Abteilung Anwendersoftware
Institut für parallele und verteilte Systeme
Universitätsstraße 38, D-70569 Stuttgart

Lehrstuhl für theoretische Computerlinguistik
Institut für maschinelle Sprachverarbeitung
Azenbergstraße 12, D-70174 Stuttgart

Diplomarbeit Nr. 3152

EVALUATION OF AUTOMATED BUSINESS PROCESS OPTIMIZATION

KEMAL TOLGA ERGIN

STUDIENGANG	Linguistik
PRÜFER	Prof. Dr.-Ing. habil. Bernhard Mitschang
ZWEITPRÜFER	Prof. Ph.D. Hinrich Schütze
BETREUER	M.Sc. Florian Niedermann
BEGONNEN AM	07.12.2010
BEENDET AM	30.07.2011
CR-KLASSIFIKATION	H.2.4, H.2.8, H.4.1, J.1, K.4.3, K.6.1

EVALUATION OF AUTOMATED BUSINESS PROCESS OPTIMIZATION

KEMAL TOLGA ERGIN

UNIVERSITÄT STUTTGART

Fakultät für Informatik, Elektrotechnik und Informationstechnik

Institut für parallele und verteilte Systeme

Diplomarbeit Nr. 3152

IMS Nr. 97

Juli 2011

SUPERVISORS:
Prof. Dr.-Ing. habil. Bernhard Mitschang, IPVS
Prof. Ph.D. Hinrich Schütze, IMS
M.Sc. Florian Niedermann, IPVS

To nourish children and raise them against odds is in any time, any place, more valuable than to fix bolts in cars or design nuclear weapons.

— Marilyn French

Für meine Allerliebsten: *Silvia & Sarah*.

ABSTRACT

Today's highly competitive markets tend to favor enterprises, in which business processes are analyzed and optimized regularly, in order to be able to operate in accordance with their business goals. The variety of *business process management* (BPM) methods applied for this purpose, since the emergence of the concept of *business reengineering* in the 1990s, ranges from *incremental* adjustments to *radical* restructuring. In combination with contemporary workflow automation technology, modern redesign methods are powerful tools for enhancing business performance, enabling companies to maintain a winning margin. Optimization methods that deliver sustainable results using *evolutionary* approaches, however, are nowadays becoming increasingly popular — once again, two decades after continuous improvement paradigms had almost completely been abandoned in favor of *revolutionary* process redesign.

This diploma thesis explores one such *evolutionary* BPM approach employed in the deep Business Optimization Platform (dBOP), a research prototype, which assists analysts with the selection and application of suitable process improvement techniques. The present work demonstrates an evaluation of dBOP with the help of simulated business scenarios based on real case studies, and documents the types of optimization patterns most readily applied through automated process redesign. For this purpose two business processes, one from a car rental enterprise and one from a health insurance company, are modeled and deployed on a process server, and executed using web services and sample data warehouses based on actual statistics. These processes are then analyzed with dBOP, in order to compare its optimization recommendations with those expected from a human analyst's perspective.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LISTINGS

ACRONYMS

API	Application Programming Interface
BASt	Bundesanstalt für Straßenwesen
BGM	Bundesministerium für Gesundheit
BI	Business Intelligence
BIA	Business Impact Analysis
BPEL	Business Process Execution Language
BPD	Business Process Diagram
BPM	Business Process Management
BPMN	Business Process Modeling Notation
BPO	Business Process Optimization
CAD	Computer-Aided Design
dBA	deep Business Analytics
dBDI	deep Business Data Integration
dBOP	deep Business Optimization Platform
DBMS	Database Management System
DM	Data Mining
DSS	Decision Support Systems
DWH	Data Warehouse
EBM	Einheitlicher Bewertungsmaßstab
EAI	Enterprise Application Integration
EC	European Commission
ECHI	European Community Health Indicators
ERM	Entity Relationship Model
ESB	Enterprise Service Bus
EU	European Union
GOÄ	Gebührenordnung für Ärzte
GUI	Graphical User Interface

HTTP	Hypertext Transfer Protocol
ICD	International Classification of Diseases
IDE	Integrated Development Environment
IPVS	Institute of Parallel and Distributed Systems, *Institut für Parallele und Verteilte Systeme*
IS	Information System
IT	Information Technology
JDBC	Java Database Connectivity
KBA	Kraftfahrt-Bundesamt
OASIS	Organization for the Advancement of Structured Information Standards
OLAP	Online Analytical Processing
OMG	Object Management Group
PAIS	Process-Aware Information System
PIR	Process Insight Repository
PKV	Verband der privaten Krankenversicherung e.V.
PM	Process Mining
SCA	Service Component Architecture
SGML	Standard Generalized Markup Language
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SQL	Structured Query Language
TAM	Technology Acceptance Model
UML	Unified Modeling Language
WfMC	Workflow Management Coalition
WHO	World Health Organization
WSBPEL	Web Services Business Process Execution Language
WSDL	Web Services Description Language
XML	Extensible Markup Language
XSD	XML Schema Definition

Part I

INTRODUCTION

1 INTRODUCTION

Over the last two decades, the notion of *business process management* (BPM) has evolved from a concept of *radical* business reengineering to a blend of methodologies that by now embrace *incremental* process improvement techniques just as well, rendering a broader applicability, and producing better sustainable results. Business process *redesign* has thus become an *evolutionary* approach to enhancing business performance, in contrast to formerly popular *revolutionary* restructuring endeavors.

Despite increasing process efficiency with the help of ever advancing information technology, past redesign methods have largely focused on improving *static* models and redeploying the process, whenever the business goals changed. A newly emerging line of process management research, however, has developed methods for analyzing and adjusting process structures *dynamically* during actual operations, i.e., without always having to redesign the process as a whole.

This diploma thesis explores these *dynamic* computational methods in *business process optimization* and how they can be utilized to investigate real-life business scenarios. After defining and formally modeling business processes based on two case studies, the process patterns therein are evaluated on an automated optimization platform, particularly with regard to possible improvements. Each formal scenario definition includes the associated data storage dimension, in which a data warehouse is generated containing operational and process data in accordance with relevant statistical patterns, aiming to achieve a realistic business process simulation.

Finally, the output of automated optimization is compared to the expected transformations leading to an improved process, as would have been suggested by a process analyst.

1.1 OBJECTIVES

The present diploma thesis attempts to demonstrate the functional capabilities of the dBOP, which is currently under development as part of an ongoing research project titled *Business Impact Analysis* at the Institute of Parallel and Distributed Systems, *Institut für Parallele und Verteilte Systeme* (IPVS), accommodated at the University of Stuttgart.

The main objective of this thesis is to inspect the analytic outcome delivered by dBOP and its contingent recommendations for optimization, particularly with regard to theoretical redesign presumptions. The analysis platform itself and related architectures are described in detail in Section 1.3.1.

The text at hand is a report on the methods deployed for evaluating the optimization features of dBOP. The preliminary strategy pursued in seeking to analyze the optimization platform, is to devise business process schemes based on real-life enterprise operations, which are then investigated on the basis of optimality considerations. These process schemes, or *scenarios*, are deliberately left in a suboptimal state for purposes of the present inquiry, and modeled accordingly. It is then expected that the analysis of such misarranged schemata on dBOP would yield corrections leading to an overall more efficient process, perhaps even to a potentially optimal structure.

In order to be able to extract and analyze process data from a given business scenario, the corresponding business process is modeled at an appropriate level of detail, and deployed on an IBM WebSphere® Process Server, where all process activities, including human tasks, are executed by simulated participants. The simulated operations are performed by web services implemented in `Java`[1]. These web service operations replace the *actors* in a process. Section 3.2.3 elaborates on the fundamentals of web service technologies, and Chapter 3 encompasses further implementation details specific to the evaluation of business scenarios, providing a comprehensive account on development and deployment.

Based on the insights attained through the conclusions of process analysis, it can be possible to detect patterns in the distribution of process tasks which bear further potential for optimization. Improvements in a given business process can be attained by displacing or otherwise altering individual tasks or groups of tasks. Certain process activities, on the other hand, can in some cases even turn out to be entirely expendable. Particulars of the optimization techniques applied in the present work are discussed in Section 2.3.

The following section provides an overview of the project context in which the present work emerged, giving a brief introduction to previous work related to the subject matter.

1.2 OUTLINE

The present document consists of three main parts: *introduction*, *method*, and *conclusion*, followed by an *appendix* containing method-specific details.

1 `Java` is a registered trademark of Sun Microsystems, a subsidiary of the Oracle Corporation

This introductory chapter gives a brief overview of the empirical investigation conducted to evaluate a compound software architecture devised to support process analysts in reenginering tasks. It presents paradigms related to workflow technologies and their applications, with primary focus on process automation and improvement, set in the specific context of automated analysis and dynamic optimization. The next section in this chapter states the particular objectives of the present diploma thesis and provides a general description of concepts essential to comprehending its underlying principles. Section 1.3 gives an overview of the specific research context in which the present work is accommodated.

The introductory part of the document at hand is complemented by a further chapter on *fundamental concepts*: Chapter 2 illuminates supplementary fundamental ideas and reflects on associated concepts and definitions. The center of attention throughout this report is largely concentrated on the notion of Business Process Management (BPM). However, the subject matter is related to a wide range of research fields, both in a historical sense as well as in terms of technical proximity. Sections 2.1 and 2.2 attempt to illuminate these relations, and describe the concept *business process* in an academic framework, adopting principal definitions originating in workflow-related research literature. The subsequent passages concisely illustrate common techniques utilized to evaluate and optimize given process structures.

The second part of the document encompasses the main body of work carried out in producing this thesis, and describes the *methodology* applied in investigating automated process optimization. A general theoretical account of the chain of procedures and development steps leading to an *executable* business process, is provided in Chapter 3, describing the standards used in *modeling* a process, illuminating techniques for providing relevant *data resources*, and characterizing the category of software methods employed in business process *simulation*. Section 3.2 presents the *implementation* details of the software environment in which the systematic evaluation of the business scenarios takes place.

Chapter 4 provides comprehensive accounts of real-life business scenarios which are evaluated for possible structural improvement. Finally, the data drawn upon for this purpose, and its preparation, is explained in Chapter 5.

The concluding part of the thesis discusses its *results* in Chapter 6 followed by a closing *outlook* into possible benefits of the present work, and reflects on its relation to current research on the subject matter.

1.3 RELATED WORK

The following sections provide an overview of the specific research context in which the present work is accommodated. As already mentioned in Section 1.1, the major context of the present study is the BIA project, which will be briefly introduced here, followed by a general account on research topics, to which this thesis is related.

1.3.1 *Business Impact Analysis*

The Business Impact Analysis (BIA) project started in June 2006, aiming to develop a heuristic approach to automated integration of process execution data and operational business data in enterprises, in order to enable a more informative and comprehensive analysis of business processes.[2]

The aforementioned IPVS project incorporates several research tasks that address the following issues:

- Development of an adequate ontology for semantic annotations
- Semantic annotation of process data
- Semantic annotation of operational data
- Automatic integration of process data and operational data: Development of reasoning and schema matching algorithms
- Development of analysis techniques for the integrated data using Online Analytical Processing (OLAP) and Data Mining (DM)
- Optimization of business processes

The present thesis is predominantly associated with the last mentioned item, i.e., *process optimization*. One of the essential components of the BIA project, is research conducted on the deep Business Optimization Platform (dBOP), which is described in Section 1.3.1.2.

Further details on the semantic annotation approach and automated data integration methods are described in [Radeschütz 08b] and [Radeschütz 08a], in the framework of *deep business analysis*. Implementation details concerning the matching techniques for process and operational data using the BIA-Editor can be found in [Radeschütz 10].

2 Further details on the BIA project are available on the official project homepage at http://www.ipvs.uni-stuttgart.de/abteilungen/as/forschung/projekte/BIA/

1.3.1.1 *dBOP System Architecture*

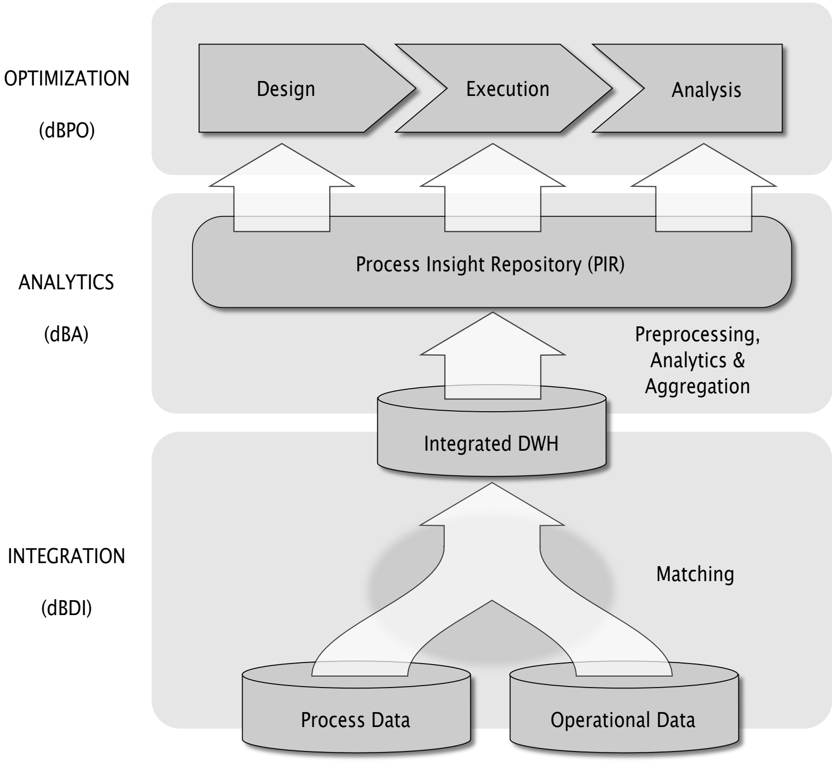


Figure 1: The three architectural layers of dBOP.

To emphasize their point of focus in the motivation for dBOP, the developers assert in [Niedermann 10], that

> *" . . . current, largely manual process optimization techniques based on non-integrated data present a significant obstacle on the way to achieving superior business process performance."*

The deep Business Optimization Platform (dBOP) developed at IPVS, provides an environment for the (semi-)automated optimization of processes, aiming to address the challenge of reducing manual adjustments during process redesign.

The Platform provides the ability to reveal *deep* insights that would remain concealed in most cases, in which the analysis is fed by a single data source. Customized DM techniques allow even deeper insight into the available data. Unlike simple queries on the process database, dBOP provides analysts with information regarding behavioral correlations between process factors. The results achieved by means of extended analysis are coupled with a pattern repository to support decision-making, by providing the analyst with guidance on how to obtain process improvement, thus enabling *"a fast and cost-efficient process optimization"* to achieve high process quality.

The dBOP architecture is composed of three main layers, as illustrated in Figure 1:

1. *Data Integration* is the initial processing step, and takes place in the deep Business Data Integration (dBDI) layer. As data relevant to the process and its decisions can be spread over a variety of heterogeneous data sources, this lowest platform layer provides the means to match and integrate process data with other data sources. The methods employed are discussed in [Radeschütz 10].

2. *Process Analytics* employ a range of specific preprocessing and analytics procedures, which are synthesized in the deep Business Analytics (dBA) layer. In order to achieve meaningful optimization results, process specific *insights*, i.e., information that can be readily transformed into concrete process optimization measures, are required to be extracted from the integrated data layer. The additional information retrieved is then stored in a Process Insight Repository (PIR). A comprehensive account on the analysis methods can be found in [Radeschütz 09].

