented in [130] accordingly. Moreover, the practices presented in [130] are mostly derived from a review of related literature, and then verified regarding their prevalence with practitioners.[1] Invariably, the various sources employed in this approach entail overlaps between practices assessed. Accordingly, Figure 11.1 presents a consolidated view on the reengineering best practices discussed in [130].

The practices in Figure 11.1 are labeled with their original association with "framework components" as described in [130], and have been consolidated as follows:

- The reengineering best practices *Control relocation*, *Contact reduction*, *Integration*, and *Interfacing* deal with the split of responsibilities with customers or other external parties (i.e., process choreographies [22]). Since these practices require adaptation of the business objective, i.e., alterations to the intended results of corresponding business processes, they have been consolidated to the *Business objectives reengineering practices* category. Since the notion of BP quality advocated in this thesis stipulates that business objectives are not an object of quality assessment, these practices are not in the scope of the analysis conducted here (cf. Chapter 5).

- The reengineering best practices *Flexible assignment*, *Centralization*, *Extra resources*, *Outsourcing*, *Trusted party*, and *Empower* deal with the quality and availability of resources. Note that, in this case, only personnel resources are addressed. Accordingly, these practices pertain to the quality of the external environment of the business process or the quality of BPM procedures. Again, the notion of BP quality pursued in this thesis stipulates that the external environment in which processes are designed, implemented and enacted must not impact process quality appraisal. Accordingly, *BPM / external environment practices* are not further considered.

[1] Note that this approach follows a natural science paradigm as opposed to the design science methodology of this thesis (cf. Chapter 3). In other words, it asks which practices are there (in literature or in practice), not which practices are required to attain certain criteria.

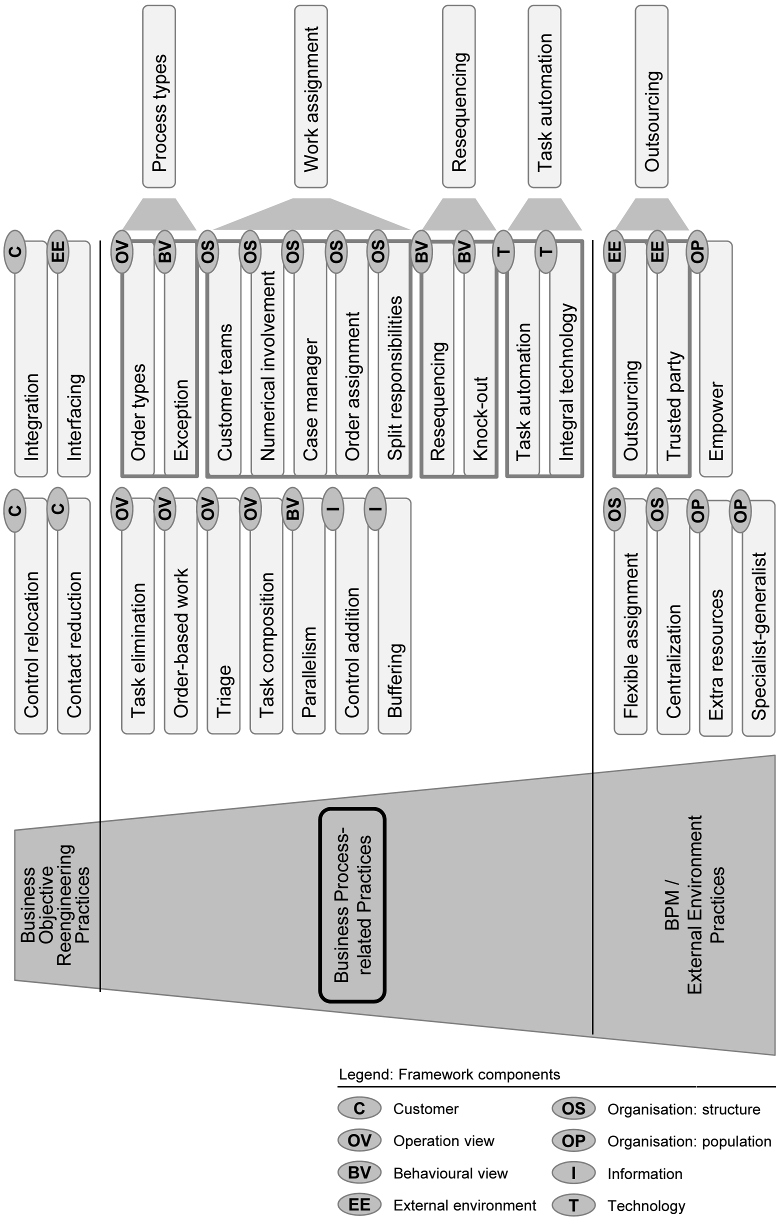



Figure 11.1: Consolidated View on Reengineering Best Practices

- The *Order types*, *Triage*, and *Exception* best practices pertain to the categorization of tasks or entire processes according to variations comprised in the respective business objective. *Order types* stipulates classification according to the type of customer order, while *Exception* stipulates separate tasks or processes for process variants which occur rarely. *Triage* covers the general case. Since these practices will result in specialized sub-types of the business process, they can be jointly considered as the *Process types* practice.
- The *Customer teams*, *Numerical involvement*, *Case manager*, *Order assignment*, and *Split responsibilities* are all based on the assignment of tasks to organizational roles in a business process. *Customer teams*, *Case manager*, and *Order assignment* pertain to creating organizational roles to handle particular customers, case types, or orders. *Numerical involvement* and *Split responsibilities* both relate to reducing the number of contact partners involved in process instances (comparable to *Case manager*). Accordingly, these practices can be subsumed as the *Work assignment* practice.
- The *Knock-out* practice [210] pertains to ordering tasks in decision processes involving multiple stages (e.g., auditing the dimensions of an object in more than one dimension) in a way to minimize effort required. As opposed to the description in [130], this is understood as ordering activities with increasing enactment effort and decreasing probability to reach an early decision (i.e., the decision can be achieved with as little effort as possible). This is a special case of the *Resequencing* practice which pertains to improving the order of activities in a process model in general.
- The *Task automation* and *Integral technology* practices both address the use of technology to automate process activities, thereby reducing manual effort during process enactment. Therefore, they may be consolidated into the *Task automation* practice.
- Both the *Outsourcing* and *Trusted party* practices pertain to using third party resources to enact parts of a business process. In the case of *Outsourcing*, the third party is employed as a paid service provider while the exact (contractual) relationship with a *Trusted party* is not further clarified. Since in both cases a governing agreement (i.e., an SLA, cf. Section 9.4) can be assumed to be in place, this distinction is irrelevant, and both practices can be viewed as *Outsourcing*.

In the next step, the resulting set of consolidated reengineering best practices can be compared to the quality attributes discussed in Chapter 8 and further detailed in Appendix C. Note that only *BP-related practices* are considered since only these support the notion of BP quality pursued in this thesis.

Moreover, out of the full set of quality attributes presented in Chapter 8, only quality drivers (i.e., inductive quality attributes which pertain to the design & implementation lifecycle stage and can be assessed by inspecting the actual process model) are relevant because the reengineering practices do not address process enactment results neither. Table 11.1 presents the respective results. For each quality driver, it shows either the corresponding (possibly consolidated) reengineering best practice according to Figure 11.1, or a short interpretation in case there is no analogy.

| **Quality Drivers** | **Reengineering Practices Analogy** |
|---|---|
| *Task Level Quality Drivers* | |
| *QA 1: Sufficiency of state operations* | *No analogy:* The quality driver is facilitated by the formal definition of business objectives (cf. Chapter 7) and by quality-aware BP modeling (cf. Chapter 7). |
| *QA 2: Effective tasks* | The *Task elimination* practice also addresses the removal of tasks which do not provide additional value. |
| *QA 3: Effective state operations* | *No analogy:* The quality driver addresses the level of individual state operations, which is not considered in the more aggregate view of reengineering practices. |
| *QA 4: Reasonable task resource requirements* | *No analogy:* There is no corresponding reengineering practice. |
| *QA 5: Task automation / use of capital investments* | The consolidated *Task automation* practice corresponds to the quality driver. |
| *QA 6: Task classification* | The consolidated *Tasks typing* practice also addresses the issue of designing alternative tasks to enable employing specialized resources. |
| *Control Flow Level Quality Drivers* | |
| *QA 7: Consideration of conditional propositions* | *No analogy:* This quality driver is enabled only if business objectives are considered. In addition, conditional and task-requisite BSDs must be reflected or formalized in the BP model. |
| *QA 8: Completeness of control flow* | *No analogy:* This quality driver is enabled only if business objectives are considered. In addition, the impact of tasks on target elements must be modeled e.g. via state operations. |
| *QA 9: Effective target aspects* | *No analogy:* This quality driver is enabled only by matching target aspects in business objective models and virtual control flow elements as the result of consolidated possible enactment paths (cf. Section 7.4). |
| *QA 10: Effective and efficacious conditional splits* | *No analogy:* This quality driver is enabled only by combining business objective models and quality-aware BP models. |
| *QA 11: Sequential tasks composition* | The *Task composition* practice reflects this quality driver without, however, considering the aspect of task-requisite BSDs that determine future enactability. |
| *QA 12: Parallel tasks composition* | The *Parallelism* practice addresses the same underlying design paradigm. However, while the quality driver stipulates “less parallelism” to potentially reduce resource consumption, the reengineering practice stipulates “more parallelism” to potentially shorten cycle times. To fully address this topic, cf. the quality meter *QA 27: Timely process enactment.* |
| *QA 13: Alternative tasks composition* | *No analogy:* This quality driver is not reflected in reengineering practices. |

| | |
|---|---|
| *QA 14: Mitigation of repetitive loops* | *No analogy:* This quality driver is not reflected in reengineering practices. It is partially enabled by quality-aware process modeling in terms of identifying non-value creating loops. |
| *QA 15: Early approval or disapproval* | The consolidated *Resequencing* practice addresses this quality driver, application scenarios are in particular reflected in the *Knock-out* practice as one of its constituents. |
| *QA 16: Early failure* | The consolidated *Resequencing* practice reflects this driver in part. However, the underlying principle of considering task-requisite BSDs' possibility of failure can only be fully addressed by using the quality-aware modeling approach. |
| *QA 17: Effective tasks in enactment paths* | *No analogy:* This quality driver is enabled only by utilizing quality-aware process modeling and business objective models. |
| *QA 18: Effective state operations in enactment paths* | *No analogy:* Like *Effective tasks in enactment paths*, this quality driver is based on business objectives and quality-aware BP modeling. |
| *QA 19: Routing automation* | This quality driver is partially reflected in the consolidated *Task automation* reengineering practice which, however, does not explicitly cite routing or workflow automation. |
| *Conceptual Level Quality Drivers* | |
| *QA 20: Consideration of good practices* | *No analogy:* This quality driver pertains to domain-specific practices and is therefore not included in the general set of reengineering practices. |
| *QA 21: Additional control procedures* | The *Control addition* also addresses supplementary controls within process models. However, the quality driver also reflects the impact of early controls on resource consumption, which is not considered in the reengineering practice. |
| *QA 22: Appropriate organizational responsibilities* | The quality driver is reflected in the consolidated *Work assignment* practice which integrates a total of five original practices. |
| *QA 23: Functional integration* | Like this quality driver, the *Order-based work* practice pertains to balancing the triggering of individual activities between starting the activity for each individual process instance ("order-based") against consolidating the activity for multiple process instances ("functional integration"). Interestingly, the former is propagated by the reengineering practice, presumably to minimize cycle times, while the resource-based view of the quality driver rather calls for the latter principle. |
| *QA 24: Overall efficacy and efficiency* | *No analogy:* this quality driver is not reflected in reengineering practices since it pertains to the final overall consideration of quality relations (cf. Figure 7.1). |

Table 11.1: Comparing Reengineering Practices to Quality Drivers

In addition to the reengineering practices included in Table 11.1, the *Buffering* practice is not reflected in the set of quality attributes. *Buffering* pertains to replacing activities aimed at obtaining information from sources external to the organization by instead referring to internal "information buffers". Presumably, this practice is aimed at reducing cycle times instead of resource requirements. According to the view on cycle times argued in this thesis (cf. Section 8.4.2), it has not been included as a quality attribute. Nevertheless, it must be conceded that *Buffering* potentially alters the nature of resource requirements posed to achieving a business objective, and might thus be a valuable consideration beyond the quality attributes presented here. Nevertheless, it requires making additional assumptions regarding the external environment (i.e., that the information required can be "subscribed to"), and probably the implementation of a separate "buffering" process to "outsource" information gathering from the originally considered process. This also reflects the observation that the buffering practice could not be validated in literature or empirically.

Out of a total of 24 quality drivers, 11 are comparable to reengineering practices. Note that in some cases, however, quality drivers refer to concepts developed in Part II of this thesis to enhance the underlying analysis in terms of content and rigor. In the case of *QA 12: Parallel tasks composition* and *QA 23: Functional integration*, the quality drivers even come to differing conclusions on how to improve business processes in comparison to the reengineering practices. The major reason behind this is that the notion of quality contained in Definition 1 is based on containing resource requirements in contrast to minimizing cycle times. 13 quality drivers are not comparable to available reengineering best practices. Therein, nine cases are based on business objectives modeling and / or quality-aware BP modeling. This observation further stresses the role of formalized business objectives and quality-aware BP modeling for the deduction of appropriate quality attributes.

## 11.2 Wide vs. Focused Business Process Quality

This section compares the set of quality attributes in this thesis to earlier results in the field of BP quality. Chapter 4 stipulated that the *top-down approach* used in this thesis and based on deducting quality attributes through appropriate typing (cf. Section 8.2) would deliver more comprehensive results than the *bottom-up approach* of transferring quality attributes from related fields or personal experience.

To validate this stipulation, this section refers to the results presented in [123, 125]. Both have been shortly discussed in Section 4.2 already, and will now be assessed in more detail in the light of the observations made in Part II of this thesis. Interestingly, both approaches take a wider view of BP quality than the notion pursued in this thesis, since the respective understanding of process quality comprises the quality of the *outer environment* (cf. Section 5.1) as well:

- In [123], the "dimensions of BP quality" include not only "function quality", but also "input / output quality", "non-human resource quality" and "human resource quality".
- In [125], "BP quality characteristics" pertain to "resource characteristics", "actor characteristics", and "information and physical object characteristics" besides "activity characteristics".

As discussed in Chapter 5, this thesis does not concur with that view. In the light of the definition of BP quality as a design artifact [103], both views have to be appraised according to their merits as management instruments. The *wide view on BP quality* endorsed in [123]

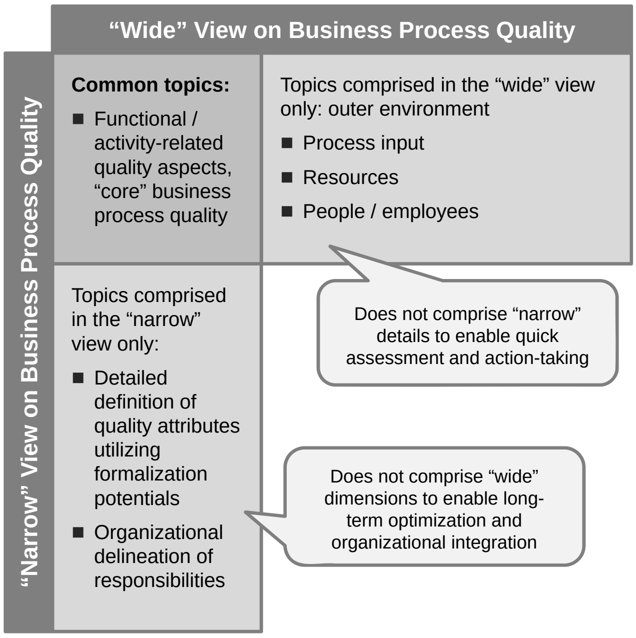



Figure 11.2: Wide vs. Focused Views on Business Process Quality

and [125] aims at an easy-to-handle tool to identify and eliminate current issues in business processes and the related BPM framework. From this perspective, it makes sense to include the outer environment of a business process as well, since issues should be resolved quickly and comprehensively (cf. the Process Root Cause Analysis approach [269]). The underlying notions of quality, business processes, and actors' roles are of secondary concern.

The *focused view of BP quality* pursued in this thesis aims at the ongoing improvement of business processes. To this end, it is necessary to delineate the "core" quality of a business process from the quality of its outer environment including, for example, the BPM framework of the organization. Otherwise, it will not be possible to integrate BP quality management into today's organizational frameworks. These considerations are reflected in the discussion of effectiveness criteria for BP quality concepts in Section 3.1 on the basis of related work dealing with managerial control and governance. This also leads to a more detailed and concise definition of individual quality attributes based on business objectives (cf. Chapter 6) and quality-aware BP modeling (cf. Chapter 7). Figure 11.2 provides a summary overview on both perspectives. Example 54 illustrates the observation that both views are useful for particular purposes, respectively.

**Example 54 (Wide vs. Focused Views on Quality).** Consider the automotive industry. From a manufacturer's perspective, it is sensible to assess the quality of produced cars in order to control both its research and development function as well as its manufacturing plants. However, the quality of a car is not determined by the quality of its driver. While car designs need to take into account drivers' capabilities, the quality of streets etc., these topics are part of the requirements definition for the cars manufactured. In the sense of aspects of a *business objective* (cf. Chapter 6), they need to be considered, but better streets will not raise the quality of the car as a product, and should therefore not be a factor in employee incentivation or other managerial methods.

On the other hand, from the perspective of traffic authorities, it also makes sense to appraise the quality of driving as a whole, taking into account not only cars, but also drivers, streets, traffic volume etc. This way, it is possible to identify the "weakest link" of factors which impact the quality of driving, and direct measures accordingly.

Based on these considerations, the remainder of this section is structured in two steps:

1. *Consolidate* the aspects described by Heravizadeh, Mendling and Rosemann on the one hand, and Heinrich and Paech on the other hand into one common set. Only the quality dimension "function quality" and the quality characteristic "activity characteristics", respectively, are considered, since issues related to the outer environment are specific to the wide view on BP quality.
2. *Compare* the consolidated set of quality aspects from related approaches to the set of quality attributes developed in Chapter 8. Additional details besides the semantic content of quality attributes (in particular, quality criteria and predicates) are not considered since they are specific to the narrow view on BP quality.
3. *Discuss* results to obtain relevant insights.

To obtain a viable base for the comparison, the aspects described by Heravizadeh, Mendling and Rosemann on the one hand, and Heinrich and Paech on the other hand are first consolidated into one common set. Heravizadeh et al.'s view on BP quality is based on the Process Root Cause Analysis approach developed by the same authors [269]. While this approach pertains to identifying impacting factors on violations of *soft goals* [199], the respective topics relevant to BP quality in the "function" dimension are identified on the basis of work looking into software engineering quality [261, 270].

Heinrich and Paech also adopt a view on BP quality that reflects common notions on software quality. Accordingly, "quality characteristics" represent common categories of software quality. These are refined into more detailed "quality attributes" and "measures" grounded on the authors' personal experience. Again, quality attributes for the "activity" characteristic are based on the ISO 9126 standard on software quality [261].[2] Any adaptations to reflect BPM specifics are marked by the authors. Accordingly, it is not surprising that quality aspects cited in both approaches do not differ widely. Figure 11.3 presents an overview, linking aspects considered in both approaches.

According to Figure 11.3, all aspects comprised in the "function" dimension of [123] are comprised in the "activity" characteristic of [125] as well. In the following, the latter approach is therefore used as a basis for comparison. Note that the "portability" attribute has been removed for the comparison since, for the purposes of BPM, there is a substantial overlap to "maintainability". From a practical perspective, it makes no difference whether

[2] This standard has by now been been replaced by the ISO 25000 family of standards. More specifically, the ISO model for software quality can be found in ISO 25010 [271]. The new norm retains the general hierarchy used in its predecessor [272].

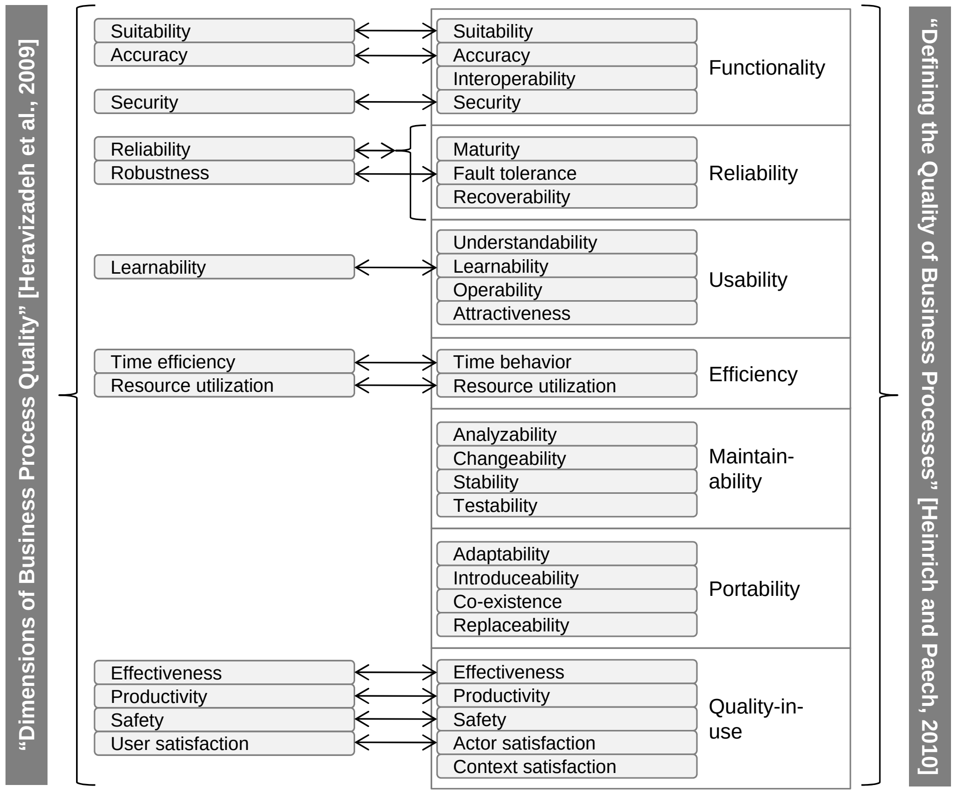



Figure 11.3: Function- and Activity-related Quality Aspects in "Wide View" Business Process Quality Approaches

a process needs to be adjusted because of a new context of use, or for another reason. In addition, the set of quality attributes and measures in [125] comprises a number of topics which do not pertain to the quality of a business process (cf. Section 2.1), but to the quality of the underlying BPM methods, tools, and systems. Note that in terms of software quality, this distinction is not made, since decisions on development environment, programming languages, software engineering tools etc. are – within a framework given by the organization – usually taken as part of the software engineering project [273]. For the design and implementation of individual business processes, this is generally not the case. Rather, the *BPM environment* and *individual business processes* are managed in a layered model comparable, for example, to multi-layer architecture patterns in software engineering [274]. Accordingly, including the respective topics would result in a violation of the principle of *exclusive coverage* (cf. Section 3.1):

- Within the "functionality" characteristic, "security" pertains to appropriate access restrictions to physical or information objects. This is not an issue of the business process, but of its underlying BPM environment or even the physical environment of the organization.

- The "reliability" characteristic addresses the stability of process enactment in case of faulty operation ("maturity" and "fault tolerance"), and the capability to recover from failures. Again, these topics are typically not addressed through the design and enactment of an individual process, but by the underlying BPM environment (e.g., by a WfMS). Note that contemporary process modeling languages provide functionality to model error handling, but these are rarely used in practice, and rely on the capabilities of the underlying BPM environment [80].
- Within the "usability" characteristic, "learnability" and "operability" refer to enabling users to learn how activities are enacted, and to actually enact activities. For processes implemented in a PAIS or through other capital goods within a BPM environment, these topics are largely determined by the underlying system instead of the design of the business process.
- The "maintainability" characteristic pertains to whether activities can be adapted to new requirements. Again, this characteristic is determined by the BPM environment of the business process, but not by the business process itself. In this context, the field of BP flexibility constitutes an important area of research [260, 25]. Note that the same considerations apply to the "portability" characteristic, which has been excluded as discussed above.
- Within the "quality-in-use" characteristic, "safety" reflects the issue of avoiding harm to people, property, or the environment. This topic is also mainly determined by the underlying BPM system of the business process.[3]

Figure 11.4 summarizes the considerations made above, and provides an initial example of how the topics reflecting BP quality, but not BPM quality can be matched against the quality attributes developed in Chapter 8.

In Figure 11.4, some BP characteristics are identified by a question mark to indicate the requirement of a more detailed discussion since they cannot be reconciled with the view on quality propagated in this thesis:

- Within the "functionality" characteristic, "accuracy" pertains to whether activities deliver "the needed degree of precision" in terms of their results. While this is an important issue for software used in research and development, manufacturing, and similar fields, it is not a topic apt to be applied to the general and administrative processes which constitute the scope of this thesis and BPM methods applied in practice. However, if precision requirements should arise, they can be modeled as part of business objectives, and are covered by efficacy-related quality attributes.
- Within the "functionality" characteristic, "interoperability" refers to the capability of activities to be "executed before or after one or more other specified activities". Since the issue of resource interdependencies is already covered by "suitability", the meaning behind this aspect does not become entirely clear. It can therefore not be reconciled with the view on BP quality assumed in this thesis.
- Within the "usability" characteristic, "understandability" addresses an activity' capability "to enable the actor to understand whether it is suitable, and how it can be

[3] Note that this assumes that safety requirements towards the process are modeled with the business objective, i.e, that they are recognized as requirements towards the process depending on the environmental elements handled. This view concurs with the observations made in Chapter 6.

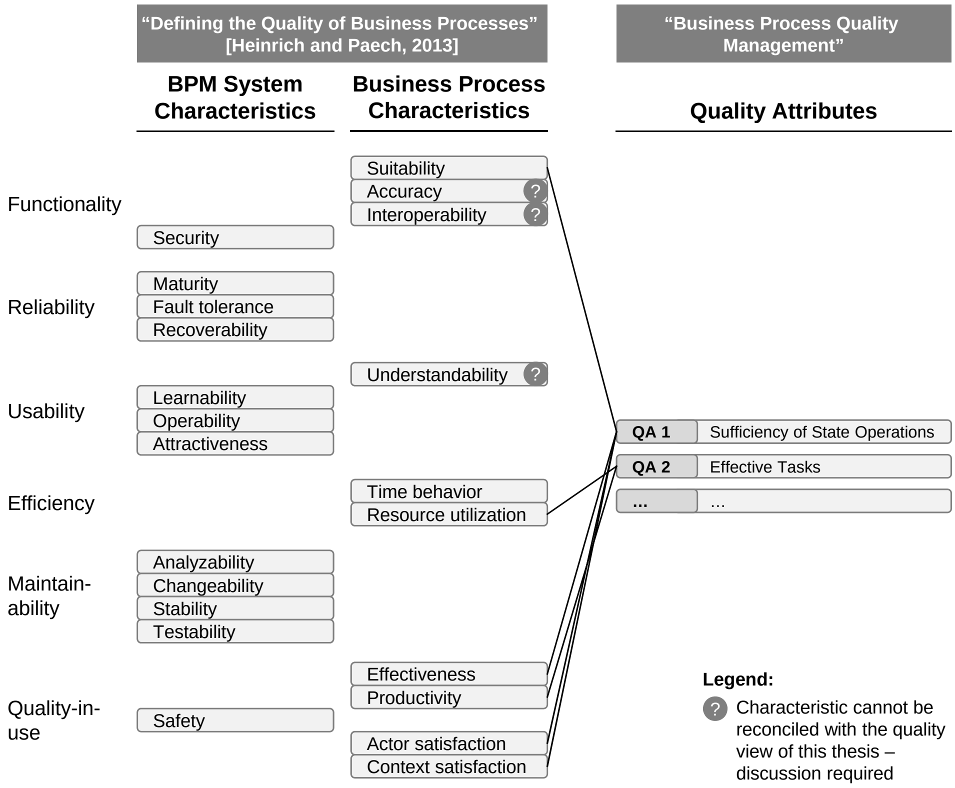



Figure 11.4: Matching BPM System Characteristics and Business Process Characteristics

executed in a particular context of use". In particular when considering WfMSs, this is one of the major reasons for organizations to implement BPM concepts – BPM facilitates triggering appropriate activities by users independently of the content of the respective activity or the judgment of an user whether it might be appropriate. Hence, this aspect is not to be considered as a matter of the quality of an individual business process. However, its purpose becomes clear when considering its original context in software development.

In addition, Figure 11.4 comprises the first two quality attributes from Chapter 8 as examples for reconciliation. Notably, the relations between the two sets of quality aspects are rather complex. That is, aspects stipulated in this thesis are reflected in many aspects of [125], and vice versa. In this context, it is instructive to revert to the quality relations regarding efficacy and efficiency (cf. Figure 7.1). As shown in Figure 11.5, relevant quality aspects from [125] can be assigned to the *efficacy* relation, i.e., the question whether and under which circumstances a business objective is achieved, or to the *efficiency* relation, i.e., the question which resources must be consumed to achieve a business objective.

On that basis, it is possible to discuss the underlying reasons for the deviations between the structure of quality aspects stipulated in [125] (or, ultimately, [261] as the ISO norm on software engineering quality), and the 28 quality attributes of Chapter 8. In this regard, two

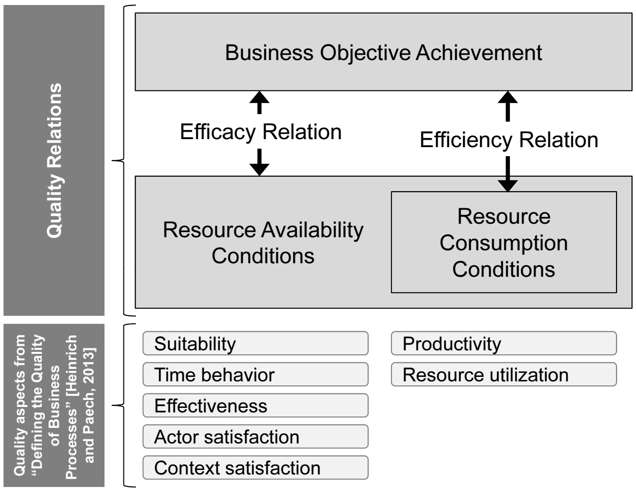



Figure 11.5: Quality Aspects and Quality Relations

topics are relevant: firstly, the distinction between various dimensions of business objectives, and, secondly, the aim of supporting BP improvement. The first topic reflects the impact of formal business objective modeling as enabled by Chapters 6 and 7. Since business objective models bundle requirements towards the outcomes of business processes, it is not necessary to distinguish general "suitability", "effectiveness"[4], "context satisfaction" etc. as reflected in [125]. Rather, these aspects are subsumed in the *efficacy* of a business process.

The second topic, i.e., the support provided to BP improvement within a quality-aware BP lifecycle (cf. Chapter 9), entails the differing level of detail encountered in the two approaches, and the scope of the underlying quality model. The focused approach pursued in this thesis explicitly aims at supporting the improvement of business processes (cf. Chapter 1). To this end, all design & implementation and enactment factors of business processes are analyzed in order to identify 28 quality attributes which are amended with quality criteria and quality predicates (cf. Figure 8.2). This approach results in a differentiated picture reflecting the root causes of BP quality [237].[5] The set of quality characteristics in [125], on the other hand, is oriented at the perspectives from which quality issues may emerge. This approach is, for example, reflected in the differentiation between "actor satisfaction" and "context satisfaction": rather than describing the underlying issues which may lead to deviations between process outcomes and requirements, the two aspects pertain to the fact that deviations may become eminent with users or customers of a business process.

In the light of Section 8.2's discussion of quality drivers and quality meters, it may therefore be concluded that the quality models of related work [123, 125] focus on *quality meters* in the sense of attributes that can be appraised by considering the affecting and the affected environment of a business process, but not the internal mechanisms of the business pro-

[4]Note that the term "effectiveness" propagated in the respective quality aspect in [125] corresponds to the term "efficacy" used in this thesis.

[5]The Process Root Cause Analysis approach pursued by Heravizadeh et al. has not resulted in a similar structure. Rather, the authors also adopt software engineering quality characteristics, as discussed above.

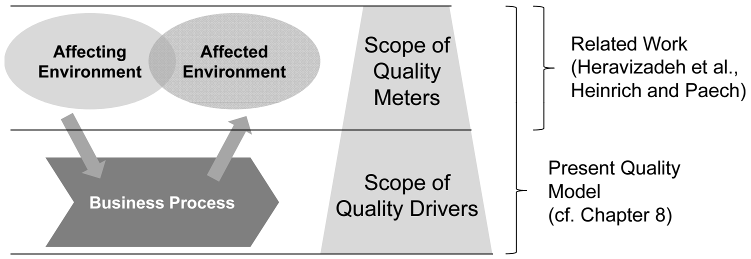



Figure 11.6: Quality Models Scope

cess itself in the sense of *quality drivers*. The latter are, however, required to drive process improvement. The resulting differences regarding the scope of quality models are summarized in Figure 11.6. In this context, note that the quality model of this thesis stipulates that quality meters build upon the content of quality drivers in order to ensure appropriate delineation of responsibilities in a quality-aware BP lifecycle (cf. Chapter 9).

In summary, the contribution of the quality model of this thesis beyond related work based on software engineering results relates to the following aspects:

- The quality model of this thesis is based on considerations spanning organizational targets, common business process and BPM concepts, managerial analysis and control, and the resulting effectiveness criteria in the sense of the design science approach (cf. Section 3.1). Hence, it results in a set of quality attributes where the contribution of each attribute towards BP quality becomes apparent. This is not the case for all quality aspects transferred from standards towards quality in software engineering.
- The deductive approach used in this thesis to derive a quality model for business processes ensures that the impact of business processes on organizational targets is covered more comprehensively. This is reflected in the scope of the quality model, which covers not only quality meters, but also quality drivers.
- This thesis provides a more detailed definition of quality attributes including the associated quality criteria and predicates (cf. Appendix C). This is an important factor to enable BP improvement.

## 11.3 Conclusion

This chapter compared major results of this thesis to related work from the fields of BP reengineering and BP quality management. Respective differences could thus be identified and discussed. In turn, this enabled determining whether the concepts developed in this thesis deliver an additional contribution in comparison to what has been available already, as required by the design science approach (cf. Chapter 3 and [103]).

To this end, quality drivers addressing the design & implementation of business processes were compared to a summary collection of reengineering practices, resulting in a total of 13 out of 24 quality drivers which are not yet reflected in common reengineering practices.

The main reason behind this is that the concepts of business objective and quality-aware BP modeling (cf. Chapters 6 and 7) developed in this thesis enable alternative and additional perspectives on BP quality. In addition, Section 11.2 discussed differences between approaches towards BP quality based on related work from the field of software engineering and the results of the methodology of the present thesis.

In both comparisons, it became apparent that the structured approach towards business objectives and quality-aware BP modeling developed in Chapters 6 and 7 is a key factor in analyzing BP quality beyond the previous state of the art. This characteristic can be considered as an additional demonstration of the "value or utility" of these design artifacts in the sense of the design science paradigm [77]. Accordingly, facilitating the validation of the concepts developed for business objective and quality-aware BP modeling constitutes a further aspect of the contribution of this chapter.

# 12 Demonstrating the Effectiveness of Quality Improvement Measures

As an additional building block to foster the practical applicability of the concepts developed in this thesis, this chapter deals with validating process quality improvement measures. In this context, it is important to recognize fundamental limitations regarding the empirical validation of the set of quality attributes presented in Chapter 8. These are described in the following.

In practice, it is not feasible to obtain a real-world example process where the business value of each individual quality attribute can be demonstrated by revealing substantial BP improvement potentials. The reason behind this is that real-world business processes will exhibit quality issues regarding particular aspects, but not with respect to the entire range of conceivable quality attributes at the same time. Individual case studies or experience reports will thus not suffice to validate the entire set of quality attributes.

Moreover, taking into account quality-aware modeling when identifying quality attributes delivered a quality model adding new content to the available body of knowledge (cf. Chapter 11). It is thus not possible to validate the quality model through a literature research compiling available empirical results, e.g. on past implementation projects.[1]

To address this issue, this chapter provides a generic approach that can be used to individually validate quality attributes in terms of their contribution to BP improvement based on a specific application scenario. The approach developed in this chapter is demonstrated along a substantial and existing real-world business process. It thus needs to be applicable regardless of formalized process quality management methods that are not present in a commercial context yet, such as business objective modeling and quality-aware BP modeling. Hence, it also provides a bridge to implement "lightweight" BP quality management, even if methods and systems for business objective and quality-aware BP modeling are not in place. It thus contributes to straightforward practical accessibility of process quality concepts, and may provide an incentive to further invest into BP quality management.

This also means that the approach presented here is not only applicable to the set of quality attributes deducted from the considerations on business objectives and quality-aware processes in this thesis. Rather, it can also accommodate other process quality characteristics or process reengineering practices that can be used for process improvement. To reflect

This chapter is based on the following referred papers:

Lohrmann, M., Reichert, M.: Demonstrating the effectiveness of process improvement patterns. In: Proc. 14th Working Conf. on Business Process Modeling, Development, and Support (BPMDS'13). Volume 147 of LNBIP, Springer (2013) 230–245

Lohrmann, M., Reichert, M.: Effective application of process improvement patterns to business processes. Software & Systems Modeling (2014) DOI 10.1007/s10270-014-0443-z

[1] Note that this approach has been applied by Limam Mansar and Reijers to validate *existing* process redesign best practices [130].

this characteristic, the following sections refer to *"process improvement patterns"* instead of quality attributes.

## 12.1 Introduction

Research on BPM and PAISs has resulted in many contributions that discuss options to improve the quality, performance, and economic viability of business processes [275]. Examples range from individual "best practices" [129] to comprehensive BP quality frameworks [123, 7]. In this context, this chapter refers to *process improvement patterns (PIPs)* as generic concepts for enhancing particular aspects of business processes. As an example, consider decision processes that require to appraise various decision criteria. The respective appraisal tasks can be arranged to reach a decision with as little effort and as quickly as possible. This can be achieved by executing tasks with a high probability of providing sufficient information for a decision and with comparably low execution effort earlier in the process. This principle is known as "knock-out" [210]. It constitutes a first example of a process improvement pattern.

**Example 55 (Knock-out principle).** Consider a process for handling invoices received from suppliers. To determine whether the invoice should be paid, we want to check whether it is in line with purchase order data. In addition, we need to ensure that there is a sign-off from the responsible manager. The former check can be fully automated in the context of ERP systems, and therefore be executed with little effort. Thus, it makes sense to execute this check first, and possibly "knock out" the invoice before incurring the much greater effort of (manual) sign-off.

To ensure practical relevance, the actual *business value* of PIPs needs to be demonstrated to practitioners, thus enabling reasonable implementation decisions. In the context of this issue, there exist many propositions for empirically establishing the effectiveness of PIPs. These include anecdotal evidence [17], case studies [276], and surveys [130]. Commonly, these approaches are based on *ex-post* (i.e., hindsight) appraisal of qualitative evidence given by process managers or other stakeholders to obtain general insights applicable to comparable cases.