3. *Process Optimization* is based on insights from the corresponding PIR generated in the preceding analysis layer. Next to customized analysis techniques, the platform also contains a broad set of formalized best practice process optimization patterns, stored in the *pattern catalogue* (see Section 2.3.3 for an overview. Implementation details and application methods are discussed in [Niedermann 11b]). These patterns, such as parallelization, task elimination or decision automation, utilize the analysis results to determine which modification of the process are most beneficial with regard to a certain goal function specified by the process analyst.

How decision-making can be supported based on the above process analysis methods, is discussed in [Niedermann 11a] (along with related contributions in [Abramowicz 11]). Specific aspects of the methods employed to optimize business processes are presented in Section 2.3.

1.3.1.2 *Deep Business Process Optimization*

A comprehensive account on the optimization layer of dBOP can be found in [Niedermann 11d], where practical applications of the findings in BPO research are discussed in detail. How these techniques are utilized to support analysts during business process design are treated in [Niedermann 11c].

General terminology, emergence and evolution of the concept of *business process optimization* is described in Chapter 2, with a detailed discussion of related research literature.

1.3.1.3 *dBOP-Modeler*

The dBOP-Modeler (formerly called dBOP-Editor) is the central process analysis tool of the deep Business Optimization Platform (dBOP). It is a custom implementation, based on standard process modeling tools. The intuitive GUI allows for straightforward process modeling and can be used to visualize analysis results and optimization recommendations, allowing the analyst to investigate various process configurations. Important features and some implementation details of the dBOP-Modeler are discussed in [Niedermann 11d].

1.3.2 *Simulating Business Processes*

Manuel Laguna and Johan Marklund devise a simulation-based framework for business process design projects in their recently published textbook, [Laguna 11]. They provide comprehensive illustrations of both *analytical queuing* methods, as well as *computer-based simulation* methods, used for incorporating variability into business process models.

Earlier attempts to formalize simulation methods and terminology from an industrial engineering perspective can be found in [Harrell 95], and a related demonstration of the applicability of these methods in business process analysis and simulation in [Tumay 96], elaborating further on analogous techniques discussed in [Gladwin 94].

Recent research on the topic in view of experimental approaches is illuminated in [van der Aalst 10a], with special attention turned to precise resource modeling for efficient allocation. Furthermore, the human dimension of business process simulation and particulars of human behavior in an IT-driven process environment is described using a socio-technical approach in [Gregoriades 08].

An account of the subject matter with regard to methods employed in the present study is given in the following chapter, in Section 2.4.

2 FUNDAMENTAL CONCEPTS

This chapter provides an overview of common terminology relevant to the principal subject matter, with brief remarks on its origins. Section 2.1 attempts to contemplate on Business Process Management (BPM) in a historical context. Technical terms and related standards are specified in Section 2.2, followed by a detailed account of optimization *"best-practices"* in Section 2.3, derived from case studies and descriptive categories in contemporary research literature on process improvement and business reengineering. The final section of the present chapter discusses the role of case studies in the evaluation of Information Technology (IT) systems and architectures.

2.1 BUSINESS PROCESS MANAGEMENT AND OPTIMIZATION

The assessment and representation of business processes are not ends in themselves, but rather essential means to statically or dynamically implement transformation activities for organizational improvement. There are several names for this kind of venture, such as *Process Innovation*, *Business Process Improvement*, *Business Process Redesign*, *Business Reengineering* and several other combinations of these key terms; whereas all endeavors related to process design and redesign are generally referred to as *Business Process Management*.

The past two decades have witnessed an increasing interest in BPM and related fields by a growing community of business people and academics alike. The body of knowledge on the subject matter and the range of methodologies, tools and techniques have also expanded substantially. However, it remains *"[a] challenge [. . .] to provide concise and widely accepted definitions, taxonomies, and overall frameworks for Business Process Management"*, as noted accurately in [vom Brocke 10]. For the sake of clarity, the present work adapts a specific definition of the term BPM which is pretty much confined to the domains of *scientific management* and *organization theory*:

defining BPM

BUSINESS PROCESS MANAGEMENT includes concepts, methods, and techniques to support the design, administration, configuration, enactment, and analysis of business processes (adopted from [Weske 07]).

2.1.1 *Recent History of Business Process Improvement*

origins of BPM

The emergence of the idea of attempting to improve the performance of business enterprises by looking deeper into labor patterns, and analyzing the interactions of the participants of an assignment, is a relatively recent phenomenon. The approach became popular in the early 1990s due to economic pressure on corporations to increase productivity or to otherwise enhance their efficiency.

Progress in information technology in the late 1980s was already leading to increased efforts among industrial engineers and business analysts to utilize the newly emerging tools and techniques for improving enterprise structures — an occupation which is nowadays known as *process engineering*.[1]

IT and process improvement

Thomas Davenport and James Short explore in detail, in their 1990 article on new paradigms in industrial engineering, the relationship between information technology and business process redesign (see [Davenport 90]). The term *information technology* is defined by the authors as *"the capabilities offered by computers, software applications, and telecommunications"* whereas *business process redesign* is depicted as *"the analysis and design of work flows and processes within an organization"*.

Shortly after the publication of his article with Short, Davenport writes in a comprehensive account on *process innovation* (see [Davenport 93]) the following enthusiastic lines:

> *Today firms must seek not fractional, but multiplicative levels of improvement — 10× rather than 10%. Such radical levels of change require powerful new tools that will facilitate the fundamental redesign of work.*

After having observed that it became common practice in many corporations to utilize these tools on a single dimension, i.e., concentrating solely on reducing costs, he emphasized in his more recent publications that

> *"[...] traditional, engineering-based approaches to knowledge work are incompatible with the autonomy and work approaches of many knowledge workers."*

Therefore, he suggests

> *" a variety of alternative process-oriented approaches to knowledge work [with emphasis] on differentiating among*

1 The term *process engineering* denotes a rather interdisciplinary profession that incorporates a variety of occupational groups, ranging from business administrators to systems analysts and information technology specialists. Process analysis and optimization in an industrial context has, however, been a traditional domain of *industrial engineering*. Nowadays, the two career terms are often used interchangeably.

> *different types of knowledge work and applying process interventions that are more behaviorally sensitive."*
> (quoted from [Davenport 10])

misuse of process improvement methods

Previously having expressed his concern on the misuse of process improvement principles for justifying massive lay-offs, Davenport refers to the methodology, or rather to misinterpretations thereof, critically as a *"fad"* that ignores the human dimension of organizations (see [Davenport 95]).

The years following the pioneering literature on *process improvement* — most of which were from an engineering perspective — brought a myriad of academic articles, technical reports and business research communications addressing the subject matter. A significant portion of these publications reflected on the methodologies of *business reengineering*, which is introduced in the following section.

2.1.2 *Business Reengineering*

Among the many systematic contributions to methods for invoking and sustaining corporate progress, accompanied by a growing pool of research literature on business process improvement, an extraordinarily influential discourse on the subject can be found in a frequently cited book by Michael Hammer and James Champy, published in 1993 (1st edition, [Hammer 93]).[2] The authors often receive credit for having coined the term *reengineering*, which they define in the following manner:

reengineering is the fundamental rethinking and radical redesign of business processes to achieve dramatic improvements in critical, contemporary measures of performance, such as cost, quality, service, and speed.

Hammer and Champy emphasize the significance of considering structural alterations in the process itself, rather than merely automating existing activity patterns, relating to several case studies in an enterprise context, as discussed in [Hammer 90]. Among a rich variety of tools and methods, modern information technology (IT) is by and large appreciated by the authors as an "*enabler*" in reengineering, being vastly important to improving processes and restructuring organizations.

However, organizations are warned against "*thinking that technology is the only essential element in reengineering*". Accelerating a business process or any corporate operation as it is by exploiting

2 In a more recent reprint of the book, the authors discuss the influence of their work on contemporary business process engineering since the release of the first edition. They also introduce new enterprise case studies that verify the significance of business reengineering (see [Hammer 03]).

computational methods, without prior structural optimization, i.e.,

> "[...] *merely overlaying new technology on old ways of doing business* [...] *allowing us to make worse decisions sooner*".

This widespread type of misguided approach to reengineering is referred by Hammer and Champy, humorously, as "*paving cowpaths*".

reengineering vs. optimization

Another commonly used term in the context of reengineering is *business process optimization*. Although the expression is often regarded in business jargon as synonymous to *business reengineering*, there is a formal distinction between the two approaches. Strictly speaking, the term *business process optimization* is attributed to rule-based methods that are applied to improve formalized process models, whereas reengineering generally refers to overall restructuring efforts in an organization, which are often conducted in a radical manner.

revolution vs. evolution

Hence, Business Process Optimization (BPO) should not be regarded as a paradigm in its own right, but a set of methods and tools for assisting not only *revolutionary* approaches, such as business reengineering, but contrasting *evolutionary* concepts like *continuous improvement* as well. Section 2.3 dwells deeper into the theoretical foundations of these optimization methods and their applications.

Business processes are traditionally carried out manually in accordance with regulations and guidelines established in an organizations managing instances. How well the distinct constituents of an organization comply with the procedures already installed depends highly on the knowledge level and skills of the participants, i.e., of the employees, and on their degree of conformity with rules and standards, as noted in [Weske 01].

Weske further states that "*enterprises can achieve additional benefits if they use software systems for coordinating the activities involved in business processes*", suggesting that information technologies can provide an improved means of reinforcing operational compliance. Based on this notion, the following definition can be adopted:

BUSINESS PROCESS MANAGEMENT SYSTEM A *business process management system* is a generic software system that is driven by explicit process representations to coordinate the enactment of business processes.
(adopted from [Weske 07])

Having defined the concepts fundamental to BPM, it is appropriate to emphasize that the terms *business process* and *workflow*

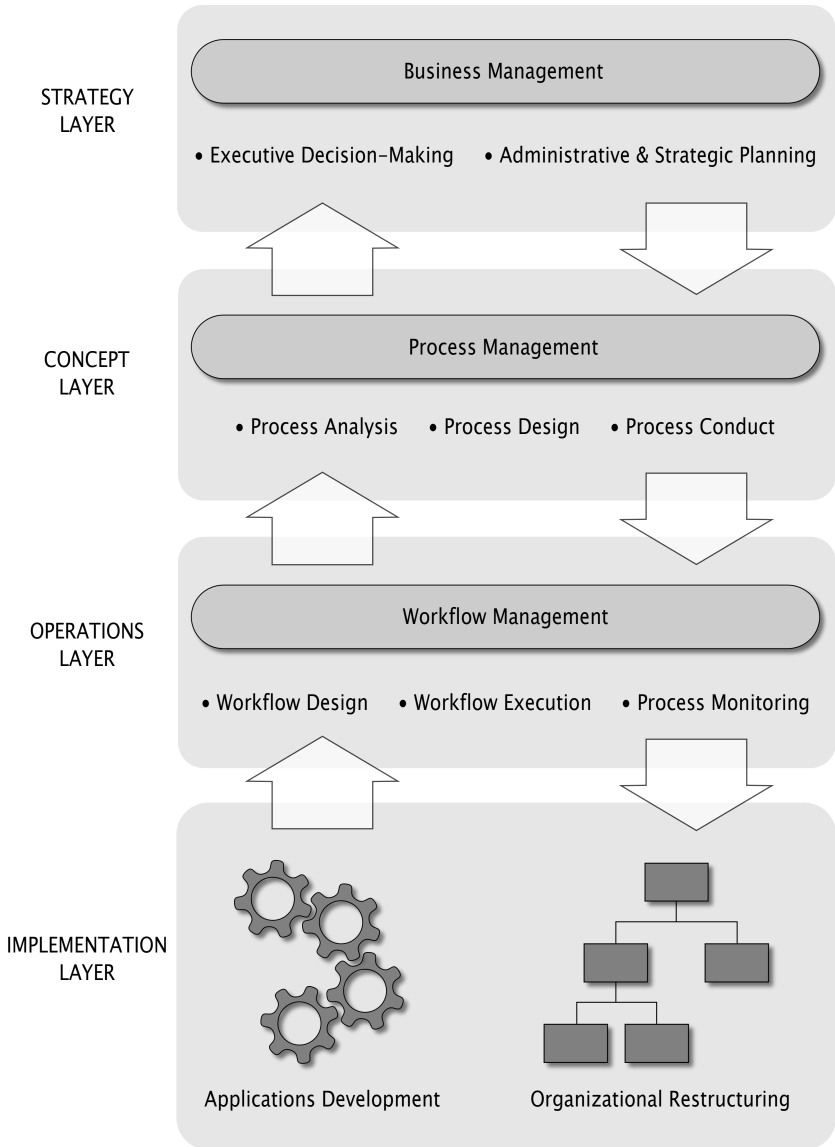


Figure 2: Illustration of the typical stages of business process development, based on a model of *management layers*, adapted from [Gadatsch 10]. Interaction among the layers is ideally bidirectional. Information flow through all layers in both directions enables an organization to evaluate the process framework and restructure parts of the system, where required.

are placed into a specific hierarchical context in the present thesis. It becomes clear how these notions fit into an organizations anatomy, when they are contemplated as integrated methodologies applied at various functional levels.

2.1.3 *Process Development*

The emergence of processes in an organization follows a particular series of development patterns. These motives comprise an own field of study, traditionally in the domain of *organization theory*, as described in [Daft 09]. This section provides a brief account of common forms of process-oriented organizational structures and the typical stages of process management therein.

process design stages

The description of a business process begins at the *strategy layer* of an organizational archetype according to the layer model in [Gadatsch 10]. This is where the decision-making in a typical corporation takes place, usually as part of a top-down managerial hierarchy. Figure 2 illustrates the subsequent steps that contribute to the realization of management decisions.

The main strategic question at this stage concerns which processes define the core functionality of an organization, and which ones can be outsourced to be carried out by external instances. Having defined the range of internal activities, strategic management determines how the processes should be structured and what methods are to be applied. Solving such decision problems typically run through four distinct phases, as distinguished in [van der Aalst 04]:

1. *Definition* involves establishing exactly what the problem is and, in particular, within what scope a solution to it must be found. Drawing up optimization criteria often forms part of this phase.

2. *Creation* involves formulating one or more solutions which fall within the scope defined, or which satisfy an optimization criterion.

3. *Evaluation* involves assessing different solutions, for instance by multi-criteria analysis.

4. *Selection* involves choosing one solution that works, which is in turn actually implemented.

Once a strategic road map is laid out in the strategy layer, it is passed on to the subjacent *concept layer*. This specific development stage incorporates *tactical management*, which takes on the task of formalizing the asserted processes and molding strategic requirements into a standardized process model. This mission requires that the process be *analyzed*, resulting in a structural

management level	decision type
Strategic	process design and resource types
Tactical	capacity planning and budgeting
Operational	resource assignment
Real-Time	equipment control

Table 1: Decision types in four interleaved process management levels, based on [van der Aalst 04].

outline based on real-life constraints and other strategic considerations. The indicated outline is then *modeled* in accordance with a technical norm assigned in advance. Section 3.1 explains how a business process is systematically modeled, and presents commonly applied rules and standards.

Finally, the business process model is *implemented*. This final stage, i.e., *process deployment*, can be preceded by several redesign cycles depending on the complexity of the targeted business process. In the classical fourfold division of process management levels, the implementation layer is referred to as the *real-time management level*, as referred in [van der Aalst 04].

At this critical position in the chain of command, there is a transition of perspective from the *administration tier* to the *technical infrastructure* embodied in the same organization. As soon as the standard model of a business process is available, it can be assigned to operating instances in the next layer. Gadatsch calls this domain the *operations layer*.

Depending on the readily available information technology infrastructure, the set of normalized business process models can now be, partially or in whole, translated into automated workflows. Thus, the operational layer incorporates both a partly conceptional *workflow management* and the necessary implementation technology.

impact of decisions

Each level of process management incorporates specific decision types, as summarized in Table 1. These decision types differ from one another, with regard to their variable scope on the time scale and their expected impact on operation results. Due to the bidirectional nature of interaction among subjacent management layers, it can generally be assumed that process exceptions or deviations form business rules in one layer, are bound to have operational consequences for all other planning stages, and have to be dealt with in all layers accordingly.

Decision-making at the lowest level, i.e., in the real-time management domain, e.g., by automated tasks or supporting computer applications, takes place very frequently, ranging from microseconds to hours. The outcomes of these decisions generally have short term effects. Nevertheless, their financial impact can be tremendous, as discussed in [van der Aalst 04].

At the top end of the process management structure, however, strategic decisions are taken once every few years, but they do have a much wider scope, with notable influence on organizational performance over long periods of time, typically for several years.

It is, therefore, essential to put substantial effort into conceptual planning in order to achieve a high degree of process optimality in the modeling stage. The outcomes should then be carefully monitored during actual operations, possibly leading to modifications in the process structure for subsequent cycles. The following section discusses several methods for improving process schemes with respect to performance measures in relevant operational settings.

2.2 PROCESSES AND WORKFLOWS

In many instances of organized entities in an enterprise setting, it is a specified requirement to document frequently repeated operations and transactions. One of several purposes for this common practice is the necessity to secure and transfer institutional know-how. Another important motive arises from the requirement to attain sufficient compliance with legal obligations due to contemporary jurisdiction.

Moreover, it is likewise justifiable to keep precise records of corporate activities in order to support management by reinforcing decision-making tasks. In other words, process documentation and analysis are prerequisites to maintaining a healthy organization and possibly accomplishing significant structural improvements.

Before elaborating further on the matter, it is appropriate to become familiar with conventional terminology related to the major subject at hand. The following section is an effort to narrow down the vast variety of definitions related to BPM, aiming to extract the most beneficial occurrences in contemporary scientific and technical literature. Descriptions of the more methodical concepts related to process modeling, workflow design and simulation, are contemplated in Chapter 3.

2.2.1 *Definitions*

A *process* (also called *procedure*) can simply be defined as a series of activities that generate a certain result, i.e., produce an output, given a particular input, as defined in [Hammer 03]. A process consists of a number of *tasks* that need to be carried out and a set of *conditions* that determine the order of the tasks, where a *task* is a logical unit of work executed as a single whole by one resource (see [van der Aalst 04]).

The smallest basic unit of work in this context is often referred to as an *atomic process*. An atomic process, therefore, denotes a single step performed by one resource. A *resource* can be any combination of devices or supporting means, such as machines and persons, that carry out specific tasks. A person who is assigned a task is regarded as a process resource, and is sometimes referred to as the *contractor*.

process resources

The Workflow Management Coalition (WfMC)[3] specifies a rather technical definition of the term *process*, which is somewhat condensed and recursive, nevertheless fairly precise:

PROCESS A *process [model]* is a formalized view of a business process, represented as a co-ordinated (parallel and/or serial) set of process activities that are connected in order to achieve a common goal.
(see [WfMC 99])

A typical example of a process as defined by the WfMC is illustrated in Figure 3. The same specification additionally suggests synonyms for the term defined above, which may be preferred in different contexts:

- Activity Network
- Directed Graph
- Petri Net
- Model
- Instruction Sheet

synonyms for "Formalized Process"

SUB-PROCESS (also *Subflow*) A process that is enacted or called from another (initiating) process (or sub-process), and which forms part of the overall (initiating) process. Multiple levels of sub-process may be supported.

The above definition of the concept *sub-process* implies that this particular type of process can be formalized in its own process representation, and may include parameters passed on its initiation and completion. A sub-process is useful for defining reusable

3 Founded in 1993, the Workflow Management Coalition (WfMC) is a global organization of adopters, developers, consultants, analysts, as well as university and research groups engaged in workflow and BPM. The WfMC creates and contributes to process related standards, educates the market on related issues, and is [according to the organizations self-portrayal] the only standards organization that concentrates purely on process. Further details on the coalition and its publications can be found at `http://www.wfmc.org`

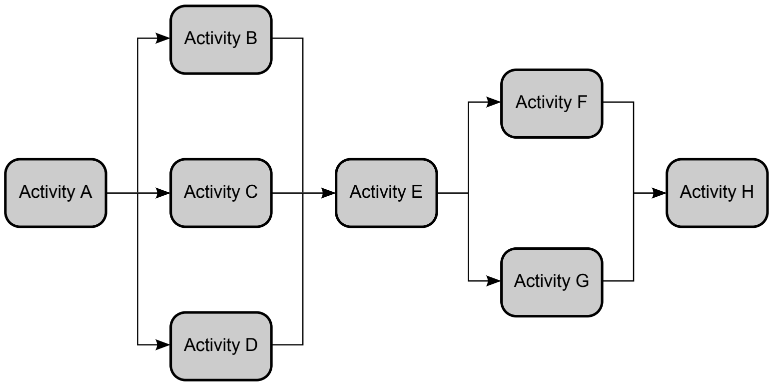


Figure 3: A formalized process with eight activities. The diagram presents serial operations as well as parallel activities, which may or may not be linked logically. The illustrated model does not actually conform to any particular standard or modeling specification. The control structure of the displayed process is highly underspecified, i.e., decisions that would possibly lead to varying sets of activity paths, are not explicitly annotated in this exemplary graph.
(adopted from [WfMC 99, Workflow Terminology and Glossary])

components within other processes, and can be integrated into a nested or chained composition.

WORKFLOW (also *Case Management*) The automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules.

A workflow, therefore, is the description of the steps and rules involved in a process. Roles and interactions of the participants in a process model are characterized in the corresponding workflow model, whereas the *process model* is meant to describe the structure of a business process in the real world. A *workflow model*, on the other hand, is commonly defined as a representation of the automated parts of a business process, i.e., a model of operations that are run on computer systems, as described in [Leymann 00].

Weske suggests a broader definition for processes in a business environment:

BUSINESS PROCESS A *business process* consists of a set of activities that are performed in coordination in an organizational and technical environment. These activities jointly realize a business goal. Each business process is enacted by a single organization, but it may interact with business processes performed by other organizations.
(see [Weske 07])

In the context of analyzing and optimizing processes by employing computational methods, the terms *business process* and *workflow* may fairly be regarded as interchangeable. The present thesis, however, does make a technical distinction between the two concepts, as previously described in Section 2.1.

2.2.2 *Standards*

Business processes are regarded as *value-adding* activities that organizations execute to accomplish a particular objective for a particular customer, whereas the highly diverse nature of processes makes their management and (re)design a cumbersome and complex challenge, as depicted in [Davenport 05, cited in [Rosenkranz 10]].

It is, therefore, highly important to rationalize and systemize the practice of process (re)design by adopting appropriate standards. Such standards are not only meant to enable organizations to analyze their own business capabilities and improve them, but also to help process analysts compare external service providers and evaluate the costs vs. the benefits of, e.g., outsourcing.

The activity of creating a *standard* is often called *standardization*. A formal definition for the concept of *standardization* is derived in [de Vries 97, cited in [Blind 03]] and explicated further in [Slob 02] by comparing existing definitions in the technical literature, along with inquiries on approaches of the term's usage in practice:

STANDARDIZATION is generally defined as the *"activity of establishing and recording a limited set of solutions to actual or potential problems directed at benefits for the party or parties involved, balancing their needs and intending and expecting that these solutions will be repeatedly or continuously used during a certain period by a substantial number of parties for whom they are meant"*.

Nevertheless, it is essential to note, that not all standards are equally applicable in any context, nor is every standard commonly accepted. When discussing about *standards*, it must be taken into consideration, for the purpose of clarity, that there are basically two types of specification modes, as distinguished in [Leymann 10]:

DE FACTO STANDARD A *de facto standard* is defined by a single vendor or small group of vendors, and by the joint market share of these vendors, the specification becomes a standard within that market segment.

DE JURE STANDARD A *de jure standard* is defined by a public, i.e., official, standardization body consisting of many dif-

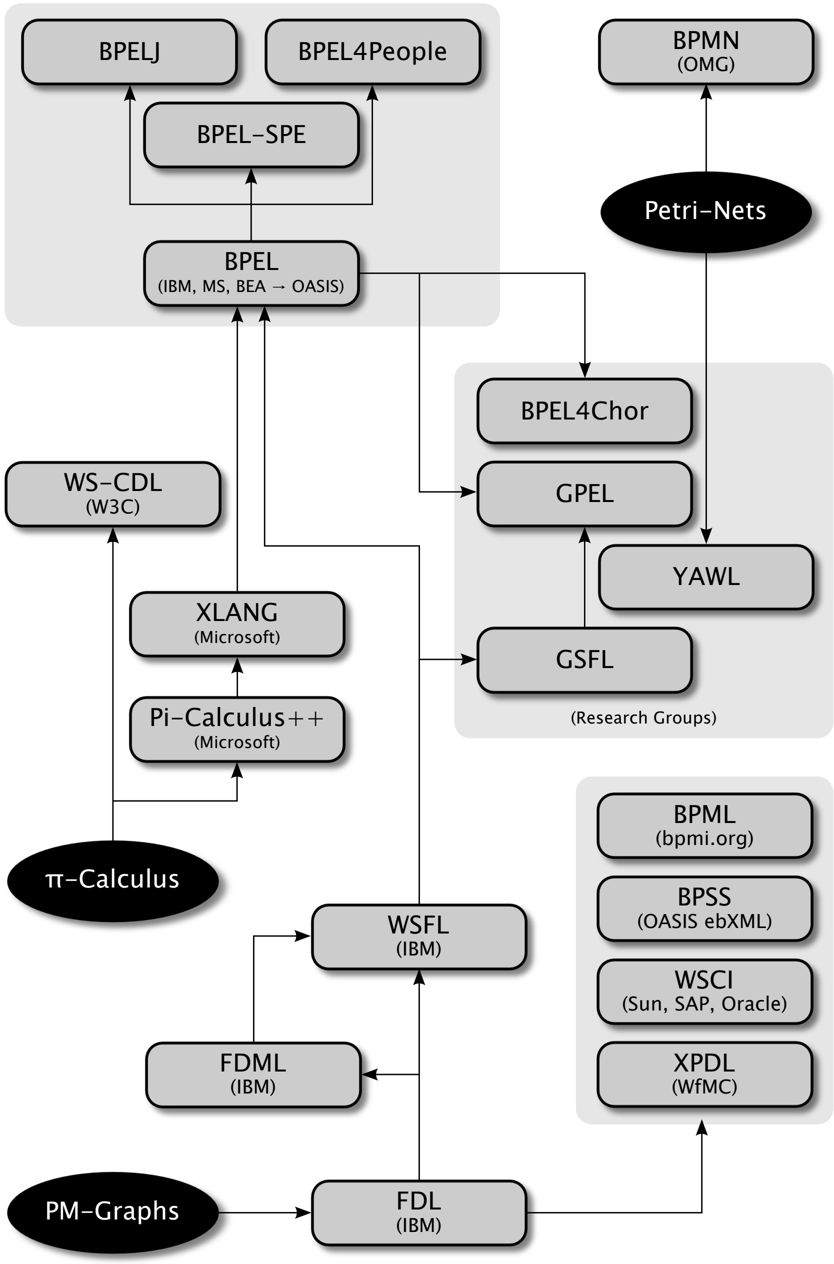


Figure 4: Relations between some language standards for defining business processes and their mutual influence patterns, adopted from [Leymann 10]. IBM®, Microsoft® (MS), Sun® Microsystems, SAP®, Oracle®, and BEA® Systems are registered trademarks.

ferent vendors and interest groups who jointly work on a specification, and release it as a standard, based on majority agreement.

Leymann emphasizes, however, that no general conclusion can be drawn as to whether a certain standard can be regarded as being routinely subject to reliable support, in terms of its implementation or employment, merely based on the circumstance that the standard in question happens to be *de jure* by definition. The relevance of a standard, he therefore argues, should be determined by the number of vendors supporting it.

maturity of standards

Moreover, standards are not always fully specified from the very beginning of their existence, but rather go through varying maturity levels, as depicted in [OMG 08]. A common framework for standardization efforts is suggested in [Rosenkranz 10]. It is argued with the help of detailed case studies in standardization processes, that a widely accepted framework would simplify the implementation of business processes and allow organizations to reach organizational excellence by defined measures.

These typical patterns of emergence can also be observed in the development of BPM standards, whereas commercial and scientific influences display varying effects over time. The origins and evolution of the most widespread BPM description language standards are illustrated in Figure 4.

The implementation techniques employed in this thesis for the definition an execution of business processes, depend predominantly on the usage of the Business Process Execution Language (BPEL), a standard currently specified and maintained by the Organization for the Advancement of Structured Information Standards (OASIS)[4].

2.3 Optimization Techniques

The principal concern of the present work is optimality in business process design. The systematic development of a business process typically amounts to one or several of the following procedures:

- An existing and operational business process is *formalized* by capturing real-life enterprise activities, e.g., for documentation purposes or for further analysis of the process itself.

4 OASIS was founded in 1993 under the name *SGML Open* as a consortium of vendors and users devoted to developing guidelines for interoperability among products that support the Standard Generalized Markup Language (SGML). OASIS changed its name in 1998 to reflect an expanded scope of technical work, including the Extensible Markup Language (XML) and other related standards (see [OASIS 11]).

- A business process is *designed* from scratch in order to implement a functional structure that has the objective of fulfilling a specific business commitment.
- An existing business process model is *redesigned* and transformed into a modified or entirely renewed process model, aiming to accomplish improvements in performance after redeployment.
- A functional business process is *optimized dynamically*, i.e., while in actual operation, in accordance with particular performance indicators.

optimization stages

The business process optimization platform evaluated in this document is one that performs dynamic optimization. The procedures employed for improving a business process, specifically on the dBOP during execution, and generally for dynamic process optimization, involve the following three stages, as distinguished in [Niedermann 10]:

1. *Design:* The initial stage consists of modeling a formalized inventory of the process, prior to its execution. The primary objective in the design stage is an optimal model based on general *best practice* methods and patterns in analogous processes.
2. *Execution:* In this particular processing phase, optimization is essentially about making the most advantageous decisions within a given process structure during its actual operation, e.g., allocating resources optimally.
3. *Analysis:* Once the outcome of a process is put into quantifiable terms, the process design can be restructured and ideally transformed into an optimal model in agreement with valid business goals.

process-aware systems

These stages roughly correspond to the phases in the life cycle of a Process-Aware Information System (PAIS), as described in [Dumas 05]. A PAIS is defined as

> *[...] a software system that manages and executes operational processes involving people, applications, and/or information sources on the basis of process models.*

Similar to the aforementioned processing phases, design or redesign is based on a requirement analysis (see Figure 5). The consecutive implementation phase typically involves configuring a generic software infrastructure, which hosts the operational process model. After having carried out the process in the enactment phase, the data obtained by analyzing the operation can

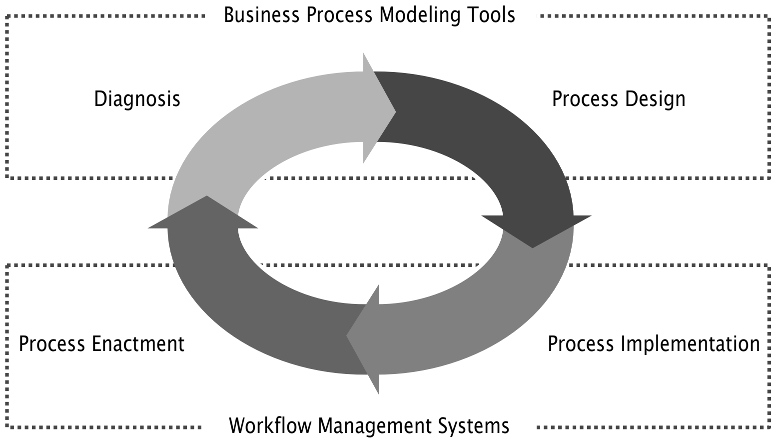


Figure 5: The PAIS life cycle, as described in [Weske 04] and [Dumas 05].

be utilized for identifying possible problems and finding aspects that are prone to improvement.