However, there still exists a gap regarding the *a-priori* (i.e., in advance) assessment of PIPs considering a *particular application scenario*, which may range from an organization's strategy and goals to its existing business process and information systems landscape. In particular, this gap should be bridged for the following reasons:

- Similar to *design patterns* in software engineering [277], PIPs constitute abstract concepts that may or may not be useful in a particular context. Experience from other scenarios, which may widely differ from the one at hand, is thus not sufficient to take reasonable decisions on the implementation of organizational changes or PAISs.
- *Ex-post evidence* is usually obtained from persons involved in the respective implementation projects. In turn, this leads to a source of bias. Moreover, *a-priori* assessment allows addressing a far wider spectrum of PIPs. In particular, it is not necessary to complete implementation projects before a PIP can be assessed.

- *Combining PAISs with process intelligence tools* [35, 36, 28, 25] opens up new opportunities to quantitatively and qualitatively gauge real-world business processes. This should be leveraged for scenario-specific PIP assessment.
- *Effective PAIS development* requires to consider process improvement potentials *before* any implementation effort is incurred. Accordingly, PAIS development should start with a requirements definition which, in turn, is based on adequate process design considering relevant PIPs.

To enable *a-priori* PIP assessment, this chapter tackles the following challenges:

- *Challenge 1.* Describe an approach towards *a-priori* PIP assessment reflecting and summarizing common practice in the field.
- *Challenge 2.* Evaluate the approach by applying it to a substantial real-world case.
- *Challenge 3.* Reconcile the approach to scientific standards by applying guidelines for empirical IS research.

The remainder of this chapter is structured as follows: Section 12.2 describes the sample process used to illustrate the approach. Section 12.3 presents the approach towards PIP assessment. In the sense of an experience report, Section 12.4 describes the results obtained when applying the approach to the sample process from Section 12.2. Section 12.5 discusses the state of the art in PIP assessment as well as other related work. Finally, Sections 12.6 and 12.7 evaluate the results obtained referring to the challenges discussed above, and conclude the chapter.

## 12.2 Sample Case: Applications Management Process

The business process used to illustrate the concepts presented in this paper stems from the field of human resources management. It addresses the handling of incoming job applications to fill open positions in a professional services firm. Figure 12.1 describes the business objective of this process according to a notation developed in [8]. The objective of the process is to achieve one of two states for each job application: either the application is refused, or a job offer is sent to the applicant. A job offer shall be sent if the following conditions are met: (1) The application documents have been accepted in terms of quality (e.g., with regard to the CV), (2) an interview has taken place with a positive feedback, (3) basic conditions have been agreed on between both parties, and (4) senior management approval has been obtained. If one of these requirements is not met, a letter of refusal has to be sent.

Based on discussions with stakeholders and the results of process mining, it becomes possible to model the business process implementing this business objective using BPMN (cf. Figure 12.2, [80]). For the sake of brevity, the model is slightly simplified, and a detailed description of its elements is omitted. As an example of the relation between the business objective and the process model, consider the conditions the business objective poses towards sending a job offer. The process model transforms these conditions into respective checking activities (e.g., *Technical quality* into *Check documents*) and XOR decision gateways. Note that there is not necessarily a one-on-one relation between conditions and checking

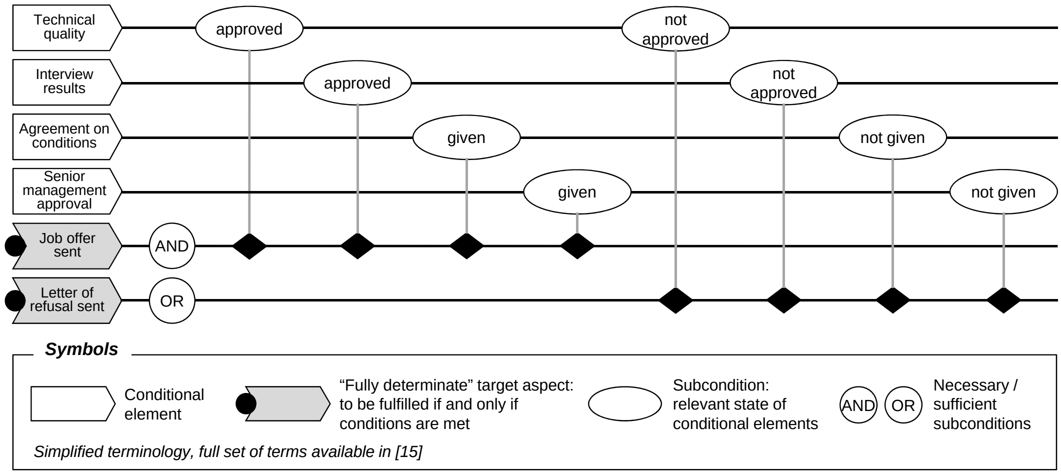



Figure 12.1: Sample Business Objective: Handling Incoming Applications

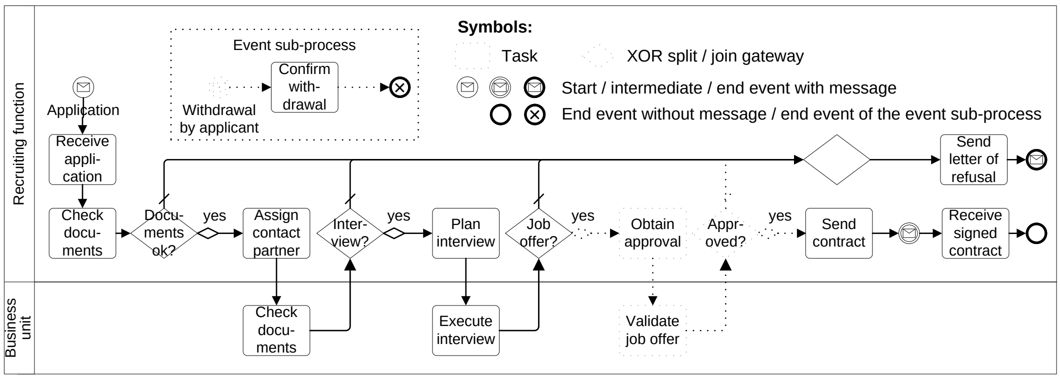



Figure 12.2: Sample Process: Handling Incoming Applications (BPMN notation)

activities. Further, there may be multiple process implementation alternatives for a given business objective (e.g., multiple conditions may be checked within one activity).

Figure 12.3 breaks down the total number of applications handled in a time period of one fiscal year into the number of applications for each possible termination state of the process. Note that the termination states from Figure 12.3 correspond to potential paths through the process model from Figure 12.2. This overview will be referred to when discussing the research execution in Section 12.4. A corresponding data sample of 27,205 process instances was obtained from the log database tables of the PAIS supporting the business process (in this case, an SAP ERP system). Each process instance covers one application. Thus, 1,972 out of the 29,177 applications of Figure 12.3 are not included in the data sample. These comprise, for example, applications handled in the business units without involvement of the HR function. These applications are not traceable in the PAIS.

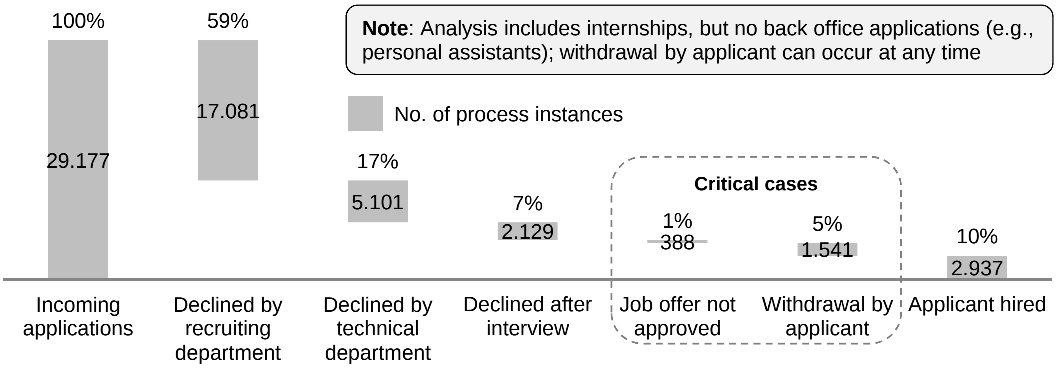



Figure 12.3: Termination States of the Application Process: One Fiscal Year Sample

Figure 12.4 shows a process map generated with the Disco process mining tool [278] when applying it to the data sample.[2] For the sake of readability, this process map has been filtered to solely comprise enactment traces that occur frequently, and events that are relevant for our analyses. The process map is an example of the results that can be generated with process mining tools. In the following, process mining and other techniques are used to analyze the log data sample with respect to process improvement potentials. The process map should be considered as an amendment, but not as a replacement of "traditional" process models such as the one presented in Figure 12.2:

- The process map is based on events logged in the PAIS. Not all events directly reflect a corresponding activity in the process model, and identifiers of events might differ from the ones of corresponding activities. There may be activities not reflected in a logged event or events not triggered by an activity from the process model.

- The process map shows the actual frequency of events in the data sample. Thus, it reflects *as-is* process execution, which may differ from *to-be* process design as recorded in the process model.

- The process map needs to be interpreted with the support of experienced stakeholders. In the sample case of this chapter, for example, application refusal events are used to purge the database of received applications to comply with privacy regulations. Further, not all hirings are handled through the corresponding end events. Issues like these need to be understood when interpreting the process map. However, this understanding is useful for process improvement as well.

[2]The field of process intelligence deals with analyzing the actual enactment of business processes [35]. In this context, *process mining* refers to using processing events logged with a timestamp to generate *process maps*, i.e., graphic representations of actual process enactment traces, and additional process information [28]. Note that Disco was selected as a representative of a number of tools available to practitioners in commercial settings today. Alternatives like ProM [279] or Celonis Discovery [280] might be used as well.

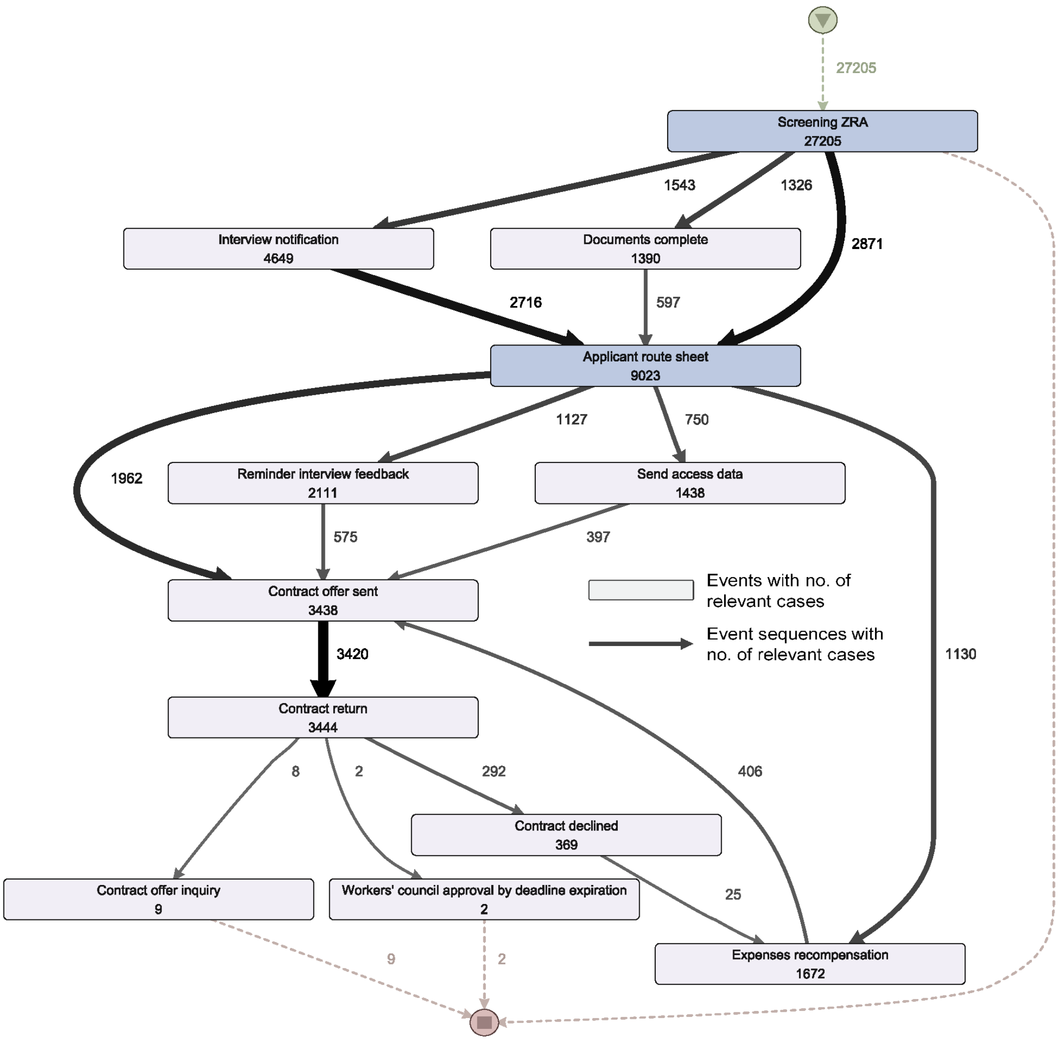



Figure 12.4: Filtered Process Map: One Fiscal Year Data Sample

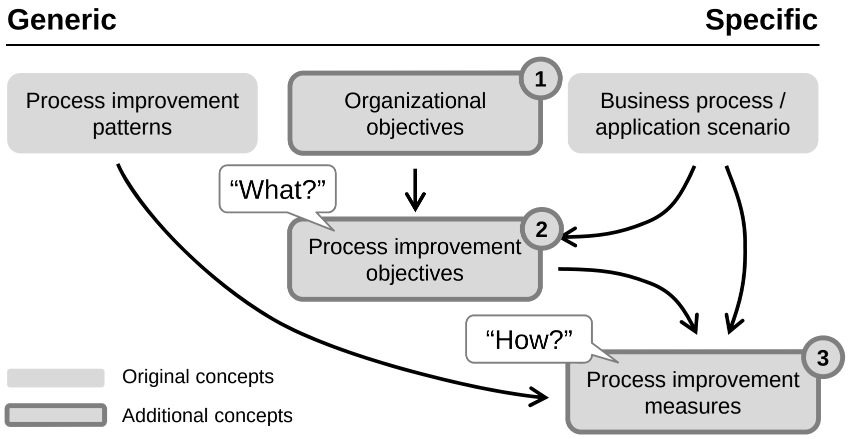



Figure 12.5: Extended Conceptual Framework

## 12.3 Methodology

Like other IS artifacts, PIPs constitute *goal-bound artificial constructs* in the sense of the design science paradigm [103] to be evaluated in terms of "value or utility" [77]. In the context of this chapter, this results in a particular challenge. While PIPs are abstract concepts applicable to a broad range of scenarios, their *business value* must be determined considering the specific use case to enable a decision whether the PIP should be implemented. To this end, an extended conceptual framework as summarized in Figure 12.5 is used.

Beyond the concepts of PIPs and business processes or application scenarios, *organizational objectives*, *process improvement objectives*, and *process improvement measures* are introduced as additional terms:

1. ***Organizational objectives* reflect strategic goals an organization wants to achieve with respect to an application scenario**. Examples of organizational objectives which apply to many scenarios include the effectiveness of process output, cost savings, or compliance with regulations [275, 155]. Note that these examples can be used as a starting point to identify organizational objectives relevant to a particular application scenario. In principle, such objectives are generic, but how they are prioritized against each other is specific to an organization's strategy.

2. ***Process improvement objectives (PIOs)* comprise characteristics that enhance a process considering organizational objectives.** PIOs can be viewed as a refinement of organizational objectives considering the particular challenges associated with a concrete application scenario. In a step-by-step approach, PIOs can be refined into a tree structure, as will be exemplified when discussing our application scenario in Section 12.4. The resulting top-down model is a useful mental technique to ensure a *comprehensive* perspective on process improvement. Note that similar considerations are used in goal-oriented requirements engineering (cf. Section 12.5.3) and value-based management [281]. This procedure can be aborted as soon as the resulting PIOs are sufficiently granular to allow for the application of PIPs. PIOs

thus constitute the "bridge" between abstract organizational objectives and concrete PIPs. The relevance of PIOs to organizational objectives may be evident, or it may require additional validation. As an example of immediately evident PIOs, consider the elimination of obviously redundant tasks to reduce costs. As an example of PIOs that require validation, consider short cycle times. It is not necessarily a strategic goal to enact processes as fast as possible. However, this may be a PIO if a link between *cycle times* and a particular organizational objective (e.g., *reducing costs*) can be demonstrated. PIOs thus provide an additional layer of abstraction as a "shortcut" between improvement measures and organizational objectives. For the above example, potential improvement measures might be validated by demonstrating a positive impact on cycle times instead of overall cost. PIOs can also be viewed as a tool to identify PIPs relevant for the application scenario: Available PIPs are considered with regard to whether they can contribute to a PIO. For example, the parallel execution of formerly sequential tasks constitutes a PIP that may contribute to shorter cycle times as an exemplary PIO. Note that the concept of PIOs corresponds to the identification of stakeholders' goals, which has been proposed as a requirement for empirical IS research in [266].

3. ***Process improvement measures (PIMs)* are bundles of actions considered for joint implementation.**[3] They reflect the application of PIPs to a specific process in order to realize PIOs. Several PIPs may be bundled into one PIM for joint implementation, depending on the given application scenario. As an example of a PIM, consider the implementation of a new workflow tool, which may incorporate multiple abstract PIPs. A PIM thus applies one or more PIPs to a specific business process to address one or more particular PIOs. Assessing PIPs for a particular application scenario thus amounts to the assessment of the business value of corresponding PIMs considering relevant PIOs.

Note that, considering the arrows, Figure 12.5 may also be read as a *top-down method* for process improvement. Section 12.4 further describes its application: General organizational objectives are refined to PIOs specific to the considered business process or application scenario. Then, PIPs relevant to the concrete scenario are selected from a generic set of generally available PIPs, and bundled into concise PIMs. Specifically to the application scenario, PIMs are described in sufficient detail to enable discussing and deciding on their implementation.

Business processes and PIMs, as our unit of study, are implemented by means of PAISs. To maintain scientific rigor, their assessment should take into account requirements known from the empirical evaluation of propositions in software engineering or IS research. In [266], the authors subsume requirements in terms of scientific methodology for evaluation approaches in IS research. Figure 12.6 provides an overview on the basic concepts described there. In the following, this chapter's approach is aligned to [266] by describing how each component is reflected in the present proposition. Note that the (general) statements made should be further refined for each application scenario. From a practical perspective, this will ensure a common understanding by all project participants. Thus, respective considerations are included in the following paragraphs as well.

[3] Note that in the given context, the term "measure" is *not* to be understood as a means of measuring something (e.g., a performance indicator) or as a unit of quantity, but as a coordinated set of activities.

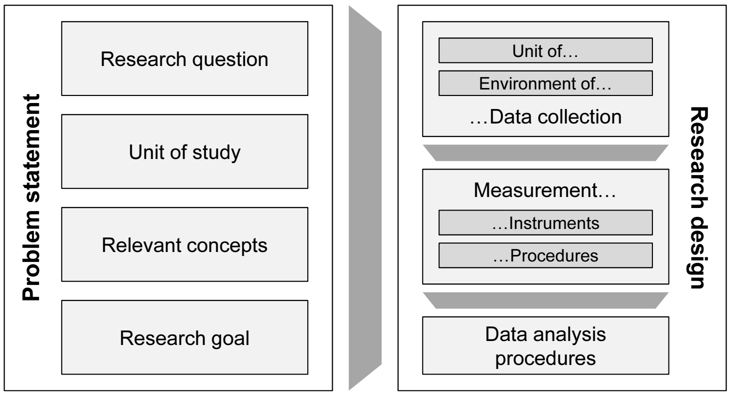



Figure 12.6: Problem Statement and Research Design: Required Components

### 12.3.1 Problem Statement

The first four components addressed constitute the *problem statement* according to [266].

***Research question*** *("What do we want to know?").* Should PIMs be implemented to better meet organizational objectives? Note that this research question refers to PIMs instead of PIPs in order to reflect the goal of *scenario-specific* assessment.

For the sample case, the research question can be refined to the question whether PIMs should be implemented to reduce cost per hire (cf. Section 12.4.1).

***Unit of study*** *("About what?").* The business process to be improved and the proposed PIMs comprising PIPs constitute the unit of study. Effectively selecting PIPs and bundling them into scenario-specific PIMs requires the participation of knowledgeable, but also creative project members. For example, the participants of workshops to discuss PIMs should be carefully selected. In this regard, researchers may contribute a valuable "outside-in view" based on, for example, experience from other scenarios.

Regarding the sample case, the application management process and the proposed PIMs as the unit of study are described in detail in Sects. 12.2 and 12.4.3, respectively.

***Relevant concepts*** *("What do we know in advance?").* Related work to be considered generally includes conceptual work on PIPs, case studies on comparable processes, and benchmarks available for the application scenario. In this regard, it is helpful to ensure proper research of available literature as well as a thorough use of available organizational knowledge (e.g, through selection of appropriate interview partners).

For the exemplary application scenario, we use a framework of process redesign practices [129], own research into PIPs, a cost per hire benchmark, and available research on "knock-out" processes [210].

***Research goal*** *("Why do we want to know?").* Implementing PIMs will result in cost and risks incurred (e.g., process disruptions). To avoid unnecessary cost and risks, implementation decisions should be based on appropriate investigation of whether implementing PIMs will enable better meeting organizational objectives. Implementation decisions should consider not only benefits in day-to-day process operations, but also required investments and future operating cost or total cost of ownership.

### 12.3.2 Research Design

The five components described in the following constitute the *research design* of the approach in terms of *data collection*, *measurement*, and *data analysis*.

***Unit of data collection.*** Understanding the application scenario requires an as-is process description to reflect process design, a process instances sample to reflect process execution, and PIOs to reflect refined organizational objectives. Depending on the application scenario and practical considerations, the process instances sample can be given as a PAIS data extract, as a set of interviews with involved people, as a set of cases directly observed, or as a combination thereof. Assess PIPs requires descriptions of available PIPs, and scenario-specific propositions of PIMs. Note that data collection should cover both process design and actual process execution. This way, PIOs can be identified prospectively (based on process design) *and* retrospectively (based on process execution). Immediate observations are preferable to indirectly related process information. Depending on the application scenario and practical considerations, the process instances sample can be given as a PAIS data extract, as a set of interviews with involved people, as a set of cases directly observed, or as a combination thereof.

Regarding the sample application scenario, it was possible to refer to a business objective model and a flowchart of the process, a statistic on the results of process execution, and a substantial PAIS execution data extract (cf. Section 12.2). To assess PIPs, PIP descriptions available in literature and from our own research, and PIMs as described below (cf. Section 12.4.3) were used.

***Environment of data collection.*** This chapter's proposition primarily aims at improving existing business processes. Hence, data is collected *in the field* to reflect the actual situation as best as possible. The environment of data collection thus generally comprises process stakeholders (i.e., contact partners involved in process execution, recipients of process output, or suppliers of process input) as well as relevant documentation and PAISs. The environment of data collection should be as broad as practically reasonable in order to facilitate identifying all PIOs that are relevant to organizational objectives, and to enable appropriate assessment of PIPs and PIMs.

Regarding the sample case, the environment of data collection comprised the head of recruiting, a business unit HR partner, business unit team managers, the PAIS administrator, and recruiting team members as process stakeholders. In terms of documentation, it was possible to use regular recruiting management reports and PAIS status codes. The PAIS used to support the business process was available as well. As a limitation to the sample environment of data collection, applicants as a group of process stakeholders were not represented in the environment of data collection due to practical requirements. Because of privacy regulations, applicants' contact data may only be used to process the application, but not for other purposes.

***Measurement instruments.*** The approach presented here is based on elaborating PIOs and PIMs in a step-by-step approach. Draft PIOs and PIMs are thus used to document input from the environment of data collection, and constitute measurement instruments comparable to semi-structured questionnaires. These are amended with the proceedings documentation from interviews and workshops (see *measurement procedures* below). In addition, depending on the process instances sample, process execution tracing capabilities in PAISs or SPC procedures also need to be considered. Note that measurement instruments should consider usability criteria with regard to stakeholders involved in measurement procedures. For example, this requires using terms customary to the organization when phrasing PIOs and PIMs.

Regarding the sample application scenario, PIOs and PIMs used as measurement instruments are described in Sections 12.4.2 and 12.4.3. In addition, workshop proceedings, confirmation letters on results reconciliation (via email), and procedures to extract execution data from the PAIS used to manage incoming applications were used.

***Measurement procedures.*** Depending on the application scenario and practical considerations, relevant measurement procedures comprise stakeholder interviews, stakeholder workshops, and questionnaire procedures. Process mining can be used if the sample of process instances is based on a PAIS data extract. Measurement procedures should take into account customary practices of the organization, e.g. by using standard templates for meeting proceedings. On-site measurement procedures (i.e., observing the process in its operations environment) can help to identify additional PIOs to be addressed for process improvement by giving a clearer picture of day-to-day process issues.

Regarding our sample case, telephone and face-to-face interviews with follow-up reconciliation of proceedings, a recruitment center site visit, and process mining with Disco were used.

***Data analysis procedures.*** In general, relevant data analysis procedures include qualitative analysis of workshop and interview results, and quantitative analyses of process instance samples depending on the measurement instruments applied. Note that data analysis procedures need to be flexibly adapted to the step-by-step refinement of PIOs and PIMs, and to the form of quantitative data available on the process instances sample. In practice, this may lead to a mix of tools actually applied. In this context, for example, statistical analysis tools can significantly reduce quantitative analysis effort, and therefore enable enhancing the search scope for relevant PIOs.

Regarding our sample case, a qualitative analysis was conducted together with stakeholders as described in Section 12.4.3. In turn, the quantitative analysis comprised filtering of subprocess views in a process mining tool (Disco), re-extraction of filtered samples and import into a spreadsheet application, conversion of the event log into a “case log” (i.e., an array of events for each process instance), computation of cycle time attributes for each case, and statistical analysis with Minitab.

## 12.4 Sample Case: Process Improvement Patterns Assessment

The extended conceptual framework comprising organizational objectives, PIOs, PIPs, and PIMs (cf. Figure 12.5) as well as the research design are now applied to the sample application scenario. Further, observations regarding the use of tools and systems for empirically analyzing the sample process are summarized. These may be useful for further developments in this regard.

### 12.4.1 Organizational Objectives

As discussed, obtaining clarity about the content and business value of organizational objectives constitutes a fundamental prerequisite to ensure the relevance of PIP assessment. In the sample application field (i.e., recruiting), organizations strive to fill vacant positions quickly, cost-effectively, and with suitable candidates. To achieve these goals, personnel marketing is responsible to generate a sufficient number of suitable applications, while the purpose of our sample process (i.e., managing job applications) is to convert suitable applications into actual hires.

Thus, *organizational objectives* for the sample application scenario include *reducing the time needed until open positions are filled*, *reducing cost per hire*, and *improving the quality of hired applicants*. Out of this set of objectives, *reducing cost per hire* is well suited for illustrating our approach. In particular, the issue of cost is transferable to many other scenarios. More precisely, the following considerations apply for our sample process:

**Reducing cost per hire as organizational objective.** The *cost per hire* key performance indicator captures the total cost of both personnel marketing and applications management. While recruiting cost spent per application is proprietary data, based on experiences from projects with clients an amount of about 400 Euros is assumed. In our sample scenario, generating and managing about 29,000 applications per year would thus result in 11.6m Euros total cost, with cost per hire at around 4,000 Euros. Since hiring cost for talent in professional services will be higher than in, for example, manufacturing, this value corresponds well to the average of 4,285 USD reported as cost per hire for larger organizations by a benchmarking organization [282]. Further, it seems rather conservative considering that professional recruiting consultants commonly charge half a year's salary for successful hires, depending on industry. This calculation demonstrates the high relevance of reducing cost per hire through an improved application handling process.

Note that while *reducing cost per hire* has been chosen to illustrate the approach, the other objectives remain highly relevant. In particular, they need to be kept in mind when designing PIMs to avoid improving the process towards one objective at the expense of others. As an example, improving recruitment cost should not result in eliminating face-to-face interviews with candidates since this would probably reduce cost at the expense of the quality of applicants hired.

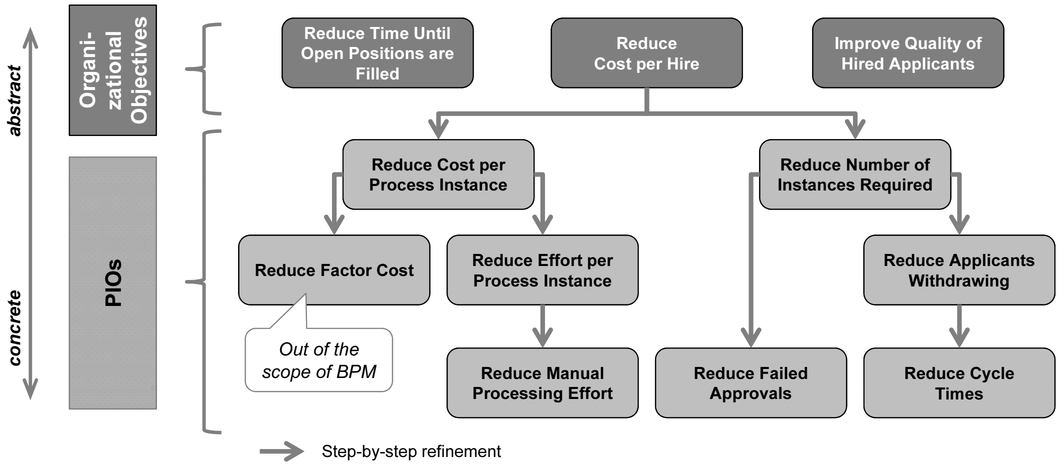



Figure 12.7: Deriving Process Improvement Objectives

## 12.4.2 Process Improvement Objectives (PIOs)

PIOs pertain to characteristics of the business process that affect the organizational objectives one wants to improve on. They serve as a "shortcut" to facilitate discussing the business value of PIMs without reverting to high-level organizational objectives. In the sample case, the organizational objective to *reduce cost per hire* are refined in a step-by-step approach by asking the question what drives cost per hire or, subsequently, the resulting lower-level PIOs.

Figure 12.7 presents relevant aspects of the resulting tree structure: Initially, total cost per hire is driven by the cost associated with each process instance (i.e., with each individual application), and, since it is possible that multiple process instances are needed to fill one position, with the overall number of process instances required. Both aspects may be optimized to reduce cost per hire, but are still rather abstract and will not allow applying PIPs without further refinement.

On the one hand, *cost per process instance* is determined by the cost of production factors (e.g., the cost of employees' working time or the cost of IT systems used) and the amount of effort spent on each process instance. Both drivers will occur in any tree of PIOs dealing with cost reduction. Factor costs, however, are generally unsuitable as a PIO since they are not governed by process designers and managers. Therefore, they cannot be subject to process improvement efforts. Rather, they should be considered as a factor given externally which may affect assessment results. As an example, consider the impact of location decisions on labor costs:

**Example 56 (The impact of factor costs on PIP evaluation).** Particularly in large organizations, it is a common practice to bundle administrative business processes into “shared services organizations” [3]. In this context, labor costs constitute an important factor when deciding on the location of the shared services organization. In turn, this decision impacts considerations on the economic viability of process improvement measures. For example, when considering capital investments to automate manual activities, like matching incoming payments on bank statements against invoices issued to customers, lower labor costs will increase the payback time of the investment, thus rendering its implementation less attractive.

On the other hand, *cost per process instance* is determined by the quantity of production factors associated with each process instance. In the sample process, factors besides manual labor can be neglected, as will be the case for most administrative business processes. Accordingly, *reducing effort per process instance* pertains to *reducing manual processing effort*. This PIO lies at the core of many PIPs commonly used in practice, such as *task elimination*, *task automation*, or *knock-out* [129], and has thus reached a sufficient degree of refinement.

Besides reducing *cost per process instance*, *cost per hire* might be improved by *reducing the number of process instances* required over time. This option corresponds to the elimination of *in-efficacious process instances* that do not terminate in a desirable state according to the underlying business objective. It closely resembles methods applied in common quality management approaches that aim at reducing “causes of poor quality” [283]. In particular, every in-efficacious process instance can be viewed as a quality issue in the business process. Note that the overall effect of quality management on cost and firm performance has been well-recognized and empirically demonstrated [52]. This option is particularly interesting since associated measures can often be implemented with limited investments. Hence, further considerations on the sample case will focus on this PIO.

In the sample case, *cost per hire* is driven by the general efficiency of the application management process, but also by the frequency of process instances terminating in one of the states marked as “critical” in Figure 12.3. The following considerations apply in this regard:

- *Not approving a job offer* after a successful interview may be caused by defective steering of capacities (i.e., job vacancies), defective communication of terms to be offered, or defective review of application documents.
- *Job offers declined by applicants* mostly means that the applicant does not approve of conditions offered, did not have a good impression during the application process, or has decided to take another job offer.

Since terminating the process in these states means that significant effort has been incurred while still failing to hire a promising candidate, organizational objectives are clearly violated: On average, only one out of six applications will successfully pass interviews. However, considering critical cases with defective termination events (cf. Figure 12.3), only one out of ten applicants can be hired. In other words, if the process enactment defects lined out could be fully eliminated, only about 18,000 applications would have to be acquired and managed to cover demand. This would reduce total hiring cost by about 4.6m Euros.

| PIOs | Rationale |
|---|---|
| *Reduce manual processing effort* | Emerging potentials in terms of reducing process enactment effort per instance should be addressed. |
| *Reduce failed approvals* | Final approval of job offers by senior management fails if there are issues regarding vacancy management, reconciliation of terms, or checking of documents. The probability of these "late defects" should be addressed. |
| *Reduce cycle times* | The probability of applicants' obtaining and taking alternative job offers increases with time. Therefore, cycle times between applications being received and job offers made should be as short as possible. |

Table 12.1: Sample Case: Process Improvement Objectives

Based on the considerations made above, the following sections focus on PIOs to reduce effort per process instance or to reduce the probability of the defective process outcomes described. Table 12.1 summarizes the resulting topics.

Note that for the first PIO (i.e., *Reduce manual processing effort*) there is an evident link to our organizational objective of *reducing cost per hire*. However, the second and third PIOs (i.e. *Reduce failed approvals* and *Reduce cycle times*) are based on *hypotheses* on what can be done to reduce process enactment defects affecting the organizational objective. Accordingly, they require qualitative or quantitative evidence to corroborate their relevance for reducing defects, and thus improving cost per hire.

For the second PIO (i.e., *Reduce failed approvals*), qualitative evidence was obtained by interviewing responsible managers, which confirmed the topics described in Table 12.1. Since the reasons for failed approvals are not captured in the PAIS used to manage the application process, quantitative evidence is not available.

For the third PIO (i.e., *Reduce cycle times*) the causal link to its underlying defect of applications withdrawn by candidates is not as obvious. Further quantitative analysis is thus required. Figure 12.8 summarizes the duration between interviews and job offers for the sub-sets of applicants accepting and declining their offer in a boxplot (this part of total cycle time will later be the subject of process improvement, cf. Section 12.4.3). In the figure, the differences between sub-sets regarding quartiles, median, and mean values appear as relatively small. However, a correlation between cycle times and the probability of a candidate to accept or decline a job offer can be statistically demonstrated:

**Correlation between Job Offers Declined and Cycle Times.** It is to be determined whether there is a significant influence of cycle time between application receipt and job offer in weeks on the probability of an applicant accepting or declining a job offer. Accordingly, a binary logistic regression test is used to evaluate the influence of a metric independent variable on a binary dependent variable. For this test, a sample of 2,721 job offers representing about 70% of the annual volume (cf. Figure 12.3) is used. The sample consists of instances fully covered in the PAIS (not all interviews and feedbacks are documented in the PAIS).

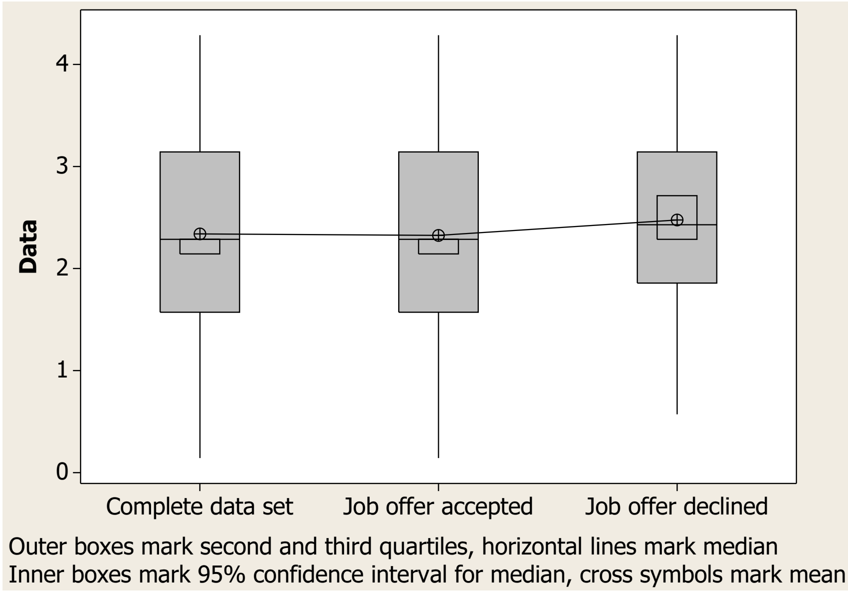



Figure 12.8: Boxplot: Duration Interview to Job Offer in Weeks vs. Acceptance of Job Offers

```
Logistic Regression Table
                                                          Odds
Predictor            Coef        SE                  P   Ratio
Constant         -2.58986   0.169500   -15.2     0.000
Duration_weeks   0.144378  0.0635831    2.27     0.023    1.16     1.02    1.31
```

**P-Value:** Probability of duration *not* being a relevant factor

**Odds Ratio:** Lowering duration by one week expected to reduce withdrawal risk by 16%

Figure 12.9: Minitab Output Excerpt: Binary Regression Test

The sample contains 261 cases where the job offer was eventually declined by the applicant. This is the latest point in the process where withdrawal by the applicant is possible, and a significant amount of effort will have been spent on each respective case. The two samples are independent from each other and have a size of more than 100 cases each. Thus, the binary logistic regression can be applied.

Figure 12.9 shows an excerpt from the output of the statistical software package we used (Minitab). The p-value of less than 0.05 indicates sufficient evidence to assume a significant correlation between the occurrence of withdrawal and cycle time. Regarding the question whether this correlation can be interpreted as as a causal link of cycle times impacting the probability of withdrawal, the following considerations apply:

- Cycle time is measured between receipt of the application and the ultimate feedback to the candidate, whereas the withdrawal sample refers to candidates that declined a job offer *thereafter*. An impact of the occurrence of a withdrawal on cycle time can therefore be ruled out.

- There is a plausible explanation for longer cycle times causing withdrawals: It is possible that candidates find another job while waiting for feedback after an interview. This explanation is substantiated by data on withdrawal reasons collected for a sample of 51 withdrawals between October 2013 and January 2014 for one business unit. The sample covers only cases where a reason was given for the withdrawal. In 33 out of 51 cases, the reason cited was a job offer by a third party, and one may assume that longer cycle times provide more opportunities for candidates to find alternative employment.
- Potential additional independent variables with a positive effect on both cycle times and the probability of withdrawal with HR management were discussed as well. Evidence on such variables could, however, not be obtained. HR managers even mentioned that particularly sought-after candidates, who can be expected to quickly obtain alternative job offers, are handled with higher priority by business units. This effect might even "hide" part of the correlation between cycle times and probability of withdrawal. However, quantitative evidence on this issue is not available.

According to the "Odds Ratio" column, a one week delay can thus be expected to increase the probability of an applicant declining a job offer by 16%.

The significant correlation between cycle times and the probability of withdrawal did not become obvious when just considering median and mean values, but only when executing the binary logic regression test. This observation stresses the necessity to use both sufficient sample sizes as well as appropriate statistical methods when dealing with empirical data on BP enactment.

### 12.4.3 Process Improvement Measures (PIMs)

PIPs relevant to the present application scenario have been selected by considering a "longlist" of known PIPs in terms of potential contributions to the PIOs described above. In this case, the "longlist" comprised PIPs from a framework by Reijers and Limam Mansar on process redesign practices [129] (these are marked with an asterisk "*") as well as from own ongoing research on improving BP quality as detailed in previous chapters of this thesis. However, organizations are not limited with regard to the sources of PIPs that can be used. PIOs are thus used as a mental technique to guide the identification of patterns that are useful for the organization. Relevant PIPs are then bundled into PIMs specific to the application scenario. Table 12.2 summarizes PIOs and corresponding PIMs as bundles of PIPs.

In actual design and implementation projects, it is common to document and track individual PIMs through *measure cards*. In the following, the PIMs from Table 12.2 are described in more detail using this metaphor. For each PIM, a short content description – with PIPs involved marked as *italic* –, and additional details on the following issues are given:

- **Implementation** describes steps required to realize the measure.
- **Business value** appraises the expected implications considering the impact on PIOs as well as implementation effort.

[4] According to [129], knock-out decisions should be ordered "in a decreasing order of effort and in an increasing order of termination probability". Since this would contradict the goal to knock out respective instances with as little effort as possible, it is assumed that knock-out decisions should be arranged in reversed order.

| PIOs | Applicable PIMs with Comprised PIPs |
|---|---|
| *Reduce manual processing effort* | *PIM 1 (Application management automation):* Task automation*, routing automation |
| *Reduce failed approvals* | *PIM 2 (Utilization and capacity management):* Empowerment*, knock-out* [4]<br>*PIM 3 (Standardized terms and conditions):* Triage*, buffering* |
| *Reduce cycle times* | *PIM 4 (Managing interview feedback cycle times for successful applicants):* Control addition*, routing automation, escalation procedure<br>*PIM 5 (Improving application routing):* Case manager*, knock-out*, mitigation of repetitive loops |

* Process redesign practices comprised in [129]

Table 12.2: Defining Process Improvement Measures for Process Improvement Objectives

- **Stakeholder verification** describes the results of validating the PIM through interviews with respective stakeholders.

Note that the PIMs presented here are, by definition, specific to our application scenario. However, their structure as well as the underlying methodology are generally applicable. In terms of content, they exemplify issues commonly found in BP improvement projects, such as the evaluation of IT implementation effort. Moreover, beyond the scope presented here, actual *measure cards* comprise additional information relevant to project management. This includes project planning, project organization, key milestones with "traffic light" status, risks, next steps, and decision requirements. Reporting on measure cards usually takes place in steering committee meetings of senior management.

**PIM Card 1 (Application management automation).** *Our sample process contains manual activities which might be automated using the PAIS. This pertains to the* Assign contact partner *and* Plan interview *tasks enacted by the recruiting function. According to the recruiting statistics presented in Figure 12.3, these tasks occurred in 12,096 and 6,995 process instances, respectively. In this regard, further automation could be achieved by using master data already available in the PAIS.*
***Implementation.*** *The manual assignment of an interview partner might be eliminated by implementing* routing automation *instead. Each process instance in the PAIS is assigned to a job advertisement. This is done either by the candidate when entering his application data via a web platform provided by an external service provider, or by the recruiting function. Since each job ad refers to a set of business unit contact partners, this data might be used to implement* routing automation. *However, automating the assignment of contact partners requires implementing tight control over master data quality to ensure that appropriate contact partners are maintained for all job ads. Currently, this topic remains challenging due to the quantity of job ads, which are specific to service, location, degree of job experience and other factors.*

*In addition, the* Plan interview *task might be* automated *by replacing the verbal feedback on candidates given by business units with structured data including the relevant interview partner.*

*In this case, however, each job ad should be assigned to only one business unit contact partner in order to avoid the need of coordinating feedback from multiple sources.*
***Business value.*** *The manual effort involved in routing an application to a contact partner is estimated to be about 5 minutes by the head of recruiting. Accordingly, about 90 routings can be handled in one working day, resulting in a capacity reduction potential of 0.7 full time equivalents (FTEs) for about 12,000 routings per year. This would amount to about 49,000 Euros annual cost savings. Implementation of automated routing (including the required function of defining deputies for contact partners) is estimated to require about 20 consultant days at a cost of 21,800 Euros. In addition, the necessary master data cleanup of currently about 400 job ads is expected to take about 150 person days including project setup and reconciliation with business units, corresponding to 52,500 Euros one-off cost. Moreover, implementing a workflow to ensure future master data quality when defining job ads is expected to require 35 consultant days at a cost of 38,150 Euros.*

*Implementing automated interview planning builds on the improved quality of contact partner master data, and would require an additional 10 consultant days or 10,900 Euros to implement the required structured interview partner feedback. Interview coordination is currently estimated to take about 15 minutes per process instance, corresponding to 1.1 FTEs or 77,000 Euros annual cost.*

*In total, recurring annual cost savings of 126,000 Euros are to be matched against one-off implementation cost of 123,350 Euros.*
***Stakeholder verification.*** *When discussing this measure with stakeholders, the required change to the assignment of contact partners to job ads proved as the most challenging issue. In particular the definition of location-specific job ads raises concerns with stakeholders that this might lead to candidates applying for multiple job ads simultaneously. With regard to the current policy of defining job ads, the implementation decision on this measure card was postponed to late 2013, because the organization plans to re-examine its entire set of job ads with the goal of substantial reduction. Stakeholders assume that this will improve the "customer experience" of candidates. However, it would require to retain manual assignment of contact partners based on application documents. Note that, counterintuitively, in this case simplification of master data would thus lead to increased manual effort.*

PIM Card 2 addresses the *reduction of failed approvals* as one of the critical cases identified in Figure 12.3.

**PIM Card 2 (Utilization and capacity management).** *Among other reasons, senior managers refuse to approve job offers when the business unit that wants to hire a candidate has excess capacity. In the present case, this can be traced by monitoring personnel utilization, i.e. the rate of hours booked on client projects. If utilization falls below a certain level, there are not enough client projects for present staff, i.e., there is excess capacity. While refusal reasons are not tracked in the PAIS, stakeholder interviews resulted in an estimate of about 30% of total refusals to be caused by this issue. Since candidates' qualifications, in particular in graduate recruiting, are mostly not specific to particular business units, the recruiting department can be* empowered *to route applications to more appropriate teams from the start on. This results in an early* knock out *of applications that would, in the end, be declined because of low utilization.*
***Implementation.*** *To enable utilization-based routing decisions, a new report on utilization per team must be integrated into the application management workflow. Since relevant data is available, and is routinely retrieved for other reporting processes, implementation effort has been estimated to be 25 consultant days or 27,500 Euros. In addition, relevant utilization thresholds*

*must be agreed and communicated. The recruiting center routes about 12,000 applications per year. If the additional operating effort for the utilization check can be assumed to be 10 minutes per application, this results in an overall additional capacity requirement of about 1.2 full time equivalents (FTEs), amounting to approximately 84,000 Euros annual cost.*
***Business value.*** *The PIM is expected to reduce the "late refusal" rate by about 30% or 120 cases per year. Assuming a rate of job offers declined by the applicant of 7% (cf. Section 12.4.2), this would reduce the number of applications to be generated and managed to achieve a constant volume of hires by about 1,200 per year. As we assumed the cost per application to be about 400 Euros, an annual savings potential of 480,000 Euros compares to 27,500 Euros one-off cost and 84,000 Euros operating expenditure per year.*
***Stakeholder verification.*** *When discussing the PIM with senior stakeholders, its business value appeared as rather clear. However, the distribution of utilization data emerged as a "political" issue. Considering present organizational culture, the PIM will not be implemented right away, but the basic capability to add utilization control functionality to the PAIS will be included with the requirements definition for the new PAIS solution to be completed by early 2013.*

The abbreviated measure cards presented above exemplify how PIM implementation benefits can be projected and matched against expected implementation effort. However, beyond this quantitative reasoning, qualitative (or "political") topics may play a role in taking implementation decisions as well, as will be exemplified with PIM Cards 3 and 4.

**PIM Card 3 (Standardized terms and conditions).** *In our sample process, extending contract offers to successful candidates requires final senior management approval. This pertains not only to the candidate, but also to the terms and conditions offered. In this regard, salaries are of particular importance, since this topic is often discussed with the candidate during the interview or in a follow-up conversation. In some cases, this leads to senior management approval not being granted because of terms which have been offered to candidates in verbal agreements. Again, refusal reasons are not tracked in the PAIS, but this issue is estimated to cause about 25% of refusals. To remedy this topic, binding standard terms and conditions could be made available to business unit interview partners. This would implement the information* buffering *PIP since information is provided upfront and does not need to be actively obtained by contact partners. In addition, an additional process step of discussing potential terms in advance between HR and the business unit might be included for particularly relevant candidates (e.g., applicants with job experience). This would implement the* triage *PIP since it would include an additional task for some cases.*
***Implementation.*** *Implementation of this PIM might be achieved by organizational measures without changes to the underlying PAIS, since the upfront discussion of potential conditions can be included in the existing workflow between HR and the business unit. Figure 12.10 presents the corresponding change to the underlying process model (note that the final control flow elements of the process model remain unchanged, and have been omitted). Note that without implementing additional workflow activities or status codes, this change could not be ascertained by process mining. The number of interviews on manager level is available in the data sample, and makes up 5.9% of total volume. Contact partners estimate that about the same relevant volume will be caused by other candidates. Accordingly, approx. 3,200 discussions on terms would have to be conducted annually. Assuming that about 10 discussions can be handled in one working day by HR, this results in an additional required capacity of 1.8 FTEs at about 126,000 Euros cost per year. Additional effort on part of the business unit is not considered. Finally, the cost of compiling and communicating standardized terms and conditions can be*

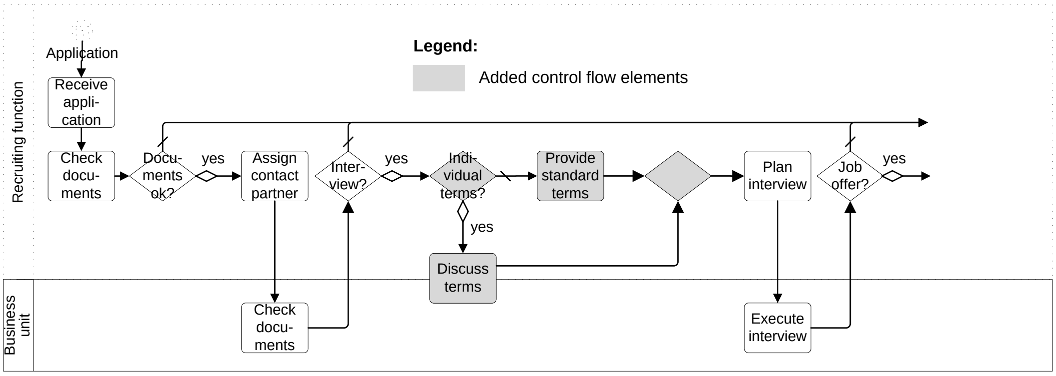



Figure 12.10: Changes to the Sample Process when Implementing PIM 3

*neglected.*
***Business value.*** *The PIM is expected to reduce the "late refusal" rate by about 25% or 100 cases per year, with an annual savings potential of 400,000 Euros according to the calculation presented with PIM 2, in comparison to 126,000 Euros additional operating expenditure per year.*
***Stakeholder verification.*** *The proposition of fully standardized terms and conditions raised concerns with stakeholders that this loss in flexibility might negatively impact chances to win candidates. However, due to privacy regulations, no empirical evidence regarding the impact of terms and conditions (namely, salaries) on the probability of candidates to accept contract offers could be obtained. Therefore, the decision to implement this PIM was postponed until conclusions from an internal confidential assessment are available, which will be additionally reconciled with an HR consultancy.*

PIM Card 4 addresses the *Reduce cycle times* PIO by dealing with one of the underlying drivers for unnecessarily long cycle times.

**PIM Card 4 (Managing interview feedback cycle times for successful applicants).** *The time span between successful interviews and job offers can be reduced by implementing a* control addition*. This means that additional control flow elements are included to ensure the correct enactment of the process. Triggered through* routing automation*, the recruiting department will call the interviewer directly when feedback is not available five business days after an interview. If the interviewer cannot be reached within two business days, an* escalation procedure *will take place by calling the respective supervisor. If no feedback can be obtained through these PIPs within ten business days, a letter of refusal will be sent.*
***Implementation.*** *To implement the PIM, comprehensive tracing of interview dates and an additional workflow with corresponding triggering mechanisms must be implemented in the PAIS. This results in an estimated cost of approx. 38,500 Euros for 35 consultant days. In the data sample used for the binary regression test (cf. Section 12.4.2), about 51% of cases would fall under the proposed regulations. Thus, a total volume of 7,000 interviews conducted annually (cf. Figure 12.3) would result in about 3,600 escalation procedures. On the one hand, this number can be expected to decline over time. On the other hand, multiple phone calls might*

*become necessary for one escalation case. Hence, it is assumed that 20 escalation procedures can be handled in one person day, i.e., an additional 0.9 FTEs are required, resulting in about 63,000 Euros annual cost.*
***Business value.*** *Based on a binary logical regression analysis (cf. Section 12.4.2), we reconciled with stakeholders that the maximum interview feedback time can be reduced to two weeks based on an escalation process. Applying the corresponding odds ratio (cf. Section 12.4.2) to all cases in our sample exceeding this timeframe results in a reduction of 39.2 cases of "late withdrawals" (cf. Figure 12.3). This would reduce the number of applications to be generated and managed by about 390 per year, corresponding to 156,000 Euros in annual savings. Considering additional operating expenditures of 63,000 Euros results in a total annual cost reduction of 93,000 Euros versus a one-off cost of 38,500 Euros.*
***Stakeholder verification.*** *During stakeholder interviews, implementation cost was validated with the application workflow administrator, additional processing effort at the recruiting center with the head of recruiting, and overall viability of the new process with the head of recruiting and the business unit HR partner. The escalation procedure to provide timely feedback was challenged by the business unit HR partner, but not by team managers. Final consent on the positive business value of the PIM could be achieved by discussing the quantitative analysis of the underlying PIO.*

The final PIM identified exemplifies an issue that occurs regularly in a top-down approach as employed in the present case: Since it is possible that similar PIPs can be used to address various PIOs, overlaps in PIMs content may emerge. This topic needs to be considered in implementation planning and when consolidating recommended PIMs into a "management summary" view (e.g. in terms of overall implementation cost and cost savings potentials). In general, it is preferable to consolidate corresponding PIMs into one overall PIM addressing multiple PIOs. However, even in this case, the top-down approach facilitates obtaining a clear overall picture of potentials to be realized by PIP implementation.

**PIM Card 5 (Improving application routing).** *Similar to PIM 1,* Application Management Automation*, this PIM deals with the implications arising from defects in application routing. Besides additional manual effort, loops in application routing, i.e. forwarding applications from one contact partner to another, extend cycle times. Unfortunately, the practice of forwarding application data is currently not tracked in the analyzed PAIS since documents are mostly forwarded manually via email. Accordingly, implications of this practice cannot be quantified. This issue underlines an additional benefit of comprehensive implementation and use of PAISs: achieving transparency over actual process enactment, and thus improvement potentials.*
***Implementation.*** *In terms of implementation, this PIM closely corresponds to PIM 1,* Application Management Automation*, since it requires improving the assignment of business unit contact partners through appropriate master data. Again, implementation necessitates defining one* case manager *contact partner for each job advertisement. Job ads would have to be defined in a manner sufficiently granular to enable this 1:n relation between ads and contact partners, specifying service, grade of job experience, and location. Thus, implementation would enable us to* mitigate repetitive loops *by assigning the right contact partner from the beginning on, and to achieve early* knock-out *of unsuitable candidates.*
***Business value.*** *Based on the available empirical data set, the business value of implementing this PIM cannot be quantified since repetitive loops are not consistently tracked in the analyzed PAIS. To enable tracking, the manual forwarding of application data would have to be replaced*

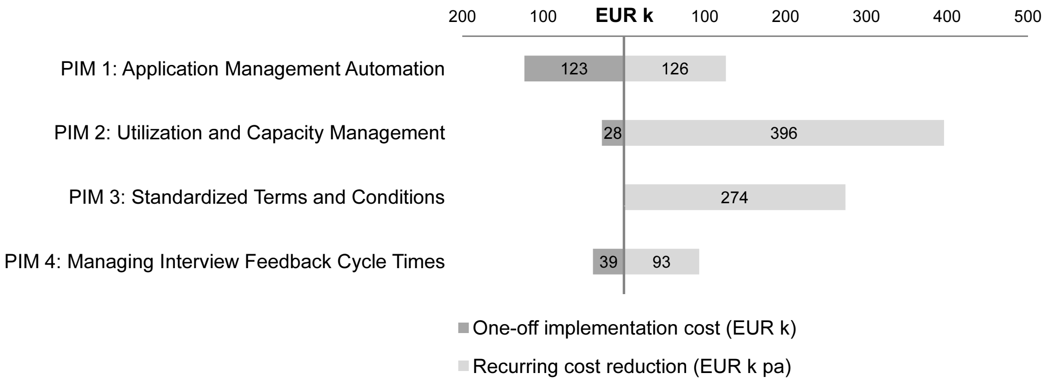



Figure 12.11: Comparison of PIMs

*by a corresponding PAIS function. Once this function has been implemented, it will be possible to assess the impact of loops on cycle times through statistical testing of the impact of the number of loop iterations on overall cycle time.*
***Stakeholder verification.*** *For the purpose of stakeholder discussion, the content of this PIM was merged with PIM 1,* Application Management Automation*. Accordingly, the same implications apply, namely to postpone final implementation decision until the overall restructuring of job ads planned for late 2013. It is expected that the implementation of a new application management PAIS will facilitate quantifying the impact of repetitive loops on cycle times and, ultimately, candidates' probability of withdrawing their application by then.*

Figure 12.11 compares PIMs 1-4 in terms of one-off implementation cost and recurring savings potential per year (i.e., gross cost reduction minus additional operating effort, PIM 5 could not be quantified in this respect). Note that the presented PIMs exhibit a fairly positive business case, with ratios between implementation cost and total annual savings below one year, and that the most positive business cases can be realized by implementing organizational measures without expensive IT implementation. They constitute good examples of a phenomenon often encountered in practice: in many cases, it is interesting to first identify and resolve existing process defects within the framework of available technology before additional process automation is implemented at huge cost. Once these "quick win" potentials have been leveraged, further process automation should be considered.

### 12.4.4 Implementation Results

The process improvement project facilitating our sample case has been concluded in early 2013 with the go-live of the newly implemented PAIS. This section briefly revisits the PIMs discussed above with regard to the results actually achieved. Statements are based on follow-up interviews conducted with stakeholders in March 2014, i.e. about one year after go-live. Interview partners included the head of recruiting operations, the application management process manager, the HR partner of a business unit, and two business unit team managers involved in recruiting (e.g., as interviewers of applicants). To structure the interviews, the available PIM cards were used to collect feedback on their implementation and corresponding results.

**PIM Card 1 (Application management automation).** Discussing this measure with stakeholders during our analysis phase resulted in postponement of the implementation decision because of the required re-structuring of job ads. By now, it has been decided to implement the PIM with slight changes as discussed in the following. This decision has been taken because the demonstration of business value documented in PIM Card 1 has led to increased management attention regarding the avoidable manual effort spent on routing applications. The organization is currently undergoing an effort to significantly reduce variability in job ads. In the future, each job ad will have exactly one contact partner from a business unit assigned who will automatically receive screened applications. If the contact partner wants to pass the application to her colleagues for an interview, she will choose the appropriate person from a contact partner data base. This way, manual routing of applications can be largely eliminated.

**PIM Card 2 (Utilization and capacity management).** The issue of utilization and capacity management has been referred to an entirely new "workforce management system" currently under development. This system will interface with the application management PAIS to avoid the issue of routing applications to teams with limited utilization. Note that this functionality will build on the implementation of PIM 1 as discussed above by controlling proposed contact partners from the contact partners data base.

**PIM Card 3 (Standardized terms and conditions).** Terms and conditions offered to candidates have been reconciled with an HR consultancy. This assessment has led to the result that current terms and conditions are in line with applicable benchmark values. On that basis, a new policy has been issued that requires deviating terms to be reconciled with business unit management. According to our interview partners, this policy has reduced corresponding cases to a minimum, which has led to a significant reduction in "late refusals" as proposed in the PIM.

**PIM Card 4 (Managing interview feedback cycle times for successful applicants).** According to application management reporting, interview feedback cycle times could be reduced to an average value of 1.4 weeks based on implementing this PIM. Concurrently, the share of "late withdrawals" could be reduced to 7.7% for the timespan of October 2013 to March 2014, in comparison to 9.6% in the original data sample we analyzed. However, one needs to keep in mind that this reduction might be caused by varying reasons, such as changes in the labor market environment. Nevertheless, interview partners confirmed their impression that reducing cycle times significantly contributed to this development.

**PIM Card 5 (Improving application routing).** As proposed, this measure is being implemented in conjunction with PIM Card 1, *Application Management Automation*, namely in relation to managing job ads master data. Extensive loops and cycle times during application are now controlled by maintaining the responsibility of initial contact partners for timely feedback, even if the application is passed on to colleagues within the business unit. This way, contact partners have an incentive to avoid redundant loops. Outstanding feedback is then escalated to senior management.

### 12.4.5 Deployment of Tools for Empirical Process Analysis in Practice

This section amends the experience report on the sample case with practical challenges observed when using available technology to drive the empirical analysis of process data. It can be argued that the present sample scenario is fairly typical in this regard, and that the issues described may be useful for the further development of corresponding tools and systems. In the empirical analysis, three types of technology were used: first, a process-aware ERP system, second, a process mining tool, and third, a tool for statistical analysis.

When using the ERP system of the sample process as a source of information for process improvement, it emerged as rather challenging to extract and interpret empirical data on process enactment. The major issues in this respect lay in the complexity of the underlying data model, which was partly subject to customization, its available documentation, and the availability of corresponding analysis and extraction reports. Thus, the usability of ERP systems in this respect might be improved by providing corresponding standard reports aiding systems administrators. In particular, this issue pertains to combining all relevant tables for particular application scenarios, and to the matching of events to underlying process instances. In this case, the latter issue was exacerbated by the use of differing primary keys in related database tables. It may be difficult to judge whether the resulting complexity of the data model is really required. Still, investigating its actual necessity might be worthwhile.

With respect to available process mining tools, it was found that there are still certain functions that might be integrated in such systems to improve their effectiveness. This relates not only to process improvement projects, but also to other settings such as compliance management [155] or benchmarking [45]. However, it needs to be stressed that these issues pertain to commercially available tools in general. Disco, the tool used in the present project, was chosen as it represents the state of the art of commercially available tools, in particular regarding ease of use and speed of deployment. Beyond the issues discussed here, [284] comprises a more detailed summary of process mining success factors based on multiple case studies. The following topics should be considered as functions not yet fully implemented:

- *Pattern analysis* allows comparing multiple process enactment variants [29, 33] including their actual frequency. For example with regard to repetitive loops (cf. PIM card 5), this functionality would be very useful to identify and prioritize process variants that should be restricted or eliminated.
- *Compliance rules modeling* allows describing relevant regulations for business processes in a way sufficiently formalized to automatically check whether these regulations have been adhered to in a process enactment data sample [155, 156]. As an example, consider the requirement of obtaining approval before job offers are issued.
- *Approximation of manual effort* facilitates amending event-based process maps with the underlying manual processing effort. This would tremendously enhance the utility of corresponding analyses, and could be achieved by enriching event types with assumptions on the corresponding activities. By matching a material sample log against total capacity used for processing (the so-called "baselining" in practice), the required degree of validity for the assumptions made could be achieved.
- *Automated regression analysis* allows finding correlations between characteristics of data samples (e.g., between the number of contact partners involved and cycle times).

If characteristics are derived from PIOs, respective drivers for process improvement can be identified automatically.

- *Sample delineation* addresses the issue of restricting a data sample to exclude process instances without particular characteristics, such as the presence of start and end events. Since this topic is important to ensure the validity of analyses, tool developers might want to consider guiding users through the sample delineation procedure by way of an appropriate user interface.

Out of the topics listed above, *compliance rules modeling* and *sample delineation* can also be addressed through *pattern analysis*, which constitutes the basic functionality to enable process enactment optimization. Like in the sample case, process improvement projects utilizing empirical process enactment data will employ spreadsheet applications if pattern analysis is not readily available in a process mining tool.

In addition, Minitab was used as an exemplary statistical tool to support process improvement, e.g., with regard to analyzing the correlation between cycle times and candidates' probability to withdraw their applications. This class of tools can be considered as advanced today, and will generally provide accurate implementations of the relevant statistical tests.

## 12.5 Related Work

Besides approaches directly addressing the topic, the assessment of PIPs relates to a broad array of fields. These range from general process improvement and quality to considerations on empirical research on information systems, and are shortly described below.

### 12.5.1 Validation of Process Improvement Patterns

Approaches aiming at empirical validation of PIPs can be traced back to quality management and BP reengineering approaches which have evolved since the 1950s and the early 1990s, respectively.

In terms of quality management, Six Sigma [283] is particularly interesting because it aims at eliminating errors in manufacturing processes through a set of quantitatively oriented tools used to identify and control "sources of poor quality". While the scope of BPM usually lies in administrative processes instead, there are important analogies since Six Sigma is based on step-by-step optimization of production processes through experimental changes to parameters. This means that measures are subject to *a-priori* assessment before they are implemented.

BPR, as exemplified in [17, 38], aims at optimizing processes "in the large" instead of implementing incremental PIPs. Transferring process enactment to an external supplier or customer constitutes a good example of this paradigm. While the potentials of this disruptive approach may seem tempting, more recent empirical evaluation has shown that the risk of projects failing is substantial [85]. Thus, incremental implementation of individual PIPs remains a valid approach.

In contemporary BPM, [130] proposes a framework to select and implement redesign practices. As opposed to our research, this approach does not aim at assessing individual PIPs for a specific applications scenario, but at efficiently appraising a broad framework of practices in order to identify the most relevant propositions. Earlier results from the same authors were used as a source of PIPs to be assessed in more detail [129].

The same authors also developed an approach to appraise BPR practices [285] based on the analytic hierarchy process (AHP) [100]. This proposition aims at ranking PIPs (or "best practices") according to their importance for the organization. This enables limiting further considerations to a prioritized set of PIPs. In contrast, the goal of the approach proposed in this chapter is to assess individual PIPs for a given application scenario based on a total set of PIPs that should, in the end, be as large as possible. However, the approach of step-by-step refinement of organizational objectives and PIOs might be used as input to the AHP in terms of scenario-specific impact criteria.

In addition, [286] proposes PIPs to tackle the findings of an earlier case study on workflow implementation regarding issues with team collaboration. While it is not the objective of this chapter to document a general methodology, it can nevertheless be viewed as an approach to prospectively appraise the implementation of PIPs for a particular application scenario.

### 12.5.2 Identification of Process Improvement Patterns

In the following, the relevant state of the art with regard to identifying PIPs that may be subject to assessment is shortly summarized.

Besides leveraging PIPs that emerged from the BPR wave of the late 1990s and early 2000s, there also exist more recent attempts to provide a basis for process improvement by appraising perspectives on *BP quality* based on software quality [123, 125] as well as the approach based on managerial analysis and control presented in this thesis. These approaches result in sets of quality attributes or characteristics for business processes. Since measures that aim at fulfilling quality characteristics constitute process improvement measures, quality characteristics can be viewed as PIPs as well. In this context, comprehensively validating the set of quality characteristics provided by an approach through empirical analysis remains challenging, because it will be virtually impossible to find practical cases where the entire set of quality characteristics "adds value". In this regard, the present approach instead enables organizations to validate quality characteristics to be improved specifically for a particular application scenario.

*Benchmarking* constitutes an approach widely employed in practice today [45]. Organizations seek to identify "best practices" to improve their business processes by comparing implementation options and results with "peers". With regard to specific industries or application fields, the resulting "best practices" have also been compiled into specialized frameworks for process management and improvement such as ITIL for information management [118].

In general, contemporary quality management methods used in manufacturing and logistics (e.g., Poka yoke to eliminate potential sources of errors [287]) can provide interesting pointers for improving business processes. A respective summary of the state of the art in "total quality management" (TQM) can be found in [47]. As an example for a TQM approach, the European Foundation for Quality Management (EFQM) proposition for achieving "organisational excellence" can be based on a BP perspective [288]. However, the underlying evaluation dimensions, which can be used to identify process improvement potentials, are rather abstract, and require general and scenario-specific interpretation.

Note that research on PIPs addresses the quality of process models and process implementations in the sense of *business content*. In contrast, [135, 174, 160, 147, 165, 289] exemplify propositions addressing *process model quality* in terms of structure, comprehensibility etc., i.e., the proper representation of actual business content by model elements.

### 12.5.3 Additional Aspects

In IS research, there have been diverse propositions to ensure common standards of scientific rigor in empirical research such as field experiments or case studies [15, 290]. Hevner et al. summarized empirical "design evaluation methods" for information systems research [105]. As a basis of this chapter, the requirements summary by Wieringa et al. [266] was chosen due to its concise, checklist-based character, which makes it readily applicable to research as well as to discussion with practitioners.

Gregor provided a taxonomy on various types of theory in information systems research [291]. In this context, PIPs would fall in the category of "design and action" theories since they give prescriptions on how to construct artifacts (in this case, business processes). This perspective is interesting for the purposes of this paper, since it highlights the limitations of treating a PIP as an object of validation, and hence as a theory, on its own. Rather, whether a PIP is valid as a prescription to implement or change a business process clearly depends on the relevant application scenario and organizational context. In line with the propositions of this chapter, a corresponding scenario-aware assessment method is required. This method then constitutes a "design and action" theory.

The top-down approach of deriving PIOs and PIMs from organizational objectives is similar to techniques for eliciting requirements in goal-oriented requirements engineering like KAOS [206] or i* [205]. Propositions in this respect are based on working with stakeholders to identify goals to be met by a system [292]. Goals are refined step by step until a level is reached that is suitable for technical implementation. We stipulated that a structure of PIOs based on organizational objectives is useful to avoid redundant discussions of the business value of lower-level PIOs. Similarly, the state of the art in requirements engineering recognizes the practical benefits of a "goal refinement tree" linking strategic objectives to detailed technical requirements [194]. In this respect, the concept of organizational objectives compares well to the notion of "soft goals" [199]. The step-by-step refinement of PIOs corresponds to the basic AND/OR decomposition of goals which has been extended to common notations for goal documentation such as KAOS [206]. These parallels are based in the shared underlying notion of breaking down a larger problem, like overall cost improvement, into more manageable chunks. This principle can also be found in contemporary approaches towards project management, e.g. in software implementation [273].

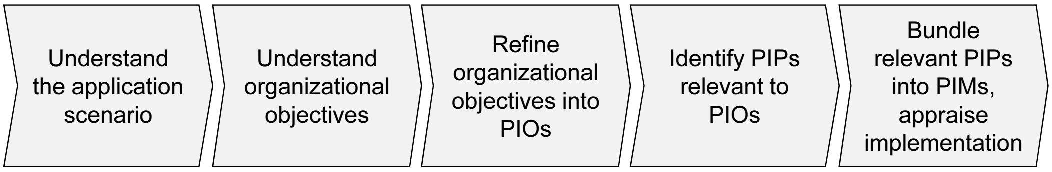



Figure 12.12: Approach Overview

## 12.6 Discussion

When motivating this paper, three challenges were identified to enable *a-priori* PIP assessment. This section discusses how these challenges could be addressed. Further, it discusses relevant limitations of the approach and presents recommendations that may be useful for similar projects. For quick reference, Figure 12.12 recapitulates the proposition of this chapter as a simplified approach overview: first, a profound understanding of the considered application scenario including the corresponding organizational objectives is developed. The organizational objectives are then refined into PIOs that are sufficiently granular to allow identifying relevant PIPs in the next step. Finally, relevant PIPs are bundled into PIMs, and are appraised to enable implementation decisions.

### 12.6.1 Revisiting Research Challenges

The sections described an approach towards *a-priori* PIP assessment. With respect to Challenge 1 (cf. Section 12.1), this approach is arguably better suited to reflect common practice in the field than the available state of the art in IS research (cf. Section 12.5.1). While there have been propositions towards *ex-post* empirical validation of PIPs in the past, to the author's knowledge, the approach presented in this paper constitutes the first proposition towards *a-priori* assessment of PIPs in the area of BPM. In particular, the two perspectives on PIP appraisal differ in their treatment of the available set of PIPs:

- The *ex-post* perspective seeks to narrow down the set of PIPs to a selection of aspects with demonstrable empirical relevance in a wide variety of application scenarios, thus following common scientific practice.
- The *a-priori* perspective seeks to accommodate a comprehensive set of PIPs, but limits assessment to one particular application scenario. It thus "constructively validates" only a limited set of PIPs at a time.

Without doubt, the first perspective reflects common scientific practice, since it enables generic statements on PIPs that are *independent* of a particular context. Nevertheless, working on a real-world process improvement project showed that the second perspective tends to be more in line with the expectations of practitioners. Arguably, this reflects a central characteristic of PIPs, and the corresponding implications for their practical adoption: As becomes clear when considering PIOs for various application scenarios, characteristics that drive organizational objectives may differ substantially for varying sample processes.

Hence, a validation of PIPs based on *other* application scenarios is of limited value for implementation decisions. In this context, the practitioners interviewed for this chapter observed that a pre-selection of PIPs will be effective only in the case of frameworks addressing a particular field of application. As examples, consider industry-specific "best practices" such as ITIL for the field of information management [118], or guidelines for dermato-oncology in medicine [43].

Assessing PIPs for each individual project requires substantial effort by qualified personnel to understand the application scenario, identify and refine organizational objectives and PIOs, select appropriate PIPs, and finally bundle them into implementable PIMs. Whether this effort can be justified *before* initiating the assessment depends on the creation of business value that may be reasonably expected. Organizations should consider three topics before starting a PIPs assessment project:

- Is the business process substantially relevant to the organization, e.g. with regard to the output produced or the cost volume incurred?
- May the organization assume that there are improvement potentials in the process, for example when considering existing problem reports or benchmarks [45]?
- Are there particular circumstances that require analyzing the process anyway such as, in the sample case used here, intentions to replace the underlying PAIS?

In the sample case, overall annual process cost of about 11.6m Euros could be assumed (cf. Section 12.4.1). Thus, it became clear that even minor cost potentials identified would suffice to cover assessment effort.

Based on these observations, the approach towards PIP assessment presented here is better aligned with common practice in the field, and thus better suited to address Challenge 1 (cf. Section 12.1) than the previous state of the art.

Regarding Challenge 2, evaluating the approach through a substantial experience report, Sections 12.2 and 12.4 presented the real-world case used to this end as well as the results of applying this chapter's propositions. The sample case dealt with a business process found in most organizations, and comprised 27,205 cases managed through a standard ERP system. It exposed typical issues when dealing with empirical analysis of real-world process enactment data, such as the complexity of extracting and interpreting data, as well as relevant process characteristics that do *not* become apparent by analyzing transactional data. It is thus stipulated that this experience report represents common practical problems fairly well, and has been suitable to evaluate the proposed approach.

To address Challenge 3, the reconciliation of our propositions to applicable scientific standards, a framework by Wieringa et al. [266] was used to guide the description of the approach. This enables tracing all relevant components of an approach that fulfills scientific criteria, and simplifies the appraisal whether the corresponding requirements can be considered as fulfilled. It was found that the more rigorous documentation of problem statement and research design demanded by scientific rigor required some additional effort in comparison to what is usually found in practice. However, this task proved still worthwhile, since it simplified final discussion of proposed PIMs with stakeholders. As an example, consider the impact of cycle times on the probability of candidates to decline job offers, which could only be demonstrated through rigorous statistical testing.

Overall, the research challenges of this chapter can be considered as met based on the above discussion. However, there are still some relevant limitations to be outlined in the following.

### 12.6.2 Relevant Limitations

The application of this chapter's approach has been demonstrated on the basis of a substantial real-world business process with 27,205 process instances. Nevertheless, the first issue that needs to be discussed with respect to limitations of the approach pertains to its evaluation on the basis of only one application scenario, and thus a limited set of relevant PIPs. As a limitation to the environment of data collection (cf. Sec. 12.3.2), applicants could not be interviewed because of privacy regulations. It will thus be useful to apply the approach to additional scenarios to extend the set of PIPs actually applied. Of course, this will require access to additional real-world process improvement projects with substantial sets of empirical data on business processes. To draw meaningful conclusions, these processes should be comparable to what is commonly found in other organizations. Accordingly, additional experience reports (with the potential to extend the underlying approach) shall be an issue for future work taking up such opportunities.

A second topic relates to the availability of a comprehensive set of PIPs to be applied to PIOs. In principle, this does not affect the validity of our approach. However, it impacts its practical effectiveness, since it will determine the actual business value of PIMs identified. In this respect, much work has been undertaken by Reijers and Limam Mansar [129, 130]. In addition, this thesis developed an approach to conceptually derive a *comprehensive* set of quality attributes that may be used as PIPs (cf. Chapter 8).

On a more abstract level, the third issue pertains to methodological limitations with respect to empirically validating PIPs. In this context, PIPs can be viewed as a prediction or theory dealing with the impact of certain process characteristics on process performance. However, comparable to design patterns in software engineering [277], PIPs do not constitute a self-contained concept for the following two reasons. As discussed above, currently no approach is available to demonstrate that a set of PIPs is comprehensive. In addition, the degree of utility of any given PIP is highly specific to the application scenario considered. Thus, it is virtually impossible to validate an *entire set* of PIPs by means of empirical information systems research such as field experiments, participative research, or case studies [293]. Note that this topic has also been included with regard to quality attributes as a limitation to the overall proposition of this thesis (cf. Chapter 13). As discussed in Section 12.5, researchers have addressed this issue by conducting meta-studies on a broad range of PIPs [130]. This, however, means that individual PIPs are validated based on widely varying research designs. The approach presented in this chapter also cannot resolve this issue. Still, it constitutes a generally applicable and re-usable approach to assess PIPs for given application scenarios, which can contribute to harmonize PIP appraisal designs.

A fourth issue that needs to be discussed concerns inherent limitations with regard to demonstrating the general validity of the assessment approach proposed. The approach results in recommendations on which PIPs to implement. However, the question is how one can ensure that these recommendations are well-founded. This challenge is exacerbated by two topics:

- On a more detailed level, the business value of PIPs is appraised considering the business process and the scenario addressed. That is, the general assessment approach is refined specifically for each application scenario. Thus, it is not possible to fully replicate the same assessment approach in other settings, which limits the possibility of empirical validation. In other words, the validity of predictions on the business value of a particular PIP in a particular setting cannot provide assurance on the validity of predictions on other PIPs in other settings.

- Revisiting PIMs after implementation will only allow identifying "false positives", i.e., PIMs that did not deliver the business value expected. "False negatives", i.e., PIPs not chosen for implementation which *would* have delivered a positive business value, will always remain undetected.