Accordingly, the term *process-awareness* indicates a shift of attention from data storage and retrieval to information systems driven by process models. Benchmarking the latter, which display a higher magnitude of complexity by nature, requires a different approach than solely gauging the performance of the technical infrastructure. The following section introduces a framework for evaluating the effectiveness of a business process, and illuminates relevant performance characteristics.

2.3.1 *Performance Indicators*

Measuring the consequences of redesigning a process is essential to understanding the correlation between business goals and the applied optimization methods. Figure 6 illustrates a conceptual framework that distinguishes four principal performance dimensions for measuring process performance: *time*, *cost*, *quality*, and *flexibility*. The effects of optimization operations on the outcome of a business process can be evaluated on the basis of these four key dimensions.

measuring business performance

Process improvement schemes typically aim to handle business tasks in less time, draw on less resources and reduce costs, increase the quality of products and services, and enhance the organizations ability to adapt to changing business demands and varying market conditions.

However, adjustments in the arrangement of a process often influence not only the targeted performance dimension, but may induce responses in all other dimensions as well, possibly leading

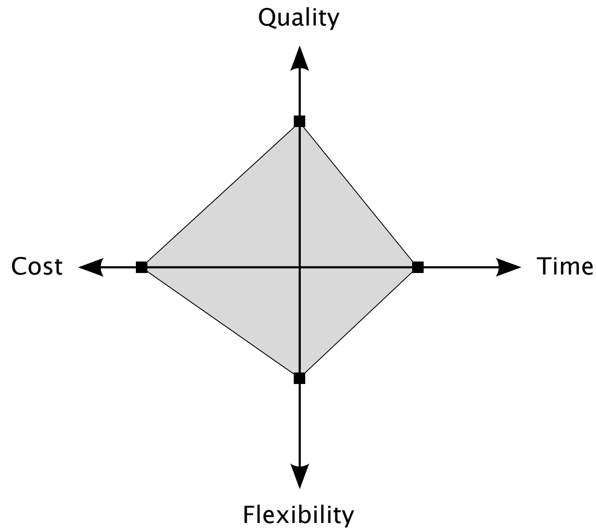


Figure 6: The Devil's Quadrangle, adopted from [Reijers 05a, originally attributed to [Brand 95]].

to undesired afflictions. Implementing a process improvement scheme can, therefore, bring up the necessity to involve trade-offs between conflicting goals. Thus, the name *"devil's quadrangle"* for the four-dimensional performance scale, adopted from Brand and Kolk (see [Brand 95, as cited in [Reijers 05a]]).

The core attributes of these performance indicators, based on [Reijers 05a, Reijers 05b], can be summarized as follows:

A. The *lead time* (or *cycle time*, *throughput time*, etc.) of a process denotes the duration of handling a case from initialization to completion.

   In a typical business process, *service time*, i.e., active time spent by the participants on actually handling a particular case, contributes only a fraction of the total duration. The remainder commonly consists of *wait time* and *queue time*, in some cases at proportions of up to 95% of total lead time.

   Redesign generally intends to reduce absolute processing time, either on *average*, or with respect to a *maximum* value for a series of process instances. An alternative approach is to define a tolerable *interval* for lead time, in order to attain a certain standard of service reliability.

B. *Process cost* is the financial dimension of performance evaluation. The term *cost* denotes process relevant expenses, and does not explicitly involve accounting factors, such as profit or revenue, although these financial criteria are highly correlated with process costs.

The predominant concern of process redesign with respect to the dimension *cost*, is its reduction. However, a deliberate increase in *operational costs*, e.g., induced by engaging a more specialized work force or by utilizing expensive equipment to enhance productivity, can lead to an improvement in overall process quality, and, as a consequence, increase profits.

c. The *quality* of a business process subsumes two possibly conflicting aspects:

External quality, as perceived by the recipients of products and services, i.e., process quality from the client's perspective, and

Internal quality, as perceived by the internal process actors and resources, i.e., quality of processing conditions from the workers point of view.

Process duration is often rated as a quality trait, meaning that a decrease in processing time is generally perceived by clients as an increase in process quality. The internal perception under the same circumstances may, nevertheless, turn out to be adversely unsatisfactory.

d. *Flexibility* is the ability to react to changes at *run time*, i.e., during process execution, or at *build time*, i.e., while restructuring a process.

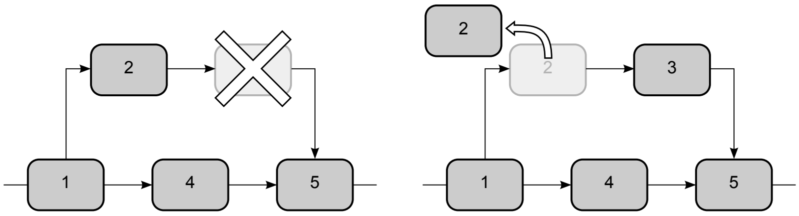
Changes in the process environment can involve new demands by the market or business partners, variations in workload or the allocation of resources, and modifications in the definition of tasks.

2.3.2 *Process Redesign Methods*

One of the most quintessential objectives of process optimization are *acceleration*, i.e., the reduction of overall process duration (*lead time*, as defined in the previous section). Another very common intention is the *improvement* of process quality. Some potential methods for attaining these goals[5] are suggested in [Gadatsch 10, adopted originally from [Bleicher 91]].

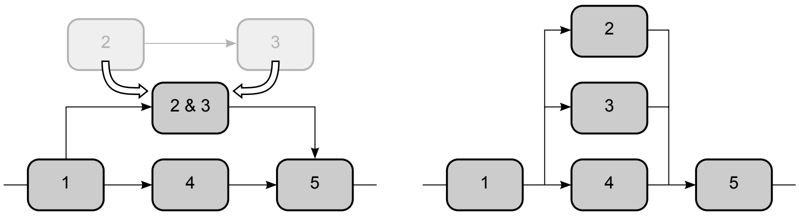
Careful analysis of a business process may occasionally lead to the conclusion that one or several activities in the process structure can be classified as *non-purposeful* with regard to the desired results. In such cases, *elimination* of these activities (see Figure

5 Obviously, any optimization redesign method may influence more than one, sometimes even all of the aforementioned performance indicators, as visualized on the *devil's quadrangle*. Earlier literature on process redesign, however, often merely mentions *process improvement* without always specifying the dimensions in which these improvements occur.



(a) Elimination of an activity

(b) Outsourcing a task to an external instance



(c) Combining several activities into one single task

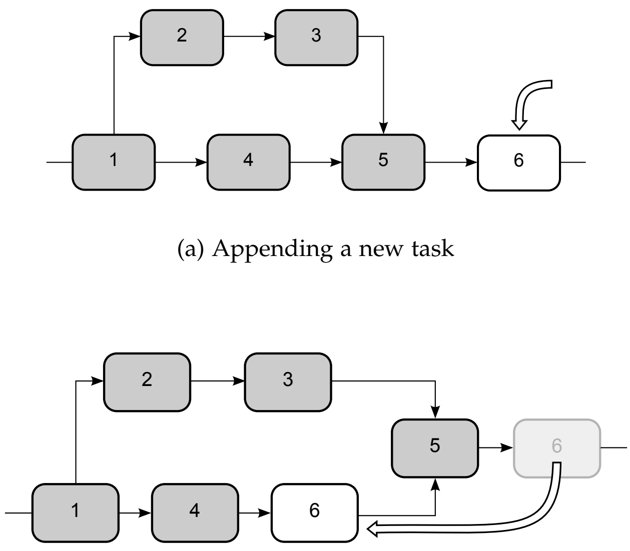
(d) Parallelizing several tasks for simultaneous processing

Figure 7: Common optimization methods, based on [Bleicher 91, cited in [Gadatsch 10], Chapter 1]

7a) can lead to an overall improvement, e.g., shorter duration. Similarly, it may be beneficial to process outcome, if certain tasks are not carried out in the process itself, but rather *outsourced* (see Figure 7b), i.e., delegated to other organizational units or external service providers.

Another commonly employed optimization method is *composition* (see Figure 7c), which consists of combining similar activities or putting together tasks that require similar resources, in order to reduce the effort necessary for setup and initialization of an activity. Combining tasks may result in decreased run-time and may, in some cases, even lead to improved process quality. Furthermore, disjunct activities, i.e., tasks that are not interdependent and that access different resources, can be executed in parallel. *Parallelization* of activities (see Figure 7d) obviously bears the potential to reduce process duration, which would otherwise amount to the total run-time of consecutive activities that are carried out serially in a typical process chain.

If a business process does not yield the desired outcome, it may be necessary to *append* complementary activities (see Figure 8a) which would assure a predefined range of results, or possibly guarantee specific quality standards. Under specific circumstances, it may likewise be convenient to *relocate* (see Figure 8b) an activity, i.e., to change the chronological order of tasks, aiming to optimize the allocation of resources. This method is closely related to a composite displacement technique known as *resequencing* (see Figure 9a).



(a) Appending a new task

(b) Relocating a task

Figure 8: Process restructuring methods based on [Bleicher 91]

The above listed methods and several other redesign techniques can be categorized on the basis of common functional classes. Depending on the performance objectives for a particular business process, the best fitting improvement methods from a class of *"best practices"* can be selected to be taken into consideration. Reijers adopts the following classification [Reijers 05a]:

best practice

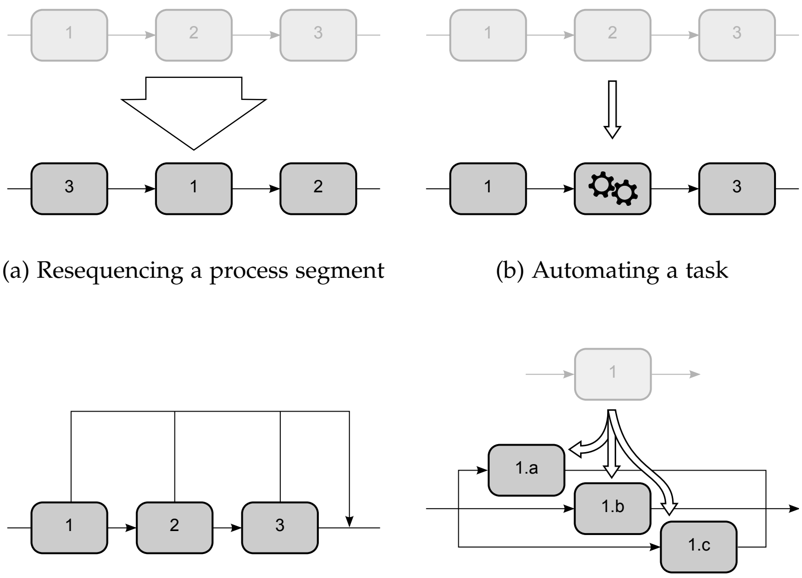
- *Task best practices*, which focus on optimizing single tasks within a business process,
- *Routing best practices*, which try to improve upon the routing structure of a business process,
- *Allocation best practices*, which involve a particular allocation of resources within the business process,
- *Resource best practices*, which focus on the types and quantity of resources,
- *Best practices for external parties*, which try to improve upon the collaboration and communication with the client and third parties,
- *Integral best practices*, which apply to the business process as a whole.

Reijers notes that this classification is to a certain degree arbitrary, and that the classes are not strictly disjunct. Table 2 gives

CLASS	METHODS
Task	Elimination
	Composition
	Automation
Routing	Resequencing
	Knockout
	Control Relocation
	Parallelism
	Triage
Allocation	Case Manager
	Case Assignment
	Customer Teams
	Flexible Assignment
	Resource Centralization
	Split Responsibilities
Resource	Numerical Involvement
	Extra Resources
	Specialist-Generalist
	Empower
External Parties	Integration
	Outsourcing
	Interfacing
	Contact Reduction
	Buffering
	Trusted Party
Process-Integral	Case Types
	Technology
	Exception
	Case-Based Work

Table 2: Overview of redesign best practices, based on [Reijers 05a]. Most methods can actually be assigned to more than one class, depending on the redesign objectives for a specific case.

an overview of the various redesign methods and the classes they are assigned to.



(a) Resequencing a process segment

(b) Automating a task

(c) Knockout

(d) Triage

Figure 9: Routing best practices for process redesign based on [Reijers 05a]

Hence, with regard to the scope of the present thesis, the following semi-formal definition can be adopted to confine the notion of optimization to a particular domain, and to clarify its purpose:

PROCESS OPTIMIZATION is the transformation of a formalized process, with the objective of achieving conformity to an explicit set of performance goals.

The immediate relationship between design methods and the intended improvements, formulated as business goals, is to be taken into consideration in all optimization efforts. One particular repository, which accommodates a comprehensive documentation of these relationships and their consequences on process redesign by means of an extensive formalism, is the *optimization pattern catalogue*, described in the following section.

2.3.3 *Optimization Pattern Catalogue*

The deep Business Optimization Platform (dBOP), as described in Section 1.3.1.2, utilizes a formalized collection of frequent optimization patterns, such as the techniques listed in the previous section, mostly derived from verbal descriptions in research literature on process design, through interviews with production engineers and BPM consultants, as well as by means of extensive analysis of successful BPM projects [Niedermann 11b].

The optimization logic is contained in the *pattern catalogue*, in which patterns are defined using a *process meta-model* developed by Niedermann et al., based on a process graph model by Leymann and Roller [Leymann 00]. A predefined *goal function*, in accordance with business objectives as described in Section 2.3.1, sets the reference frame for the intended optimization measures. The pattern catalogue describes the relationship between patterns and the goal function, and between the patterns themselves. Formal methods for pattern detection and the application logic for each pattern is likewise specified in the catalogue.

goal function

However, classification of optimization patterns within the catalogue follows a slightly different approach than those described in the preceding section. Patterns in the catalogue are distinguished by the following set of criteria, as discussed in [Niedermann 10]:

- *Optimization Scope* is the effective application range of a pattern, reaching from a single process activity to an entire process.
- The *Degree of Automation* indicates whether a pattern is applied automatically, manually, or semi-automatically involving interaction with a process analyst.
- The *Application Stage* denotes the phase in which a pattern is applied (see *optimization stages* on page 24).
- *Data Requirements* determine, on the one hand, whether the pattern in question requires model data or instance data, and on the other hand, whether the requirement is limited to process data, operational data, or both.

Figure 10 illustrates an excerpt of the pattern catalogue that was classified based on the above mentioned criteria. These patterns are an integral part of the *optimization layer* in dBOP. Every pattern is described in a formal manner specified in a template that embodies the following three elements:

1. *Specification* Lists the meta-information of the pattern that is used by the optimizer with respect to the target parameters of the goal function. The specification helps to ensure that the analyst employs the pattern properly, e.g., by ensuring that only appropriate patterns are considered and that they are executed in the right order.
2. *Detection* Here lies the algorithm that specifies how instances of the pattern are retrieved. It returns a set of pattern instances identified as potential optimizations, and interacts with the business analysts, where necessary.

3. *Application* This segment embodies the algorithm that describes the transformation logic to be applied when a pattern instance is confirmed, hence enabling the restructuring of a process to implement the desired improvements.

2.4 THE ROLE OF SCENARIOS AND SIMULATION

This section discusses the role of scenarios in the evaluation of software architectures, and provides an overview of the principles behind business process simulation.

2.4.1 *Case Studies as Simulation Objects*

Using case studies for research purposes, although very challenging, is quite commonplace not only in the social sciences, but also increasingly in business studies, economics, as well as computer sciences. Like in most other methods of doing scientific research, the primary goal is to collect, evaluate and report data. A good case study design is, therefore, essential to conducting sound research in accordance with scientific criteria.

Robert Yin suggests in [Yin 02] — from a social scientist's perspective — that *"case studies are [in general] the preferred strategy when 'how' or 'why' questions are being posed, when the investigator has little control over events, and when the focus is on contemporary phenomenon within some real-life context"*. Further, he urges investigators to exercise great care in designing and conducting case studies to overcome the traditional criticisms of the method.

Unfortunately, this *explanatory* approach to case studies is suitable only to a limited extent in its applicability to process analysis. To be specific, the investigation of process structures that have probably been modeled by the investigator himself, is unalike a setting in which the object of scrutiny is an autonomous system, such as a social group that is being studied. Yin does, however, emphasize that are two complementary research types: *exploratory* and *descriptive* case studies. Thus, modeling a business scenario based on a real-life case, including all assessment activities leading to the model, can certainly be regarded as *descriptive*, whereas investigating the particulars of this model, e.g., through simulations in which the boundary conditions and other internal parameters are varied in a controlled analysis, the case study can be considered to have a more *exploratory* nature.

2.4.2 *Limitations of Simulation-Based Process Evaluation*

Designing an accurate process model, with the highest possible resemblance to the original operations it mimics, is the fundamental

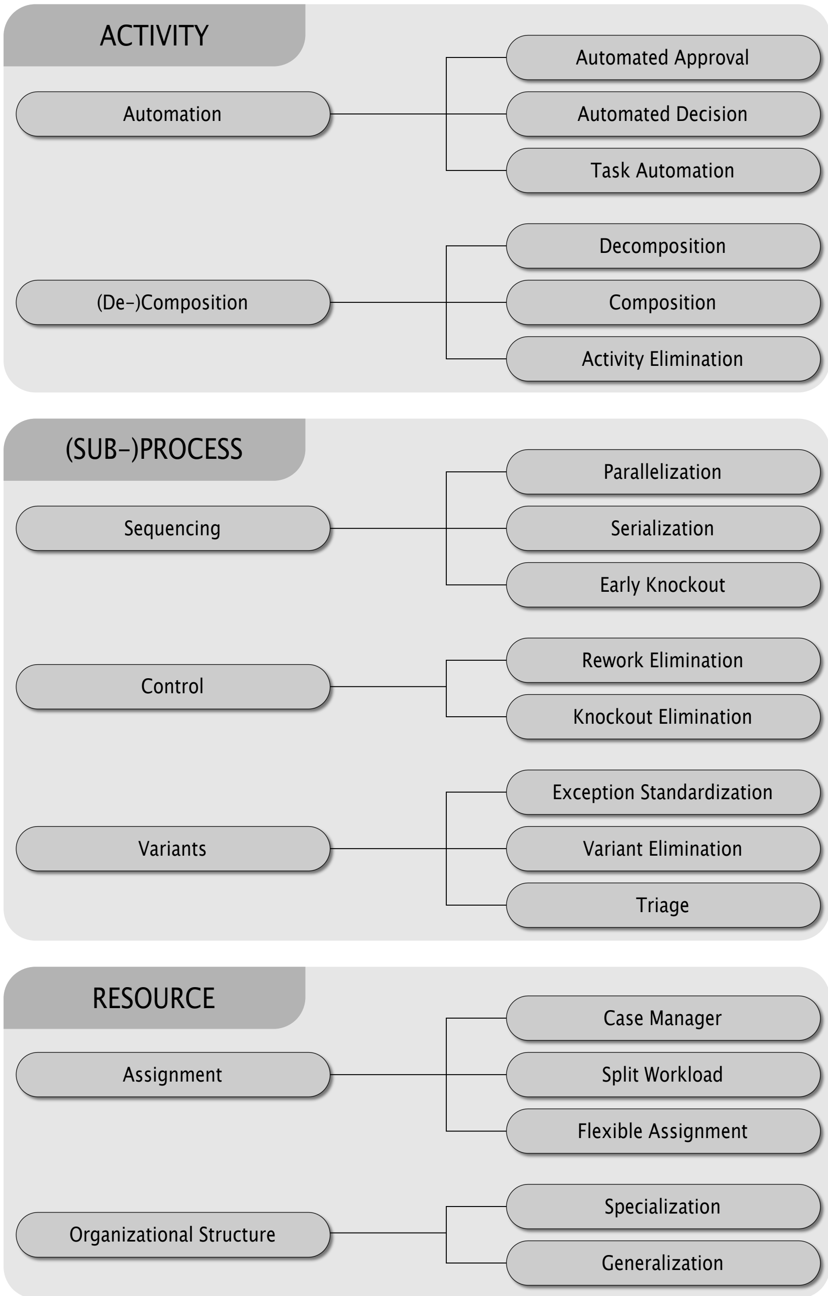


Figure 10: An excerpt from the optimization pattern catalogue, as employed in the deep Business Optimization Platform (dBOP), with typical patterns for a selection of optimization categories in three different scope classes: *Activity*, *Process*, and *Resource* (based on [Niedermann 11b]).

prerequisite to achieving useful conclusions through simulation-based process evaluation. There are, however, limits as to how precise the behavior of a process and its resources can be imitated using computational methods.

In a recent account on *business process simulation*, it is argued, that, although widely applied, contemporary simulation approaches rarely deliver the desired results. Inadequate reliability of simulation results is assumed to be mostly due to limitations linked to a set of common *pitfalls*, as enlisted in [van der Aalst 10a]:

1. Focus on design rather than operational decision making,
2. Modeling from scratch rather than using existing artifacts, and
3. Incorrect modeling of resources.

As argued in [Modarres 06], the value of simulation and modeling technology tends to be contingent on creating models that can offer a systematic and well defined way of representing firm's structure and business processes. However, complexity in hierarchical business processes and the introduction of random changes within such business processes can create operational instability in the process models, hence leading to insufficient predictability via simulation. These random elements are assumed to intrude any given business process model in three possible ways:

1. Through consideration of interarrival times of new entities into the system,
2. By routing items or resources to different processes or subprocesses, and
3. Due to inaccurate modeling of error margin in flow times for individual process activities.

Despite its drawbacks, automatic analysis and simulation methods are considered to be highly important to providing valuable insights into business processes. The trend in software systems design is to provide methods powerful enough to capture the rich reality of business systems, putting the focus on the behavior and interactions of users, as suggested in [Barjis 08].

Part II

METHOD

3 TECHNICAL FRAMEWORK

The technical foundations for combining IT capabilities with BPM methods were not always as readily available and as rich in diversity, as they are today. The evolution of relevant technologies can be regarded as consisting of a number of concentric layers, resembling the hierarchic configuration of a typical information system. Figure 11 shows an abstraction of how information systems developed in a historical reflection, as described in [Dumas 05]. The innermost layer represents the first stage of IT development in the 1960s, where information systems comprised merely small operating systems with very limited functionality.

Only after the emergence of *generic applications* in the following decades, contained in the second IS layer, organizations could regularly employ common computational tools, such as text editors, spreadsheets and database management systems (DBMS). A further layer containing *domain-specific applications* provided specialized business instruments, like decision support systems (DSS) and computer-aided design (CAD) tools.

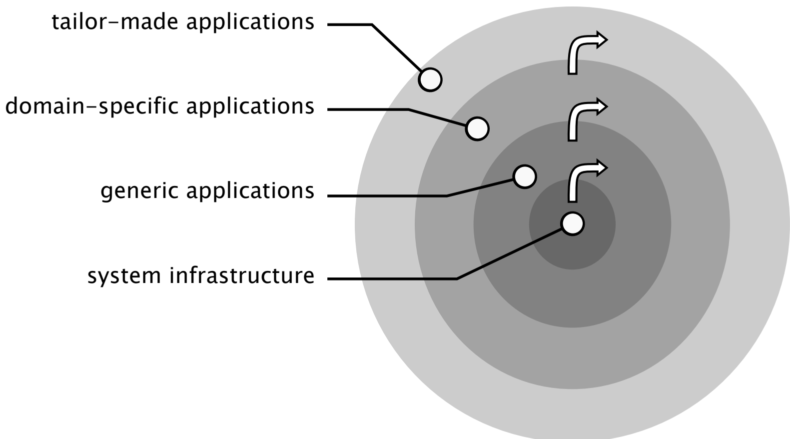


Figure 11: IT trends relevant to business process management, as described in [Dumas 05].

from data orientation to process orientation

Following the persistence of *data-driven* architectures, further reinforced by the emergence of a fourth IS layer embodying *tailor-made applications* for enterprise-specific purposes, a trend shift towards *process-driven* systems arose in the early 1990s, as already discussed in Section 2.1.1. Today there is an increasing number of process-aware information system (PAIS; see Section 2.3

for a detailed description) architectures based on a variety of technological infrastructure types.

The choice of appropriate technologies and methods in the context of enterprise application integration (EAI) usually adheres to certain patterns of user acceptance and organizational consent, such as those predicted by the technology acceptance model (TAM).[1] The present chapter does not dwell deeper into the specific aspects of the selection criteria, but briefly presents the methods and standards utilized in constructing business process simulations for the present thesis.

The following sections provide an overview of the particular techniques employed in this study to design processes, provide process data, orchestrate process services, and to carry out simulations.

3.1 PROCESS DESIGN WITH BPMN

A wide variety of approaches and notations have been developed, and are being currently used for BPM and workflow design. Building on the basis provided by a number of predecessing language specifications, the Business Process Modeling Notation (BPMN) has been promoted and suggested as a standard (see Section 2.2.2) over the last years. BPMN has been devised to meet a diverse set of needs, predominantly to create process models that are easily comprehensible for humans, and nevertheless remain machine readable, for sense-making, quality management, simulation, and execution.

A very common approach to process execution is capturing process definitions using BPMN, and mapping these models to the Business Process Execution Language (BPEL), as asserted in [Aagesen 10]. Particulars of mapping BPMN models to BPEL are discussed in the OMG standard documents for BPMN (see, e.g., BPMN Version 2.0 [OMG 11]).

3.1.1 *Language Elements and Standards*

BPMN provides an extensive set of language constructs, which can be used to satisfy a wide range of process modeling demands. The aforementioned official language standard defines three process model types, or sub-classes:

1. *Descriptive* models contain visible elements and attributes used in high-level modeling.

1 For a comprehensive account of the third and current interpretation, TAM3, refer to [Venkatesh 08].

2. *Analytic* models include the descriptive type, and extend it further to include additional elements based on BPMN training experience and an analysis of user-patterns.

3. The *common executable* sub-class focuses on elements needed to describe executable process models.

The present work makes use of a very small number of elements, in order to prevent complexity issues in the process simulations by keeping the process models topologically as simple as possible. Table 3 and Table 4 illustrate event and connection elements actually used for modeling the business scenarios simulated in this study. An interesting account on general usage of BPMN in an enterprise context can be found in [zur Muehlen 08], where the authors assert that most analysts use a small subset of elements.

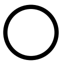
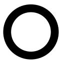
SYMBOL	NAME	FUNCTION
	Start	Process start without a specific trigger
	End	End of process without a defined result
	Terminate	All activities in the process are ended immediately without subsequent event handling
	Exclusive	XOR gateway; creates alternative flows where only one path is taken
	Parallel	AND gateway; creates parallel paths or to combine parallel flows
	Inclusive	OR gateway; creates parallel paths where all conditions ara evaluated

Table 3: Examples of BPMN events and gateway elements, as specified in the OMG standard documents ([OMG 09] and [OMG 11]). The enlisted elements make up a very small subset of the descriptive model sub-class, particularly, only those actually required for the business process models in the present study.

3.2 SIMULATING PROCESS ACTIVITIES VIA WEB SERVICES

The simulation and analysis of business scenarios is conducted in the dBOP *executionn layer*, which is built on top of IBM WebSphere®. Formally modeled business processes can be deployed on IBM WebSphere® Process Server. Supporting run-time services such as the *classifier service* that handles decision automation, or the

SYMBOL	NAME	FUNCTION
——▶	Sequence Flow	Shows the order, in which activities are performed
o- - - - - - - - - -▷	Message Flow	Shows the flow of messages between participants
···················>	Association	Links information and artifacts with elements

Table 4: Connection types in BPMN: Sequence flow, message flow, and association.

resource manager responsible for resource allocation are implemented as application components on IBM WebSphere® Applications Server, as described in [Niedermann 11d].

The following sections introduce and depict in detail the methods and software used in implementing and deploying the business process models described in Chapter 4.

3.2.1 *Development Environment*

IBM WebSphere® Integration Developer[2] is an Integrated Development Environment (IDE) for integrating service oriented applications. The tool is typically utilized to create solutions for BPM and to integrate them in a framework based on Service-Oriented Architecture (SOA). Business functions are encapsulated in service components, which are then packaged into an integrated service. The IDE provides process modeling tools in compliance with BPEL standards.

A top-down approach is used to generate an integrated frame for the process server, followed by the implementation of the involved service components and process activities. The IDE further includes a test environment, which allows debugging and performance analysis prior to deployment of the integrated service application on a real-time system.

3.2.2 *Service Integration*

The run-time environment for integrated process applications is provided by IBM's WebSphere® Process Server. The server

2 The implementation of business process models and the related web service programming in the present thesis were carried out using IBM WebSphere® Integration Developer Version 7.0

allows the assignment of tasks to contributors, as well as the coordination of workflows. It ensures that business rules are properly applied to all process stages.

An important function of the process server is the mediation of messages among service providers and requesters. These messages may occasionally have to be evaluated or manipulated based on process data and business functions involved in relevant process activities. Such technical actions are handled in a lower process level known as the Enterprise Service Bus (ESB).

The process implementation utilizes a predefined set of elements intrinsic to the development framework. These elements can be shortly described as follows:

MODULES are subprojects, which define the way how process resources are organized and administered. The standard module, also known as the integration module, is an application package implemented on the process server to enable the execution of business processes. `ExampleModule` in Figure 12 is a simplified representation of such an application element.

A second type of module, the so-called mediator, provides a means of connecting and integrating services. Modules operate on messages and contain further components, which share process data using bindings.

LIBRARIES store resources, which are referenced by modules or by other libraries. Unlike modules, libraries contain only public artifacts.

ASSEMBLY DIAGRAMS provide a means to create and organize services, processes and mediation chains, which can be further specified by appending artifacts from an embedded catalogue.

COMPONENTS are the basic building blocks in a business solution. Modules are built by combining components, which resemble the actual application flow or service routine. These so called Service Component Architecture (SCA) elements consist of an implementation, at least one interface and at least one link. Components can be made available to other modules by exports. A component can access the services of another module using imports.