Nevertheless, it is still good organizational practice to track the results of PIM implementation. This provides an incentive to involved stakeholders to apply due diligence during PIPs assessment. However, since only "false positives" can be tracked, one should be aware that this might lead to overly risk-averse assessment practices. Setting top-down process improvement targets (e.g., via quantitative benchmarking) can be a way to respond to this challenge.

A final limitation takes up the issue of "false negatives" described above. It pertains to the degree of control we have with regard to the procedure of selecting PIPs and proposing PIMs for an application scenario. It needs to be kept in mind that this procedure depends on the knowledge, experience, and creativity of project participants. In other words, if no project participant can think of a way in which a PIP could be used to address a PIO, the PIP will not be considered in PIM propositions. However, this does not mean that the PIP *cannot* provide value in the application scenario. Arguably, the step-by-step refinement of PIOs is a useful technique to address this issue since it helps to focus efforts on relevant aspects. However, it cannot provide formal assurance on this issue.

### 12.6.3 Recommendations for Implementing the Method Presented

When working with practitioners to identify and assess PIPs applicable to the sample scenario, several general issues and recommendations emerged that should be considered when applying PIPs in process improvement projects. These observations were discussed with interview partners in the course of the respective steps in the approach of this chapter. On that basis, a number of *project recommendations* could be phrased. These recommendations were reconciled with management level interview partners, and may be viewed as guidance for researchers and practitioners when setting up and executing comparable projects. Readers familiar with these topics may wish to skip this section.

The first recommendation pertains to the overall structure of the proposed approach, and to the "research design" component as required in [266].

**Project Recommendation 1 (Top-down process improvement methodology).** *Top-down process improvement* refers to methods based on an initial definition of and agreement on the goals to be pursued, which are then further elaborated and amended with corresponding measures in a step-by-step approach. As a general principle, earlier decisions are refined to a more detailed level in later project phases. Top-down approaches address challenges resulting from two topics: First, process improvement projects typically require effective collaboration between multiple parties in an organization. However, these may tend to "sub-optimize" by focusing on individual interests instead of overall organizational objectives. As an example, consider the recruiting department and the various business units in our sample scenario. To "sub-optimize" its own effort in application handling, the recruiting department might pass applications not to the best, but to the most accessible contact partner in a business unit, thus impeding the goals of the organization as a whole. To realize the full potential of process improvement, parties need to be aligned towards clearly defined common goals and decisions as early as possible. Second, projects without a top-down decision structure may be obstructed by re-discussing goals and decisions multiple times. Besides the additional effort caused, this may lead to inconsistencies in the project. As an example, consider multiple measures addressing cycle times. Without a general understanding that cycle times are an objective of process improvement, this discussion will be led for each corresponding measure individually. With a top-down approach, earlier goals and decisions can serve as a gauge to appraise later decisions and measures.

The top-down principle is reflected in the approach presented in this chapter. First, early senior management agreement on the general "call for action" is required (see the concept of organizational objectives). This is then refined into process-related objectives (see the PIOs concept), and finally into individual improvement measures. This way, the discussion of individual measures focuses on *how* things are to be achieved instead of *what* to achieve in general. To implement this recommendation, agreed project results should be strictly documented, e.g. in a decision log.

The second recommendation is applicable to the "unit" and "environment of data collection" components as described in [266].

**Project Recommendation 2 (Identification of potential PIOs, PIMs, and PIPs based on process design and enactment).** Potentially applicable PIOs, PIMs, and PIPs should be identified not only by considering the process model, but also by analyzing empirical data on actual process enactment. This is crucial to focus on topics of actual value potential. For example, consider the selection of critical cases in Figure 12.3, which is reflected in the PIOs for the sample case.

The third and fourth recommendations address data gathering and analysis procedures required to appraise PIOs and PIMs for a particular application scenario. In terms of [266], they qualify "measurement" and "data analysis procedures".

**Project Recommendation 3 (Appropriate qualitative or quantitative demonstration of business value).** For each PIO to be addressed by PIMs, the underlying business value must be empirically demonstrated based on proper qualitative or quantitative analyses with respect to organizational objectives. Likewise, the business value of PIMs must be made transparent through appropriate analyses.

The specific analytic approach for individual PIOs and PIMs must consider the actual application scenario, balancing expected insights against analysis efforts. For example, the omission of tasks which obviously do not contribute to the business objective of the process can be justified by a short qualitative description. In contrast, the introduction of additional control tasks to diminish defects later on in the process will require careful quantitative weighing of pros and cons.

**Project Recommendation 4 (Identifying relevant stakeholders as interview partners).** To ensure the validity of measurement procedures, proper selection of interview partners is particularly relevant for PIOs and PIMs that should be validated qualitatively. For BPM scenarios, it is important to interview experienced senior personnel overlooking the end-to-end business process, and to represent both the "supplier" and the "customer" perspective to avoid lopsided optimization. For the sample process, the following contact partners were interviewed: the head of recruiting operations and the administrator of the application management process from the "supplier" side, and the HR partner of a business unit as well as team managers from the business unit from the "customer" side.

The fifth and sixth recommendations concern the final assessment of PIMs. Hence, they refer to "data analysis procedures" as well [266].

**Project Recommendation 5 (Considering implementation effort in business value appraisal).** When discussing the business value of particular PIMs for a business process, the respective implementation effort must be taken into account. This includes measures required, cost, time, and change management issues (e.g., training personnel to enact new activities). A PIM will only provide business value if implementation effort is justified by realized process improvement potentials. For example, an organization may demand that the required investment must not exceed three times the projected annual cost savings when appraising operational cost optimization measures.

**Project Recommendation 6 (Leveraging "quick win" potentials).** In many practical scenarios, it is possible to identify "quick win" PIMs that can be implemented with limited effort and should thus be given higher priority than others, in particular in comparison to full-scale PAIS implementation measures which are usually very costly. Examples include the elimination of process defects caused by process participants' behavior, interface issues between departments, or issues of data quality. Note that these topics are often identified through empirical analyses (e.g., using process mining).

## 12.7 Summary and Outlook

This chapter described an approach for *a-priori*, scenario-specific assessment of process improvement patterns based on organizational objectives, process improvement objectives, and process improvement measures. The approach leverages available work on generic requirements towards empirical research in IS engineering [266]. It could thus be demonstrated how these principles can be applied to practical cases while ensuring the general appeal of the approach.

The approach was applied to a real-world business process, including validation of the respective results with practitioners. This led to the identification of five potential process improvement measures that bundle and refine individual process improvement patterns for the given application scenario. Matching the expected gains against implementation and operating efforts, the organization was enabled to take well-informed implementation decisions. Revisiting the proposed process improvement measures more than one year after initial data collection confirmed that these decisions could be used to guide further development of the business process in practice.

The approach presented in this chapter is based on applying PIPs to concrete business processes. The quality attributes developed in Chapter 8 can be viewed as a particular form of PIPs. Hence, the proposition of this chapter contributes to demonstrating how quality attributes can be used in practice.

Moreover, as will be discussed in Chapter 13, it is not feasible to validate the entire set of quality attributes through application to individual BP scenarios. The reason for this is that only particular quality attributes will induce improvements in a particular scenario. The same consideration generally applies to any given set of PIPs. In this regard, the approach presented in this chapter contributes by providing a method to validate the applicability of individual quality attributes or PIPs to a process scenario in the field. The more comprehensive the set of quality attributes or PIPs used as input to this method, the more improvement potentials can be identified. Again, this highlights the significance of *comprehensive coverage* as an aspect of Effectiveness Criterion 1, which has been discussed in Section 3.1.1.

# 13 Discussion

According to the design science paradigm, design artifacts – such as concepts addressing the research objectives of this thesis – are validated by demonstrating "value or utility" beyond the previous state of the art [103, 77]. To support the determination of "value or utility", Section 3.1 derived *effectiveness criteria* from literature relevant to the research objectives defined. Thus, validating the contribution of this thesis amounts to appraising the concepts developed both in terms of the research objectives set out *and* in terms of the effectiveness criteria stipulated. The research methodology underlying this thesis has been adapted to address aspects relevant to research objectives and effectiveness criteria, but not yet covered by previous art (cf. Chapter 4). Nevertheless, there are still limitations in this regard that need to be discussed. On that basis, the following sections reflect the results achieved in this thesis along research objectives while considering effectiveness criteria. This way, they summarize both the contributions delivered as well as limitations yet to be addressed in future work.

## 13.1 Enabling Business Process Quality Analysis

Research Objective 1, *Enabling BP quality analysis*, constitutes the core requirement to drive BP quality management in a structured and well-founded manner. Concepts addressing this research objective need to be *holistic* in the sense of encompassing all relevant aspects of BP quality (cf. Effectiveness Criterion 1, Section 3.1), and *generic* in the sense of applicable to all kinds of business processes (cf. the motivational theses described in Sections 1.2 and 1.4). In summary, they should facilitate performance evaluation of parties involved in both BP *design & implementation* as well as *enactment*. At the same time, they should consider the requirements discussed in relevant literature on managerial analysis and control (cf. Effectiveness Criteria 2 and 3).

In this regard, the *quality model* developed in Chapter 8 assumes a central role. Applying it to a BP model and to a corresponding sample of BP instances will enable analyzing process quality while considering the common *split of responsibilities* encountered in practical settings. As discussed in Section 3.1, this constitutes a key requirement of effective managerial analysis. Without the split of responsibilities, quality management cannot approach responsible contact partners with analysis results. In this case, the impact of quality analysis results on the actual behavior or involved parties will be rather limited. Figure 13.1 provides an overview on the relevant aspects addressed.

- With regard to *lifecycle stages*, the quality model addresses both design & implementation as well as enactment. This reflects the notion that a business process can be understood as a model and its implementation *or* as a set of actual instances enacting an explicit or implicit model (cf. Section 2.1).

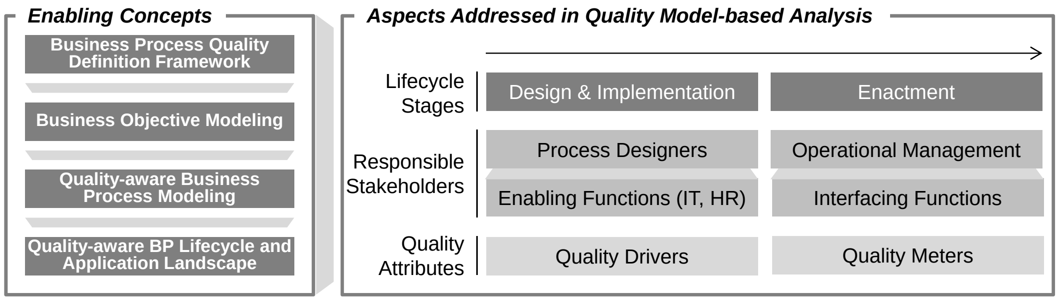



Figure 13.1: Quality Model-based Analysis

- With regard to *responsible stakeholders*, the quality model pertains to process designers as well as operational process management. In general, this "split of responsibilities" reflects the differing lifecycle stages: On the one hand, process designers and enabling functions are not responsible for the later enactment of processes. On the other hand, operational process management as well as interfacing functions have limited influence over their design and implementation. In this context, *enabling functions* are to be understood as functions (i.e., specific roles or departments) within an organization which provide the required infrastructure to design and implement business processes. As examples, consider IT (e.g., for the design and implementation of PAISs) or HR (e.g., for the definition and implementation of organizational rules and guidelines). In contrast, *interfacing functions* deliver upstream business processes as part of an end-to-end process orchestration, with output required as a resource for process enactment. Interfacing functions are specific to the business process in question. As an example, consider the pharmacy delivering drugs in Sample Process C (cf. Figure 2.7).

  Appropriate quality criteria will facilitate appraising process quality without bias resulting from the performance of interfacing functions (cf. Section 8.4). However, enabling functions should be subject to the governance of process designers during design & implementation. For example, it should be possible for process designers to sign off IT implementations. It is essential for the effectiveness of quality analysis as a management tool that the respective scope of responsibilities is reflected appropriately [113]. Accordingly, the split between process designers and operational process management is followed through in the structure of the quality model by relating quality attributes to the corresponding lifecycle stages.

- In terms of *quality attributes*, the quality model finally reflects design & implementation as well as enactment quality with the associated organizational responsibilities through *quality drivers* and *quality meters*, respectively. Quality drivers constitute the basis of quality analysis since they enable analyzing the quality of a business process as an abstract model, without considering an actual set of process instances. To ensure comprehensive coverage as required to fully achieve Effectiveness Criterion 1, *Congruence to organizational targets*, quality drivers are deducted using a sub-structure of quality driver types. On that basis, quality meters are designed to consider organizational responsibilities with regard to process design & implementation, which delineates overall enactment performance from the performance of upstream processes (cf. Example 57). Each quality attribute is amended with quality criteria and quality predicates. The former enable quality appraisal, while the latter translate appraisal results to readily applicable instructions to improve quality where feasible. Together, these constructs provide organizations with a means to conduct quality analyses.

**Example 57 (Requirement to Delineate Enactment Quality from Design & Implementation and Upstream BP Quality).** Consider Sample Process A as described in Figure 2.5, the management of incoming supplier invoices. When analyzing enactment quality, it is necessary to consider the parameters and framework conditions given by the IT implementation of process automation potentials and by upstream process quality delivered by the procurement function.[1]

In terms of possible defects, it is conceivable that the IT function failed to implement a state-of-the art intelligent scanning solution which will lead to substantial rework requirements during process enactment. Likewise, missing supplier master data, which has to be provided as a resource by upstream procurement processes, might impede speedy processing during invoice handling. Both defects must be delineated from quality appraisal of process enactment. In other words, while the defects mentioned above might impact process performance in comparison to peer organizations [45], it is still possible that process enactment exhibits good quality.

In addition to the quality model, the concepts developed in Chapters 6, 7 and 9 provide enabling techniques to quality model-based analysis:

- The *BP quality definition framework* provides a short and concise notion of BP quality which can be matched against common quality notions (cf. Section 2.2) and discussed with stakeholders (cf. Chapter 10). This is an important step to focus stakeholders on a common understanding which is, in turn, essential to successfully integrate quality analysis results into BP management.
- *Business objective modeling* (cf. Chapter 6) enables formally defining desired outcomes of a business process. This is the key prerequisite to discuss the efficacy of a business process in the sense of whether a process actually achieves what the organization intends it to. Without modeling objectives, a major aspect of quality as "the degree to which a set of inherent characteristics fulfills requirements" [89] therefore could not be addressed.
- *Quality-aware BP modeling* (cf. Chapter 7) closes the gap between business objective models and BP models, and enables deducting resource requirements in terms of availability and consumption posed by a process towards a given business objective. The quality-aware BP meta-model transcends common process modeling approaches by making prerequisites and impact of conditional branches and tasks explicit, and by enabling to consolidate possible paths through a process model while maintaining semantic consistency. The resulting *quality relations* of *efficacy* and *efficiency* reflect the definition framework developed in Chapter 5.
- A *quality-aware BP lifecycle and application landscape* (cf. Chapter 9) facilitates quality analysis by embedding the required procedures into the stages of the BP lifecycle,

[1] Upstream business processes constitute preceding processes delivering input for the process considered (cf. Section 2.1.1).

| **Effectiveness Criteria** | **Critical Appraisal** | **Conclusion on Limitations** |
|---|---|---|
| *Congruence to organizational targets* | Requires comprehensive, but exclusive coverage of organizational targets, in particular through quality definition and quality attributes | Comprehensive coverage can be reasonably fostered, but cannot be proven |
| *Transparency and retraceability* | Requires clear analysis methods and quality criteria for each quality attribute | Relative quality attributes cannot be fully formalized, analysis procedures with subject matter experts might not be fully transparent |
| *Cost effectiveness* | Requires efficient procedures, use of automation potentials, and relevance of quality analysis results | Quality-aware PAISs and WfMSs are not commercially available yet, potential additional modeling effort |

Table 13.1: Enabling Business Process Quality Analysis: Limitations

and into process-aware application landscapes. The former, for example, pertains to considering quality-aware concepts during modeling, while the latter comprises the provision of logging facilities to trace actual process enactment.

In summary, the described concepts enable analyzing the quality of business processes while considering the requirements posed towards managerial application scenarios. Remaining limitations in this regard with respect to effectiveness criteria are summarized in Table 13.1, and further described in the following paragraphs.

In terms of *Congruence to organizational targets*, the sub-criterion of exclusive coverage can be fulfilled by deducting quality attributes from the quality definition framework and other concepts. This way, there is an "audit trail" available that allows tracing back each quality attribute to the underlying aspects of the quality notion assumed in this thesis. However, there is no way to provide ultimate assurance regarding the sub-criterion of comprehensive coverage since the definition of quality attributes still involves a degree of human creativity. This issue is contained by the methodical approach used, e.g., by employing *guiding questions* based on quality attribute types, but cannot be fully eliminated.

*Transparency and retraceability* in BP quality analysis can be ensured by providing clear appraisal methods and quality criteria for each quality attribute. In particular, this can be achieved for absolute quality drivers, i.e., quality attributes that can be formally appraised without referring to subject matter experts on the basis of quality-aware process modeling. For relative quality drivers, transparency and retraceability can be supported by enabling formal (and thus, ultimately, automatable) support for quality appraisal as far as possible. This issue has been considered in the descriptions of quality drivers in Appendix C.

*Cost effectiveness* in process quality analysis pertains to both the effort incurred for analysis procedures and the perceived "business value" of analysis results. Similarly to *Transparency and retraceability*, the former topic can be addressed by leveraging automatable assessment methods as described for various quality attributes. However, this requires investing to implement corresponding IT systems as described in Section 9.4 since fully quality-aware PAISs and WfMSs are not commercially available yet. Moreover, it needs to be kept in mind that automating quality assessment procedures builds on business objective models and quality-aware process models. The business value of quality assessment results needs to justify the additional modeling effort incurred. Whether this is the case needs to be assessed for each specific application scenario. Nevertheless, quality management concepts can also be used *without* incurring additional system implementation or modeling effort. A corresponding approach to scenario-specific assessment of process improvement potentials based on quality attributes has been described in Chapter 12.

## 13.2 Enabling Business Process Quality Control

As a management instrument, quality analysis will only create value if organizations succeed in leveraging analysis results to actually improve the quality of business processes. Beyond BP quality analysis, it is thus one of the research objectives of this thesis to enable *quality control* in the sense of providing organizations with the means to not only recognize quality issues, but also to define and implement appropriate improvement measures.

In this context, effective quality control in the sense of improving actual process models and organizational behavior relevant to process enactment must consider not only the technical content of process improvement measures (cf. Chapter 12), which are delivered through a quality model covering criteria and attributes (cf. Chapter 8), but change management issues as well [294, 118]. In this regard, two topics are of paramount importance:

- *Embedding quality-related concepts into a BP lifecycle* [13, 22] integrates observations made in quality analysis to the evolution of BP design, implementation, and enactment in the sense of a feedback cycle (cf. Chapter 9).
- *Considering Effectiveness Criteria 1 (Congruence to organizational targets) and 2 (Transparency and retraceability)* is essential to ensure *Perceived fairness* [113] as a prerequisite to appropriately motivate responsible stakeholders towards quality improvement.

While the topic of BP lifecycle embedding has been described in detail in Chapter 9, the topic of *perceived fairness* constitutes a pervasive leitmotif for considerations underlying this thesis. In Chapter 5, a definition framework for BP quality has been deducted from the outer environment of business processes, organizational targets, and the scope of influence exerted during process design & implementation as well as enactment. The resulting short and concise definition of BP quality (cf. Definition 1) constitutes a focal point for stakeholder discussions. This fosters the common understanding necessary to drive BP quality throughout the organization, and has been validated through the field experiment described in Chapter 10.

In addition, the approach towards modeling and assessing business objectives, BP efficacy, and BP efficiency as developed in Chapters 6 and 7 provides a new way to document requirements and characteristics relevant to quality. This enables organizations to discuss desired BP outcomes on a formal basis, and to provide stakeholders with a-priori indications on the quality impact of design & implementation decisions, for example with regard to design alternatives.

| Effectiveness Criteria | Critical Appraisal | Conclusion on Limitations |
|---|---|---|
| *Congruence to organizational targets* | Requires appropriate definition of quality predicates for all quality attributes | n/a |
| *Transparency and retraceability* | Requires clear quality criteria as a basis to determine quality predicates for a given business process | Quality criteria of relative quality attributes require subject matter experts' interpretation |
| *Cost effectiveness* | Requires deduction of effective process improvement measures from quality predicates | Process improvement measures are specific to an application scenario |

Table 13.2: Enabling Business Process Quality Control: Limitations

Finally, the quality model of Chapter 8 defines quality attributes which can be traced back to the quality definition framework. Each quality attribute is amended with a description of how it can be assessed as well as corresponding quality criteria and predicates. Quality predicates, in particular, can be readily transformed into instructions to process designers and managers. Thus, they facilitate actual quality control.

As a general consideration, quality control builds on quality analysis. Therefore, limitations in terms of Research Objective 1 (*Enabling BP quality analysis*) apply to *Enabling BP quality control* as well. Thus, only additional limitations beyond what has been discussed in the previous section are described here. Relevant limitations that might apply with respect to the effectiveness criteria defined for this thesis are shortly summarized in Table 13.2, and described in more detail below.

With regard to *Congruence to organizational targets*, quality predicates could be defined for all quality attributes. Assuming that complete coverage could be achieved in terms of quality attributes (see above), therefore, this criterion could be fulfilled for quality control.

Considering *Transparency and retraceability*, the deduction of applicable quality attributes for a given business process requires to determine whether quality criteria are fulfilled. For absolute quality drivers, determination can be formalized, and can therefore be considered as transparent and retraceable. However, relative quality drivers and quality meters require the support of subject matter experts to appraise quality criteria. In this case, it is possible that appraisal results may not be transparent and retraceable for stakeholders such as process managers. As a consequence, process managers or other stakeholders may be unable to relate to suggested process improvement measures. This issue can be addressed by carefully managing the quality appraisal process. As an example, consider the approach presented towards process improvement measures in Chapter 12: the documentation of measures via "measure cards" requires including the underlying rationale of measures. This enables discussing and reconciling measures with relevant stakeholders, thus fostering *Transparency and retraceability* of related quality appraisal procedures.

Process quality control procedures can be considered as *cost effective* if the process improvement measures triggered can be viewed as economically reasonable taking into account quality analysis and control efforts. Whether this is the case depends not only on arrangements towards executing quality analysis and control, but also on whether the application scenario considered provides substantial leverage for process quality improvement. While the former topic can be addressed through effective implementation of a quality-aware BP lifecycle supported by a corresponding system landscape (cf. Chapter 9), the latter issue is beyond the control of quality managers. Therefore, it is not possible to demonstrate the impact of process quality management in general. Nevertheless, Chapter 12 provides a substantial example of a real-world business process that could be materially improved by implementing process improvement measures which are, in turn, based on quality attributes.

## 13.3 Enabling Economically Reasonable Practical Adoption

To enable economically reasonable practical adoption of BP quality analysis and control concepts as developed in this thesis, two major considerations have to be taken into account. These pertain to a reasonable level of assurance regarding the validity of propositions, and to the availability of methods sufficiently elaborated for transfer into practice.

With regard to the first consideration, reasonable validity assurance, the deductive methodology used in this thesis makes its results traceable and understandable for practitioners and researchers while maintaining scientific rigor at the same time. It spans

- the deduction of effectiveness criteria applicable for managerial analysis and control of BP quality (cf. Section 3.1),
- the derivation of a concise BP quality definition framework as a basis of discussion (cf. Chapter 5),
- and, finally, the discussion of a corresponding quality model as well as its transfer into management procedures and a quality-aware PAIS landscape (cf. Chapters 8 and 9).

In this context, the approach to selectively assess quality attributes with regard to their business value for particular applications scenarios described in Chapter 12 plays an important role. It enables organizations to benefit from BP quality management practices without incurring the effort of implementing a full quality-aware BPM landscape first. This way, it is possible to initially appraise the business value of process quality management on the basis of a well-defined application scenario or individual process improvement project. The confidence of stakeholders towards BP quality management implementation decisions can thus be enhanced.

The second consideration, i.e., the availability of methods that are transferable into practice, has been addressed by formalizing concepts on business objectives modeling, quality-aware BP modeling, derivation of quality relations, and quality attributes (cf. Chapters6, 7 and 8). This enables integrating these concepts into quality-aware BPM system landscapes (cf. Section 9.4). Particularly challenging topics in the context of deriving quality relations have also been addressed in a prototypical implementation of corresponding algorithms (cf. Appendix B).

In terms of limitations, Research Objective 3 constitutes a "meta-objective" concerning the (practical) validity and applicability of the concepts developed to address the other two objectives. It is therefore not appropriate to use the effectiveness criteria of Section 3.1 as a basis to identify limitations to the research objective considered in this section. Instead, discussing limitations that apply to the achievement of Research Objective 3 amounts to discussing limitations to integrating quality management concepts into BPM procedures and system landscapes as described in Chapter 9, and to the validation approaches presented in Chapters 10 to 12. In this regard, two major aspects are of particular relevance: On the one hand, substantial effort will be required to fully implement the concepts developed in PAISs and WfMSs suitable for use in the field. On the other hand, the validation of the set of quality attributes developed in this thesis remains challenging. Both aspects result from the breadth and complexity of issues that need to be addressed in BP quality management, and will be further discussed in the following:

- A prototypical implementation of fundamental functionality to derive quality relations from quality-aware process models has been described in Section 7.5 and Appendix B. Still, a full (commercially viable) implementation of PAISs or WfMSs leveraging automation potentials for quality analysis and control is still outstanding. Addressing this limitation through a corresponding implementation project will entail efforts common to the development or extension of PAISs or WfMSs with commercial aspirations. Therefore, this topic cannot be addressed within the scope of this thesis. Rather, this thesis is to be understood as an approach to provide a conceptual foundation for such endeavors. In addition, Chapter 12 has described and demonstrated an approach to leverage quality management concepts *without* first implementing IT systems for quality-aware process modeling or process quality analysis and control.

- It is not possible to validate the "business value" of individual quality attributes as a basis for quality analysis and control independently of a concrete application scenario. Rather, the individual contribution of quality attributes for a particular process in a particular organization must be determined on a case-by-case basis, since only a subset of quality attributes will contribute to quality improvement in a certain application scenario. This topic has been discussed in detail in Section 12.6.2. It constitutes a limitation that is inherent to the nature of business processes and quality attributes. Hence, Chapter 12 provided an approach towards assessing the scenario-specific contribution of individual quality attributes *before* implementation effort is incurred.

Note that the second issue is exacerbated when considering that, in real-world enterprises, it is often not possible to measure the economic impact of individual process improvement measures, because other influencing factors cannot be dismissed. Rather, the impact of individual measures must be approximated by making assumptions based on, for example, the observed effort incurred for individual process instances. Example 58 further illustrates these considerations.

**Example 58 (Process Volume and Factor Costs Impact).** Consider Sample Process A from Figure 2.5. The initial handling of incoming supplier invoices constitutes a set of tasks which is widely automated through "intelligent scanning" solutions, or even replaced by automated EDI interfaces [295] in the sense of a process choreography [22].

While the economic viability (i.e., the underlying business case) of these measures in the sense of Quality Attribute 5, *Task Automation / Use of Capital Investments*, can be easily made plausible, its actual quantified economic impact will depend on factors which are individual to an organization. These comprise, for example, the number of process instances to be handled, labor costs per working hour, and the existing IT landscape. These may constantly change as operational, day-to-day management decisions are taken. For example, larger organizations today employ “shared service centers” (cf. Chapter 9) with a so-called “follow the sun” strategy of maintaining operations for 24 hours per day by using locations in different time zones. The the cost of manually handling a process instance may thus change depending on which location is currently being used. It is thus generally not possible to directly quantify the economic impact of measures addressing a quality attribute by assessing its impact on balance sheet and profit and loss statement. Moreover, it becomes clear that it is not feasible to generally validate the quality attribute by appraising an implementation business case for one or more individual enterprises.

As an approach to deal with this challenge, Chapter 12 proposed a top-down methodology of breaking down general organizational targets into scenario-specific *process improvement objectives* where the impact of a process improvement measure can be clearly demonstrated. Following this approach enables organizations to take individual, case-by-case decisions on economically reasonable adoption of quality attributes.

# 14 Summary and Outlook

This thesis addressed the issue of BP quality based on a notion of quality as a means of aligning processes towards organizational targets. To this end, Chapters 1 and 2 of Part I first discussed relevant motivational theses, research objectives, and basic concepts of BPM. On that basis, Chapter 3 developed a research methodology oriented at the design science paradigm and corresponding effectiveness criteria to appraise design results. Chapter 4 assessed the current state of the art with regard to effectiveness criteria, thus identifying open issues to focus further progress.

In Part II, Chapter 5 derived a concise definition of BP quality based on its purpose as a steering tool within BPM. In this context, modeling business objectives and assessing BP efficacy emerged as an open issue not yet addressed in today's BPM state of the art. Accordingly, Chapter 6 developed corresponding concepts. As a prerequisite to formally underpin quality management for business processes, Chapter 7 integrated concepts relevant to quality into process modeling approaches, and lined out an approach to deduct quality relations from business objective and quality-aware process models. The insights gained contributed to devising a BP quality model of quality attributes, criteria, and predicates in Chapter 8. As a final concept, Chapter 9 reflected how process quality management can be integrated into today's BP lifecycles and system landscapes.

Part III subsumed considerations to validate and discuss the concepts developed in this thesis. In this regard, Chapter 10 reported on the validation of the BP quality definition of this thesis with practitioners. Chapter 11 compared the contributions of this thesis to available approaches towards BP quality in order to elaborate its additional contribution. Chapter 12 developed a method for validating process quality improvement measures derived from the quality model provided by this thesis in the context of given application scenarios, and demonstrated its application to a substantial real-world business process. Finally, Chapter 13 revisited the research objectives set out for this thesis, and discussed contributions achieved as well as limitations that are still remaining.

While developing concepts towards BP quality, a number of additional topics and questions have arisen as challenges for future work in this direction. The following paragraphs shortly discuss each challenge by presenting a short description of the motivation underlying the respective issue, as well as criteria and initial ideas for a satisfying solution.

Figure 14.1 provides an initial overview on challenges by arranging them according to their perceived ease of implementation and value as BPM concepts. While, at this stage of research, this "challenge map" can only represent the author's personal judgment based on initial reflection and own experience, it may serve as an impulse for researchers and practitioners to further discuss the relative merits of research challenges presented in this section. The first five challenges have been marked as the "quick win cluster" since these challenges might be addressed with rather manageable effort while still maintaining a significant impact.

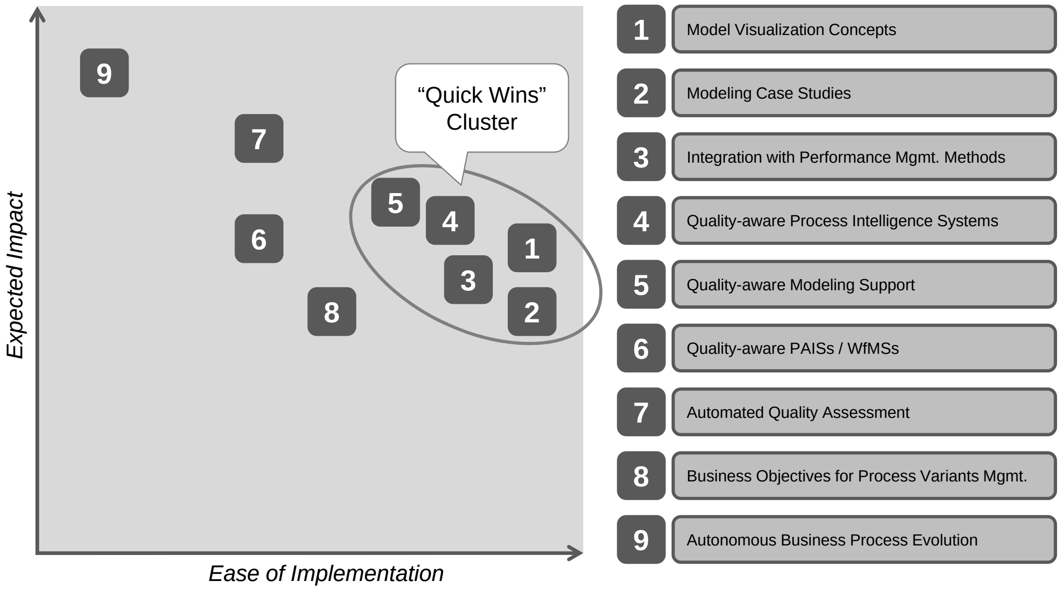



Figure 14.1: Initial Challenge Map

The first proposed challenge aims at improving the usability of modeling concepts through enhanced graphical representation:

**Challenge 1 (Model Visualization).** The approaches towards business objective modeling and quality-aware BP modeling presented in Chapters 6 and 7 include initial propositions with regard to visualizing corresponding models. However, these propositions have been designed mainly for illustrative purposes, and do not represent the full body of knowledge available with regard to BPM model visualization yet (e.g., [296, 82, 297, 84]). Accordingly, practical applicability and acceptance of concepts should be enhanced by developing additional visualization approaches.
**Satisfaction Criteria:** Criteria towards effective visualization of business objective and quality-aware BP models can be readily deducted from the corresponding state of the art with regard to human-machine interaction, software ergonomics, and user interface design [298].
**Implementation:** Besides the available state of the art within the BPM community, the implementation of visualization concepts should in particular consider interoperability with existing tools and methods as well as the possibility of empirical evaluation and stepwise refinement of propositions. The latter is enabled by corresponding research techniques such as *grounded theory* [299].

The second challenge pertains to additional qualitative empirical research into practical adoption of proposed concepts:

**Challenge 2 (Modeling Case Studies).** In Chapters 6 and 7, sample business objective models and quality-aware BP models were used to illustrate concepts. For reasons of simplicity and to foster readers' understanding, these models were based on simplified process examples corresponding to real-world business processes. In addition to this, case studies based on non-simplified real-world business processes and application scenarios would be useful to validate modeling propositions, and to highlight additional requirements.
**Satisfaction Criteria:** For the case studies to be conducted, general requirements for effective case study research apply [268]. Following the grounded research paradigm [299], these comprise "fit", "relevance", "workability" and "modifiability".
**Implementation:** Besides appropriate research design and execution, the contribution of case studies is determined by the representative character of cases selected. In the context of this thesis, this means that the business processes under assessment should reflect common application scenarios in one or more industries. This applies to most general and administrative processes (cf. Section 2.1). Further, they should exhibit sufficient complexity to enable non-trivial insights, and they should be manageable as a clearly delimited business process to obtain a clearly defined research object [266]. In addition, required deviations from standard case study procedures are to be highlighted and discussed to maintain scientific rigor.

The third proposed challenge seeks to further bridge the gap between BPM and managerial performance management by leveraging BP quality concepts:

**Challenge 3 (Integration with Performance Management Methods).** The use of BP quality analysis results as a means of performance management for process managers and other stakeholders was discussed as part of the motivational theses presented in Chapter 1. This constitutes an indirect method of BP quality control, since managers will be incentivized to improve process quality [109]. Hence, the conceptual integration of BP quality into common performance management methods will enhance the practical impact of quality analysis concepts.
**Satisfaction Criteria:** The effectiveness criteria discussed in Section 3.1 are fully applicable to managerial performance management methods. Accordingly, research in this direction can be oriented at the discussion presented there including the underlying literature.
**Implementation:** BP quality analysis concepts contribute additional content or input to performance management methods. As an example, consider the balanced scorecard approach [122]: the fundamental dimensions of a balanced scorecard and the associated management methods will not change, but the "process" dimension can now be addressed in much more detail. Thus, implementation may focus on possibly required adaptations to performance management methods, and on the management of possible overlaps and contradictions between existing and new (process quality) content. The latter issues should be avoided to uphold effectiveness criteria.

The fourth, fifth, sixth, and seventh challenge address the implementation of concepts proposed in this thesis in information systems used in the context of BPM (cf. Section 9.4):

**Challenge 4 (Quality-aware Process Intelligence Systems).** Process intelligence systems deliver *ex-post* analyses of business processes on the basis of sets of actual process instances that have been logged, for example, in a PAIS [35]. This functional scope can be leveraged to assess quality meters as discussed in Chapter 8. Accordingly, a process intelligence system is quality-aware if it supports the analysis of quality meters by using a relevant set of data to provide analytical functions addressing quality meters.

An additional process intelligence capability that would be useful to foster practical implementation of quality concepts in BPM relates to the recognition of recurring patterns in process traces. Once amended with resource requirements and target aspects addressed, these patterns could be re-used in quality-aware process modeling. This approach could help to reduce the effort of obtaining quality-aware BP models.
**Satisfaction Criteria:** The effectiveness of quality-aware process intelligence systems can, on the one hand, be appraised by matching capabilities implemented against the set of quality meters defined in Section 8.4. On the other hand, considerations with regard to general software quality apply as well [261].
**Implementation:** As a requirements definition, quality meters have to be analyzed, and corresponding data requirements have to be deducted. Design and implementation phases can then be executed accordingly. As a particular success factor, early interaction with stakeholders of managerial analysis and control of business processes such as shared service managers should be pursued.

**Challenge 5 (Quality-aware Modeling Support).** As discussed in Chapters 6 and 7, quality-aware BP modeling and, as a prerequisite, the modeling of business objectives entails efforts many organizations will be reluctant to expend. As a remedy to this issue, efficient corresponding modeling techniques can be developed. Examples in this respect include the re-use of "snippets" (i.e., fragments of quality-aware process or business objective models that occur more often) stored in model repositories [300], or the use of corresponding process mining technology [28].
**Satisfaction Criteria:** Effective approaches towards supporting quality-aware modeling will, on the one hand, make appropriate use of available techniques and technologies. On the other hand, proper integration into the specific BP lifecycle of the organization (cf. Chapter 9) is required. Overall success can be appraised by comparing modeling efforts incurred with and without the supporting methods and tools.
**Implementation:** Implementation should first assess the overall modeling demand in terms of quantity, complexity and available capabilities in the organization. For example, organizations might want to look into the capabilities of available process mining and modeling tools. On that basis, the most promising approaches are to be selected, implemented, and leveraged.

**Challenge 6 (Quality-aware PAISs / WfMSs).** Analysis and control functions in the sense of a BP lifecycle [13] generally constitute only a fringe aspect of common PAISs and WfMSs. Instead, the focus of such systems lies on enabling the enactment of business processes. Nevertheless, expanding these systems to consider the requirements of BP quality management would be useful as a functional addition. Corresponding requirements mainly

relate to the handling of business objective and quality-aware BP models, and to appropriate enactment data logging capabilities, which provide the input to quality-aware process intelligence systems.
**Satisfaction Criteria:** Comparable to quality-aware process intelligence systems, effectiveness requirements pertain to the degree of implementation of concepts developed in Chapters 6, 7, and 8, and have been discussed in more detail in Section 9.4. Note that implementing business objective modeling and quality-aware BP modeling is not required for quality-aware process enactment as the scope of PAISs and WfMSs. Nevertheless, these systems are often used as a tool for modeling tasks in the context of BPM as well. If this is the case, the respective meta-models should be considered. With respect to quality meters, data required for ex-post analysis should be logged during enactment. In addition, general software quality criteria apply [261].
**Implementation:** Projects may follow any appropriate software engineering methodology covering requirements analysis, design, and implementation [273]. However, quality and performance managers should be considered as an additional user group relevant to the PAIS or WfMS in question.

**Challenge 7 (Automated Quality Assessment).** Automated quality assessment builds on quality-aware PAISs, WfMSs, and process intelligence systems. In this regard, Appendix C discusses formalizable and therefore automatable aspects of quality assessment for each quality attribute. Note that this does not only pertain to formal efficacy-related quality attributes. In addition, there is a substantial share of relative quality attributes where final assessment requires the support of subject matter experts, but assessment can be prepared and supported by deducting and processing relevant characteristics from quality-aware process models and enactment logs. Since quality-aware process intelligence systems provide the means to analyze quality meters, it is well conceivable to further extend these systems accordingly.
**Satisfaction Criteria:** Requirements for automated quality assessment can be deducted from the detailed descriptions of quality attributes in Appendix C. These pertain to data requirements to be fulfilled through appropriate interfaces to quality-aware PAISs and WfMSs as well as analytic capabilities specific to individual quality attributes.
**Implementation:** Prior to software implementation, individual requirements of quality attributes should be refined by appropriately modeling data requirements, and by further formalizing assessment methods so that these can be readily transferred to algorithms. User acceptance testing should then involve subject matter experts to appraise individual quality attributes with the automated quality assessment system as a preliminary tool.

The eighth proposed direction of research pertains to the management of process variants as a specific topic of current BPM research which might be enhanced by using business objective modeling:

**Challenge 8 (Business Objectives for Process Variants Management).** Managing BP variants currently receives much attention in the BPM community (e.g., [301, 30, 29]). In this context, formal business objective models as presented in Chapter 6 provide a means

to assess whether two potential variants actually address a common business objective. This constitutes a particular application of the business objective meta-model which is not directly linked to BP quality, but should be considered nevertheless. Approaches following this direction should assess the "objective equivalence" of process models as well as the scope of process alterations possible while maintaining the business objective. Implementing a relaxed form of business objective models might also be a consideration in this regard.
**Satisfaction Criteria:** Effective approaches towards objective-based process variants management will take into regard application scenarios and requirements elaborated in the corresponding field of research. In addition, tight integration with quality-aware modeling tools will reduce implementation effort and thus foster the appeal of methodological propositions.
**Implementation:** Requirements from a variants management perspective need to be elicited from available literature. On that basis, implementation should cover methods and tools to determine the "objective equivalence" of process models, possibly by formal testing methods. Deriving objective model propositions from available process models, e.g. by augmenting subject matter experts' work, might be an addition driving practical appeal and adoption.

The ninth and final proposition for further research appears to be the most challenging, but also rewarding topic – leveraging concepts proposed in this thesis towards autonomous improvement of business processes:

**Challenge 9 (Autonomous Business Process Evolution).** Autonomous evolution of business processes by PAISs, WfMSs or other BPM tools might deliver a valuable contribution, for example considering the continuous process improvement (CPI) paradigm employed in many manufacturing organizations today [237]. Hitherto, the capabilities of BPM tools in this direction, for example, with respect to re-arranging the sequence of tasks in a process model, are severely limited. One of the reasons behind this is that there is no formal way to determine whether an evolutionary step would impair the ability of a business process to achieve its business objective. The concepts towards business objective modeling and efficacy-aware BP modeling presented in this thesis (cf. Chapters 6 and 7) may pave the way to overcome this challenge. In addition, integrating efficiency-aware process modeling, thus implementing the full set of quality-aware process modeling capabilities, would also enable predicting the impact of process evolution on resource consumption, thereby proceeding towards autonomous BP optimization.
**Satisfaction Criteria:** Criteria for a satisfying solution pertain to the maintenance of consistency with business objectives during BP evolution (i.e., automated appraisal of formal efficacy), to autonomous judgment whether an evolutionary step will improve full efficacy and efficiency, and to ease of use for an implementing organization (e.g., in terms of possible user interaction or automated integration with ERP systems).
**Implementation:** On the basis of a quality-aware (or, at least, efficacy-aware) WfMS, the system first needs to be enhanced by implementing an automated check for formal efficacy. Then, capabilities to autonomously perform evolutionary changes to process models under management need to be added. In this respect, it is conceivable to implement a "natural selection" evolutionary approach, in particular for high-volume processes. This would require devising quality meters for ex-post appraisal of instances to judge whether evolutionary

progress has been achieved. It is to be expected that for generic (i.e., scenario-independent) use, these quality meters would have to be rather simple, such as cycle time. Alternatively, the system would have to be provisioned with the capability to discern better and poorer process *designs*. In this respect, the set of absolute quality drivers related to resource consumption as described in Section 8.3 will be useful, since they do not require subject matter experts' appraisal.

Together, the propositions for future work presented in this section show how the field of BP quality management can further evolve, both in terms of its practical value as well as with regard to its integration with related areas of research. These considerations give confidence that the notion of BP quality developed in this thesis will continue to prove its contribution to the area of BPM, and to achieve practical adoption by organizations striving for excellence in this field.

# References


[1] Lohrmann, M., Reichert, M.: Effective application of process improvement patterns to business processes. Software & Systems Modeling (2014) DOI 10.1007/s10270-014-0443-z.

[2] Lohrmann, M., Reichert, M.: Efficacy-aware business process modeling. In: Proc. 20th Int'l Conf. on Cooperative Information Systems (CoopIS'12). Volume 7565 of LNCS, Springer (2012) 38–55

[3] Lohrmann, M., Reichert, M.: Demonstrating the effectiveness of process improvement patterns. In: Proc. 14th Working Conf. on Business Process Modeling, Development, and Support (BPMDS'13). Volume 147 of LNBIP, Springer (2013) 230–245

[4] Lohrmann, M., Reichert, M.: Modeling business objectives for business process management. In: Proc. 4th S-BPM ONE – Scientific Research. Volume 104 of LNBIP, Springer (2012) 106–126

[5] Lohrmann, M., Riedel, A.: Requirements towards effective process mining. In: Proc. On the Move to Meaningful Internet Systems Workshops (OTM Workshops '12). Volume 7565 of LNCS, Springer (2012) 64–68

[6] Lohrmann, M., Riedel, A.: Process Quality and Performance in Shared Services Organizations. In: Finance Bundling and Finance Transformation. Springer Gabler (2013) 225–251

[7] Lohrmann, M., Reichert, M.: Understanding Business Process Quality. In: Business Process Management: Theory and Applications. Volume 444 of Studies in Computational Intelligence, Springer (2013) 41–73

[8] Lohrmann, M., Reichert, M.: Formalizing concepts for efficacy-aware business process modeling. Technical Report UIB-2012-05, Ulm University, Germany (2012)

[9] Lohrmann, M., Reichert, M.: Basic considerations on business process quality. Technical Report UIB-2010-04, Ulm University, Germany (2010)

[10] Ellis, C.A., Nutt, G.J.: Office information systems and computer science. ACM Computing Surveys **12**(1) (1980) 27–60

[11] Jablonski, S., Bussler, C.: Workflow Management: Modeling Concepts, Architecture and Implementation. International Thomson Publishing Services (1996)

[12] Reichert, M., Dadam, P.: Geschäftsprozessmodellierung und Workflow-Management – Konzepte, Systeme und deren Anwendung. Industrie Management **16**(3) (2000) 23–27

[13] van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M.: Business process management: A survey. In: Proc. 1st. Int'l Conf. on Business Process Management (BPM'03). Volume 2678 of LNCS, Springer (2003) 1–12

[14] Workflow Management Coalition: Workflow management coalition terminology & glossary 3.0 (1999) http://www.wfmc.org/resources, document reference WFMC-TC-1011, accessed on 17/01/2015.

[15] Benbasat, I., Zmud, R.W.: Empirical research in information systems: The practice of relevance. Management Information Systems Quarterly **23**(1) (1999) 3–16

[16] Dumas, M., van der Aalst, W.M.P., ter Hofstede, A.H.M., eds.: Process-aware Information Systems: Bridging People and Software through Process Technology. John Wiley & Sons (2005)

[17] Davenport, T.J., Short, J.E.: The new industrial engineering: Information technology and business process redesign. Sloan Management Review **31**(4) (1990) 11–27

[18] Hammer, M.: Reengineering work: don't automate, obliterate. Harvard Business Review **68**(4) (1990) 104–112

[19] Gartner, Inc.: CIO Agenda (2008-2010) http://www.gartner.com/technology/cio/cioagenda.jsp, accessed on 16/01/2015.

[20] McKinsey & Company: McKinsey Global Survey results: A rising role for IT (2011) http://www.mckinsey.com/insights/business_technology/a_rising_role_for_it_mckinsey_global_survey_results, accessed on 17/01/2015.

[21] Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A.: Fundamentals of Business Process Management. Springer (2013)

[22] Weske, M.: Business Process Management. Springer (2007)

[23] Pedrinaci, C., Domingue, J., Alves de Medeiros, A.K.: A core ontology for business process analysis. In: Proc. 5th European Semantic Web Conf. (ESWC'08). Volume 5021 of LNCS, Springer (2008) 49–64

[24] Anthony, R.N., Dearden, J., Bedford, N.M.: Management Control Systems. 6th edn. Irwin (1989)

[25] Reichert, M., Weber, B.: Enabling Flexibility in Process-aware Information Systems: Challenges, Methods, Technologies. Springer (2012)

[26] Towers Watson: Survey: Companies tighten link between executive pay and performance (2010) http://www.towerswatson.com/en/Insights/IC-Types/Survey-Research-Results/2010/07/Survey-Companies-Tighten-Link-Between-Executive-Pay-and-Performance, accessed on 19/01/2015.

[27] Turner, K., Belfield, R.: What makes remuneration committees effective? (2011) http://www.towerswatson.com/en/Insights/IC-Types/Survey-Research-Results/2011/09/What-makes-remuneration-committees-effective, accessed on 19/01/2015.

[28] van der Aalst, W.M.P.: Process Mining: Discovery, Conformance and Enhancement of Business Processes. Springer (2011)

[29] Li, C., Reichert, M., Wombacher, A.: Mining business process variants: Challenges, scenarios, algorithms. Data & Knowledge Engineering **70**(5) (2011) 409–434

[30] Hallerbach, A., Bauer, T., Reichert, M.: Configuration and Management of Process Variants. In: International Handbook on Business Process Management. Springer (2010) 237–255

[31] Hallerbach, A., Bauer, T., Reichert, M.: Capturing variability in business process models: The Provop approach. Journal of Software Maintenance and Evolution: Research and Practice **22**(6-7) (2010) 519–546

[32] Torres, V., Zugal, S., Weber, B., Reichert, M., Ayora, C., Pelechano, V.: A qualitative comparison of approaches supporting business process variability. In: Proc. Business Process Management Workshops (BPM Workshops '12). Volume 132 of LNBIP, Springer (2013) 560–572

[33] Ayora, C., Torres, V., Weber, B., Reichert, M., Pelechano, V.: VIVACE: A framework for the systematic evaluation of variability support in process-aware information systems. Information and Software Technology **57**(1) (2015) pp. 248–276

[34] Castellanos, M., Casati, F., Dayal, U., Shan, M.C.: A comprehensive and automated approach to intelligent business processes execution analysis. Distributed and Parallel Databases **16**(3) (2004) 239–273

[35] Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M., Shan, M.C.: Business process intelligence. Computers in Industry **53**(3) (2004) 321–343

[36] Scheer, A.W., Jost, W., Heß, H., Kronz, A., eds.: Corporate Performance Management. Springer (2006)

[37] Davenport, T.J.: Process Innovation: Reengineering Work through Information Technology. Harvard Business School Press (1993)

[38] Hammer, M., Champy, J.: Reengineering the Corporation: A Manifesto for Business Revolution. HarperBusiness (1993)

[39] Kettinger, W.J., Teng, J.T.C., Guha, S.: Business process change: A study of methodologies, techniques and tools. Management Information Systems Quarterly **21**(1) (1997) 55–80

[40] Simon, H.A.: On the concept of organizational goal. Administrative Science Quarterly **9**(1) (1964) 1–22

[41] Reichert, M., Rinderle-Ma, S., Dadam, P.: Flexibility in process-aware information systems. LNCS Transactions on Petri Nets and Other Models of Concurrency (ToPNoC) **II** (2009) 115–135

[42] Lenz, R., Reichert, M.: IT support for healthcare processes – premises, challenges, perspectives. Data & Knowledge Engineering **61**(1) (2007) 39–58

[43] van der Geer, S., Reijers, H.A., Krekels, G.A.M.: How to run an effective and efficient dermato-oncology unit. Journal of the German Society of Dermatology **8**(1) (2010) 15–18

[44] Bobrik, R., Reichert, M., Bauer, T.: Requirements for the visualization of system-spanning business processes. In: Proc. 1st Int'l Workshop on Business Process Monitoring and Performance Management (BPMPM'05), IEEE Computer Society Press (2005) 948–954

[45] Camp, R.C.: Benchmarking: the Search for Industry Best Practices that Lead to Superior Performance. Quality Press (1989)

[46] Software AG: Annual reports 2006-2013 `http://www.softwareag.com/corporate/inv_rel/finpub/annualreport/default.asp`, accessed on 17/01/2015.

[47] Dale, B.G., van der Wiele, T., van Iwaarden, J., eds.: Managing Quality. 5th edn. Wiley-Blackwell (2007)

[48] Phillips, L.W., Chang, D.R., Buzzell, R.D.: Product quality, cost position and business performance: A test of some key hypotheses. Journal of Marketing **47**(2) (1983) 26–43

[49] Capon, N., Farley, J.U., Hoenig, S.: Determinants of financial performance: a meta-analysis. Management Science **36**(10) (1990) 1143–1159

[50] Flynn, B.B., Schroeder, R.G., Sakakibara, S.: The impact of quality management practices on performance and competitive advantage. Decision Sciences **26**(5) (1995) 659–692

[51] Forker, L.B., Vickery, S.K., Droge, C.L.: The contribution of quality to business performance. International Journal of Operations & Production Management **16**(8) (1996) 44–62

[52] Adam, E.E., Corbett, L.M., Flores, B.E., Harrison, N.J., Lee, T.S., Rho, B.H., Ribera, J., Samson, D., Westbrook, R.: An international study of quality improvement approach and firm performance. International Journal of Operations & Production Management **17**(9) (1997) 842–873

[53] Reijers, H.A.: Implementing BPM systems: the role of process orientation. Business Process Management Journal **12**(4) (2006) 389–409

[54] Kohlbacher, M.: The effects of process orientation: a literature review. Business Process Management Journal **16**(1) (2010) 135–152

[55] Parasuraman, A., Zeithaml, V.A., Berry, L.L.: A conceptual model of service quality and its implications for future research. Journal of Marketing **49**(4) (1985) 41–50

[56] Harvey, J.: Service quality: a tutorial. Journal of Operations Management **16**(5) (1998) 583–597

[57] Porter, M.E.: Competitive Advantage: Creating and Sustaining Superior Performance. Free Press, New York (1985)

[58] Reijers, H.A.: Design and Control of Workflow Processes: Business Process Management for the Service Industry. Springer (2003)

[59] Miller, J.G., Vollmann, T.E.: The hidden factory. Harvard Business Review **63**(5) (1985) 142–150

[60] Emore, J.R., Ness, J.A.: The slow pace of meaningful change in cost systems. Journal of Cost Management **5**(4) (1991) 36–45

[61] Küting, K., Lorson, P.: Grenzplankostenrechnung versus Prozeßkostenrechnung. Der Betriebs-Berater **21**(46) (1991) 1421–1433 in German.

[62] Hollingsworth, D.: Workflow management coalition: The workflow reference model, issue 1.1 (1995) `http://www.wfmc.org/resources`, document reference WFMC-TC00-1003, accessed on 17/01/2015.

[63] International Organization for Standardization: ISO 9001:2005: Quality management systems – Requirements (2008)

[64] Elzinga, D.J., Horak, T., Lee, C.Y., Bruner, C.: Business process management: Survey and methodology. IEEE Transactions on Engineering Management **42**(2) (1995) 119–128

[65] Lindsay, A., Downs, D., Lunn, K.: Business processes – attempts to find a definition. Information and Software Technology **45**(15) (2003) 1015–1019

[66] Nordsieck, F.: Die schaubildliche Erfassung und Untersuchung der Betriebsorganisation. C. E. Poeschel (1932) in German.

[67] Thom, L., Reichert, M., Iochpe, C.: Activity patterns in process-aware information systems: Basic concepts and empirical evidence. International Journal of Business Process Integration and Management **4**(2) (2009) 93–110

[68] van der Aalst, W.M.P., Berens, P.J.S.: Beyond workflow management: product-driven case handling. In: Proc. Int'l ACM SIGGROUP Conf. on Supporting Group Work (GROUP'05), ACM (2001) 42–51

[69] Reijers, H.A., Limam, S., van der Aalst, W.M.P.: Product-based workflow design. Journal of Management Information Systems **20**(1) (2003) 229–262

[70] Müller, D., Reichert, M., Herbst, J.: Data-driven modeling and coordination of large process structures. In: Proc. 15th Int'l Conf. on Cooperative Information Systems (CoopIS'07). Volume 4803 of LNCS, Springer (2007) 131–149

[71] Weber, B., Mutschler, B., Reichert, M.: Investigating the effort of using business process management technology: Results from a controlled experiment. Science of Computer Programming **75**(5) (2010) 292–310

[72] Künzle, V., Reichert, M.: PHILharmonicFlows: towards a framework for object-aware process management. Journal of Software Maintenance and Evolution: Research and Practice **23**(4) (2011) 205–244

[73] Künzle, V.: Object-Aware Process Management. PhD thesis, Ulm University (2013)

[74] Cabanillas, C., Resinas, M., Ruiz Cortés, A.: Automated resource assignment in BPMN models using RACI matrices. In: Proc. 20th Int'l Conf. on Cooperative Information Systems (CoopIS'12). Volume 7565 of LNCS, Springer (2012) 56–73

[75] Smith, A.: An Inquiry into the Nature and Causes of the Wealth of Nations. 5th edn. Methuen & Co., London (1904)

[76] Fdhila, W., Indiono, C., Rinderle-Ma, S., Reichert, M.: Dealing with change in process choreographies: Design and implementation of propagation algorithms. Information Systems **49**(1) (2015) 1–24

[77] March, S.T., Smith, G.F.: Design and natural science research on information technology. Decision Support Systems **15**(4) (1995) 251–266

[78] van der Aalst, W.M.P., ter Hofstede, A.H.M.: YAWL: Yet Another Workflow Language. Information Systems Journal **30**(4) (2005) 245–275

[79] Reichert, M.: Dynamische Ablaufänderungen in Workflow-Management-Systemen. PhD thesis, Ulm University (2000) in German.

[80] The Object Management Group: Business Process Model and Notation: Version 2.0 (2011) `http://www.omg.org/spec/BPMN/2.0`, accessed on 17/01/2015.

[81] Reijers, H.A., Mans, R.S., van der Toorn, R.A.: Improved model management with aggregated business process models. Data and Knowledge Engineering **68**(2) (2009) 221–243

[82] Reichert, M., Kolb, J., Bobrik, R., Bauer, T.: Enabling personalized visualization of large business processes through parameterizable views. In: Proc. 27th ACM Symposium On Applied Computing (SAC'12), 9th Enterprise Engineering Track (EE'12), ACM Press (2012) 1653–1660

[83] Smirnov, S., Reijers, H.A., Weske, M.: From fine-grained to abstract process models: A semantic approach. Information Systems **37**(8) (2012) 784–797

[84] Kolb, J., Reichert, M.: A flexible approach for abstracting and personalizing large business process models. Applied Computing Review **13**(1) (2013) 6–17

[85] Cao, G., Clarke, S., Lehaney, B.: A critique of BPR from a holistic perspective. Business Process Management Journal **7**(4) (2001)

[86] Hammer, M.: The process audit. Harvard Business Review **85**(4) (2007) 111–123

[87] Porter, M.E.: What is strategy? Harvard Business Review **74**(6) (1996) 61–78

[88] Dale, B.G.: The Received Wisdom on TQM. In: Managing Quality. 5th edn. Wiley-Blackwell (2007) 58–73

[89] International Organization for Standardization: ISO 9000:2005: Quality management systems – Fundamentals and vocabulary (2005)

[90] Guler, I., Guillén, M.F., Macpherson, J.M.: Global competition, institutions, and the diffusion of organizational practices: The international spread of ISO 9000 quality certificates. Administrative Science Quarterly **47**(2) (2002) 207–232

[91] Neumayer, J., Perkins, R.: Uneven geographies of organizational practice: explaining the cross-national transfer and diffusion of ISO 9000. Economic Geography **81**(3) (2005) 237–259

[92] International Organization for Standardization: The ISO Survey - 2007 (2008) `http://www.iso.org/iso/survey2007.pdf`, accessed on 17/01/2015.

[93] Garvin, D.A.: What does "product quality" really mean? Sloan Management Review **26**(1) (1984) 25–43

[94] Garvin, D.A.: Competing on the eight dimensions of quality. Harvard Business Review **65**(6) (1987) 101–109

[95] Sower, V.E., Fair, F.K.: There is more to quality than continuous improvement: listening to Plato. Quality Management Journal **12**(1) (2005) 8–20

[96] Christensen, C.M., Rosenbloom, R.S.: Explaining the attacker's advantage: technological paradigms, organizational dynamics, and the value network. Research Policy **24**(2) (1995) 233–257

[97] Bower, J.L., Christensen, C.M.: Disruptive technologies: catching the wave. Harvard Business Review **73**(1) (1995) 43–53

[98] Reeves, C.A., Bednar, D.A.: Defining quality: alternatives and implications. Academy of Management Review **19**(3) (1994) 419–445

[99] Leffler, K.B.: Ambiguous changes in product quality. American Economic Review **72**(5) (1982) 956–967

[100] Saaty, T.L.: How to make a decision: The analytic hierarchy process. European Journal on Operations Management **48**(1) (1990) 9–26

[101] Feigenbaum, A.V.: Quality Control: Principles, Practice, and Administration. McGraw-Hill, New York (1951)

[102] Gospel, H., Sako, M.: The unbundling of corporate functions: the evolution of shared services and outsourcing in human resource management. Industrial and Corporate Change **19**(5) (2010) 1367–1396

[103] Simon, H.A.: The Sciences of the Artificial. 3rd edn. MIT Press (1996)

[104] Wieringa, R.: Design science as nested problem solving. In: Proc. 4th Int'l Conf. on Design Science Research in Information Systems and Technology (DESRIST'09), ACM (2009) 1–12

[105] Hevner, A.R., March, S.T., Park, J., Ram, S.: Design science in information systems research. Management Information Systems Quarterly **28**(1) (2004) 75–105

[106] Epstein, M.K., Henderson, J.C.: Data envelopment analysis for managerial control and diagnosis. Decision Sciences **20**(1) (1989) 90–119

[107] Kennerley, M., Neely, A.: Performance Measurement Frameworks: A Review. In: Business Performance Measurement: Theory and Practice. Cambridge University Press (2002) 145–155

[108] Healy, P.M.: The effect of bonus schemes on accounting decisions. Journal of Accounting and Economics **7**(1-3) (1985) 85–107

[109] Prendergast, C.: The provision of incentives in firms. Journal of Economic Literature **37**(1) (1999) 7–63

[110] Thommen, J.P.: Gabler Wirtschaftslexikon, Keyword: Wirtschaftlichkeitsprinzip (2010) `http://wirtschaftslexikon.gabler.de/Archiv/8349/wirtschaftlichkeitsprinzip-v5.html`, accessed on 17/01/2015, in German.

[111] Vancil, R.F.: What management control do you need? Harvard Business Review **51**(2) (1973) 75–86

[112] International Accouting Standards Board: Framework for the Preparation and Presentation of Financial Statements (1989) `http://eifrs.ifrs.org/eifrs/Menu`, accessed on 17/01/2015.

[113] Eisenhardt, K.M.: Control: Organizational and economic approaches. Management Science **31**(2) (1985) 134–149

[114] Ouchi, W.G.: A conceptual framework for the design of organizational control mechanisms. Management Science **25**(9) (1979) 833–848

[115] Simon, H.A.: On how to decide what to do. Bell Journal of Economics **9**(2) (1978) 494–507

[116] March, J.G.: Bounded rationality, ambiguity, and the engineering of choice. Bell Journal of Economics **9**(2) (1978) 587–608

[117] IT Governance Institute: CobiT 5 Framework (2012) `https://cobitonline.isaca.org/`, accessed on 17/01/2015.

[118] Arraj, V.: ITIL: The Basics (2011) `https://www.axelos.com/case-studies-and-white-papers/itil-the-basics-white-paper`, accessed on 14/01/2015.

[119] APICS Supply Chain Council: Supply Chain Operations Reference (SCOR) Model `http://www.apics.org/sites/apics-supply-chain-council/frameworks/scor`, accessed on 18/01/2015.

[120] Küting, K., Weber, C.P., Boecker, C.: Die Bilanzanalyse: Beurteilung von Abschlüssen nach HGB und IFRS. 8th edn. Schäffer-Poeschel (2006) in German.

[121] Nestic, S., Stefanovic, M., Djordjevic, A., Arsovski, S., Stojanovic, S.: An assessment and optimization of quality of strategic process. International Journal for Quality Research **7**(4) (2013) 465–478

[122] Kaplan, R.S., Norton, D.P.: The balanced scorecard: Measures that drive performance. Harvard Business Review **70**(1) (1992) 71–79

[123] Heravizadeh, M., Mendling, J., Rosemann, M.: Dimensions of business processes quality (QoBP). In: Proc. Business Process Management Workshops (BPM Workshops '08). Volume 17 of LNBIP, Springer (2009) 80–91

[124] Heravizadeh, M.: Quality-aware Business Process Management. PhD thesis, Queensland University of Technology, Australia (2009)

[125] Heinrich, R., Paech, B.: Defining the quality of business processes. In: Proc. Modellierung 2010. Volume P-161 of LNI, Bonner Köllen (2010) 133–148

[126] Heidari, F., Loucopoulos, P., Kedad, Z.: A quality-oriented business process meta-model. In: Proc. 7th Int'l Workshop on Enterprise and Organizational Modeling and Simulation (EOMAS'11). Volume 88 of LNBIP, Springer (2011) 85–99

[127] Heidari, F., Loucopoulos, P., Brazier, F.: Business process modelling for measuring quality. International Journal on Advances in Intelligent Systems **6**(3-4) (2013) 342–355

[128] Ohno, T.: Toyota Production System: Beyond Large-scale Production Processes. 3rd edn. Productivity Press (1988)

[129] Reijers, H.A., Limam Mansar, S.: Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics. Omega **33**(4) (2005) 283–306

[130] Limam Mansar, S., Reijers, H.A.: Best practices in business process redesign: use and impact. Business Process Management Journal **13**(2) (2007) 193–213

[131] Jansen-Vullers, M.H., Kleingeld, P.A.M., Loosschilder, M.W.N.C., Netjes, M., Reijers, H.A.: Trade-offs in the performance of workflows – quantifying the impact of best practices. In: Proc. Business Process Management Workshops (BPM Workshops '07). Volume 4928 of LNCS, Springer (2008) 108–119

[132] Nissen, M.E.: Redesigning reengineering through measurement-driven inference. MIS Quarterly **22**(4) (1998) 509–534

[133] Nissen, M.E.: Knowledge-based knowledge management in the reengineering domain. Decision Support Systems **27**(1-2) (1999) 47–65

[134] Speck, M., Schnetgöke, N.: To-be Modeling and Process Optimization. In: Process Management: A Guide for The Design of Business Processes. 2nd edn. Springer (2011) 157–185

[135] Becker, J., Rosemann, M., von Uthmann, C.: Guidelines of Business Process Modeling. In: Business Process Management: Models, Techniques, and Empirical Studies. Volume 1806 of LNCS Springer (2000) 241–262

[136] The Object Management Group: Business process maturity model (BPMM) (2008) `http://doc.omg.org/formal/08-06-01.pdf`, accessed on 17/01/2015.

[137] Smith, H., Fingar, P.: Process management maturity models (2004) `http://www.bptrends.com/bpt/wp-content/publicationfiles/07-04%20COL%20Maturity%20Models-%20Smith-Fingar.pdf`, accessed on 17/01/2015.

[138] Rosemann, M., vom Brocke, J.: The Six Core Elements of Business Process Management. In: Handbook on Business Process Management 1. Springer (2010) 107–122

[139] Lindland, O.I., Sindre, G., Sølvberg, A.: Understanding quality in conceptual modeling. IEEE Software **11**(2) (1994) 42–49

[140] Krogstie, J., Lindland, O.I., Sindre, G.: Defining quality aspects for conceptual models. In: Proc. IFIP Int'l Working Conf. on Information System Concepts (ISCO3), Chapman & Hall (1995) 216–231

[141] Krogstie, J., Jørgensen, H.D.: Quality of interactive models. In: Proc. 21st Int'l Conf. on Conceptual Modeling Workshops (ER Workshops '02). Volume 2503 of LNCS, Springer (2002) 351–363

[142] Reijers, H.A., Mendling, J., Recker, J.: Business Process Quality Management. In: Handbook on Business Process Management 1. Springer (2010) 167–185

[143] van der Aalst, W.M.P.: The application of Petri nets to workflow management. Journal of Circuits, Systems and Computers **8**(1) (1998) 21–26

[144] Mendling, J.: Detection and Prediction of Errors in EPC Business Process Models. PhD thesis, Wirtschaftsuniversität Wien, Austria (2007)

[145] Hallerbach, A., Bauer, T., Reichert, M.: Guaranteeing soundness of configurable process variants in Provop. In: Proc. 11th IEEE Conf. on Commerce and Enterprise Computing (CEC'09), IEEE Computer Society Press (2009) 98–105

[146] Weber, B., Reichert, M.: Refactoring process models in large process repositories. In: Proc. 20th Int'l Conf. on Advanced Information Systems Engineering (CAiSE'08). Volume 5074 of LNCS, Springer (2008) 124–139

[147] Weber, B., Reichert, M., Mendling, J., Reijers, H.A.: Refactoring large process model repositories. Computers in Industry **62**(5) (2011) 467–486

[148] Li, C., Reichert, M., Wombacher, A.: Discovering reference models by mining process variants using a heuristic approach. In: Proc. 7th Int'l Conf. on Business Process Management (BPM'09). Volume 5701 of LNCS, Springer (2009) 344–362

[149] Li, C., Reichert, M., Wombacher, A.: The MinAdept clustering approach for discovering reference process models out of process variants. International Journal of Cooperative Information Systems **19**(3-4) (2010) 159–203

[150] Weber, B., Sadiq, S., Reichert, M.: Beyond rigidity - dynamic process lifecycle support: A survey on dynamic changes in process-aware information systems. Computer Science - Research and Development **23**(2) (2009) 47–65

[151] Rinderle, S., Weber, B., Reichert, M., Wild, W.: Integrating process learning and process evolution - a semantics based approach. In: Proc. 3rd Int'l Conf. on Business Process Management (BPM'05). Volume 3649 of LNCS, Springer (2005) 252–267

[152] Sadiq, S., Governatori, G., Namiri, K.: Modeling control objectives for business process compliance. In: Proc. 5th Int'l Conf. on Business Process Management (BPM'07). Volume 4714 of LNCS, Springer (2007) 149–164

[153] Governatori, G., Sadiq, S.: The Journey to Business Process Compliance. In: Handbook of Research on Business Process Modelling. Hershey (2009) 426–454

[154] Ly, L.T., Rinderle, S., Dadam, P.: Integration and verification of semantic constraints in adaptive process management systems. Data & Knowledge Engineering **64**(1) (2008) 3–23

[155] Ly, L.T., Knuplesch, D., Rinderle-Ma, S., Goeser, K., Pfeifer, H., Reichert, M., Dadam, P.: SeaFlows toolset - compliance verication made easy for process-aware information systems. In: Proc. CAiSE'10 Forum - Information Systems Evolution. Volume 6412 of LNCS, Springer (2010) 332–346

[156] Knuplesch, D., Reichert, M., Fdhila, W., Rinderle-Ma, S.: On enabling compliance of cross-organizational business processes. In: Proc. 11th Int'l Conf. Business Process Management (BPM'13). Volume 8094 of LNCS, Springer (2013) 146–154

[157] Knuplesch, D., Reichert, M., Ly, L.T., Kumar, A., Rinderle-Ma, S.: Visual modeling of business process compliance rules with the support of multiple perspectives. In: Proc. 32nd Int'l Conf. on Conceptual Modeling (ER'13). Volume 8217 of LNCS, Springer (2013) 106–120

[158] Gebhart, M., Mevius, M., Wiedmann, P.: Application of business process quality models in agile business process management. In: Proc. 6th Int'l Conf. on Information, Process, and Knowledge Management (eKNOW'14), IARIA (2014) 152–158

[159] Moody, D.L.: Theoretical and practical issues in evaluating the quality of conceptual models: current state and future directions. Data & Knowledge Engineering **55**(3) (2005) 243–276

[160] Mendling, J., Reijers, H.A., van der Aalst, W.M.P.: Seven process modeling guidelines (7PMG). Information and Software Technology **52**(2) (2010) 127–136

[161] Mendling, J., Recker, J., Reijers, H.A.: On the usage of labels and icons in business process modeling. International Journal of Information Systems Modeling and Design **1**(2) (2010) 40–58

[162] Reijers, H.A., Freytag, T., Mendling, J., Eckleder, A.: Syntax highlighting in business process models. Decision Support Systems **51**(3) (2011) 339–349

[163] La Rosa, M., ter Hofstede, A.H.M., Wohed, P., Reijers, H.A., Mendling, J., van der Aalst W. M. P.: Managing process model complexity via concrete syntax modifications. IEEE Transactions on Industrial Informatics **7**(2) (2011) 255–265

[164] La Rosa, M., Wohed, P., Mendling, J., ter Hofstede, A.H.M., Reijers, H.A., van der Aalst, W.M.P.: Managing process model complexity via abstract syntax modifications. IEEE Transactions on Industrial Informatics **7**(4) (2011) 614–629

[165] Reijers, H.A., Mendling, J.: A study into the factors that influence the understandability of business process models. IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans **41**(3) (2011) 449–461

[166] Reijers, H.A., Mendling, J., Dijkman, R.M.: Human and automatic modularizations of process models to enhance their comprehension. Information Systems **36**(5) (2011) 881–897

[167] Kock, N., Murphy, F.: Redesigning Acquisition Processes: A New Methodology Based on the Flow of Knowledge and Information. Defense Acquisition University Press (2001)

[168] Kock, N.: Changing the focus of business process redesign from activity flows to information flows: A defense acquisition application. Acquisition Review Quarterly **8**(2) (2001) 93–110

[169] Kock, N.: Communication-focused business process redesign: Assessing a communication flow optimization model through an action research study at a defense contractor. IEEE Transactions on Professional Communication **46**(1) (2003) 35–54

[170] Danesh, A., Kock, N.: An experimental study of process representation approaches and their impact on perceived modeling quality and redesign success. Business Process Management Journal **11**(6) (2005) 724–735

[171] Guceglioglu, A.S., Demirors, O.: Using software quality characteristics to measure business process quality. In: Proc. 3rd Int'l Conf. on Business Process Management (BPM'05). Volume 3649 of LNCS, Springer (2005) 374–379

[172] Cardoso, J.: Business process quality metrics: Log-based complexity of workflow patterns. In: Proc. 15th Int'l Conf. on Cooperative Information Systems (CoopIS'07). Volume 4803 of LNCS, Springer (2007) 427–434

[173] Vanderfeesten, I.T.P., Cardoso, J., Mendling, J., Reijers, H.A., van der Aalst, W.: Quality Metrics for Business Process Models. In: Workflow Handbook 2007. Future Strategies (2007) 179–190

[174] Vanderfeesten, I.T.P., Reijers, H.A., van der Aalst, W.M.P.: Evaluating workflow process designs using cohesion and coupling metrics. Computers in Industry **59**(5) (2008) 420–437

[175] Mendling, J.: Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness. Springer (2008)

[176] Buijs, J.C.A.M., van Dongen, B.F., van der Aalst W. M. P.: On the role of fitness, precision, generalization and simplicity in process discovery. In: Proc. 20th Int'l Conf. on Cooperative Information Systems (CoopIS'12). Volume 7565 of LNCS, Springer (2012) 305–322

[177] Vergidis, K., Tiwari, A., Majeed, B.: Business process analysis and optimization: Beyond reengineering. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews **38**(1) (2008) 69–82

[178] van der Aalst, W.M.P., van Hee, K.M.: Business process redesign: A Petri-net-based approach. Computers in Industry **29**(1-2) (1996) 15–26

[179] Oliveira, C.A.L., Lima, R.M.F., Reijers, H.A., Ribeiro, J.T.S.: Quantitative analysis of resource-constrained business processes. IEEE Transactions on Systems, Man, And Cybernetics – Part A: Systems and Humans **42**(3) (2012) 669–684

[180] Hofacker, I., Vetschera, R.: Algorithmical approaches to business process design. Computers and Operations Research **28**(13) (2001) 1253–1275

[181] Glykas, M.M.: Effort based performance measurement in business process management. Knowledge and Process Mgmt. **18**(1) (2011) 10–33

[182] Grigori, D., Casati, F., Dayal, U., Shan, M.C.: Improving business process quality through exception understanding, prediction, and prevention. In: Proc. 27th Int'l Conf. on Very Large Data Bases (VLDB'01), Morgan Kaufmann (2001) 159–168

[183] Heckl, D., Moormann, J.: Process Performance Management. In: Handbook on Business Process Management 2. Springer (2010) 115–135

[184] Rosemann, M., de Bruin, T., Hueffner, T.: A model for business process management maturity. In: Proc. 15th Australasian Conf. on Information Systems (ACIS'04), AIS (2004) Paper 6.

[185] De Bruin, T., Rosemann, M.: Towards a business process management maturity model. In: Proc. 13th European Conf. on Information Systems (ECIS'05), AIS (2005) Paper 37.

[186] De Bruin, T., Rosemann, M.: Towards understanding strategic alignment of business process management. In: Proc. 17th Australasian Conf. on Information Systems (ACIS'06), AIS (2006) Paper 82.

[187] International Accouting Standards Board: International Accounting Standard 16: Property, Plant and Equipment (2003) `http://eifrs.ifrs.org/eifrs/Menu`, accessed on 17/01/2015.

[188] Kilger, W., Pampel, J.R., Vikas, K.: Flexible Plankostenrechnung und Deckungsbeitragsrechnung. 12th edn. Gabler Verlag (2007) in German.

[189] Lanen, W., Anderson, S., Maher, M.: Fundamentals of Cost Accounting. 3rd edn. McGraw-Hill (2010)

[190] Oxford University Press: Oxford Dictionaries Online, Keyword: Efficacy `http://oxforddictionaries.com`, accessed on 17/01/2015.