INTERFACES define the methods, which operate on components. Typical operations are functions and queries provided by a service component. Components exchange data by transferring business objects through I/O operations. The Web Services Description Language (WSDL) Interface and the

Java™ Interface are the two main types, while Java™ components may consist of both. It is common practice to organize interfaces in libraries in order to make them readily accessible for several modules.

BUSINESS OBJECTS are containers for application data based on a XML Schema Definition (XSD) structure.

IMPORTS allow a module to utilize external functions, whereas

EXPORTS provide public interfaces to components, making their services available to other modules.

Figure 12: Example Module.

A typical integrated solution would incorporate several modules, which are utilized to drive an external service, as illustrated in Figure 15. Such visual representations provide an outline of the relationships between relevant projects and components in an integrated solution. In this particular example, `ExampleModule` comprises a business process, that operates on an external web service named `ExampleExternalService`.

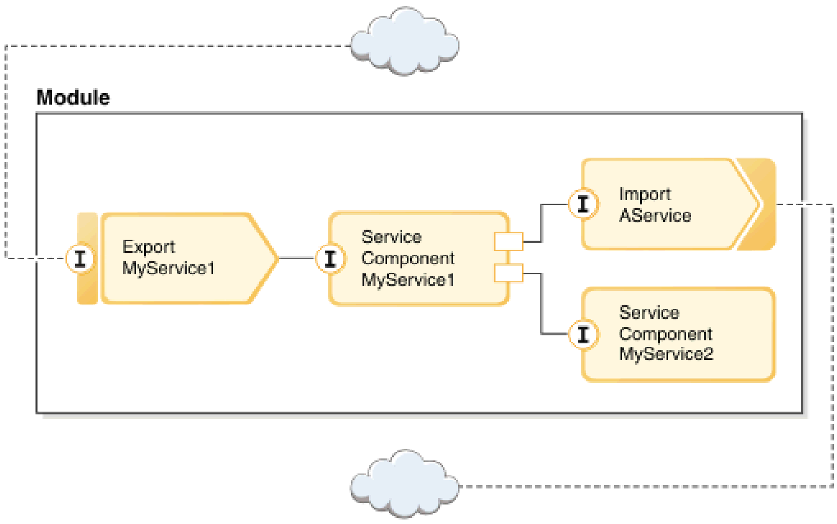


Figure 13: Example Module, © IBM.

We can generally assume, that the data formats employed in a business process model vary in most cases from those of the corresponding web services, which is the major reason why the interaction between these elements are required to be mediated by one or more intermediate modules, similar to the rightmost component group in the illustration, tagged `ExampleMediationModule`. Functional reviews and batch testing can likewise be managed

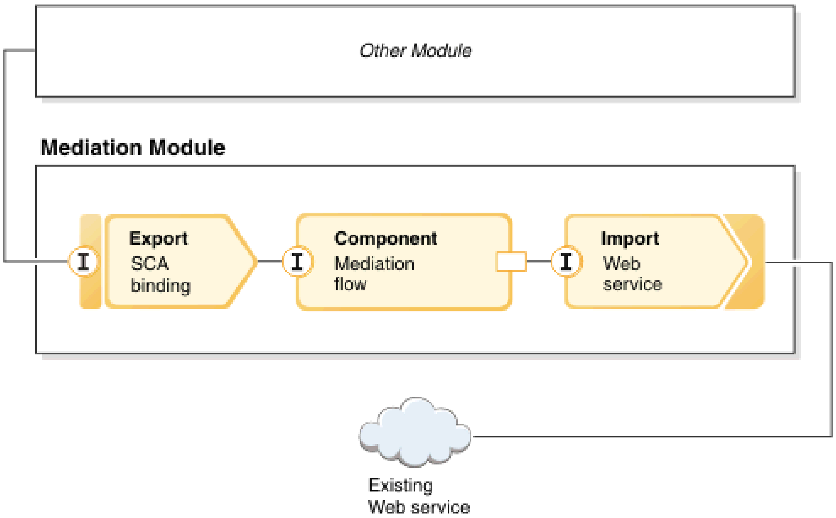


Figure 14: Example Mediation Module, © IBM.

using a separate module dedicated to these tasks, analogous to `ExampleComponentTest`.

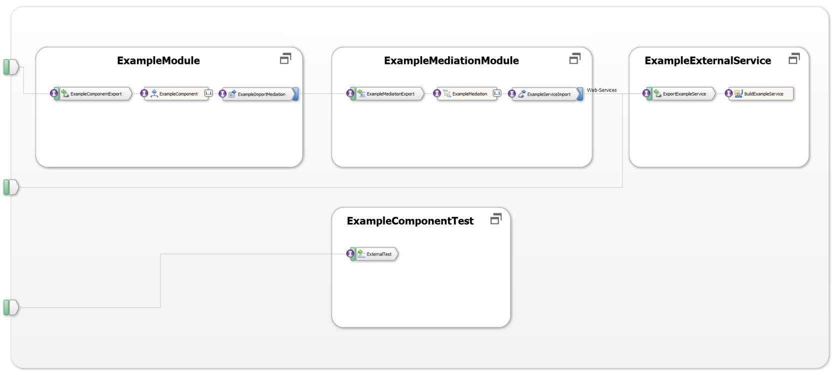


Figure 15: Example Solution.

3.2.3 *Web Services*

Having referred several times to the concept of *web services*, it is appropriate to provide a technical definition of the term for the sake of completeness. The glossary section of the standard document on web services (see [W3C 04]) provides the following explanation:

> *There are many things that might be called "Web services" in the world at large. However, for the purpose of the [W3C Web Services Architecture] Working Group and [the Web services] architecture, and without prejudice toward other definitions, [the authors] will use the following definition:*

WEB SERVICE A *Web service* is a software system designed to support interoperable machine-to-machine interaction over

> a network. It has an interface described in a machine-processable format (specifically WSDL). Other systems interact with the Web service in a manner prescribed by its description using Simple Object Access Protocol (SOAP)-messages, typically conveyed using Hypertext Transfer Protocol (HTTP) with an XML serialization in conjunction with other Web-related standards.

3.2.4 *BPEL*

The concept of Service-Oriented Architecture (SOA) has been attracting a lot of attention from every realm of the IT industry. Propelled by standards-based technologies like XML, web services, and SOAP, SOA has moved quickly from pilot projects to mainstream applications critical to business operations. One of the key standards that have helped to accelerate the adoption of SOA is BPEL.[3]

BPEL was created to address the requirements of composing web services in a service-oriented environment. In the past few years, BPEL has effectively become the most significant standard to elevate the visibility of SOA from IT to business level. BPEL now enables organizations to automate their business processes by orchestrating services within or across established structures, as discussed in [Gaur 06].

As already mentioned in Section 2.2.2, BPEL is currently the BPM standard that is supported by most vendors. The specification began as a *de facto standard* and transitioned into a *de jure standard*. This transition process took place in order to enable the reflection of input from as many participants as possible, aiming to cover requirements from a vast diversity of application domains, as described in [Leymann 10].

3 The most recent OASIS standard for BPEL is the Web Services Business Process Execution Language (WSBPEL) version 2.0 [OASIS 07, WSBPEL 2.0].

4 CASE STUDIES

The central technical requirement for the realization of the present study consists of designing realistic business scenarios for a simulated analysis environment. The significance of case studies in the context of software architecture evaluation, and related technical background on the simulation of business processes were already discussed in Chapter 2, Section 2.4. Laying out a solid empirical foundation for the previously mentioned methods requires, that real-life settings be pre-analyzed and accordingly documented.

This chapter describes the methods utilized to obtain process details for the case studies investigated in the present thesis. Following an overview of the chosen scenarios, each case is illustrated in detail along with the corresponding business process models.

4.1 SCENARIO DESIGN

The business scenarios discussed in the following sections have been assembled from a blend of sources. The process details originate from interviews conducted to support consulting activities related to reengineering efforts at the represented business enterprises. The process information collected in this manner has already been for the most part thoroughly documented and formally modeled for further analysis in related research literature, previously introduced in Section 1.3.

Parts of the investigated business process models have been modified for several reasons:

- *Reduction of process complexity* with the objective to allow for a controlled investigation of causal relationships between simulation parameters,
- Adherence to *data protection obligations* by omitting company specific elements and annotation,
- *Generalization* of the case studies for extended applicability to real-world situations, not only in affiliated industries, but also in a wider spectrum of business branches,
- Providing *improved intelligibility* of process structures with the help of abridged illustrations, and
- Elimination of *incompatible process patterns* from the formal model, in the face of graph-theoretic considerations, and

regarding certain technical constraints on the deep Business Optimization Platform. Loops, iterative or recursive activities, and complex data interdependencies are typical examples of candidate patterns subject to elimination, which in most cases are not completely omitted, but converted to functionally equivalent linear structures.

Therefore, the resulting process models analyzed in the present work are not merely simplified in their structure compared to their original counterparts, but also partly fictional in their content.

4.1.1 *Synopsis*

This section provides an overview of the business scenarios chosen for analysis. The following two real-life cases are investigated and formalized for automated optimization:

A. CAR RENTAL This scenario describes a common car rental business and an exemplary process, which represents the activities leading to a rental contract, initiated by a customer request. The corresponding process model originates from a German car rental company, and is discussed in several BIA publications as a motivational example (see related work in Section 1.3).

B. HEALTH INSURANCE CLAIMS When an insuree files a claim for medical expenses, several approval transactions must be carried out by the insurance carrier, before the requested sum can be paid out. The business scenario analyzed in the present study, is based on the contemporary claims process at a representative health insurance company. The sample process is adopted from the core models employed in a patented system for electronically reviewing and adjudicating medical insurance claims, as described in [Peterson 02].

The following sections elaborate on these business scenarios, and comment on selected formalized processes. The names of the companies have been changed due to data protection.

4.2 SCENARIO A: CAR RENTAL

A typical car rental business consists of a vehicle fleet and several service points for customer interaction. There are commonly various additional internal services for fleet management tasks, such as vehicle maintenance, transfer and disposition.

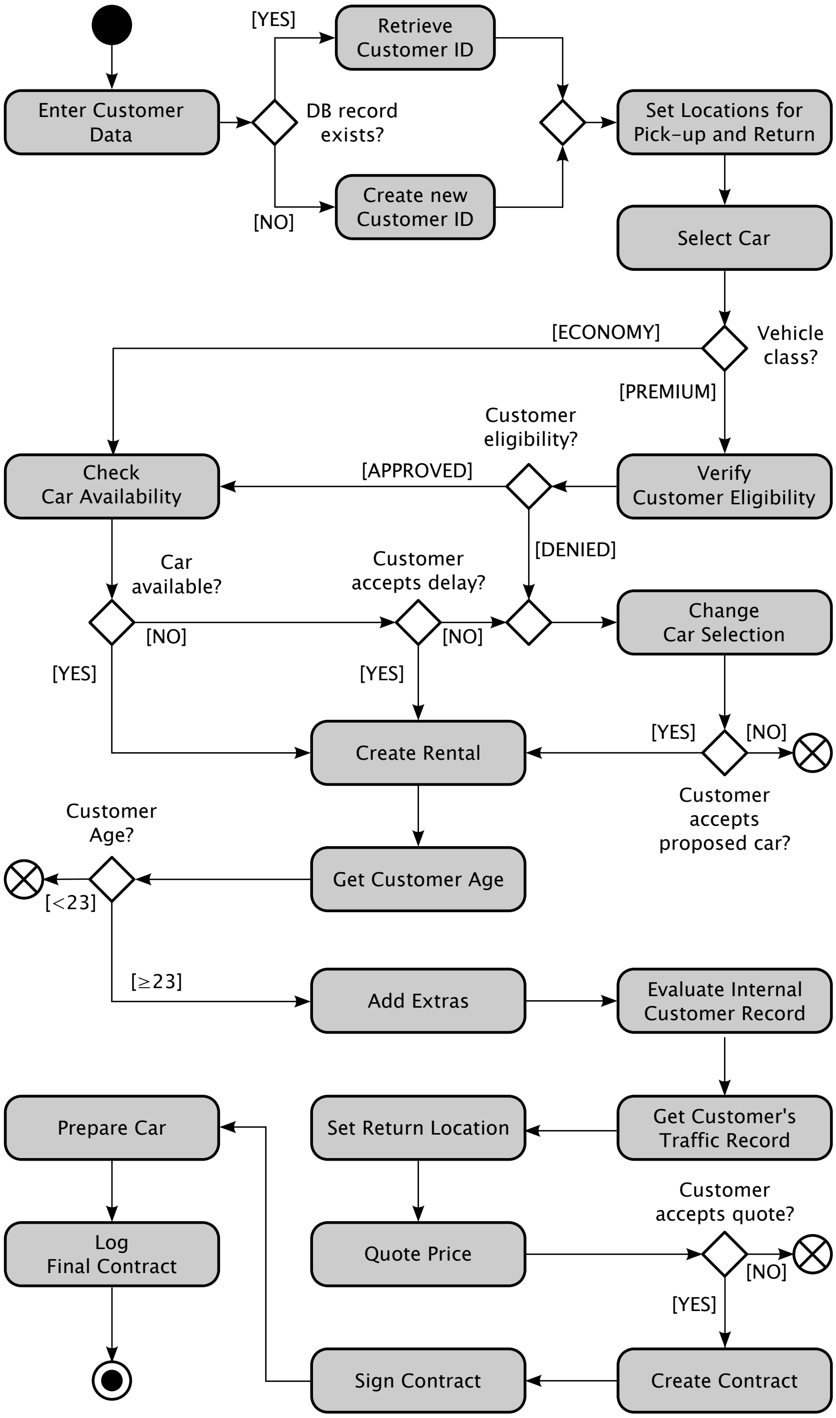


Figure 16: Rental process as currently employed at *Car Rental GmbH*. The UML based activity diagram describes the business process with four terminal points, of which three are regarded to abort the process prematurely, i.e. without having drawn up a rental contract.

The present scenario focuses on the process chain beginning with a customer request, and on the associated series of activities leading to drawing up a rental contract in due form. Figure 16 depicts a regular rental process with high-level descriptions of its routine activities. The following section provides a detailed verbal description of the rental process and its constituent tasks, as employed at an imaginary car rental company in southern Germany, as originally discussed in [Niedermann 08].

4.2.1 *Car Rental Process*

Car Rental GmbH is a fictitious enterprise, based loosely on an existing company, which started as a local car rental business in the 1980s, now grown to a medium-sized enterprise with offices all throughout Germany. However, the profitability of the company has decreased noticeably in the past few years. The decline in business performance is assumed to be caused by increasing operational costs as well as decreasing revenues, according to the directors.

The company's management has launched several projects to improve the synergy between so far separately operating departments, such as marketing, sales, and accounting, by extracting related information from the workflow audit data and from other operational knowledge sources. *Car Rental GmbH* has already installed a sophisticated Data Warehouse (DWH) and an accompanying Business Intelligence (BI) system for this purpose. This technical infrastructure aims to obtain highly correlated data patterns, which could be beneficial to a deeper understanding of the interrelations between process structures and their outcomes. Nonetheless, such information can not be readily acquired by classical Process Mining (PM) applications, similar to those employed at *Car Rental GmbH*.

The aim of the present work is to devise a concise model of one of the company's core processes for further analysis and optimization. Figure 16 illustrates the rental procedure by means of a UML based activity diagram, that provides an outline of the business process by which a client is provided with a vehicle on request. The process is assumed to be insufficient in terms of the company's performance goals. Therefore, the process model is mapped into BPMN/BPEL to be analyzed on the deep Business Optimization Platform (dBOP).