[191] Scheer, A.W., Thomas, O., Adam, O.: Process Modeling Using Event-Driven Process Chains. In: Process-aware Information Systems: Bridging People and Software Through Process Technology. John Wiley & Sons (2005) 119–145

[192] Ponnalagu, K., Narendra, N.C., Ghose, A., Chiktey, N., Tamilselvam, S.: Goal oriented variability modeling in service-based business processes. In: Proc. 11th Int'l Conf. Service-Oriented Computing (ICSOC'13). Volume 8274 of LNCS, Springer (2013) 499–506

[193] Hull, E., Jackson, K., Dick, J.: Requirements Engineering. 3rd edn. Springer (2011)

[194] van Lamsweerde, A.: Goal oriented requirements engineering: a guided tour. In: Proc. 5th IEEE Int'l Symp. on Requirements Engineering (RE'01), IEEE (2001) 249–262

[195] Sølvberg, A.: Data and What they Refer to. In: Conceptual Modeling. Volume 1565 of LNCS Springer (1999) 211–226

[196] Krogstie, J., Sindre, G., Jørgensen, H.: Process models representing knowledge for action: a revised quality framework. European Journal of Information Systems **15**(1) (2006) 91–102

[197] Kueng, P., Kawalek, P.: Goal-based business process models: creation and evaluation. Business Process Management Journal **3**(1) (1997) 17–38

[198] Neiger, D., Churilov, L.: Goal-oriented business process modeling with EPCs and value-focused thinking. In: Proc. 2nd Int'l Conf. on Business Process Management (BPM'04). Volume 3080 of LNCS, Springer (2004) 98–115

[199] Soffer, P., Wand, Y.: On the notion of soft-goals in business process modeling. Business Process Management Journal **11**(6) (2005) 663–679

[200] Lin, Y., Sølvberg, A.: Goal annotation of process models for semantic enrichment of process knowledge. In: Proc. 19th Int'l Conf. on Advanced Information Systems Engineering (CAiSE'07). Volume 4495 of LNCS, Springer (2007) 355–369

[201] Markovic, I., Kowalkiewicz, M.: Linking business goals to process models in semantic business process modeling. In: Proc. 12th Int'l IEEE Enterprise Distributed Object Computing Conf. (EDOC'08), IEEE (2008) 332–338

[202] Cardoso, E.C.S.: Towards a methodology for goal-oriented enterprise management. In: Proc. 17th IEEE Int'l Enterprise Distributed Object Computing Conf. Workshops (EDOC Workshops '13). (2013) 94–103

[203] Goedertier, S., Vanthienen, J.: Designing compliant business processes with obligations and permissions. In: Proc. Business Process Management Workshops (BPM Workshops '06). Volume 4103 of LNCS, Springer (2006) 5–14

[204] Goedertier, S., Haesen, R., Vanthienen, J.: Rule-based business process modelling and enactment. Int. J. Business Process Integration and Management **3**(3) (2008) 194–207

[205] Yu, E.S.K., Mylopoulos, J.: Using goals, rules, and methods to support reasoning in business process reengineering. International Journal of Intelligent Systems in Accounting, Finance and Management **5**(1) (1996) 1–13

[206] Dardenne, A., van Lamsweerde, A., Fickas, S.: Goal-directed requirements acquisition. Science of Computer Programming **20**(1-2) (1993) 3–50

[207] Engelsman, W., Quartel, D., Jonkers, H., van Sinderen, M.: Extending enterprise architecture modelling with business goals and requirements. Enterprise Information Systems **5**(1) (2011) 9–36

[208] van der Aalst, W.M.P.: On the automatic generation of workflow processes based on product structures. Computers in Industry **39**(2) (1999) 97–111

[209] Reijers, H.A.: Product-based design of business processes applied within the financial services. Journal of Research and Practice in Information Technology **34**(2) (2002) 34–46

[210] van der Aalst, W.M.P.: Reengineering knock-out processes. Decision Support Systems **30**(4) (2001) 451–468

[211] Pesic, M., van der Aalst W. M. P.: A declarative approach for flexible business processes management. In: Proc. Business Process Management Workshops (BPM Workshops '06). Volume 4103 of LNCS, Springer (2006) 108–119

[212] van der Aalst, W.M.P., Pesic, M.: Decserflow: Towards a truly declarative service flow language. Technical Report BPM-06-21, BPM Center (2006) `http://bpmcenter.org/wp-content/uploads/reports/2006/BPM-06-21.pdf`, accessed on 19/11/2014.

[213] Westergaard, M.: Declare renaming (2011) `http://www.win.tue.nl/declare/2011/11/declare-renaming/`, accessed on 22/11/2014.

[214] Clarke, E.J., Grumberg, O., Peled, D.: Model Checking. MIT (1999)

[215] Pichler, P., Weber, B., Zugal, S., Pinggera, J., Mendling, J., Reijers, H.A.: Imperative versus declarative process modeling languages: An empirical investigation. In: Proc. Business Process Management Workshops (BPM Workshops '11). Volume 99 of LNBIP, Springer (2012) 383–394

[216] Zugal, S., Soffer, P., Haisjackl, C., Pinggera, J., Reichert, M., Weber, B.: Investigating expressiveness and understandability of hierarchy in declarative business process models. Software & Systems Modeling (2013) DOI 10.1007/s10270-013-0356-2.

[217] Haisjackl, C., Barba, I., Zugal, S., Soffer, P., Hadar, I., Reichert, M., Pinggera, J., Weber, B.: Understanding declare models: Strategies, pitfalls, empirical results. Software & Systems Modeling (2014) DOI 10.1007/s10270-014-0435-z.

[218] Hildebrandt, T.T., Mukkamala, R.R.: Declarative event-based workflow as distributed dynamic condition response graphs. In: Proc. 3rd Workshop on Programming Language Approaches to Concurrency and communication-cEntric Software (PLACES'10). (2011)

[219] The Object Management Group: Semantics of Business Vocabulary and Business Rules (SBVR): Version 1.2 (2013) `http://www.omg.org/spec/SBVR/1.2/PDF/`, accessed on 17/01/2015.

[220] Petri, C.A.: Kommunikation mit Automaten. PhD thesis, Bonn University, Germany (1962) in German.

[221] van der Aalst, W.M.P., Stahl, C.: Modeling Business Processes: A Petri Net-Oriented Approach. MIT Press (2011)

[222] Scheer, A.W.: Wirtschaftsinformatik: Referenzmodelle für industrielle Geschäftsprozesse: Studienausgabe. Springer (1995) in German.

[223] van der Aalst, W.M.P., ter Hofstede, A.H.M., Kiepuszewski, B., Barros, A.P.: Workflow patterns. Distributed and Parallel Databases **14**(1) (2003) 5–51

[224] Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms. 3rd edn. MIT Press (2009)

[225] Rinderle, S.: Schema Evolution in Process Management Systems. PhD thesis, Ulm University (2004)

[226] Vanhatalo, J., Völzer, H., Leymann, F.: Faster and more focused control-flow analysis for business process models through SESE decomposition. In: Proc. 5th Int'l Conf. on Service-oriented Computing (ICSOC'07). Volume 4749 of LNCS, Springer (2007) 43–55

[227] Johnson, R., Pearson, D., Pingali, K.: The program structure tree: computing control regions in linear time. In: Proc. ACM SIGPLAN Conf. on Programming Language Design and Implementation (PLDI'94), ACM (1994) 171–185

[228] Vanhatalo, J., Völzer, H., Koehler, J.: The refined process structure tree. Data & Knowledge Engineering **68**(9) (2009) 793–818

[229] Object Management Group: OMG unified modeling language (OMG UML): Infrastructure (2011) `http://www.omg.org/spec/UML/2.4.1/Infrastructure/PDF/`, accessed on 17/01/2015.

[230] Mutschler, B., Reichert, M., Rinderle, S.: Analyzing the dynamic cost factors of process-aware information systems: A model-based approach. In: Proc. 19th Int'l Conf. on Advanced Information Systems Engineering (CAiSE'07). Volume 4495 of LNCS, Springer (2007) 589–603

[231] Mutschler, B., Reichert, M.: On modeling and analyzing cost factors in information systems engineering. In: Proc. 20th Int'l Conf. on Advanced Information Systems Engineering (CAiSE'08). Volume 5074 of LNCS, Springer (2008) 510–524

[232] Mutschler, B., Reichert, M.: Evaluation patterns for analyzing the costs of enterprise information systems. In: Proc. 21st Int'l Conf. on Advanced Information Systems Engineering (CAiSE'09). Volume 5565 of LNCS, Springer (2009) 379–394

[233] International Accouting Standards Board: International Accounting Standard 2: Inventories (2009) `http://eifrs.ifrs.org/eifrs/Menu`, accessed on 17/01/2015.

[234] Hammer, M.: What is Business Process Management? In: Handbook on Business Process Management 1. Springer (2010) 3–16

[235] Berk, J., DeMarzo, P.: Corporate Finance. 2nd edn. Pearson Education (2011)

[236] Lehner, F.: Organizational Memory: Konzepte und Systeme für das organisatorische Lernen und das Wissensmanagement. Hanser (2000) in German.

[237] Shingo, S.: A Study of the Toyota Production System from an Industrial Engineering Viewpoint. Productivity Press (1989)

[238] Gelles, G.M., Mitchell, D.W.: Returns to scale and economies of scale: Further observations. Journal of Economic Education **27**(3) (1996) 259–261

[239] Betriebsverfassungsgesetz: § 87 Abs. 1 Nr. 9. Bundesgesetzblatt (50) (2001) 2518–2551 German federal statute, in German.

[240] Gronau, N.: Enterprise Resource Planning: Architektur, Funktionen und Management von ERP-Systemen. 2nd edn. Oldenbourg (2010)

[241] Dorn, C., Edwards, G., Medvidovic, N.: Analyzing design tradeoffs in large-scale socio-technical systems through simulation of dynamic collaboration patterns. In: Proc. 20th Int'l Conf. on Cooperative Information Systems (CoopIS'12). Volume 7565 of LNCS, Springer (2012) 362–379

[242] Casati, F., Dayal, U., Sayal, M., Shan, M.C.: Business process intelligence (2002) `http://www.hpl.hp.com/techreports/2002/HPL-2002-119.pdf?jumpid=reg_R1002_USEN`, accessed on 16/01/2015.

[243] Shewhart, W.A.: Statistical Method from the Viewpoint of Quality Control. Dover Publications (2011)

[244] Institut der Wirtschaftsprüfer: IDW Prüfungsstandard: Feststellung und Beurteilung von Fehlerrisiken und Reaktionen des Abschlussprüfers auf die beurteilten Fehlerrisiken (IDW PS 261 n.F.) (2012) in German.

[245] Keuper, F., Lueg, K.E., eds.: Finance Bundling and Finance Transformation: Shared Services Next Level. Gabler (2013)

[246] Kowalkowski, C.: The service function as a holistic management concept. Journal of Business & Industrial Marketing **26**(7) (2011) 484–492

[247] Weikum, G., Vossen, G.: Fundamentals of Transactional Information Systems: Theory, Algorithms, and the Practice of Concurrency Control and Recovery. Elsevier (2001)

[248] Gray, J.: The transaction concept: Virtues and limitations. In: Proc. 7th Int'l Conf. on Very Large Databases (VLDB'81), IEEE Computer Society Press (1981) 144–154

[249] Davis, R.: Thinking end-to-end: Time for Cinderella to go to the ball? (2010) `http://www.bptrends.com/processes-in-practice`, accessed on 16/01/2015.

[250] Reiterer, H., Mann, T.M., Mußler, G., Bleimann, U.: Visualisierung von entscheidungsrelevanten Daten für das Management. HMD, Praxis der Wirtschaftsinformatik **212** (2000) 71–83 in German.

[251] Coenenberg, A.G., Fischer, T.M., Günther, T.: Kostenrechnung und Kostenanalyse. 6th edn. Schäffer-Poeschel (2007) in German.

[252] Bodenstaff, L., Wombacher, A., Reichert, M., Jaeger, M.C.: Monitoring dependencies for SLAs: The MoDe4SLA approach. In: Proc. IEEE 5th Int'l Conf. on Services Computing (SCC'08), IEEE (2008) 21–29

[253] Bodenstaff, L., Wombacher, A., Reichert, M.: Empirical validation of MoDe4SLA; approach for managing service compositions. In: Proc. 14th Int'l Conf. on Business Information Systems (BIS'11). Volume 87 of LNBIP, Springer (2011) 98–110

[254] Trienekens, J.J.M., Kusters, R.J., Rendering, B., Stokla, K.: Business objectives as drivers for process improvement: Practices and experiences at Thales Naval The Netherlands (TNNL). In: Proc. 2nd Int'l Conf. on Business Process Management (BPM'04). Volume 3080 of LNCS, Springer (2004) 33–48

[255] Stein, S., Ruh, W., Selg, M., Trost, M., Riedel, A.: Transfer Pricing Aspects of Cross-border Service Provisions. In: Finance Bundling and Finance Transformation: Shared Services Next Level. Gabler (2013) 487–523

[256] Ahlrichs, F., Knuppertz, T.: Controlling von Geschäftsprozessen: prozessorientierte Unternehmenssteuerung umsetzen. Schäffer-Poeschel (2006)

[257] Bundesministerium der Finanzen: Grundsätze ordnungsmäßiger DV-gestützter Buchführungssysteme (GoBS). Bundessteuerblatt **45** (1995) 738–754 in German.

[258] Chaudhuri, S., Dayal, U., Narasayya, V.: An overview of business intelligence technology. Communications of the ACM **54**(8) (2011) 88–98

[259] Boehm, B.W.: A spiral model of software development and enhancement. IEEE Computer **21**(5) (1988) 61–72

[260] Dadam, P., Reichert, M.: The ADEPT project: A decade of research and development for robust and flexible process support – challenges and achievements. Computer Science - Research and Development **23** (2009)

[261] International Organization for Standardization: ISO/IEC 9126-1: Software engineering – Product quality – Part 1: Quality model (2001)

[262] IBM: White paper: WebSphere MQ – the right information in the right place and time `http://www.ibm.com/software/products/en/websphere-mq/`, accessed on 17/01/2015.

[263] Anandarajan, M., Anandarajan, A., Srinivasan, C.A., eds.: Business Intelligence Techniques: A Perspective from Accounting and Finance. Springer (2003)

[264] Myers, M.D.: Qualitative Research in Business & Management. Sage Publications (2009)

[265] Zelkowitz, M.V., Wallace, D.: Experimental models for validating technology. IEEE Computer **31**(5) (May 1998) 23–31

[266] Wieringa, R., Heerkens, H., Regnell, B.: How to write and read a scientific evaluation paper. In: Proc. 17th IEEE Int'l Requirements Engineering Conf. (RE'09), IEEE Computer Society (2009) 361–364

[267] Myers, M.D., Newman, M.: The qualitative interview in is research: Examining the craft. Information and Organization **17**(1) (2007) 2–26

[268] Yin, R.K.: Case Study Research: Design and Methods. 4th edn. Volume 5 of Applied Social Research Methods. Sage Publications (2009)

[269] Heravizadeh, M., Mendling, J., Rosemann, M.: Root cause analysis in business processes (2008) `http://eprints.qut.edu.au/13572/`, accessed on 17/01/2015.

[270] World Wide Web Consortium: QoS for web services: Requirements and possible approaches (2003) `http://www.w3c.or.kr/kr-office/TR/2003/NOTE-ws-qos-20031125/`, W3C Working Group Note 25, accessed on 17/01/2015.

[271] International Organization for Standardization: ISO/IEC 25010: Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - System and software quality models (2011)

[272] Wagner, S.: Software Product Quality Control. Springer (2013)

[273] Balzert, H.: Lehrbuch der Softwaretechnik: Basiskonzepte und Requirements Engineering. 3rd edn. Spektrum Akademischer Verlag (2009) in German.

[274] Eckerson, W.W.: Three tier client/server architecture: Achieving scalability, performance, and efficiency in client server applications. Open Information Systems **3**(20) (1995) 46–50

[275] Mutschler, B., Reichert, M., Bumiller, J.: Unleashing the effectiveness of process-oriented information systems: Problem analysis, critical success factors and implications. IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews **38**(3) (2008) 280–291

[276] zur Muehlen, M., Ho, D.T.: Service process innovation: A case study of BPMN in practice. In: Proc. 41st Hawaii Int'l Conf. on System Sciences (HICSS'08), IEEE Computer Society (2008) 372

[277] Gamma, E., Helm, R., Johnson, R., Vlissides, J.: Design patterns: Abstraction and reuse of object-oriented design. In: Proc. 7th European Conf. on Object-Oriented Programming (ECOOP'93). Volume 707 of LNCS, Springer (1993) 406–431

[278] Günther, C.W., Rozinat, A.: Disco: Discover your processes. In: Proc. Demonstration Track of the 10th Int'l Conf. on Business Process Management (BPM'12). (2012) 40–44

[279] van Dongen, B.F., Alves de Medeiros, A.K., Verbeek, H.M.W., Weijters, A.J.M.M., van der Aalst, W.M.P.: The prom framework: A new era in process mining tool support. In: Proc. 26th Int'l Conf. on Application and Theory of Petri Nets (ICATPN'05). Volume 3536 of LNCS, Springer (2005) 444–454

[280] Celonis GmbH: Celonis: Our product `http://www.celonis.de/en/discover/our-product`, accessed on 16/01/2015.

[281] Ittner, C.D., Larcker, D.F.: Assessing empirical research in managerial accounting: a value based management perspective. Journal of Accounting and Economics **32**(1-3) (2001) 349–410

[282] Society for Human Resource Management: Executive brief: What factors influence cost-per-hire? (2012) `http://www.shrm.org/benchmarks`, accessed on 17/01/2015.

[283] Pyzdek, T., Keller, P.: The Six Sigma Handbook. McGraw-Hill (2009)

[284] Mans, R.S., Reijers, H.A., Berends, H., Bandara, W., Prince, R.: Business process mining success. In: Proc. 21st European Conf. on Information Systems (ECIS'13). (2013) Paper 102.

[285] Limam Mansar, S., Reijers, H.A., Ounnar, F.: Development of a decision-making strategy to improve the effciency of BPR. Expert Systems with Applications **36**(2) (2009) 3248–3262

[286] Reijers, H.A., Poelmans, S.: Re-configuring workflow management systems to facilitate a "smooth flow of work". International Journal of Cooperative Information Systems **16**(2) (2007) 155–175

[287] Fisher, M.: Process improvement by poka-yoke. Work Study **48**(7) (1999) 264–266

[288] European Foundation for Quality Management: An Overview of the EFQM Excellence Model (2012) `http://www.efqm.org/sites/default/files/overview_efqm_2013_v1.pdf`, accessed on 17/01/2015.

[289] Reijers, H.A., Mendling, J., Dijkman, R.M.: Human and automatic modularizations of process models to enhance their comprehension. Information Systems **35**(5) (2011) 881–897

[290] Jedlitschka, A., Ciolkowski, M., Pfahl, D.: Reporting Experiments in Software Engineering. In: Advanced Topics in Empirical Software Engineering. Springer (2007) 201–228

[291] Gregor, S.: The nature of theory in information systems. MIS Quarterly **30**(3) (2006) 611–642

[292] Nuseibeh, B., Easterbrook, S.: Requirements engineering: A roadmap. In: Proc. Conf. on The Future of Software Engineering (ICSE'00). (2000) 35–46

[293] Jenkins, A.M.: Research Methodologies and MIS Research. In: Research Methodologies and MIS Research. Elsevier (1985) 103–117

[294] Phillips, J.R.: Enhancing the effectiveness of organizational change management. Human Resources Management **22**(1-2) (1983) 183–199

[295] United Nations Economic Council for Europe: UN/EDIFACT directory, version D.11A (2011) `http://www.unece.org/trade/untdid/welcome.html`, accessed on 17/01/2015.

[296] Streit, A., Pham, B., Brown, R.: Visualization support for managing large business process specifications. In: Proc. Business Process Management Workshops (BPM Workshops '05). Volume 3649 of LNCS, Springer (2005) 205–219

[297] Kolb, J., Kammerer, K., Reichert, M.: Updatable process views for user-centered adaption of large process models. In: Proc. 10th Int'l Conf. on Service Oriented Computing (ICSOC'12). Volume 7636 of LNCS, Springer (2012) 484–498

[298] Preim, B., Dachselt, R.: Interaktive Systeme: Grundlagen, Graphical User Interfaces, Informationsvisualisierung. Springer (2010) in German.

[299] Glaser, B.G., Strauss, A.L.: The Discovery of Grounded Theory: Strategies for Qualitative Research. 8th edn. Aldine Publishing (1999)

[300] Buchwald, S., Tiedeken, J., Bauer, T., Reichert, M.: Anforderungen an ein Metamodell für SOA-Repositories. In: Proc. 2nd Central-European Workshop on Services and their Composition (ZEUS'10). (2010) 25–26 in German.

[301] van der Aalst, W.M.P., Dumas, M., Gottschalk, F., ter Hofstede, A.H.M., La Rosa, M., Mendling, J.: Correctness-Preserving Configuration of Business Process Models. In: Fundamental Approaches to Software Engineering. Volume 4961 of LNCS Springer (2008) 46–61

[302] Gosling, J., Joy, B., Steele, G., Bracha, G., Buckley, A.: The Java language specification, Java SE 7 Edition (2013) `http://docs.oracle.com/javase/specs/jls/se7/jls7.pdf`, accessed on 17/01/2015.

[303] van der Aalst, W.M.P.: Verification of workflow nets. In: Proc. 18th Int'l Conf. on Application and Theory of Petri Nets (ICATPN'97). Volume 1248 of LNCS, Springer (1997) 407–426

[304] RMG Enterprises: Document imaging report (2009) `http://www.documentimagingreport.com`, accessed on 17/01/2015.

[305] Ericson, C.A.: Fault tree analysis – a history. In: Proc. 17th Int'l System Safety Conf., System Safety Society (1999)

# Appendices

# A Modeling Business Objectives: Large Process Example

This appendix demonstrates the application of the business objective modeling approach developed in Chapter 6 to a more complex application scenario to be addressed by larger BP models. As an example in this regard, consider the information technology support process commonly found in enterprises today [118]. Figure A.1 demonstrates how the underlying business objective of this process can be captured, e.g., to compare possible implementations between organizations, or to enable quality management of the process.

The objective model captures characteristics that must be fulfilled by all efficacious implementations of corresponding processes, but leaves open design decisions that should be taken by the process modeler. As examples, consider the following topics:

- Before a ticket can be issued, the user's identity must be confirmed. On the other hand, an issued ticket is a prerequisite for all other target aspects (dealing with resolving an issue) except for a security breach, a simple software update, or an update of the support decision tree (the support decision tree documents issues and the respective required steps).
- Closing a ticket requires fixing the issue remotely, installing new software etc., but also a corresponding confirmation from the user.
- It is possible to deal with multiple issues within one ticket. Otherwise, the corresponding target aspects could have been modeled as mutually exclusive.
- If the support decision tree has been fully processed but an issue still cannot be resolved, an update process for the decision tree must be triggered. Requiring the ticket to be closed however, would mean that user confirmation is needed as well. The way the target aspect is modeled therefore implies that whether the update process needs to be triggered is decided solely by the IT department.
- There are threshold values with regard to the approval of hardware orders. In contrast, for software orders, the license management process must be triggered.
- The objective model does *not* specify whether the process should be triggered through a hotline or by visiting with local IT support.

In general, larger and more complex business processes will require larger business objective models. However, in most cases, this does not mean that interrelations between target aspects and BSDs will become more complex as well. Mostly, the number of respective associations will grow linearly with the number of target aspects considered. For example, larger processes can be created by merging subsequent sub-processes into one process model. However, this does not mean that the number of BSDs to be considered per target aspect will grow. Therefore, there is no impact on the complexity of business objective modeling.

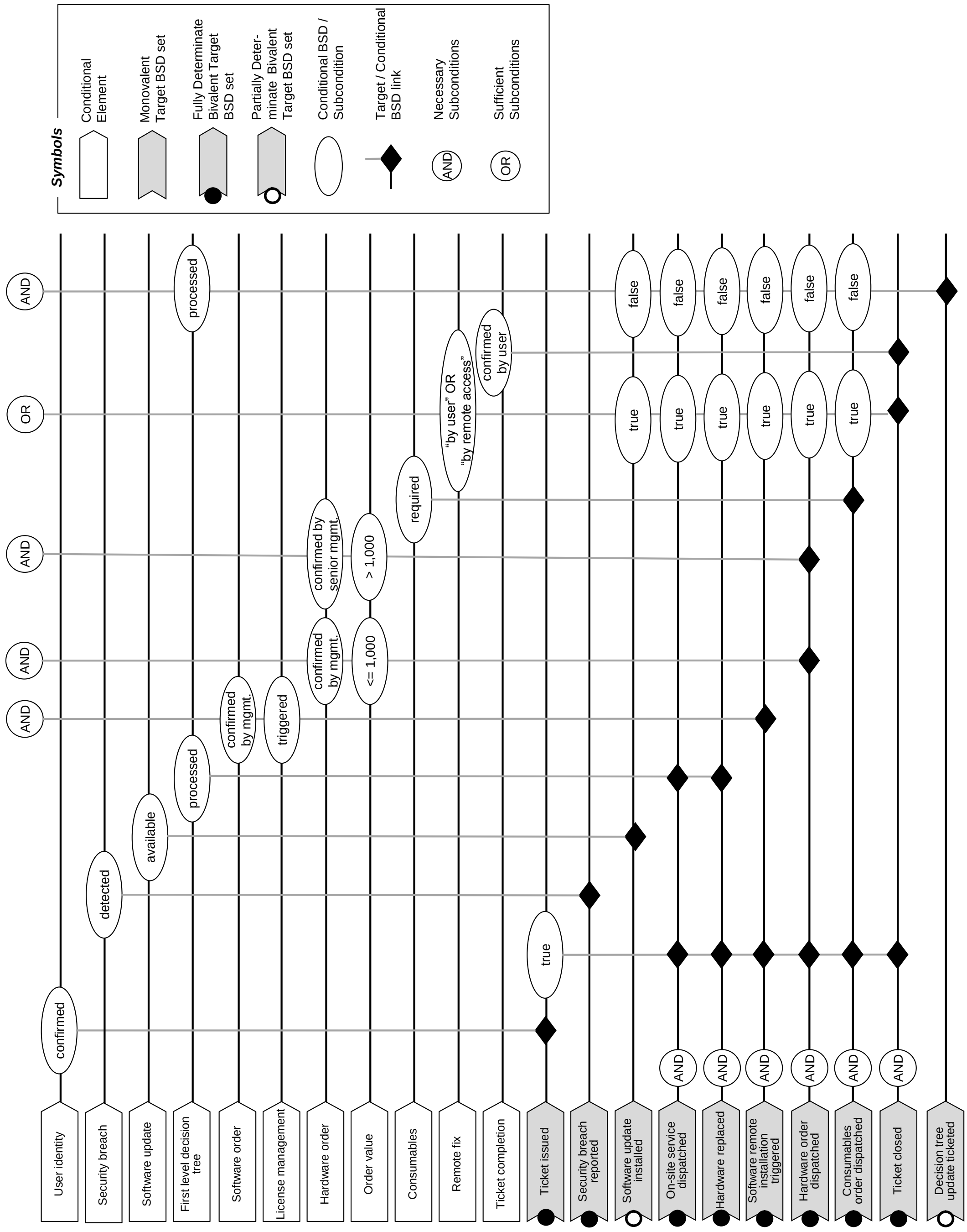



Figure A.1: IT Support Process

# B Creating Virtual Control Flow Elements from Block-structured Process Models

This appendix lists algorithms to deduct sets of VCFEs (cf. Section 7.5) for relevant workflow patterns [223] in block-structured BP models (cf. Definition 23). It more closely addresses considerations described in Section 7.5.3, reflects the results of a prototypical implementation, and provides examples for subsequent implementation projects aimed at the concepts developed in this thesis.

Note that the algorithms comprised in this Appendix are presented using the Java programming language, which was also employed to implement the corresponding prototype application [302].

The set of VCFEs for task blocks can be obtained through Algorithm B.1.

```
protected ControlFlowElementsSet
    createVirtualControlFlowElements() {

  ControlFlowElementsSet virtualControlFlowElements =
      new ControlFlowElementsSet();

  // Add an appendable CFE to represent the task block's
  // role in overall control flow
  Task t = new Task(this.getConditionalBSDs(),
      this.getStateOperations());
  t.setAppendable(true);
  virtualControlFlowElements.add(t);

  // Add a second, non-appendable CFE if the task block
  // addresses a target BSD to represent relevant PEPs
  // terminating with this task block
  if(this.addressesTargetAspect()) {
    t = new Task(this.getConditionalBSDs(),
        this.getStateOperations());
    t.setAppendable(false);
    virtualControlFlowElements.add(t);
  }

  return virtualControlFlowElements;

}
```

Algorithm B.1: *createVirtualControlFlowElements* for Task Blocks

The set of VCFEs for sequence blocks can be obtained through Algorithm B.2.

```
protected ControlFlowElementsSet
    createVirtualControlFlowElements() {

  // This set will be filled with relevant VCFEs
  ControlFlowElementsSet virtualControlFlowElements =
      new ControlFlowElementsSet();


  // The Iterator interface allows traversing a set
  // element by element
```

```
  Iterator<SubBlock> it = getSubBlocks().iterator();

  // Traverse the set of SubBlocks
  while(it.hasNext()) {

    // ``Recursively'' obtain the set of VCFEs for the
    // SubBlock
    ControlFlowElementsSet subControlFlowElements =
        it.next().getVirtualControlFlowElements();

    // The set of VCFEs of first SubBlock constitutes the
    // initial fill of the set of VCFEs of the SequenceBlock
    if(virtualControlFlowElements.isEmpty()) {
      virtualControlFlowElements = subControlFlowElements;
    } else {

      // Prepare a cloned VCFEs set for the SequenceBlock:
      // new elements can be appended to this set while
      // maintaining the ``old'' set as a repository
      // of paths to the current SubBlock
      ControlFlowElementsSet newSet =
          virtualControlFlowElements.clone(false);

      // For additional SubBlocks, the set of VCFEs of the
      // Sequence Block is repaced by a new set comprising
      // the set of existing non-appendable VCFEs, and one
      // VCFE for each each combination of an existing
      // appendable VCFE and a non-appendable SubBlock VCFE
      // -- note that the appendable characteristic is
      // determined by the last ``actual'' control flow
      // element appended to a VCFE

      ControlFlowElementsSet appendableSet =
          virtualControlFlowElements.getSubSet(true);

      // Iterator through the VCFEs of the SubBlock
      Iterator<ControlFlowElement> subIt =
          subControlFlowElements.iterator();
      while(subIt.hasNext()) {
        ControlFlowElement actElement = subIt.next();
        // Iterator through the set of current appendable
        // VCFEs of the SequenceBlock
        Iterator<ControlFlowElement> appendableIt =
            appendableSet.iterator();
        while(appendableIt.hasNext()) {
          ControlFlowElement appendable =
              appendableIt.next();
          // The existing appendable VCFE needs to be
          // cloned, since multiple VCFEs of the
          // SubBlock are to be appended in parallel
          ControlFlowElement newElement =
              appendable.clone();
          newElement.append(actElement);
          newSet.add(newElement);

        }
      }

      // The ``old'' VCFEs set is replaced by the set of
      // new VCFEs created on the basis of cloned elements
      // of the old set
      virtualControlFlowElements = newSet;

    }
  }

  return virtualControlFlowElements;

}
```

Algorithm B.2: *createVirtualControlFlowElements* for SequenceBlocks

The set of VCFEs for choice blocks can be obtained through Algorithm B.3.

```

protected ControlFlowElementsSet
    createVirtualControlFlowElements() {

  ControlFlowElementsSet virtualControlFlowElements =
      new ControlFlowElementsSet();

  // Iterate over the branches in the ChoiceBlock
  Iterator<Branch> branchIt = getBranches().iterator();
  while(branchIt.hasNext()) {

    Branch actBranch = branchIt.next();

    // If there is a SubBlock for the Branch, clone the
    // Branch for every VCFE of the SubBlock, and add this
    // to the set of VCFEs of the ChoiceBlock
    // Else, just add the Branch
    actSubBlock = this.getSubBlock(actBranch);
    if(actSubBlock != null) {
      Iterator<ControlFlowElement> vcfeIt = actSubBlock.
          getVirtualControlFlowElements().iterator();
      while(vcfeIt.hasNext()) {
        Branch clonedBranch = actBranch.clone();
        clonedBranch.append(vcfeIt.next());
        virtualControlFlowElements.add(clonedBranch);
      }
    } else {
      virtualControlFlowElements.add(actBranch);
    }

  }

  return virtualControlFlowElements;

}
```

Algorithm B.3: *createVirtualControlFlowElements* for ChoiceBlocks

The set of VCFEs for parallel blocks can be obtained through Algorithm B.4.

```
protected ControlFlowElementsSet
    createVirtualControlFlowElements() {

  // This set will be filled with relevant VCFEs
  ControlFlowElementsSet virtualControlFlowElements =
      new ControlFlowElementsSet();

  // This set of control flow element sets will hold
  // the sets of non-appendable VCFEs of the SubBlocks
  Set<ControlFlowElementsSet> tempSet =
      new HashSet<ControlFlowElementsSet>();

  // Iterator through all SubBlocks, used to fill tempSet
  // with a set of non-appendable VCFEs for each SubBlock
  // if there are such
  Iterator<SubBlock> subBlocks = getSubBlocks().iterator();
  while(subBlocks.hasNext()) {
    ControlFlowElementsSet actNonAppendables = subBlocks.next().
        getVirtualControlFlowElements().getSubSet(false);
    if(actNonAppendables.size() > 0) {
      tempSet.add(actNonAppendables);
    }
  }

  // This creates a set of relevant non-appendable VCFEs,
  // i.e. possible combinations of non-appendable VCFEs
  // of the SubBlocks -- each combination may comprise
  // zero or one VCFE from each SubBlock set
  ControlFlowElementsSet nonAppendableVCFEs =
      createNonAppendableCombinations(tempSet, null);
  if(nonAppendableVCFEs != null) {
```

```
      virtualControlFlowElements.addAll(nonAppendableVCFEs);
    }


    // The appendable VCFEs for the parallel block are created
    // via a virtual sequence block -- mutual conditional
    // independence ensures that the ordering of SubBlocks
    // is of no consequence
    SequenceBlock tempBlock = new SequenceBlock("temp");
    subBlocks = getSubBlocks().iterator();
    while(subBlocks.hasNext()) {
      tempBlock.addSubBlock(subBlocks.next());
    }
    Iterator<ControlFlowElement> tempElements =
        tempBlock.getVirtualControlFlowElements().iterator();
    while(tempElements.hasNext()) {
      ControlFlowElement tempElement = tempElements.next();
      if(tempElement.isAppendable()) {
        virtualControlFlowElements.add(tempElement);
      }
    }

    return virtualControlFlowElements;

}


// This function recursively creates the relevant set of
// non-appendable VCFEs for a parallel block, i.e., it
// transforms a set of sets of VCFEs into a ``flat'' set
private ControlFlowElementsSet createNonAppendableCombinations
      (Set<ControlFlowElementsSet> controlFlowElements,
      ControlFlowElementsSet nonAppendableVCFEs) {

    // If the set of non-appendable VCFEs for the parallel
    // block has not been created yet, this is the first
    // iteration
    if(nonAppendableVCFEs == null) {
      nonAppendableVCFEs = new ControlFlowElementsSet();
    }

    // If the set of ControlFlowElementsSets has been
    // reduced to zero elements, abort the recursion
    if(controlFlowElements.size() > 0) {

      // Take the first element in the set of sets as the
      // ``actual set'', ordering does not matter
      ControlFlowElementsSet actSet =
          controlFlowElements.iterator().next();

      // Are there any elements yet?
      if (nonAppendableVCFEs.size() > 0) {

        // For each element in the current set, the current
        // state of the set of VCFEs of the parallel block is
        // cloned, and the element is appended to each element
        // of the cloned set, reflecting the alternative
        // character of elements in the current set
        Iterator<ControlFlowElement> actSetIt =
            actSet.iterator();

        // This is an intermediate store for additions
        // to the set of VCFEs of the parallel block,
        // required since this one needs to be cloned
        // anew for each element of the current set
        ControlFlowElementsSet newElements =
            new ControlFlowElementsSet();

        while(actSetIt.hasNext()) {

          ControlFlowElement actElement = actSetIt.next();
          ControlFlowElementsSet tempSet =
              nonAppendableVCFEs.clone();

          Iterator<ControlFlowElement> tempIt =
              tempSet.iterator();
          while(tempIt.hasNext()) {
            ControlFlowElement elem = tempIt.next();
```

```
            // Caution: temporarily, set the non-appendable
            // VCFEs to an appendable state!
            elem.setAppendable(true);
            elem.append(actElement);
            elem.setAppendable(false);
          }

          //
          newElements.addAll(tempSet);

        }

        // Add the new combinations to the existing set of
        // VCFEs of the parallel block
        nonAppendableVCFEs.addAll(newElements);

      }

      // Add the individual elements of the current set to the
      // existing set of VCFEs of the parallel block as well
      Iterator<ControlFlowElement> actSetIt = actSet.iterator();
      while(actSetIt.hasNext()) {
        nonAppendableVCFEs.add(actSetIt.next());
      }

      // Remove the current set from the set of
      // ControlFlowElementsSet sets, and start a new
      // iteration with the remaining set of sets
      controlFlowElements.remove(actSet);
      nonAppendableVCFEs = createNonAppendableCombinations
          (controlFlowElements, nonAppendableVCFEs);

    }

    return nonAppendableVCFEs;

}
```

Algorithm B.4: *createVirtualControlFlowElements* for ParallelBlocks

# C Quality Attributes

In the following sections, additional details are provided for the quality attributes discussed in Sections 8.3 and 8.4. For each quality attribute, the respective content, assessment methods, quality criteria, and quality predicates are discussed, and an example is provided.

## C.1 Quality Drivers

As described in Section 8.2, quality drivers constitute inductive quality attributes. Quality drivers can be assessed based on the actual BP model, i.e., empirical data on process instances is not required. To identify a reasonably complete set of quality drivers, *guiding questions* reflect the various types of quality drivers identified. Each quality driver belongs to one quality driver type, and is therefore associated with one guiding question.

### C.1.1 Task Level Quality Drivers

This section provides additional details on quality drivers which can be assessed by considering the set of tasks comprised in the business process without taking into account control flow.

**Guiding Question 1.** *On the level of individual tasks, which characteristics are relevant regardless of the application domain to achieve formal efficacy?*

**Quality Attribute 1 (Sufficiency of State Operations).** The set of state operations associated with the set of tasks should comprehensively address the target BSDs comprised in the business objective associated with the process (cf. Definition 4).
**Assessment:** Target BSDs are matched against the affected elements of state operations (cf. Step 1 in Section 7.4). Each target BSD should comprise at least one affecting element that is the affected element of at least one state operation. Based on a formalized business objective, the procedure can be automated.

| Quality Criteria | Quality Predicates |
|---|---|
| Each target BSD is addressed by at least one state operation. | Tasks are formally sufficient towards the business objective. |
| The business objective comprises target BSDs that are not addressed by state operations within the business process. | The business process is not formally efficacious, additional state operations should be included. |

**Example 59 (Sufficiency of State Operations).** Consider Sample Process C from Figure 2.7. The results of Examination D must be available to achieve the business objective. Therefore, if there is no task where Examination D is conducted and the corresponding results are recorded as a state operation, the set of state operations in the business process are not sufficient towards the business objective.

Example 60 demonstrates how *QA 1: Sufficiency of state operations* can be inverted, as described in Section 8.2.4.