Every rental process is initiated by an incoming customer request. Although the mode of interaction can be regarded as arbitrary, it is assumed that customers approach a physical sales counter in person. Other viable modalities are intentionally left unconsidered in the present model, mainly for the sake of simplicity. The general operation pattern is of course equally applicable

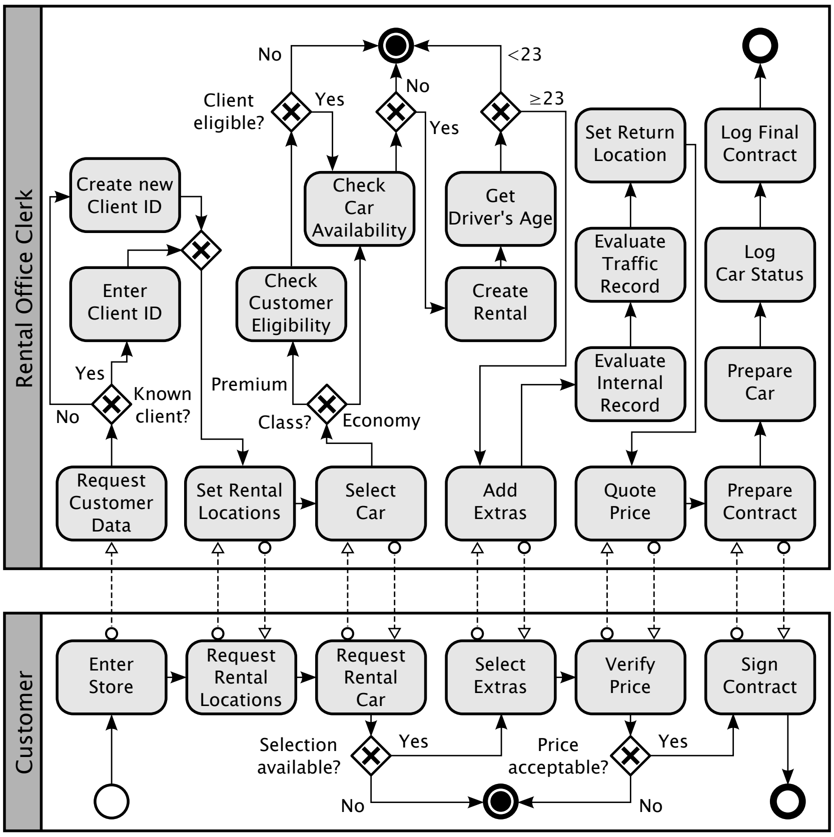


Figure 17: Business process diagram in BPMN describing the car rental process *before* optimization. The illustration combines the customer's view of the process with the front-office perspective at *Car Rental GmbH*. Database transactions and data flow inside the process are not explicitly annotated in the model. Several adjoining gateways have been likewise omitted to reduce complexity.

to any other type of customer interaction, e.g., most commonly via phone or even through a web portal. Different request modes may naturally require minor adjustments to business rules applied within the overall process.

The first activity on the part of the counter staff is an identification task. Assessing relevant customer data can be facilitated by keeping records of previous requests and contracts. If, however, a client has not used the rental service before or has not yet been registered, then a new record will have to be created in the customer base. Client records typically contain identification data and further relevant features and metrics. See Section 5.1 for a detailed account of the data warehouse, which incorporates the customer database.

All subsequent interactions are likewise documented in the form of attributes and translated into process logic, in accordance with the main objective of evaluating and adequately satisfying the customers request. The next activity would be choosing among possible locations, at which the client intends to receive and return the rented vehicle. The customer then selects a car, which may not in all cases be available, either because the particular vehicle is not at the preferred pick-up station, or because it is simply not operational due to other reasons, e.g. maintenance or repairs.

The postulated business logic demands that the client is provided with alternatives, in case his requirements can not be fulfilled sufficiently. Another feasible option is to propose delaying the service, i.e., to alter the time of beginning the rental. Depending on the customers response, the process can be terminated, or preferably continued by commencing contract formalities.

At this point of the rental process, it is desirable to generate additional revenue by offering extra services, such as insurance, unlimited range, sattelite navigation, roof racks, etc. Before the rental contract can be concluded, the client is presented a binding quotation. If he accepts the price, the rental is finalized, advancing the process regularly. Otherwise, the process is terminated without a contract.

The finalization of the rental contract consists of a small set of consecutive activities. First of all, a legally valid contract document is created and printed out, which is then signed by the contracting parties. Afterwards the rented vehicle is prepared for disposition and, if necessary, dispatched to the pick-up station agreed upon by contract. The final contract details and a status report on the rented vehicle are logged into the data warehouse, and the rental process is concluded regularly.

The evaluation of some customer specific attributes may lead to automated activities that influence the outcome of the overall process. To many penalty points in the customer's public traffic record may, for instance, cause the vehicle request to be automatically declined. On the other hand, participation in loyalty programs can lead to a discount on the regular rate, which is automatically considered in the final quotation. To reduce illustrative complexity, the automated nature of such activities is not explicitly annotated in the process model. Furthermore, an employee may be authorized to override the automated decisions, depending on his rank.

4.2.2 *Data Structure of Rental Contracts*

The data dimension of the car rental process is summarized as an entity relationship diagram in Figure 18. The data model is built around a central entity for *rental contracts*, which is essential to documenting concluded agreements. Client data is recorded in a *customer base*, including additional information, such as participation in *loyalty programs*, or possible entries in an external *traffic record*.

The contract framework must consistently assign one *employee* to each agreement, responsible for drawing up the contract and carrying out or triggering all necessary procedures. Moreover, the contract record must include complementary data as to which vehicle was rented when and where. An internal view of the inventory is tracked in a *car* database, and disposition details are related to the corresponding rental *station*.

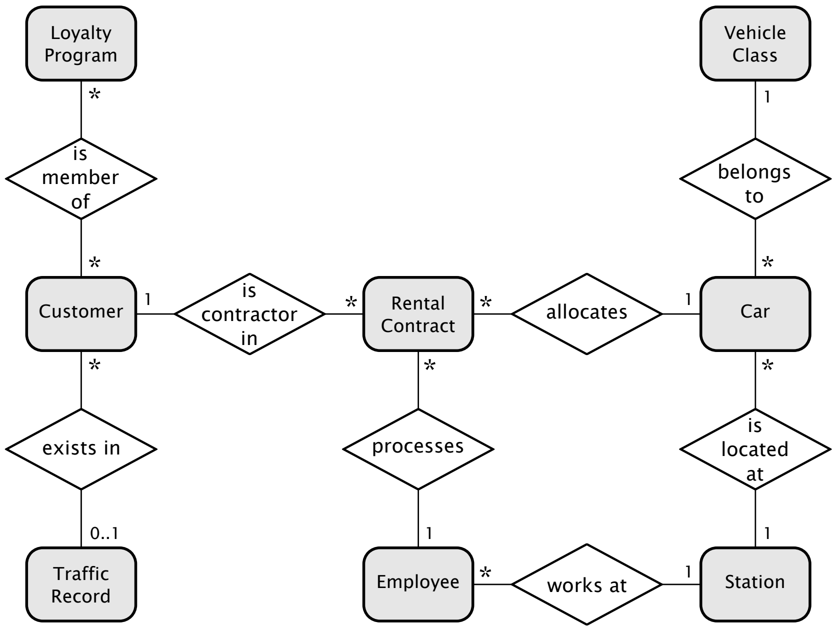


Figure 18: ERM Diagram: Car Rental Contracts. The presented extract of the model includes only process-relevant data. Entities describing insignificant details, such as category lists and minor records, have been omitted for simplicity.

Additional details on attributes and dependencies of car rental data are explained in Section 5.1, including a comprehensive account on sample data generation.

4.2.3 *Possible Improvements in the Rental Process*

The formalized model of the car rental process reveals considerable optimization potential — even without necessarily analyzing the tasks in depth. It should be obvious to an experienced process analyst, that, e.g., it would be reasonable to execute several process activities in this particular process scheme in *parallel* (see page 28) to reduce overall processing time. Typical parallelization candidates can be found among tasks involving a selection from a set of options, for which alternatives may have to be provided in case the customer disagrees with the initial offer, or if he lacks eligibility for the desired option. The query on additional options can be processed before the related tasks request a compulsory alternative, e.g., when a particular vehicle desired by the client is not available for rental.

Another promising improvement possibility is to relocate certain approval tasks which may induce an abort condition. To prevent running the process unnecessarily, it is essential to verify mandatory attributes as early as possible, e.g., whether the customer is in possession of a valid drivers license, or whether he fulfills the age requirements for rental services. Such tasks with *knockout* conditions (see page 31) can be moved to the very beginning of the process, or at least as far backwards as possible, to the point where relevant data is already accessible — provoking an *early knockout*, whenever possible.

Further improvements can be achieved by completely *eliminating* irrelevant tasks or removing duplicate activities from the business process. For instance, the above illustrated rental process contains a superfluous activity, in which a return location for the rented car is registered once again to calculate the quotation, although the client is previously expected to declare the pick-up and return locations, at the beginning of the process during requisition.

4.2.4 *Automated Optimization of the Rental Process*

Applying automated deep business process optimization to the car rental process yields the improved model illustrated in Figure 19. The process segments that have changed with respect to the source model are highlighted. The recommended optimizations can be summarized as follows:

- *Parallelization*: Evaluation of the internal customer record and data retrieval from the public traffic record do not have any data interdependency, and they do not require common resources, so that these activities can be carried out in parallel. The same principles apply to the logging tasks at the end of the process. Since the office staff is not responsible for disposition tasks, and therefore can not log the vehicle status, this activity can be executed, while the contract specifics are being recorded in the front-office.

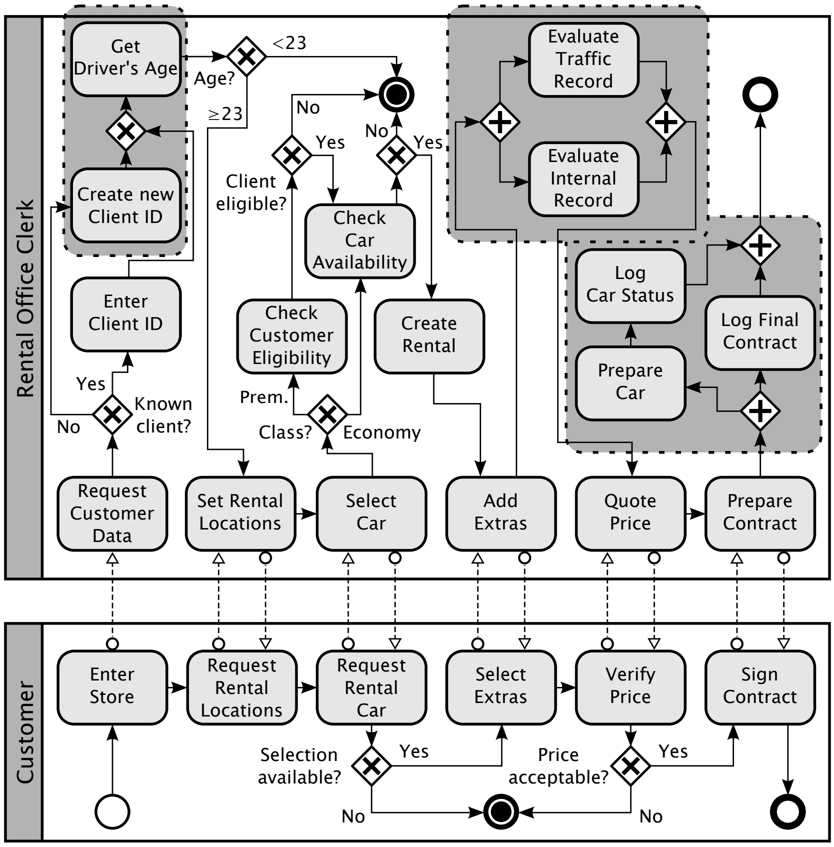


Figure 19: Business process diagram in BPMN describing the car rental process *after* automated optimization. The illustration combines the customer's view of the process with the front-office perspective at *Car Rental GmbH*. The process segments that have changed with respect to the source model are highlighted. Several *logging* activities and *data retrieval* tasks have been *parallelized* for optimal overall lead time. A redundant data entry task has been completely *eliminated* by the optimizer. Moreover, one task involving a *knockout* condition (*customer age*) has been *relocated* to be evaluated earlier in the process.

- *Task Elimination*: The second entry of the car return location following contract closure is redundant, as this information is already available due to previous requisition tasks.
- *Early Knockout*: Determining customer age earlier in the process can prevent the execution of unnecessary activities, thus reducing average lead time, regarding cases in which the customer lacks eligibility due to age restrictions.

4.3 SCENARIO B: HEALTH INSURANCE CLAIMS

The next case study consists of a simplified process model derived from a motivational business scenario, originally assessed and described by Niedermann (see [Niedermann 12]). The subject is a mid-sized private health care enterprise, which will be henceforth referred to as *Health Insurance Corporation (HI Corp.)*. The corporation has approximately 8000 employees and serves around 5 million insurees. Its core business has been growing constantly over the last years, but the company is now coping with increased administrative overhead. Furthermore, rising complexity of business operations and higher expenses in the health care system has built up pressure to reevaluate and optimize the company's core processes.

HI Corp. aims to catch up with the competition, primarily by reducing process costs, and by concurrently decreasing processing time. The predominant administrative task, as in any typical insurance company, embodies the compensation of insured expenses, specifically, payments for medical services.

The occurrence of an insured event commonly leads to a claim, filed either by the policy holder, through an authorized agent, or directly by the involved medical institutions. The insurance company subsequently investigates the details of the claim in order to decide whether the requested payout is eligible.

A formalized process for filing claims has the main objective of ensuring that the case at hand falls under covered conditions. It is likewise important to determine whether the requested payment is reasonable, taking into account the possibility of fraud by contractors or by policy holders themselves.

The following section describes the currently applied claims process at *HI Corp.*, as depicted in [Niedermann 12], which is subject to further analysis and optimization.