**Example 60 (Inverted Positive Quality Drivers).** *QA 1: Sufficiency of state operations* as described above can also be formulated negatively:

**Quality Attribute (Sufficiency of State Operations).** There may be no target BSDs comprised in the business objective associated with the process (cf. Definition 4) which is not addressed by the set of state operations associated with the set of tasks.
**Assessment:** Target BSDs are matched against the affected elements of state operations. There may be no target BSDs which do not comprise at least one affecting element that is the affected element of at least one state operation. Based on a formalized business objective, the procedure can be automated.

| Quality Criteria | Quality Predicates |
|---|---|
| The business objective comprises target BSDs that are not addressed by state operations within the business process. | The business process is not formally efficacious. |
| Each target BSD is addressed by at least one state operation. | Tasks are formally sufficient towards the business objective. |

**Guiding Question 2.** *On the level of individual tasks, which characteristics are relevant regardless of the application domain to limit resource requirements?*

**Quality Attribute 2 (Effective Tasks).** There should be no tasks without at least one state operation required to further pursue control flow or to fulfill a target BSD.
**Assessment:** State operations of each task are matched against target BSDs as well as task-requisite and branch-conditional BSDs of other control flow elements. The affected element of at least one state operation of each task should be comprised in the set of affecting elements of a target BSD or a task-requisite or branch-conditional BSD of another control flow element. Otherwise, the task will not contribute to achieving the business objective. Since it may be assumed that each task consumes resources through its implementation or its enactment, this will impede process quality. Note that this quality driver does not consider the ordering of control flow elements. Therefore, it can be assessed on the task level without deriving relevant control flow paths. Since it is a formal quality attribute, assessment may be fully automated.

| Quality Criteria | Quality Predicates |
|---|---|
| There are no state operations where the affected element is not an affecting element of a target BSD. | All tasks are ideally effective. |
| There are no tasks that do not comprise at least one state operation where the affected element is an affecting element of a target BSD. | All tasks are directly effective. |
| There are no tasks that do not comprise at least one state operation where the affected element is an affecting element of another control flow element or a target BSD. | All tasks are indirectly effective. |
| There are tasks without at least one state operation where the affected element is an affecting element of a target BSD or another control flow element. | The process comprises in-effective tasks which should be eliminated. |

**Example 61 (Effective Tasks).** Consider Sample Process B from Figure 2.6, the management of outgoing payments. All tasks comprised in the process model either directly address target BSDs (B5, B6, B7, B8, B9) or are required to further conduct control flow (B1, B2, B3, B4). Therefore, all tasks are indirectly effective. If an additional task was introduced to double-check, e.g., bank account balances although this is not a requirement of the business objective, the process would comprise an in-effective task which should be eliminated.

**Guiding Question 3.** *On the level of individual tasks, which characteristics are relevant considering the application domain to limit resource requirements?*

**Quality Attribute 3 (Effective State Operations).** *QA 2: Effective tasks* demands that each task comprises at least one state operation required in the course of control flow or affecting a target BSD. Moreover, it is also desirable that each individual state operation not modeling resource consumption instead of just at least one per task fulfills this characteristic. However, since it may be assumed that each task consumes resources while there is no resource consumption assigned to individual state operations, subject matter experts' appraisal is required to determine if redundant state operations raise resource requirements, and, accordingly, constitute a quality issue or just a technical matter. Therefore, this topic is considered as a relative quality driver.
**Assessment:** For each state operation, it is determined whether the state operation models the consumption of resources (cf. Definition 21), and whether its affected element is comprised in the set of affecting elements of a target BSD or of a task-requisite or branch-conditional BSD of another control flow element. This analysis can be automated. For the state operations which do not fulfill one of these criteria, subject matter experts' appraisal is required to determine whether they drive resource requirements. This appraisal also needs to consider whether the state operation in question arises from a justified technical necessity.

| Quality Criteria | Quality Predicates |
|---|---|
| There are no state operations where the affected element is not an affecting element of a target BSD. Note that this criterion is also assessed in the course of *QA 2: Effective tasks*. Accordingly, if all tasks are ideally effective, all state operations are ideally effective, and vice versa. | All state operations are ideally effective. |
| There are no state operations where the affected element is not an affecting element of a target, task-requisite, or branch-conditional BSD or models resource consumption. | All state operations are effective. |
| There are state operations where the affected element is not an affecting element of a target, task-requisite, or branch-conditional BSD or models resource consumption. However, as per subject matter experts' appraisal, the state operations in question do not drive resource consumption considering technical necessity. | There are no resource requirements resulting from in-effective state operations. |
| There are state operations where the affected element is not an affecting element of a target, task-requisite, or branch-conditional BSD or models resource consumption. In addition, as per subject matter experts' appraisal, the state operations in question drive resource consumption considering technical necessity. | The process comprises in-effective state operations which should be eliminated. |

**Example 62 (Effective State Operations).** Consider Sample Process C from Figure 2.7. It is conceivable that entering examination results into the IT system where patient data is managed requires changing the state of the master data record. This state change would be modeled through a state operation which affects neither target BSDs nor other control flow elements. However, it is still justifiable as a technical necessity, and would not pose additional resource requirements. Therefore, the process would still not comprise in-effective state operations to be eliminated.

**Quality Attribute 4 (Reasonable Task Resource Requirements).** There should be no tasks where the associated resource requirements are appraised as unreasonable in comparison to the desired outcome. Note that this quality driver does not address the "content" of tasks *per se*, but the question whether task-requisite BSDs are reasonable considering the task as it is. Moreover, this quality driver does not consider whether tasks are required to fulfill the business objective. Instead, resource requirements are matched against the "desired outcome" of a task. In some cases, this may constitute a deviation from the principle of not addressing business objectives as a determinant of process quality which is observed in this thesis. However, for reasons of practical applicability, there may be situations where, on the level of individual tasks, this may be justified. In these cases, the business objective may be adapted to enable elimination of the tasks in question.
**Assessment:** Resource requirements as defined by the set of task-requisite BSDs are matched against state operations *not* modeling resource consumption (cf. Table 7.2) for each task (these state operations depict the "desired outcome" of the task). For each task,

subject matter expert appraisal is required to judge whether resource requirements can be considered as reasonable on that basis. To this end, domain-specific good practices or qualitative benchmarking [45] can be employed.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| As per subject matter experts' appraisal, there are no tasks comprising unreasonable resource requirements. | Tasks comprise generally reasonable resource requirements. |
| As per subject matter experts' appraisal, tasks comprising unreasonable resource requirements are sufficiently limited not to compromise the overall resource requirements of the business process (e.g., corresponding tasks only occur as an exception in non-standard control flow paths). | Tasks mostly comprise reasonable resource requirements. |
| As per subject matter experts' appraisal, there are material tasks where resource requirements are considered as unreasonable. | The process comprises tasks with unreasonable resource requirements which should be subject to re-consideration. |

**Example 63 (Reasonable Task Resource Requirements).** Consider Sample Process A from Figure 2.5. The process model comprises a task of obtaining senior management approval for invoices above a value threshold. In many cases, obtaining such approval will require significant manual effort. Therefore, this task and the corresponding target BSD as part of the business objective may be subject to discussion if, for example, the purchase order has already been subject to senior management approval. As an exception to the general rule, discussion in this case would also consider the applicable alterations to the business objective.

**Quality Attribute 5 (Task Automation / Use of Capital Investments).** Task automation potentials should be utilized reasonably. Task automation may pertain to entire tasks, or to parts of tasks, such as the automated provision of information, or the use of machinery for manual operations. In general, it corresponds to the replacement or reduction of manual human effort by capital investments. As opposed to *QA 4: Reasonable task resource requirements*, this driver *does* address the "content" of tasks. Domain-specific appraisal must also take into account capital expenditures (and operating expenses) associated with the use of capital investments.
**Assessment:** For each individual task, subject matter expert appraisal regarding the use of capital investment potentials is executed taking into account processing volumes and capital expenditures required for individual measures, e.g., according to organization-specific guidelines for business case assessment (*efficiency assessment*). To this end, domain-specific good practices or qualitative benchmarking [45] can be employed.

| Quality Criteria | Quality Predicates |
|---|---|
| As per subject matter experts' appraisal, there are no automation or capital investment potentials on the level of individual tasks that are not utilized. | All tasks are automated to the possible extent. |
| As per subject matter experts' appraisal, there are no automation or capital investment potentials on the level of individual tasks that are not utilized although they are considered as efficient. | All tasks are automated to the efficient extent. |
| As per subject matter experts' appraisal, there are automation or capital investment potentials on the level of individual tasks that are not utilized, but considered as efficient. | There are tasks not automated to the efficient extent, capital investments should be considered. |

**Example 64 (Task Automation / Use of Capital Investments).** Consider Sample Process B from Figure 2.6. At the end of the process, open items are generated which represent the payment advices that may be sent to suppliers (B9). In a subsequent process, these are typically matched against bank statements. If an outgoing payment has been confirmed by the bank statement, the open item is "cleared". For the task of matching open items and bank statements, effective automated software solutions are available today. If these should be implemented as a capital investment in a concrete application case, however, needs to be determined based on transactional volume, processing cost per transaction, the total invest required, available means, the prioritization within the investment projects portfolio etc. To this end, a sound understanding of the application domain is required.

**Quality Attribute 6 (Task Classification).** This quality driver mainly pertains to human labor involved in tasks, but may also be relevant for other types of resources such as tools and machinery. Generally, tasks are designed to be executable if all task-requisite BSDs are fulfilled. This cause the set of task-requisite BSDs associated with a task to be more exhaustive than required for all cases. This may pertain to both the quantity and the quality of resources. Accordingly, resource requirements for some traces may be reduced by splitting the trace, which represents a class of cases, into multiple possible traces by using split gateways with appropriate conditional BSDs. Out of the new set of cases, all but one should be associated with reduced resource requirements. For implementation, the reduced resource requirements should be judged to outweigh the possibly incurred additional effort to properly route the corresponding process instances. Note that the additional effort will decrease with the degree of WfMS application.
**Assessment:** For each individual task, state operations which do not define resource consumption (i.e., the intended output of the task) are appraised by subject matter experts. The appraisal examines whether it is possible to classify the output of the task into sub-classes with partially reduced resource requirements. In many cases, classification will depend on the addressed target elements (cf. the example below). It is also possible that task classification results in a task not being required at all for certain cases. Note that it may be necessary to reconcile the proposed task classification to the business objective.

| Quality Criteria | Quality Predicates |
|---|---|
| As per subject matter experts' appraisal, there are no tasks that could be classified. | There are no additional task-classification potentials. |
| As per subject matter experts' appraisal, there are tasks that could be classified, but the additional routing effort outweighs the possible reduction in resource requirements. | There are only ineffective additional task-classification potentials. |
| As per subject matter experts' appraisal, there are tasks that could be classified, and the possible reduction in resource requirements outweighs the additional routing effort. | There are additional effective task classification potentials which should be considered. |

**Example 65 (Task Classification).** Consider Sample Process A (cf. Figure 2.5). The process comprises senior management approval for invoices with a value of more than 5,000. Senior management approval could be further classified if middle management approval would suffice for, e.g. invoices with a value of more than 5,000, but less than 15,000. Note that, in this case, the intended alteration to the process model would have to be reconciled with the business objective. With respect to the business objective, it would also have been possible to design the process with senior management approval for *every* invoice. In that case, task classification would result in eliminating the task for a class of target elements: invoices below the threshold value.

### C.1.2 Control Flow Level Quality Drivers

This section provides additional details on quality drivers which can be addressed by considering control flow spanning multiple tasks or gateways. In general, these quality drivers can be addressed by re-arranging control flow.

**Guiding Question 4.** *On the level of control flow, which characteristics are relevant regardless of the application domain to achieve formal efficacy?*

**Quality Attribute 7 (Consideration of Conditional Propositions).** According to Definition 8, formally efficacious business processes must reflect conditional BSDs given by the business objective in the control flow model. This means that tasks with associated state operations inducing target BSDs must be bound to task-requisite or branch-conditional BSDs in the process model corresponding to the conditional BSDs in the objective model considering the target BSD type. Namely, this consideration is relevant for fully determinate bivalent target BSDs. Conditional propositions required with respect to the fulfillment of target BSDs can be interpreted as compliance restrictions [155].
**Assessment:** Formal assessment of this quality attribute has been described in Section 7.4. In short, a quality-aware BP model enables consolidating possible enactment paths which induce a target BSD to a single virtual control flow element. The set of requisite BSDs for the control flow element can then be compared to the conditional proposition of the target

BSD as defined in the business objective.

| Quality Criteria | Quality Predicates |
|---|---|
| According to the process model, each target BSD will be fulfilled if and only if the corresponding conditional proposition is fulfilled, without any other prerequisites. | The business process is ideally efficacious. |
| According to the process model, each target BSD will be fulfilled only if the corresponding conditional proposition is fulfilled. However, additional prerequisites are possible. | The business process is formally efficacious. |
| According to the process model, there are target BSDs that may be fulfilled although the corresponding conditional proposition given by the business objective is not fulfilled. | The process is not efficacious due to missing conditional restrictions which should be addressed through additional task-requisite or branch-conditional BSDs. |

**Example 66 (Consideration of Conditional Propositions).** Consider Sample Process C from Figure 2.7. According to the business objective, Drug III may only be applied based on the results of previous examinations. If this condition is not met by the corresponding business process, formal efficacy is not given since compliance restrictions are not reflected. Note that, as discussed in Chapter 6, ideally efficacious processes are not possible in practice since every task poses task-requisite BSDs. In our sample process, these might be exemplified by the required availability of medical personnel, laboratory equipment etc.

**Quality Attribute 8 (Completeness of Control Flow).** In an efficacious business process, possible enactment paths through the process model in terms of control flow must be complete in the sense of addressing all relevant target BSDs, providing that resource requirements are met. While *QA 7: Consideration of conditional propositions* ensures that fully determinate bivalent target BSDs are not fulfilled if the respective environmental conditions as given by the business objective are not met, *QA 8: Completeness of control flow* addresses the issue that target BSDs must be achievable in principle. Note that, on the control flow level, this quality driver complements *QA 1: Sufficiency of state operations* on the task level: the latter ensures that the process model comprises the required set of tasks, and the former ensures that these tasks are part of possible enactment paths. In this respect, it may be assumed that all tasks comprised in a process model are reachable (cf. the concept of soundness [303, 22]). However, this does not necessarily mean that all tasks are reachable in a single enactment path. Accordingly, the process model must ensure that target BSDs which are not mutually exclusive on the basis of the respective conditional propositions as defined in the business objective model can be addressed in at least one common enactment path.
**Assessment:** As described in Chapter 6, the overall set of target BSDs can be divided into sub-sets which are *not* mutually exclusive by considering semantic interdependencies between sub-conditions (cf. Table 6.9). For each sub-set, the process model must provide for at least one possible enactment path addressing all elements of the set. Thus, for each possible enactment path, the set of target BSDs addressed is determined by matching affected elements of state operations against affecting elements of target BSDs. The resulting sets

of addressed target BSDs can then be matched against sub-sets of not mutually exclusive target BSDs as per the business objective.

| Quality Criteria | Quality Predicates |
|---|---|
| For each set of not mutually exclusive target BSDs, there is at least one possible enactment path according to the process model addressing all target BSDs in the set. A target BSD is addressed iff there is at least one state operation where the affected element is an affecting element of the target BSD. | The control flow model is complete with regard to the business objective. |
| There is at least one set of mutually not exclusive target BSDs where the process model provides no possible enactment paths addressing all target BSDs in the set. | The control flow model is incomplete with regard to the business objective. |

**Example 67 (Completeness of Control Flow).** Consider Sample Process C as described in Figure 2.7. The application of Drugs II and III is not mutually exclusive. Therefore, the control flow model provides possible traces where both target BSDs are addressed. On the other hand, it may not occur that the existence *and* the non-existence of Condition X are noted – therefore, there is no trace through the process model including both the respective tasks. This is achieved by using an XOR split gateway.

**Guiding Question 5.** *On the level of control flow, which characteristics are relevant regardless of the application domain to limit resource requirements?*

**Quality Attribute 9 (Effective Target Aspects).** *QA 8: Completeness of control flow* and *QA 7: Consideration of conditional propositions* address the ability of the process model to fulfill all relevant target BSDs without violating conditional propositions, respectively, thus ensuring formal efficacy on the control flow level. In addition to this, *QA 9: Effective target aspects* pertains to unnecessary resource requirements incurred by fulfilling target BSDs which are not required to achieve the business objective for a particular process instance. In other words, process models should be careful to avoid spending resources on target aspects irrelevant for the case at hand. As per Definition 4, this applies to partially determinate bivalent target BSDs and to trivalent target BSDs. Since it may be assumed that fulfilling target BSDs will raise resource requirements, ineffective target BSDs should be avoided in possible enactment paths. Note that, e.g. in comparison to *QA 17: Effective tasks in enactment paths*, it is in this case not assumed that target BSDs can be fulfilled without significantly raising resource requirements, unless the respective tasks are relevant to other aspects of control flow as well – otherwise, subject matter experts' appraisal would be required to weigh resource requirements incurred.
**Assessment:** It is assumed that trivalent target BSDs are modeled as a pair of partially determinate bivalent target BSDs (cf. Section 6.5). For each partially determinate bivalent target BSD, possible enactment paths are determined. For each relevant possible enactment path, it is determined whether the task inducing the target BSD in question is relevant to other elements of control flow as well. This is the case if there is at least one state operation

not modeling resource consumption (cf. Definition 21) where the affected element is an affecting element of a task-requisite or branch-conditional BSD of a subsequent task. The possible enactment paths where this characteristic applies are not further considered. For all remaining possible enactment paths, conditional consolidation is executed as described in Section 7.4. The resulting set of necessary sub-conditions should be at least as restrictive as the one given by the business objective.

| Quality Criteria | Quality Predicates |
|---|---|
| There are no partially determinate bivalent target BSDs where possible enactment paths do not fully consider the set of necessary sub-conditions modeled in the business objective. | Target aspects fulfilled are effective for every process instance. |
| There are partially determinate bivalent target BSDs where possible enactment paths do not fully consider the set of necessary sub-conditions modeled in the business objective, but the respective tasks are relevant to subsequent control flow elements. | Tasks addressing target aspects are effective for every process instance. |
| There are partially determinate bivalent target BSDs where possible enactment paths do not fully consider the set of necessary sub-conditions modeled in the business objective, and the respective tasks are not relevant to subsequent control flow elements. | Target aspects fulfilled are not effective for every process instance. |

**Example 68 (Effective Target Aspects).** Consider Sample Process C from Figure 2.7. Examination D is not harmful, but it is only required depending on the results of other examinations. Accordingly, it constitutes a partially determinate target BSD. Therefore, conducting Examination D without considering the respective necessary sub-conditions would unnecessarily raise resource consumption.

**Quality Attribute 10 (Effective and Efficacious Conditional Splits).** Control flow in BP models is governed by split and join gateways. While parallel splits are used to allow activities to be enacted concurrently, or at least without assuming a particular sequence, inclusive or exclusive “or” splits, as well as complex derivatives, are used to control which activities are enacted for a particular process instance depending on its conditional environment. In the following, these gateways are referred to as *conditional splits*. Conditional splits generally require appraisal of the conditional environment, which is usually achieved through assessment tasks. Accordingly, conditional splits drive resource requirements, and should be used only where required. This, in turn, implies that the structure of conditional splits in a process model should reflect conditional BSDs in a business objective or the provision of resources internal to a process model. The latter case applies if the process model checks whether resources are available and initiates mitigatory activities as required. Conditional splits which are not required in this sense are to be considered as in-effective. Moreover, conditional splits where not every respective branch can be enacted are to be considered as in-efficacious.
**Assessment:** The following characteristics are assessed:

- For each possible enactment path addressing a target BSD, conditional consolidation is executed, and the domains of the resulting set of conditional BSDs (of the virtual control flow element) are intersected. In the following, the result of this step is referred to as the *process conditional domain* of the target BSD. The domain for which the conditional proposition of the target BSD is fulfilled is determined as follows: first, for each necessary sub-condition, the domains of all constituting (alternative) conditional BSDs are united. Then, the resulting domains of necessary sub-conditions are intersected. The result of this step is referred to as the *objective conditional domain* of the target BSD. The set difference of the objective conditional domain and the process conditional domain should be empty, denoting that the process model considers the conditional proposition of the target BSD. Note that this is an alternative way to determine whether conditional propositions as defined in a business objective are considered (cf. *QA 7: Consideration of conditional propositions*). Then, the domains of all branch-conditional BSDs in the possible enactment path are united with the process conditional domain. For each branch-conditional BSD, the result is again compared to the objective conditional domain. If the set difference is not empty anymore, the respective conditional split is required to achieve formal efficacy. It is thus marked as effective.
- Conditional splits may also be required to address resource requirements in the course of a process model. In this case, the respective branch-conditional BSD will be "cancelled out" by state operations of succeeding tasks. Accordingly, branch-conditional BSDs are compared to succeeding state operations according to possible enactment paths. If at least one affecting element of a branch-conditional BSD is an affected element of a succeeding state operation, the respective domains are compared as follows: if the set difference of the co-domain of the state operation and the domain of the branch-conditional BSD is empty, it can be assumed that the conditional split reflects resource requirements to be fulfilled in the course of the process. It is thus marked as effective.
- Moreover, within possible enactment paths leading to a conditional split, each branch-conditional BSD should be satisfiable considering all preceding branch-conditional and task-requisite BSDs. This can be assessed by executing conditional consolidation for preceding enactment paths (cf. Section 7.4) and comparing the resulting conditional BSD set to the branch-conditional BSDs of the conditional split. If there are shared affecting elements, the domains of the shared affecting elements for which the BSDs to be compared are satisfied are intersected. If the intersection is not empty, both BSDs can be satisfied concurrently. Note that conditional consolidation allows properly considering the provision and consumption of resources in this respect. Conditional splits are marked as efficacious accordingly.

The remainder of conditional splits not marked are considered as in-effective or in-efficacious.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| All conditional splits which are required in terms of formal efficacy or to consider resource requirements. Moreover, all branch-conditional BSDs are satisfiable. | Conditional splits are effective for every process instance. |

| | |
|---|---|
| There are conditional splits which are not required in terms of formal efficacy or to consider resource requirements. | There are in-effective conditional splits. |
| There are conditional splits where not all branch-conditional BSDs are satisfiable. | There are in-efficacious conditional splits. |

**Example 69 (Effective Conditional Splits).** Consider Sample Process A from Figure 2.5. Senior management approval for invoices is required if the invoice value exceeds a threshold value. If this condition would not be present in the corresponding objective model, both the conditional split and the respective task (cf. *QA 2: Effective tasks*) would be considered as in-effective.

**Guiding Question 6.** *On the level of control flow, which characteristics are relevant considering the application domain to limit resource requirements?*

**Quality Attribute 11 (Sequential Tasks Composition).** According to the quality-aware BP meta-model, tasks are assumed to be enacted atomically, i.e., in full or not at all, in the course of a process instance (cf. Section 7.3), depending on whether task-requisite BSDs are fulfilled. This modeling presumption is required to deduct quality relations from process models. Moreover, as a prerequisite to apply WfMS (cf. [14]), tasks are structured in a way to allow assigning each task to one responsible role (e.g., an employee or information system in the case of fully automated tasks), and to accommodate control flow logic (i.e., split and join gateways). Observing this limitation, tasks should be as large as possible to avoid completing a preceding task, with the associated resource consumption, and then aborting the instance prior to a subsequent task. Accordingly, there should be no tasks joined by a sequence gateway where the following criteria apply:

1. The preceding task consumes resources.
2. The succeeding task comprises no task-requisite BSDs where an affecting element is the affected element of a state operation of the preceding task.
3. The preceding and the succeeding task are enacted by the same responsible role.
4. There are no other (domain-specific) reasons why the two tasks should be separated (e.g., required waiting times after a medical examination).

Note that this driver is especially relevant for processes managed via a WfMS which may be used to a priori determine the availability of information system or data resources. It constitutes a relative quality driver since roles are not comprised in the formalized quality-aware process model (cf. Chapter 7), and because there may be other reasons to sequentialize tasks considering domain-specific requirements.
**Assessment:** Sequential pairs of tasks are identified and assessed. Since task-requisite BSDs and state operations are formalized in a quality-aware process model, pairs not fulfilling the second criterion stated above can be eliminated automatically. For the remaining pairings, the appraisal of subject matter experts is required.

| Quality Criteria | Quality Predicates |
|---|---|
| There are no sequential task pairings. | The process is fully de-sequentialized. |
| For each sequential task pairing, the preceding task affects the set of task-requisite BSDs of the succeeding task, or does not consume resources. | De-sequentialization is not required for formal reasons. |
| For each sequential task pairing, the preceding task affects the set of task-requisite BSDs of the succeeding task, or does not consume resources, or there are domain-specific reasons for sequential tasks as per the appraisal of subject matter experts. | De-sequentialization is not advisable for domain-specific reasons. |
| The process comprises sequential tasks where there is no reason for sequential enactment. | The process comprises tasks that should be de-sequentialized. |

**Quality Attribute 12 (Parallel Tasks Composition).** Tasks or activities (in the sense of a set of composed tasks and gateways with a defined beginning and end, cf. [80]) executed in parallel branches bear the risk that one or more branches cannot be executed in full, and thus deadlock the other branches at the parallel join gateway, although resource requirements have been incurred. It is also possible that a parallel branch becomes obsolete due to the results of other branches. Accordingly, parallelization is not desirable from a design quality perspective, but may be required to ensure timely process enactment. In this context, note that timely process enactment is not a matter of process design quality, but of process performance. Example 70 discusses this matter in more detail. In general, parallel activities must be enactable independently of each other, i.e. with the exception of common resources consumed, an activity may not have affected elements which are affecting elements of task-requisite BSDs in a parallel activity. Thus, parallel activities can be sequentialized without compromising semantic requirements.
**Assessment:** Parallel activities are identified and assessed. Identification can be automated through respective tools. Subject matter experts' appraisal is required to assess whether sequentialization would endanger timely process enactment.

| Quality Criteria | Quality Predicates |
|---|---|
| There are no parallel activities. | The process is fully de-parallelized. |
| As per subject matter experts' appraisal, all cases of parallel activities are required to foster timely process enactment. | De-parallelization is not advisable for domain-specific reasons. |
| As per subject matter experts' appraisal, the process contains activities that should be de-parallelized, i.e., designed in sequence instead of in parallel. | The process comprises tasks that should be de-parallelized. |

**Example 70 (Parallel Tasks Composition).** Consider Sample Process B from Figure 2.6, the execution of payment runs. In this case, it would be possible to parallelize the approval of large payments and the check whether the respective items are due for payment already. Thus, cycle times for a process instance might be reduced. However, if management approval is obtained for line items which turn out as not due, resources have been wasted.

In this respect, one also needs to consider that parallelization of activities may shorten cycle times for individual instances, but will not lead to improved resource utilization (e.g., in terms of labor capacities). If processing speed is primarily restricted by the availability of scarce resources, parallelization might even prolong average cycle times due to possibly increased overall resource requirements. In our example, this might be the case if management is available for approval only with limitations, as is often incurred in practice.

**Quality Attribute 13 (Alternative Activities Composition).** In the case of fully determinate target BSDs, conditional split gateways are required to ensure that the target BSD is fulfilled if and only if the respective conditional proposition is fulfilled. In all other cases, conditional splits are used to minimize resource requirements for particular cases. It is, however, conceivable that the activities required to arrive at decisions for particular conditional splits, i.e. the evaluation of conditional elements, cause higher resource requirements than the activities that are possibly not executed. In this respect, it also needs to be considered that evaluation activities are enacted each time the split gateway is enacted, while the potential additional activities are only relevant in particular cases. Accordingly, overall resource requirements, and hence the quality of the business process, may be improved by replacing conditional splits and the associated branches by a single branch comprising all elements of the individual branches. Since assessment in this respect requires comparison of the economic viability of resource requirements as well as judgment regarding the relative probability of each outcome of the conditional split, this aspect constitutes a relative quality attribute.
**Assessment:** Based on a quality-aware process model, conditional split gateways are identified. If a conditional split determines whether or not a fully determinate bivalent target BSD is addressed, it is not considered further. For each remaining conditional split, the set of affecting elements of the respective branch-conditional BSDs is matched against affected elements of preceding tasks according to the set of possible enactment paths. Affecting elements which are also required for other purposes in the process model (i.e., as affecting elements of branch-conditional or task-requisite BSDs according to the set of possible enactment paths) are not considered. This also applies to affecting elements which are also comprised in target BSDs. Each match identifies an "checking" task (cf. *QA 15: Early approval / dis-approval*) required to enact the conditional split. This set of assessment tasks can be identified automatically. With respect to availability and consumption, the set of resource requirements associated with the checking tasks required for a conditional split can now be compared to the resource requirements associated with each branch. Resource requirements occurring in any case do not need to be considered. Note that block-structuring of the process model [25], i.e., guaranteeing a join gateway corresponding to the conditional split, would enable automating the derivation of comparable conditional BSD sets through conditional consolidation (cf. Section 7.4). Via subject matter experts' appraisal, it is determined whether the potentially "saved" resource requirements justify the additional requirements incurred to enable the checking tasks.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| The BP model does not comprise alternative activities which are not required to properly address fully determinate bivalent target BSDs. | Alternative activities composition is not relevant. |
| The BP model comprises alternative activities not required for reasons of fully determinate bivalent target BSDs. However, according to subject matter experts' appraisal, the additional resource requirements incurred for the respective checking tasks are justified by the differential in resource requirements between branches. | Alternative activities composition is not effective. |
| The BP model comprises alternative activities not required for reasons of fully determinate bivalent target BSDs. According to subject matter experts' appraisal, the additional resource requirements incurred for the respective checking tasks are not justified by the differential in resource requirements between branches. | Alternative activities composition should be considered. |

**Example 71 (Alternative Activities Composition).** Consider Sample Process C from Figure 2.7. The application of Drug I and Examination B depends on the result of Examination A. The result of Examination A is also required later in the process. Accordingly, the examination is not considered as an checking task with respect to the immediately subsequent conditional split. However, if the later re-use of the results of Examination A is not taken into account, and if it is assumed that both the application of Drug I and the enactment of Examination B constitute partially determinate bivalent target BSDs (which means that neither are harmful to the patient), the effort incurred for drug application and examination B is to be compared to the effort incurred for examination A. If the latter surpasses the former, both branches should be subject to alternative activities composition (i.e., drug application and Examination B are enacted in every case), and Examination A can be abolished.

**Quality Attribute 14 (Mitigation of Repetitive Loops).** Loops denote individual control flow elements being executed multiple times in the course of a process instance. This might occur for two reasons:

- According to the process model, instances comprises multiple “sub-instances” characterized by disjoint sets of environmental elements which are, however, of the same type (e.g. line items in Sample Process B, cf. Figure 2.6). In BPMN notation, this characteristic is in many cases expressed by the sub-process construct comprised in a process model. This type of loops is used if resources required are blocked by “sub-instances” (e.g., if a clerk can handle only one credit application at a time). Otherwise, there is generally no semantic interconnection between “sub-instances”. For the purpose of quality management, it is assumed that sub-processes are subject to separate quality assessment to reflect the arbitrary aggregation and dis-aggregation of process models (cf. Example 16). Thus, this type of loops is not considered further here.

- Loops may also occur when one set of environmental elements is subject to recurring control flow elements. This may happen if elements are processed through incremental steps until a condition is met, or if successive processing steps are insufficient to promote the instance, and must be repeated. In the following, the latter case is referred to as *repetitive steps*, as opposed to *incremental steps*. As examples, consider the application of veneer layers in incremental steps, and the re-routing of posted documents in *repetitive steps*. These types of loops are relevant to process quality.

Whether a loop is characterized as iterative or as repetitive depends on whether state operations induced are incremental or not. It can be assumed that incremental enactment of state operations adds value, while repetitive enactment will not. Therefore, repetitive enactment is to be avoided where possible.
**Assessment:** Loops in process models can be identified by applying corresponding graph algorithms (e.g., [224]). Incremental state operations are reflective state operations (cf. Definition 11) which do not model resource consumption. Resource consumption can be identified as described in Section 7.4, Step 3. The resulting set of repetitive loops is then appraised by subject matter experts to determine whether loops are required for domain-specific reasons, and whether repetitive enactment of tasks is mitigated sufficiently.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| The process model does not comprise repetitive loops. | The process is free of repetitive loops. |
| As per subject matter experts' appraisal, all repetitive loops comprised in the process model are required for domain-specific reasons, and sufficient mitigation to avoid repetition of tasks is in place. | Repetitive loops are employed reasonably. |
| As per subject matter experts' appraisal, the process model comprises repetitive loops not required or not sufficiently mitigated. | The process comprises repetitive loops that should be mitigated. |

**Example 72 (Mitigation of Repetitive Loops).** Consider Sample Process A from Figure 2.5 which comprises the approval of invoices. In many organizations, depending on the industry, invoices are handled not as a whole, but split into separate line items. In this case, each line item may be considered as a separate "sub-instance" of the process, which is not relevant to process quality.

Regarding approval of high-volume invoices, the process might comprise a hierarchy: the invoice is approved by a manager, and passed on for more senior approval until the respective value threshold is met. Reflecting a four eyes principle, this approach is also common in many organizations. It constitutes an incremental, value-adding loop which does not endanger process quality.

On the other hand, if invoices are passed on to contact partners from a central point of invoice receipt, as is common if a WfMS is employed, a repetitive loop may occur. This is the case if contact partners return the invoice to invoice receipt because they are not responsible, and the invoice is passed on to somebody else then. Clearly, these processing steps do not add value, and it must be judged whether invoice receipt sufficiently ensures to obtain the right contact partner in the first place, which constitutes a mitigative measure.

**Quality Attribute 15 (Early Approval or Dis-approval).** As discussed in Chapter 6, efficacious process designs can be based on a strategy to approve or disapprove target BSDs as early as possible. Approval or dis-approval in this context means that the conditional environment of the process instance has been appraised sufficiently to determine whether a target BSD is to be fulfilled or not in order to achieve the business objective as a whole. The earlier approval or disapproval is obtained, the more effort to appraise the outer environment can be avoided. "Early" in this context refers to the development of resource consumption in the course of the process, but not necessarily to the absolute number of tasks to be enacted, or to total processing time. This aspect needs to be considered in the course of the examination of approval or dis-approval strategies. Moreover, the absolute probability of individual conditional BSDs as aspects of target BSDs' conditional propositions is to be taken into account.
**Assessment:** On the basis of a business objective model as described in Chapter 6, necessary and sufficient sub-conditions can be formally determined. These form the basis for early dis-approval and approval, respectively. Necessary and sufficient sub-conditions are then matched against the respective checking activities in the process model. Checking activities are generally modeled as tasks converting conditional elements appraised to "data elements" recording the results of the appraisal, and split gateways with the respective data elements as branch-conditional elements. Thus, matching can be done by matching affecting elements of the objective model against affected elements of tasks and branch-conditional elements of split gateways. The following considerations apply:

- Only one necessary or sufficient sub-condition is required to disapprove or approve a target BSD, respectively. Thus, the corresponding checking activities should be grouped together; concurrently appraising multiple sub-conditions is only sensible if there are semantic reasons to do so (e.g. technical interdependencies of tasks).
- Necessary sub-conditions can be fulfilled by fulfilling *at least one* out of a set of one or more *alternative* conditional BSDs. Sufficient sub-conditions can be fulfilled by fulfilling *all* out of a set of one or more conditional BSDs. This is to be reflected in the process model.
- Target BSDs should be probable to be approved or dis-approved as early as possible in the sense of resource consumption (see above). This requires arranging checking activities in a sequence reflecting the probability of possible checking activity outcomes (in the sense of the enactment path pursued after the split gateway) as well as resource consumption incurred. Necessary and sufficient sub-conditions likely to disapprove or approve a target BSD, respectively, with relatively low resource consumption should be addressed first.

On that basis, the respective order of checking activities is appraised by subject matter experts.

| Quality Criteria | Quality Predicates |
|---|---|
| The objective model does not comprise necessary or sufficient sub-conditions for target BSDs. | Early approval or dis-approval is not applicable. |
| The objective model comprises necessary or sufficient sub-conditions. In the process model, the corresponding checking activities are grouped together, and sub-conditions are arranged in a way to reflect the principle of early approval or dis-approval as per subject matter experts' appraisal. | The process reflects the principle of early approval and dis-approval. |
| The objective model comprises necessary or sufficient sub-conditions. In the process model, the corresponding checking activities are not grouped together, but might be re-arranged as per subject matter experts' appraisal. | Checking activities should be grouped by sub-conditions addressed. |
| The objective model comprises necessary or sufficient sub-conditions. Checking activities are grouped sensibly as per subject matter experts' appraisal. However, as per subject matter experts' appraisal, the order of grouped checking activities should be re-arranged to reflect relative outcome probabilities and resource consumptions. | Grouped checking activities should be re-arranged to reflect the principle of early approval or dis-approval in terms of outcome probabilities and resource consumption. |

**Example 73 (Early Approval / Dis-approval).** Consider Sample Process A from Figure 2.5. Purchase order and goods receipt are checked in this order to determine whether the invoice can be approved. If one of the two checks fails, the target BSD *"Invoice approved = true"* is considered as disapproved according to the corresponding business objective. Enacting both checks in parallel, in contrast, would violate the principle of early dis-approval, since effort for both checking tasks will have been incurred in all cases, although one failed check would suffice to disapprove the invoice.

To foster early dis-approval, the two checks need to be arranged in a way to reflect each check's probability of disapproving the invoice as well as resource consumption incurred. If, for example, the probability of the goods receipt missing was higher than the probability of the purchase order missing, and, at the same time, the purchase order required manual checking while the goods receipt could be checked automatically, the goods receipt would ideally be checked first to avoid the possibly unnecessary effort of checking the purchase order. If, however, the goods receipt required manual checking and the purchase order could be matched automatically, the relative probability of dis-approval and the relative resource consumption incurred would have to be weighed against each other by subject matter experts. In general, automated checks should always be enacted before manual checks are done.

**Quality Attribute 16 (Early Failure).** Process instances which cannot be completed in the sense of achieving a business objective induce in-efficient resource requirements. Therefore, control flow should be arranged in a way to ensure that high-risk activities, i.e. tasks which are necessary to achieve a business objective, but where task-requisite BSDs have a

high risk of being not fulfilled (i.e, where a high "resource risk" is incurred), are scheduled first. This way, failing instances are probable to fail early on, which minimizes resource consumption induced.
**Assessment:** Sequences of activities can be re-arranged if subsequent tasks or gateways do not depend on the result of the state operations of preceding tasks. In other words, pairs consisting of a preceding task and a succeeding task or gateways cannot be re-arranged if an affected element of a state operation of the preceding task is an affecting element of a task-requisite BSD or a conditional BSD of the succeeding task or gateway, respectively. For each remaining pairing, subject matter experts' appraisal is required to determine whether re-arrangement would be desirable. It is possible to support the identification of high-risk activities with WfMS based on enactment statistics. In this case, a WfMS might even be extended to autonomously re-arrange activities.

| Quality Criteria | Quality Predicates |
|---|---|
| Due to formalized semantic interdependencies, the process model does not comprise pairings that could be re-arranged. | Early failure re-arrangement is not possible. |
| According to formalized process semantics, the process model comprises pairings which could be re-arranged, but, as per subject matter experts' appraisal, re-arrangement would not promote early failure due to the individual resource risks of activities. | Early failure re-arrangement is not advisable. |
| According to the appraisal of subject matter experts, the process model comprises activity pairings that should be re-arranged to promote early failure. | Early failure re-arrangement should be considered. |

**Example 74 (Early Failure).** Consider Sample Process A from Figure 2.5. Purchase order and goods receipt are checked in this order to determine whether the invoice can be approved. Both tasks are semantically not interdependent in the sense of one assuring prerequisites for the other. If, however, the goods receipt system runs a high risk of being not available, and the purchase order check requires the involvement of labor resources, it would be advisable to re-arrange the process to enact the goods receipt check first.

**Quality Attribute 17 (Effective Tasks in Enactment Paths).** *QA 2: Effective tasks* demands that all tasks comprise at least one state operation contributing to enable other tasks or conditional split gateways, or to the achievement of target BSDs. This requirement can be tightened by taking into account control flow as well. This means that there should be no possible enactment paths where the corresponding ordered set of tasks comprises individual tasks that are not necessary to fulfill the target BSDs addressed in the enactment path. At least one of the following criteria should be fulfilled for each task in an enactment path:

- The task comprises a state operation where the affected element is an affecting element of a target BSD, and the image of the state operation for the respective element is comprised in the range of values for which the target BSD can be fulfilled.

- The task comprises a state operation where the affected element is an affecting element of a task-requisite or a conditional BSD of a *subsequent* task or conditional split gateway, and the state operation's image for the respective element is comprised in the range of values for which the task-requisite or conditional BSD can be fulfilled.

Thus, possible enactment paths should always terminate with a task which addresses a target BSD. This applies to decision split gateways as well. Tasks which are not effective for any possible enactment path should be identified in the course of *QA 2: Effective tasks*. Accordingly, this quality attribute does not address the quality of tasks and decision split gateways, but the quality of the control flow model binding them together. Process optimization with respect to *QA 17: Effective tasks in enactment paths* will, in general, aim at an early differentiation of target BSDs to be fulfilled in a concrete process instance through appropriate conditional split gateways. For each relevant combination of target BSDs, this will allow including only tasks required in the concrete case. However, it needs to be considered that full differentiation might cause additional resource requirements as well, since the respective branch-conditional BSDs might have to be addressed through “information gathering” tasks (cf. the concept of “checking activities” discussed with respect to *QA 15: Early approval or dis-approval*). Moreover, it is possible that, due to technical reasons, the additional resource requirements induced by particular formally ineffective tasks would not justify to further differentiate the process model (see the example below). Thus, the effectiveness of tasks in enactment paths is to be subject to subject matter experts' appraisal, and constitutes a relative quality driver.
**Assessment:** Possible enactment paths through the process model in the sense of ordered sets of tasks and decision split gateways are determined. On that basis, the following actions are executed for each enactment path: The sub-set of tasks addressing target BSDs is determined. All tasks and decision split gateways occurring after the last task which addresses a target BSD are marked as potentially ineffective. Starting with the remaining decision split gateways and the tasks addressing a target BSD as initial effective control flow elements, the full set of effective control flow elements is recursively determined as follows. For each effective control flow element, all contributing tasks are determined by matching affecting elements of the task-requisite or branch-conditional BSDs of the effective element against affected elements of state operations of preceding tasks. State operations modeling resource consumption (cf. Definition 21) are not considered. If a match occurs, the preceding task constitutes a contributing task of the control flow element. Contributing tasks are considered as effective control flow elements as well, and recursively assessed with respect to their respective contributing tasks. The occurrence of tasks and decision split gateways which are not effective can thus be determined for each possible enactment path. For each ineffective control flow element, subject matter experts' judgment is required to determine whether the control flow element causes additional resource requirements to be addressed through further differentiation of possible enactment paths.

| Quality Criteria | Quality Predicates |
|---|---|
| Ineffective tasks in possible enactment paths do not occur. | All tasks are effective for each possible enactment path. |
| Ineffective tasks in possible enactment paths do occur. However, additional differentiation of possible enactment paths in this respect would induce unreasonable additional effort according to the appraisal of subject matter experts. | Additional differentiation of possible enactment paths to ensure task effectiveness is not advisable due to the additional resource requirements incurred. |

| | |
|---|---|
| Ineffective tasks in possible enactment paths do occur. However, according to the appraisal of subject matter experts, additional differentiation of possible enactment paths in this respect would not reasonably reduce resource requirements considering technical interdependencies between tasks and the individual resource requirements of tasks in question. | Additional differentiation of possible enactment paths to ensure task effectiveness is not advisable due to lack of resource requirements reduction potentials. |
| Ineffective tasks in possible enactment paths do occur. As per subject matter experts' appraisal, this should be remedied by appropriately differentiating possible enactment paths. | Additional differentiation of possible enactment paths should be considered. |

**Example 75 (Effective Tasks in Enactment Paths).** Consider Sample Process C from Figure 2.7. Task C8 comprises a medical examination which is only relevant if, as modeled in the subsequent task, a particular drug is to be applied. Since the medical examination is non-invasive, it might also be executed if the medical condition is not given, e.g., in conjunction with Examination A. In this case, the task would formally be considered as ineffective with regard to possible enactment paths since it also occurs for paths where the respective drug (in the sense of a target BSD) will not be applied. However, it is possible that Examinations A and D are, for technical reasons, particularly "cheap" if executed in sequence, so there might be reasons for subject matter experts to approve of this process design regardless of whether Examination D is strictly required in each case.

**Quality Attribute 18 (Effective State Operations in Enactment Paths).** Similar to *QA 2: Effective tasks* and *QA 17: Effective tasks in enactment paths*, *QA 3: Effective state operations* can be extended by taking into account possible enactment paths. However, in this context it may be assumed that state operations contributing to neither the fulfillment of target BSDs nor to enabling control flow by providing resources have already been addressed by the quality attributes mentioned. This would not be the case if there were tasks judged as effective for one or more particular state operations, but, at the same time, comprising additional "piggybacking" state operations which are ineffective with respect to particular possible enactment paths *and* driving resource requirements. This appears as a remote possibility which is therefore not further considered. However, state operations can also be considered as ineffective if they are contradictory or redundant. Assessment in this respect requires taking into account control flow, and is therefore considered as a separate quality driver.
**Assessment:** For each possible enactment path, state operations comprised are assessed as follows:

- Reflective state operations are discarded (cf. Definition 11). Note that this includes state operations modeling resource consumption (cf. Definition 21).
- Subsequent state operations addressing the same affected element are determined. Note that state operations addressing the same affected element are not possible within one task (cf. Section 7). Together with the original state operation, these are marked as a potentially conflicting set.

Each potentially conflicting set of state operations is appraised by subject matter experts with respect to the affected element in question. If the affected element is not used as a flag to govern control flow, the set of state operations comprises conflicts. These may be justified by technical reasons, for example if affected elements are initialized by the business process. Otherwise, the set of state operations comprises redundancies or even contradictions.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| There are no potentially conflicting sets of state operations. | All State Operations are effective. |
| There are potentially conflicting sets of state operations, but these are justified by technical reasons according to subject matter experts' appraisal. | In-effective state operations are justified by technical reasons. |
| There are potentially conflicting sets of state operations which are not justified by technical reasons according to subject matter experts' appraisal. | There are in-effective state operations which should be addressed through process design alterations. |

**Example 76 (Effective State Operations in Enactment Paths).** Consider Sample Process B from Figure 2.6. In this case, it would be possible to subject the payment list generated to approval after it has been initialized. In addition, the payment list is approved after it has been filled with entries to be cleared. In this case, both state operations in question would address the same environmental element (the approval flag of the payment list), and would thus be redundant or even conflicting.

**Quality Attribute 19 (Routing Automation).** In the case of alternative split gateways (i.e., OR, XOR or complex split gateways, cf. [80]), process instances are generally routed along a possible enactment path based on the state of branch-conditional BSDs. If branch-conditional BSDs refer to information elements, they can be assessed automatically by a WfMS [14], and routing can be automated accordingly. Otherwise, assessment must be modeled as a resource-consuming task preceding the split gateway, thus increasing overall resource requirements. With regard to join gateways, WfMS functionality may coordinate preceding branches to trigger subsequent processing. Again, if a WfMS is not in place, periodically checking whether a process instance can proceed constitutes a resource consuming task in principle.
**Assessment:** Alternative split gateways and join are identified based on the process model. If alternative splits and joins are not managed through a WfMS, subject matter experts are required to appraise whether WfMS implementation is judged as reasonable on the basis of the following criteria:

- WfMS implementation constitutes a capital investment to be justified on the basis of transactional volumes and reduced resource requirements per process instance. Ongoing maintenance effort, license and operating costs need to be taken into account.
- WfMS implementation may include deriving information environmental elements from tangible environmental elements, e.g. when the state of a target element is entered into an information system. This is a reverse effect in conjunction with resource

requirements reduction when a WfMS is implemented, and needs to be considered accordingly.

Note that the automation of "passing on" results in the course of a task e.g. to relevant stakeholders is considered as task automation.

| Quality Criteria | Quality Predicates |
|---|---|
| There are no alternative splits or joins. | The process is fully linear. |
| There are no alternative splits or joins which are not managed through a WfMS. | Process routing is fully WfMS-supported. |
| There are alternative splits or joins not managed through a WfMS, but WfMS implementation is considered as economically not reasonable by subject matter experts. | WfMS support for routing is not recommended. |
| There are alternative splits or joins not managed through a WfMS, and WfMS implementation is considered as economically reasonable by subject matter experts. | WfMS support for routing should be considered. |

**Example 77 (Routing Automation).** Consider Sample Process C from Figure 2.7. In this case, routing is based on the results of medical examinations. If the case modeled constitutes a standard case, processing volume will be sufficient to implement a medical information system which, in the sense of a WfMS, automatically schedules treatments on the basis of examination results that have been entered into the system. If not, subject matter experts will judge information system implementation as economically not viable, and scheduling (i.e., routing) will be done manually.

### C.1.3 Conceptual Level Quality Drivers

This section provides details on quality drivers that need to be assessed on the basis of the business process as a whole considering conceptual characteristics which cannot be altered by, for example, altering individual tasks or re-arranging control flow. Rather, requirements towards fundamental changes to the underlying idea of a business process are addressed. Accordingly, conceptual level quality drivers require close examination of the respective business objective, fundamentally inquiring whether there might be an overall process design more apt to achieve the objective at hand.

**Guiding Question 7.** *On the conceptual level, which characteristics are relevant considering the application domain to limit resource requirements?*

**Quality Attribute 20 (Consideration of Good Practices).** For many application domains, collections of good practices or process patterns are available representing the "state

of the art" for the respective application domain. These might address a business objective as a whole or just individual processing aspects, and good practices may be documented explicitly (e.g., the ITIL in the information management domain [118]) or available as organizational knowledge of subject matter experts [236]. Whether good practices and process patterns have been considered properly constitutes an important driver towards limiting resource consumption.
**Assessment:** Subject matter experts' appraisal will take into account available collections of good practices specific to the industry or to the process class, as well as "group best practices" which reflect particular requirements and available resources a group of companies (e.g., available template information systems) that may be transferred from one organizational unit to others.

| Quality Criteria | Quality Predicates |
|---|---|
| As per subject matter experts' appraisal, there are no conceptual good practices or proven process patterns applicable to the business objective. | Conceptual good practices are not available. |
| As per subject matter experts' appraisal, conceptual good practices or proven process patterns are applicable to the business objective and have been implemented. | Conceptual good practices are leveraged. |
| As per subject matter experts' appraisal, conceptual good practices or proven process patterns are applicable to the business objective and have not been implemented, but implementation is considered as inefficient. | Conceptual good practices are not advisable. |
| As per subject matter experts' appraisal, conceptual good practices or proven process patterns are applicable to the business objective, have not been implemented, and implementation is considered as efficient. | Conceptual good practices should be considered. |

**Example 78 (Consideration of Good Practices).** Consider Sample Process A from Figure 2.5. The management of incoming invoices constitutes a well-understood application domain since it is not specific to any one industry. Relevant good practices in this respect comprise the application of intelligent scanning solutions to manage paper invoices (much promoted by industry, cf. [304]), the application of credit note procedures [17, 18], and the application of EDI solutions (which is even promoted by the United Nations, cf. [295]). As opposed to topics considered in *QA 5: Task automation / use of capital investments*, these practices are not implemented by changing a single task, but require an entirely new process to achieve the business objective. Whether any or all of these practices are advisable in a concrete application scenario needs to be determined considering required capital investments, transactional volumes etc.

**Quality Attribute 21 (Additional Control Procedures).** In many business processes, there are final "overall checks" before the final tasks enact state operations impacting target elements. This approach entails the risk that the overall check fails, and the resource con-

sumption incurred up to this point is wasted. Similarly, it is possible that individual tasks fail or cannot be enacted because prerequisites (i.e., task-requisite BSDs) are not fulfilled. These topics may be addressed by implementing additional controls, preferably early on in the process.
**Assessment:** Appraisal by subject matter experts will entail identification of possible issues that might be addressed by additional control procedures such as less desirable process outcomes. In addition, the additional effort of implementing new control procedures is to be taken into account.

| Quality Criteria | Quality Predicates |
|---|---|
| As per subject matter experts' appraisal, there are no issues which might be addressed by additional control procedures. | Additional control procedures are not applicable. |
| As per subject matter experts' appraisal, there are issues which might be addressed by additional control procedures. However, the additional effort incurred is judged to exceed possible gains. | Additional control procedures are not advisable. |
| there are issues which might be addressed by additional control procedures, and the additional effort incurred is judged to be justified by possible gains. | Additional control procedures should be considered. |

**Example 79 (Additional Control Procedures).** Consider Sample Process C from Figure 2.7. It is conceivable that Drug III might be applied together with Drug I, thus saving additional effort, and that the requirement for drug III could be determined early on through a simple additional examination. Implementing the said additional examination would then constitute an additional control procedure saving effort by enabling joint application of Drugs I and III.

**Quality Attribute 22 (Appropriate Organizational Responsibilities).** Besides activities, control flow, events and artifacts, BP meta-models, and therefore BP designs, also comprise organizational responsibilities in many cases (e.g., [80, 222]). For example, these are modeled as "swim lanes" in common process modeling languages (e.g., [80]). Accordingly, process model quality must consider the appropriateness of modeled organizational responsibilities as well. In terms of quality-aware modeling, organizational responsibilities are best reflected by task-requisite BSDs and state operations modeling the consumption of labor as a resource. In this context, appropriate organizational responsibilities must be considered between the possibly conflicting priorities of labor cost and the amount of time required to enact tasks.
**Assessment:** The following factors are relevant:

- *Labor unit cost* (e.g., per hour) incurred to enact tasks should be as low as possible. Usually, this is achieved by employing resources where the level of qualification does not exceed the level required by the application scenario.
- *Productivity* in the sense of processing speed and thus the amount of labor resources consumed is partially dependent on the level of qualification of personnel employed.

Accordingly, there is a possible goals conflict between minimizing labor unit cost and the amount of labor required. Hence, this issue is to be appraised by subject matter experts.

- Balancing *economies of scale and economies of scope* refers to specializing individuals on tasks or on cases (i.e., process instances). In the former case, tasks requiring manual labor will tend to be enacted by differing personnel per instance, i.e. a process instance will be subject to many hand-overs. In the latter case, individual process instances will tend to be enacted by individual persons as far as possible. The former case fosters specialization effects [75], and the latter case avoids additional effort incurred in hand-overs (e.g. making personnel familiar with case specifics). Economies of scope orientation is also referred to as "case management" (e.g., [72, 73]).

Subject matter experts need to take into account the factors described to appraise the appropriateness of organizational responsibilities in a process model.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| As per subject matter experts' appraisal, organizational responsibilities comprised in the process model are appropriate considering labor unit cost, personnel qualification, and the balancing between economies of scale and economies of scope. | Organizational responsibilities are modeled appropriately. |
| As per subject matter experts' appraisal, labor unit cost incurred for process enactment as per modeled resource consumption might be lowered considering qualification requirements. | Labor unit cost potentials should be considered. |
| As per subject matter experts' appraisal, productivity might be increased considering the respective effect on labor unit costs. | Productivity potentials should be considered. |
| As per subject matter experts' appraisal, economies of scale should be realized by utilizing specialization effects. Accordingly, activities should be allocated to more specialized personnel taking into account increased hand-over effort. | Economies of scale should be considered. |
| As per subject matter experts' appraisal, economies of scope should be realized by strengthening case handling principles. Activities should thus be allocated to less specialized personnel taking into account decreased specialization effects and potential additional qualification requirements. | Economies of scope (case management) should be considered. |

**Example 80 (Appropriate Organizational Responsibilities).** Consider Sample Process C from Figure 2.7. In terms of organizational responsibilities, it would be possible to have the entire process enacted by a physician. However, physicians' working hours are a scarce and expensive resource, so it is to be considered whether some tasks, e.g. examina-

tions, might be carried out by nurses instead. This might reduce labor unit costs. Moreover, specialized laboratory practices could be employed for examinations to generate specialization effects and increase productivity. Both measures, however, would increase the number of required case hand-overs which might result in a contrary effect.

**Quality Attribute 23 (Functional Integration).** Functional integration pertains to the utilization of scale effects in enacting activities within business processes. The notion of scale effects refers to the observation of decreasing cost per transaction with growing transactional volume. This is caused by, for example, changeover requirements (e.g., retooling of machinery, [237]), continuous utilization of capital resources, or specialization effects with personnel. Note that realizing scale effects may entail forgoing the minimization of organizational interfaces and cycle times which is one of the key benefits of the BPM paradigm (*process integration*, c.f., e.g., common BP reengineering practices [129]). Therefore, careful appraisal of corresponding potentials by subject matter experts is required. Implementation of functional integration in BP models can be achieved by using appropriate triggering events as the precedent to activities in scope for functional integration.
**Assessment:** Tasks or (composite) activities in scope for functional integration are identified by subject matter experts. In particular, the respective resource requirements as documented in task-requisite BSDs and state operations modeling resource consumption can provide relevant pointers. Relevant tasks and activities are then appraised by subject matter experts, taking into account actual transactional volume, possible scale effects, additional interfacing efforts caused, and possibly increase cycle times (cf. *QA 27: Timely process enactment*) caused by triggering uniform activities only if a sufficient "stack" has been accumulated.

| Quality Criteria | Quality Predicates |
|---|---|
| As per subject matter experts' appraisal, the business process comprises no activities where scale effects may be realized by functional integration. | Functional integration is not beneficial. |
| As per subject matter experts' appraisal, the business process comprises activities where scale effects may be realized by funtional integration. However, the additional effort caused in comparison to process integration will not be justified. | Functional integration is not beneficial considering effects on process integration. |
| As per subject matter experts' appraisal, the business process comprises activities where scale effects that may be realized by functional integration exceed additional effort caused by waiving corresponding process integration. | Functional integration should be considered. |

**Example 81 (Functional Integration).** Consider Sample Process B from Figure 2.6. When managing outgoing payments, senior management approval may be required for reasons of compliance or governance. Instead of approving each individual outgoing payment, payments are in practice collected in a proposed payment list, which is then approved in

summary by respective managers. Accordingly, the approval activity has been subject to functional integration. It is triggered by a time event, e.g., a weekly payment run.

**Quality Attribute 24 (Overall Efficacy and Efficiency).** Besides considering drivers of resource requirements on task and control flow level, it is also possible to subject a business process on the conceptual level to subject matter experts' appraisal regarding whether resource requirements in terms of availability and consumption are considered as reasonable. Overall, it is possible that this will be the case despite minor issues with "lower level" quality drivers, for instance in the case of processes where transactional volumes do not justify prioritized optimization projects. On the other hand, it is also conceivable that, while other quality drivers will not find fault with a business process, subject matter experts still consider the process as not efficacious and / or efficient. In this case, it is probable that the business objective cannot be achieved with reasonable means and needs to be re-considered. It may then be appropriate to subject the business process to a reengineering instead a quality management approach (cf. Section 8.2.5). Accordingly, *QA 24: Overall efficacy and efficiency* constitutes a sink to address topics specific to an individual process which cannot be expressed through generic quality drivers.
**Assessment:** Resource availability and consumption requirements are deducted from a quality-aware BP model as described in Section 7.4. This procedure can be automated. The results per target BSD are subject to subject matter experts' appraisal regarding whether these quality relations can be considered as reasonable.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| Considering the business objective in question, both resource requirements in terms of availability and consumption are considered as reasonable as per subject matter experts' appraisal. | The process is fully efficacious and fully efficient. |
| Considering the business objective in question, resource consumption as imposed by the business process cannot be considered as reasonable as per subject matter experts' appraisal. | The process is not fully efficient. |
| Considering the business objective in question, resource availability requirements cannot be considered as reasonable as per subject matter experts' appraisal. | The process is not fully efficacious. |

## C.2 Quality Meters

As opposed to quality drivers, quality meters constitute deductive quality attributes (cf. Section 8.2). Quality meters can be assessed based on empirical data on process instances.

**Quality Attribute 25 (Appropriate Capital Investments).** During design & implementation, capital resources of the organization are used to implement business processes, for example by implementing information systems. An appropriate process design will weigh expected transactional volumes and automation potentials on the one hand against corresponding capital expenditures on the other hand. With respect to the "content" of capital investments regarding individual tasks and, on the conceptual level, workflow automation and the implementation of proven good practices, this topic is considered in the respective quality drivers. This quality meter compares the overall capital expenditure incurred for process implementation to transactional volumes and optimization potentials in terms of cost. Unless capital investments can also be used for purposes besides the business process under consideration, the related expenditures might be considered as "sunk cost" and therefore as irrelevant to managerial control once the process has been implemented. This is, however, not the case considering the limited lifecycle of capital goods including information systems. Once necessary re-investments are taken into account, it makes sense to consider required capital expenditures in terms of process quality as well.

**Assessment:** To appraise whether capital expenditures can be considered as appropriate, subject matter experts will compare the planned or actual amount of capital invested to expected or actual transactional volumes and enactment cost. In the context of transactional general and administrative processes which constitute the main scope of this thesis, enactment cost will usually pertain to the fully loaded[1] personnel cost of labor resources committed to the process. Accordingly, assessment will take into account the actual or expected factor cost level (depending, e.g., on geographical location and the level of qualification required) and the degree to which individual employees spend their available working time on the process in question. Based on this information, subject matter experts will evaluate whether further capital investments are to be considered as economically efficient or not. This will require either in-depth technical knowledge of the process in question or available planning on possible implementation measures including business case analyses for individual measures.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| According to subject matter experts' appraisal, capital investments planned or implemented will cause or have caused capital expenditures which are inefficient considering the expected or actual transactional volume, the corresponding reduction in manual effort per process instance, and the underlying factor costs. | Inefficient capital investments are planned or have been implemented. |
| According to subject matter experts' appraisal, capital investments planned or implemented can be considered as appropriate based on the comparison between capital expenditures incurred, expected or actual transactional volume, the corresponding reduction in manual effort per process instance, and the underlying factor costs. | Capital investments planned or implemented are appropriate. |
| According to subject matter experts' appraisal, additional capital investments would reduce manual effort to enact the business process sufficiently to justify the corresponding capital expenditures considering factor costs. | Additional capital investments should be implemented. |

[1] The term "fully loaded" usually refers to personnel costs including social contributions, insurances, and infrastructure cost directly related to personnel such as PCs.

**Example 82 (Appropriate Capital Investments).** Consider Sample Process A from Figure 2.5. Many organizations today invest in automating the management of supplier invoices by automating the entry of relevant invoice data into the ERP system through early scanning in combination with original character recognition (OCR) methods or EDI with suppliers. Whether these investments are justified, however, must be determined considering implementation cost, transactional volume, and expected effort saved per process instance. In addition, subject matter experts need to take into account alternative methods (cf. the Consideration of Good Practices quality driver), e.g. the implementation of a credit note procedure [38].

**Quality Attribute 26 (Efficacious and Efficient Enactment Performance).** According to the considerations above, analyzing traces constitutes a means to assess enactment quality independently of design quality. To this end, possible enactment paths (i.e., the "templates" for actual traces) can be assigned to one of three classes:

- Desirable traces reflect efficacious and efficient process enactment, i.e. the fulfillment of target BSDs according to the business objective, and an economic treatment of resources.
- Undesirable traces reflect defects which lead to target BSDs not being fulfilled according to the business objective, or to a failure to treat resources economically.
  - Undesirable traces may be caused by *defective provision of resources.* This occurs when resource availability requirements are not met because the relevant resources have not been provided by upstream processes (e.g., shortages of semi-finished goods), or are generally unavailable to the organization (e.g., information system downtime).
  - Undesirable traces may be caused by *defective process enactment.* This aspect is relevant to enactment quality.

Economic treatment of resources in this respect refers to avoiding resource waste in the sense of unnecessary resource consumption. Note that resource availability considerations, or full efficacy, are not relevant to determine whether a possible enactment path can be considered as desirable. This consideration reflects that resources available in the outer environment of a process instance are given prior to process enactment – generous availability of resources cannot be charged as a fault to the process instance. Resource consumption, however, constitutes another matter, since this aspect can be influenced by process performance.
**Assessment:** Assessment of the quality attribute requires two stages. The first stage, the appraisal of possible enactment paths, is performed once for the process model. The second stage, the appraisal of traces, can be performed for any set of actual traces, e.g. periodically for the traces which have occurred for a given timeframe.

1. The set of possible enactment paths is determined based on the quality-aware BP model. Possible enactment paths are amended with the corresponding set of target BSDs fulfilled. For each possible enactment path, conditional consolidation is performed, and formal efficacy as well as resource resource consumption of each path is determined (cf. Section 7.4). Formally inefficacious paths are discarded as undesirable. For the remaining paths, subject matter experts' appraisal is required to determine whether each path is desirable or not. Grouping paths based on the respective set

of target BSDs fulfilled may support this task, since resource consumption for each option can then be easily compared to other paths addressing the same set of target BSDs. Undesirable paths are then analyzed to determine whether the task in question is caused by a defective outer environment (i.e., by defective upstream processes), or by defective human effort during process enactment. In this regard, fault tree analysis or comparable methods may be employed [305].

2. For a given set of traces, the relative frequency of each underlying possible enactment path is determined. This task can be supported through process mining tools [28]. Based on the classification of enactment paths as desirable, undesirable due to defective upstream processes, or undesirable due to poor enactment performance, subject matter experts can appraise whether enactment performance can be considered as efficacious and efficient.

Figure 8.13 illustrates the two stages of appraisal described. In practical settings, a quality-aware model might not be available. In that case, it is also possible to mine process variants [148] in the sense of traces reflecting a common possible enactment path from enactment logs, and amend the results with relevant estimates.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| Undesirable traces occur, and reflect defective process enactment. This leads to target BSDs not being fulfilled. | Process enactment is inefficacious. |
| Undesirable traces occur, and reflect defective process enactment. This leads to increased resource consumption. | Process enactment is inefficient. |
| Undesirable traces occur, but do not reflect defective process enactment. Rather, undesirable traces are caused by deficiencies in upstream business processes impacting the availability of resources and the initial state of target artifacts. | The effectiveness of upstream business processes should be improved. |
| Undesirable traces do not occur. | Process enactment is efficacious and efficient. |

**Example 83 (Efficacious and Efficient Enactment Performance).** Consider Sample Process A from Figure 2.5. In many organizations, assessment of incoming invoices is carried out by sending the invoice to the department or contact partner responsible, who will then approve or decline the invoice. If, however, the invoice is sent to the wrong contact partner, it will be passed on until the responsible person has been found. Clearly, this behavior constitutes an example of inefficient process performance due to defects in the manual effort employed in the process, namely, the identification of the right contact partner from the start on.

Inefficacious process performance occurs if the responsible contact partner is not identified at all, which will cause the business objective to remain unachieved.

It is also possible that the invoice is not sent to the right contact partner because data capture in an upstream scanning process is defective. In that case, process performance cannot be designated as inefficacious or inefficient, but the effectiveness of upstream processes should be improved.

**Quality Attribute 27 (Timely Process Enactment).** Timely availability of process results, or timely fulfillment of target BSDs, is an issue of enactment performance since process design can be assumed to enable timeliness in practice. Accordingly, timeliness will reflect two aspect: the selection of an actual trace comprising the relevant control flow elements out of the set of possible enactment paths, and the cycle times of individual control flow elements. Both aspects are influenced by the availability of resources to the process instance, and by the quality of manual effort involved. Accordingly, timeliness is closely related to *QA 26: Efficacious and efficient enactment performance*, since comparable root causes need to be assessed.
**Assessment:** For all target BSDs, the timeframe to be adhered to if process enactment is to be considered as timely is determined. The timeframe will usually be given in terms of attributes of the corresponding target artifacts, and requires subject matter experts' knowledge on the application domain. For example, the timeframe might be defined in terms of a certain amount of working days after a document has been received. Based on actual cycle times for each target BSD, defects with respect to timeliness are determined. This can be achieved by considering WfMS or ERP system log data. Employing process intelligence tools will facilitate the assessment. Note that whether a target BSD is fulfilled at all is not within the scope of this quality attribute, but considered as a matter of *QA 26: Efficacious and efficient enactment performance*. Similarly to the latter quality attribute, timeliness defects are analyzed on the basis of actual traces, which reflect both resource availability and manual effort in process enactment. In addition, the cycle times of individual control flow elements can be assessed in a similar manner. In practical settings, particularly the appraisal of individual control flow elements will demand significant involvement of subject matter experts. Thus, an approach based on spot checks of defects will be appropriate in most cases.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| Defects in timeliness occur, and reflect defective process enactment with regard to the selection of possible enactment paths as actual traces. In other words, defects in manual effort during process enactment lead to traces which entail prolonged cycle times. | The quality of manual enactment effort should be addressed regarding the selection of actual traces to achieve timeliness. |
| Defects in timeliness occur, and reflect defective process enactment with regard to the cycle times of individual control flow elements. In other words, the enactment of control flow elements requiring manual effort is not completed on time. | The availability of employees to enact control flow elements should be addressed to achieve timeliness. |

| | |
|---|---|
| Defects in timeliness do not reflect defective process enactment, but issues in upstream processes. In other words, a lack of required resources provided by upstream processes leads to actual traces with prolonged cycle times or to prolonged cycle times of individual control flow elements. The latter occurs if the process holds to await the termination of upstream processes. | The effectiveness of upstream business processes should be improved to achieve timeliness. |
| Defects in timeliness do not occur. | Timeliness of process enactment is achieved. |

**Example 84 (Timely Process Enactment).** Consider Sample Process C from Figure 2.7. If there is a delay in the application of drugs causing, for example, unnecessary occupation of hospital beds, this might be caused by in-availability of nurses and physicians, but also by in-availability of medications to be provided by the upstream pharmacy process.

**Quality Attribute 28 (Trace Deviation Errors).** Trace deviation errors refer to issues occurring during process enactment which cannot be captured through the assessment and analysis of actual enactment traces as given by, for instance, WfMS or ERP log files. In principle, there are the following error possibilities in this regard:

- *Omission error:* Although a target BSD should be fulfilled as per the actual trace enacted, the corresponding modifications to the respective affected target artifact are not made or not wrongly executed (cf. Example 85).
- The following error possibilities refer to the defective approval or disapproval of fully determinate bivalent target BSDs.
  - *Alpha error* or false positive: A fully determinate bivalent target BSD is fulfilled although the respective conditional proposition is not fulfilled. Since formal efficacy is considered when assessing design quality (cf. the *Consideration of Conditional Propositions* quality driver), this issue is usually caused by defective assessment of conditions attached to conditional split gateways.
  - *Beta error* or false negative: A fully determinate bivalent target BSD is not fulfilled although the respective conditional proposition is fulfilled. Again, this issue is usually caused by defective assessment of conditions attached to conditional split gateways. Note that issues regarding instance abortion for lack of resources are addressed as a matter of the *Efficacious and Efficient Enactment Performance* quality meter.

The concept of alpha and beta errors reflect the corresponding terminology in the context of statistical analyses: an alpha error refers to the failure to accept a true null hypothesis, and a beta error refers to the failure to reject a false null hypothesis [243].
**Assessment:** Since trace deviation errors cannot be captured by analyzing traces as provided by WfMS or ERP log data, assessment must be executed by manually analyzing process instance samples in terms of the actual states of affecting and affected elements. Whether this analysis is carried out considering Effectiveness Criterion 3, *Cost effectiveness*, and which sample size will be required to achieve meaningful results must be determined considering the risk and the impact of errors. In this context, subject matter experts may,

for instance, utilize the Six Sigma methodology as a collection of tools comprising respective sampling methods [283] which allow achieving a reasonable level of assurance. To this end, the required level of confidence with regard to the omission, alpha and beta errors must be agreed by subject matter experts based on a consideration of how severe the organization will be impacted by omission, alpha and beta errors, and which mitigation measures are possible to rectify errors. The higher the possible impact of errors, the higher the required level of confidence will be.

| **Quality Criteria** | **Quality Predicates** |
|---|---|
| Based on statistical sampling guided by subject matter experts' appraisal of process characteristics and the respective impact, it cannot be confirmed with reasonable assurance that target BSDs are sufficiently fulfilled as suggested by the enactment of the respective task. | Safeguards against omission errors should be implemented. |
| Based on statistical sampling guided by subject matter experts' appraisal of process characteristics and the respective impact, it cannot be confirmed with reasonable assurance that fully determinate bivalent target BSDs are fulfilled only if the respective conditional propositions are given, although formal efficacy of the process model is ensured. | Safeguards against alpha errors should be implemented. |
| Based on statistical sampling guided by subject matter experts' appraisal of process characteristics and the respective impact, it cannot be confirmed with reasonable assurance that fully determinate bivalent target BSDs are fulfilled if the respective conditional propositions as given in the process model are fulfilled. Note that this quality criterion assumes full efficacy of the process model. In other words, it does not address whether resource requirements given by the process model can be considered as reasonable, but whether errors occur during approval or disapproval of target BSDs. | Safeguards against beta errors should be implemented. |
| Based on statistical sampling guided by subject matter experts' appraisal of process characteristics and the respective impact, it cannot be confirmed with reasonable assurance that the actual fulfillment of target BSDs reflects the logged enactment of corresponding tasks, and that fully determinate bivalent target BSDs are fulfilled if and only if the respective conditional propositions are fulfilled. | Process enactment achieves a reasonable level of assurance with regard to omission errors, alpha errors, and beta errors. |

**Example 85 (Poka-Yoke Safeguards to Address Omission Errors).** This possibility of omission errors is, for example, addressed through the quality management practice of

Poka-yoke (Japanese: “error-proof”) where tasks are amended with safeguards to ensure that they are actually enacted as specified [287].

As an example, consider the assembly of electronic circuits: within one task, resistors might be soldered on the circuit board. It is possible that single resistors are omitted by accident. In that case, the logged enactment of the task would suggest that the corresponding target BSD (*“Resistors applied = true”*) is fulfilled, which is, however, not the case in reality. A possible safeguard in this respect might be that the resistors for each circuit board are counted onto a Petri plate in advance. Thus, there is a visual and haptic control for the worker to ensure that all resistors have been applied.

# D Sample Application of Quality Attributes

This appendix demonstrates the application of the full set of quality attributes presented in Chapter 8 and refined in Appendix C to the sample business process used in Chapter 12. It exemplifies how quality attributes can be leveraged even if quality-oriented process modeling and corresponding tool support are not available yet. Moreover, it also highlights the limitations of this approach in comparison to a fully-fledged quality-aware BP lifecycle (cf. Chapter 9).

Table D.1 lists the BP quality attributes defined in this thesis, and assigns quality predicates on the basis of the sample business process described in Chapter 12 which relates to the management of incoming job applications in a professional services firm. In addition, the table flags the quality predicates that constitute a call for further action, and lists the respective PIMs as defined in Table 12.2.

| **Quality Attribute** | **Assigned Quality Predicate** | **Need for Action** |
|---|---|---|
| *Task Level Quality Drivers* | | |
| *QA 1: Sufficiency of State Operations* | Tasks are formally sufficient towards the business objective. | |
| *QA 2: Effective Tasks* | All tasks are indirectly effective. | |
| *QA 3: Effective State Operations* | There are no resource requirements resulting from in-effective state operations. | |
| *QA 4: Reasonable Task Resource Requirements* | Tasks comprise generally reasonable resource requirements. | |
| *QA 5: Task Automation / Use of Capital Investments* | There are tasks not automated to the efficient extent, capital investments should be considered. | X (PIM 1) |
| *QA 6: Task Classification* | There are additional effective task classification potentials which should be considered. | X (PIM 3) |
| *Control Flow Level Quality Drivers* | | |
| *QA 7: Consideration of Conditional Propositions* | The business process is formally efficacious. | |
| *QA 8: Completeness of Control Flow* | The control flow model is complete with regard to the business objective. | |
| *QA 9: Effective Target Aspects* | Target aspects fulfilled are effective for every process instance. | |
| *QA 10: Effective and Efficacious Conditional Splits* | Conditional splits are effective for every process instance. | |
| *QA 11: Sequential Tasks Compositions* | De-sequentialization is not advisable for domain-specific reasons. | |

| | | |
|---|---|---|
| *QA 12: Parallel Tasks Composition* | The process is fully de-parallelized. | |
| *QA 13: Alternative Activities Composition* | Alternative activities composition is not relevant. | |
| *QA 14: Mitigation of Repetitive Loops* | The process comprises repetitive loops that should be mitigated. | X (PIM 5) |
| *QA 15: Early Approval or Dis-approval* | Grouped checking activities should be re-arranged to reflect the principle of early approval or dis-approval in terms of outcome probabilities and resource consumption. | X (PIM 2, PIM 5) |
| *QA 16: Early Failure* | Early failure re-arrangement is not advisable. | |
| *QA 17: Effective Tasks in Enactment Paths* | Additional differentiation of possible enactment paths to ensure task effectiveness is not advisable due to the additional resource requirements incurred. | |
| *QA 18: Effective State Operations in Enactment Paths* | In-effective state operations are justified by technical reasons. | |
| *QA 19: Routing Automation* | WfMS support for routing should be considered. | X (PIM 1, PIM 4) |
| *Conceptual Level Quality Drivers* | | |
| *QA 20: Consideration of Good Practices* | Conceptual good practices are leveraged. | |
| *QA 21: Additional Control Procedures* | Additional control procedures should be considered. | X (PIM 2, PIM 4) |
| *QA 22: Appropriate Organizational Responsibilities* | Economies of scope (case management) should be considered. | X (PIM 3, PIM 5) |
| *QA 23: Functional Integration* | Functional integration is not beneficial considering effects on process integration. | |
| *QA 24: Overall Efficacy and Efficiency* | The process is fully efficacious and fully efficient. | |
| *Quality Meters* | | |
| *QA 25: Appropriate Capital Investments* | Capital investments planned or implemented are appropriate. | |
| *QA 26: Efficacious and Efficient Enactment Performance* | Process enactment is inefficacious. | X |
| *QA 27: Timely Process Enactment* | The availability of employees to enact control flow elements should be addressed to achieve timeliness. | X (PIM 4) |
| *QA 28: Trace Deviation Errors* | Process enactment achieves a reasonable level of assurance with regard to omission errors, alpha errors, and beta errors. | |

Table D.1: Quality Predicates for Sample Business Process

Note that the "escalation procedure" process improvement pattern cited in PIM 4 refers to process instances not completed in time because employees have not been available for tasks within the process. As noted in Table D.1, this corresponds to the quality predicate associated with QA 27. Moreover, a "need for action" has been identified with regard to QA 27, *Efficacious and Efficient Enactment Performance.* The corresponding quality predicate is not reflected in a singular process improvement measure, but represents the general issue that not all process instances are successfully completed in the application scenario. Accordingly, this topic has been included in the process improvement *objective* of reducing the number of process instances required to achieve the overall business objective (cf. Figure 12.7).

Besides the quality attributes included in Table D.1, PIM 3 also refers to the *Buffering* reengineering practice described in [130]. As argued in Section 11.1, this practice has not been included in the set of quality attributes defined in this thesis. On the basis of the sample application, it thus needs to be discussed whether *Buffering* can be subsumed as a *Task classification* (cf. QA 6), or if a new quality attribute should be introduced. In this context, consider the discussion of limitations presented in Chapter 13, in particular with regard to the completeness of the set of quality attributes identified.