4.3.1 *Claims Process*

The claims process is initiated by a formal claim, i.e., the actual application of an insuree or an authorized agent for the payment of medical expenses. First of all, particulars of the policy holder

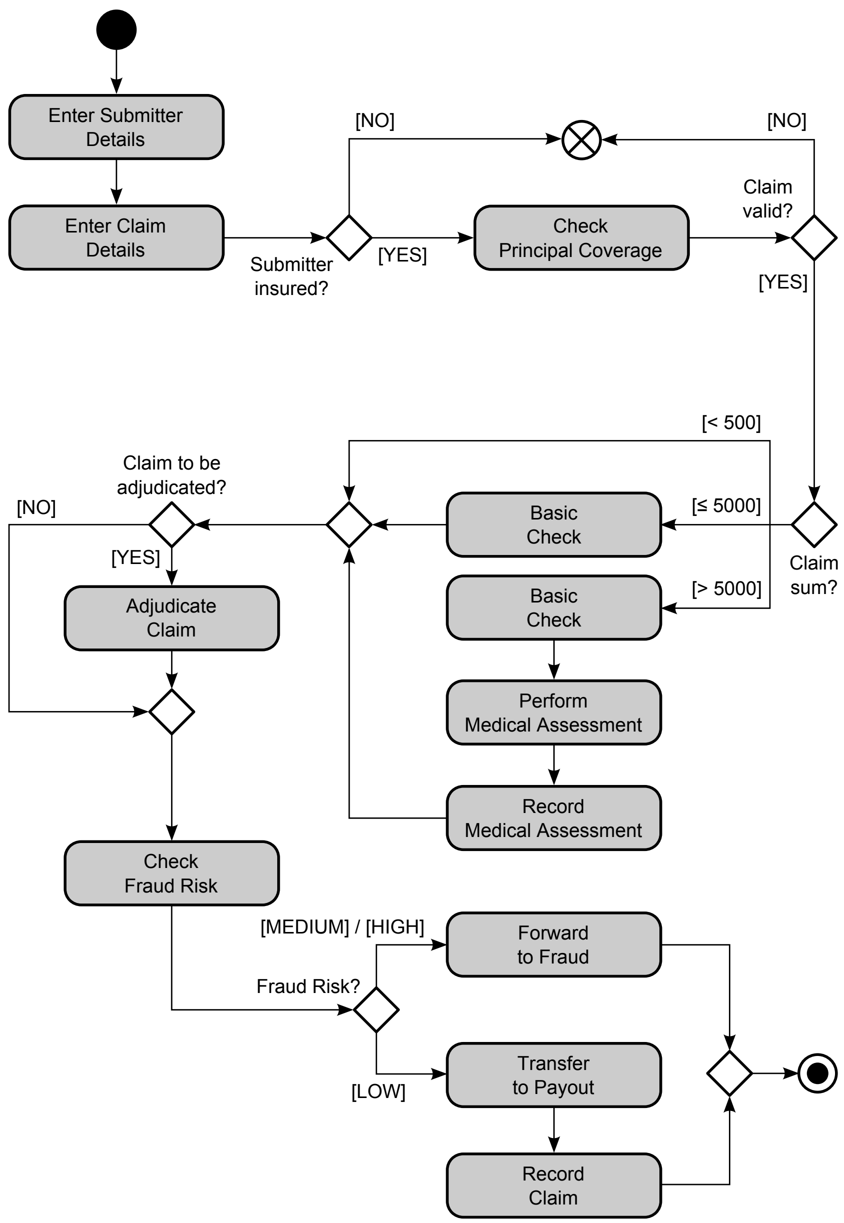


Figure 20: The business process for working through medical claims at *Health Insurance Corporation*. The UML activity diagram displays several approval stages and related tasks, mainly from the viewpoint of an insurance clerk. The process leads to payout, and is terminated regularly, only if certain prerequisites are satisfied. Declined claims cause the process to be aborted.

and claim details are registered manually by an insurance clerk. Only if the insurance in question is valid, i.e., the submitter holds an effective insurance policy by the specified date of service, the claimed compensation can be further evaluated.

The claim is then checked against coverage on the basis of the conditions agreed upon by policy. If the case does not fall under covered conditions, the payment has to be rejected. Otherwise, the claim is forwarded directly to approval instances for payment requests less than € 500. Any claim containing a payout request higher than € 500 requires that an elementary inspection (annotated as *basic check* in the process model) be performed by the assigned insurance clerk.

The elementary inspection typically involves the examination of consistency between diagnosis and treatment, and other similar routines. As soon as the claimed payment sum exceeds € 5'000, however, the case is additionally assigned to a medical expert who is authorized to perform a medical assessment. The results of the medical assessment are recorded by the clerk in charge for further processing.

The insurance clerk then applies a set of predefined rules to determine whether the claim is subject to manual adjudication. The term *adjudication* is used in the insurance business to denote parts of a claims approval process, and refers generally to an automated routine to check eligibility requirements, to request an appraisal report and to calculate deductible payments, i.e., the contingent to be paid by the insuree before coverage by the insurance company takes effect.

After the claim is adjudicated manually or automatically, the probability of fraud must be evaluated in advance of final payout approval. Therefore, the clerk performs a *fraud risk check* before assigning the claim to accounting for payment formalities, and commits a final record of the claim into the claims database (See Section 5.2 for details on the insurance data warehouse). If, however, the fraud risk is estimated to be higher than a predefined threshold, the case is forwarded to fraud investigation, and the clerk terminates that particular instance of the claims process — without an ultimate decision of approval or decline.

4.3.2 *Data Structure of Insurance Claims*

This section introduces the operational data produced during the processing of insurance claims, and supplementary data sources utilized in approval activities. as illustrated in Figure 22. The conceptual model is framed around a central entity for the filed *claims*. Every case is documented so as to include information about the submitter (*patient*), and a distinct association with one particular *insurance policy*.

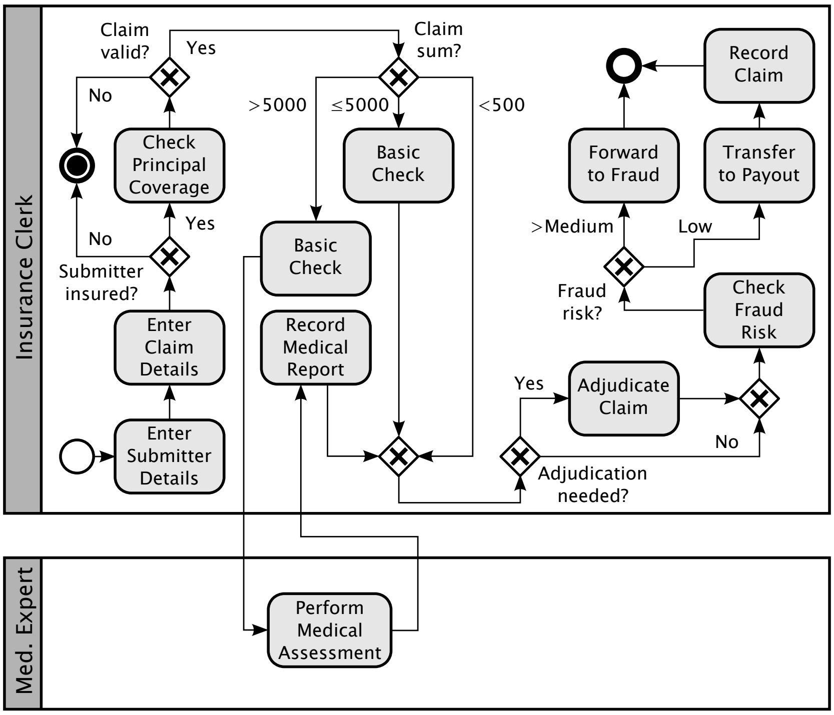


Figure 21: Business Process Diagram for health insurance claims from an internal perspective, as currently applied at *HI Corp*. Database transactions are not annotated explicitly in the process model, for reasons of graphical simplicity. Activities related to submission modalities of claims — which would have required an additional BPMN swimlane for the submitter — have also been intentionally omitted for similar reasons. The process, therefore, begins with the receipt of a medical claim by an insurance clerk.

Every medical claim requests compensation for one or more specific *services*, which can be supplied by a number of authorized *providers*. The responsible case manager (*employee*), has the duty to determine, whether the claimed services are covered by the agreed *insurance products* in the associated policy.

Patients may also receive additional benefits due to affiliation to certain *employers*. Furthermore, it is essential to incorporate public *medical records* for all insurees, in order to be able support evaluate

Additional details on attributes and dependencies of insurance claim data are presented in Section 5.2, including a comprehensive account on sample data generation.

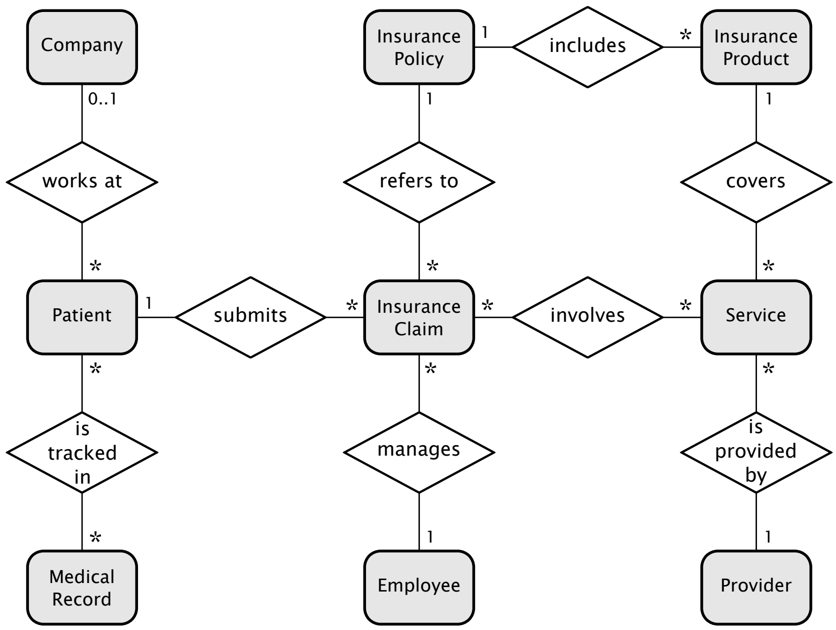


Figure 22: ERM Diagram: Health insurance claims. The model represents a generalized depiction of data recorded and retrieved during every instance of the claims process. Entities describing insignificant details, such as category lists and minor records, have been omitted for simplicity.

4.3.3 *Possible Improvements in the Claims Process*

Analogous to the previously described improvement prospects for the car rental process (see Section 4.2.3), the insurance claims process likewise consists of particular activity patterns, which seem prone to optimization, though not as numerous and not quite as obvious in comparison to the former case.

Despite the fact that all evaluation activities are mandatory for a sound approval process, it is reasonable to find a means to execute as few of these sub-processes as possible, assuming that such tasks utilize expensive resources and require special attention at all stages due to their critical nature. Provided sufficient operational data, and detailed workflow audits with decisive attributes, it could be reasonable to *resequence* (see Figure 9) parts of the process.

With this particular optimization approach in mind, one suitable method would be to execute the least expensive tasks sooner, or with higher priority in case of parallelization. For example, determining the claimed amount at an early stage of the process can help to decide whether it could in some cases be practical to prepare a medical assessment request without waiting for active approval tasks to be concluded.

A comparable effect can be observed, if activities involving compulsory approval tasks are favored at the initial phase of a process. An early decision to decline a payout request, i.e. an *early knockout*, can save valuable processing time and substantially reduce overall costs.

Further potential for time and cost optimization lies in the parallel execution of tasks, as previously mentioned. Considering the closely related natures of the *adjudication process* and *insurance fraud assessment* with regard to the common data resources required, it could be beneficial to parallelize these activities, needless to say, depending on process preconditions. In doing so, the interdependencies with other process segments must be considered, i.e., measured in a simulated environment, to guarantee an effective optimization with respect to predefined performance indicators.

Careful observation of the claims process reveals some intrinsic redundancy in data logging activities. It can generally be assumed that most, if not all, case specifics and submission data are already fed into the system, e.g., via database entries, at the very beginning of the process, when a claim is initially filed by an insurance clerk. Combining such registry tasks, involving data collection and input, by integrating related or repetitious activities — preferably at an applications level — could viably produce the desired optimization results. Certain input tasks can be eliminated altogether, provided that the data in consideration is completely redundant.

4.3.4 *Automated Optimization of the Claims Process*

Applying automated deep business process optimization to the claims process yields the improved model illustrated in Figure 23. The process segments that have changed with respect to the source model are highlighted. The recommended optimizations can be summarized as follows:

- *Parallelization*: A basic medical check requires resources other than those utilized by a medical expert during a comprehensive medical assessment. Both examination tasks are required, if the claim sum exceeds €5'000, however, these do not need to be performed consecutively. Hence, the two activities are parallelized in the optimized model.

- *Task Elimination*: Since all claim details are recorded at the beginning of every process instance, it is not necessary to enter further specifics at the conclusion of the payout approval. Therefore, the redundant logging activity at the end of the process is removed completely.

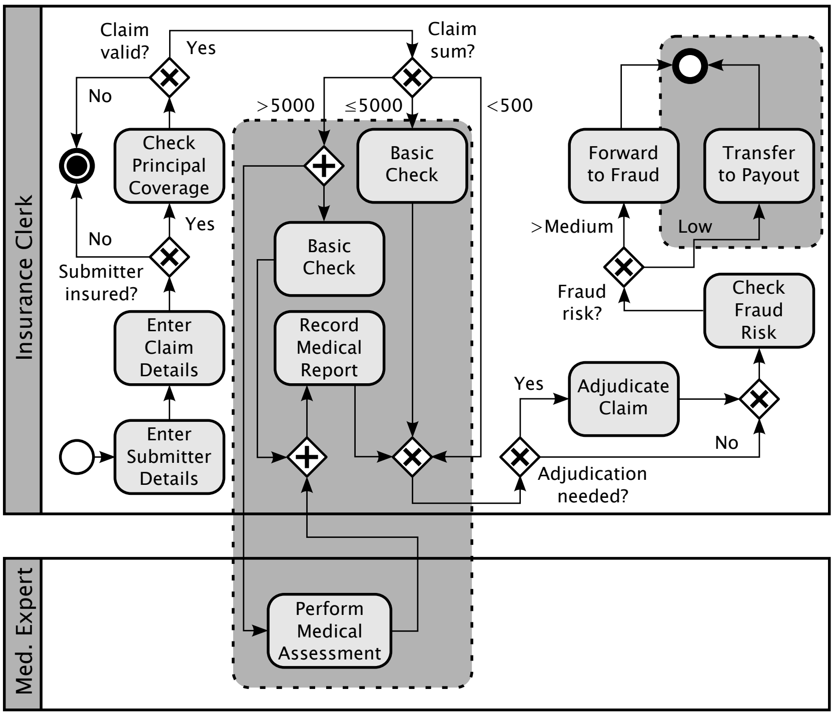


Figure 23: Revised business process model for health insurance claims from an internal perspective, as optimized by dBOP. The process segments that have changed with respect to the source model are highlighted. The business process diagram BPD now includes a new `AND` gateway, at which the activities *basic check* and *medical assessment* are initiated in *parallel*. Furthermore, a redundant logging activity at the end of the process has been *eliminated*.

5 GENERATING DATA SAMPLES

In order to achieve a high degree if realism in process performance behavior while simulating the business scenarios constructed in Chapter 4, it is essential to provide an accordingly realistic operational data warehouse with respect to overall volume. Furthermore, the semantic basis of data attributes must be as close to reality as possible, with almost real-life value distribution. The scenarios simulated for analysis in the present study, although based on real data, nevertheless remain representative approximations of their archetype counterparts.

Generating test data merely by means of a simple randomizing algorithm would, however, not fully meet the purpose of performing realistic simulations, taking into account that real-life data collections tend to display certain statistical distribution patterns. Subsequently, the process data, and their underlying database concepts have been modeled based on publicly available analytic material, e.g., demographical data on age distribution or statistics concerning public traffic records.

With the exception of category tables and complementary data base entities, all sets of sample data were generated using simple `Java`™ methods, primarily to execute SQL statements utilizing the database API[1] to query the target data warehouse. Database connections were established using a universal Java Database Connectivity (JDBC) driver[2] for IBM® DB2™. The following sections illustrate the database structures associated with the simulated business processes, and provide specific aspects of the sample data therein.

5.1 CAR RENTAL DATA WAREHOUSE

The data warehouse for car rental operations consists of a simplified relational database containing the most essential process related data and some complementary entities. The present data model is based on an account of the car rental scenario used as a motivational example for an integrative process analysis approach in [Niedermann 08]. Similar entity relationship models have been employed in several subsequent BIA publications (see,

1 Package: `java.sql`

2 Driver tag: `com.ibm.db2.jcc.DB2Driver`. For more information on connecting to a data source using the DriverManager interface with the IBM DB2 Driver for JDBC, see http://publib.boulder.ibm.com/infocenter/db2luw/v9/topic/com.ibm.db2.udb.apdv.java.doc/doc/cjvintro.htm

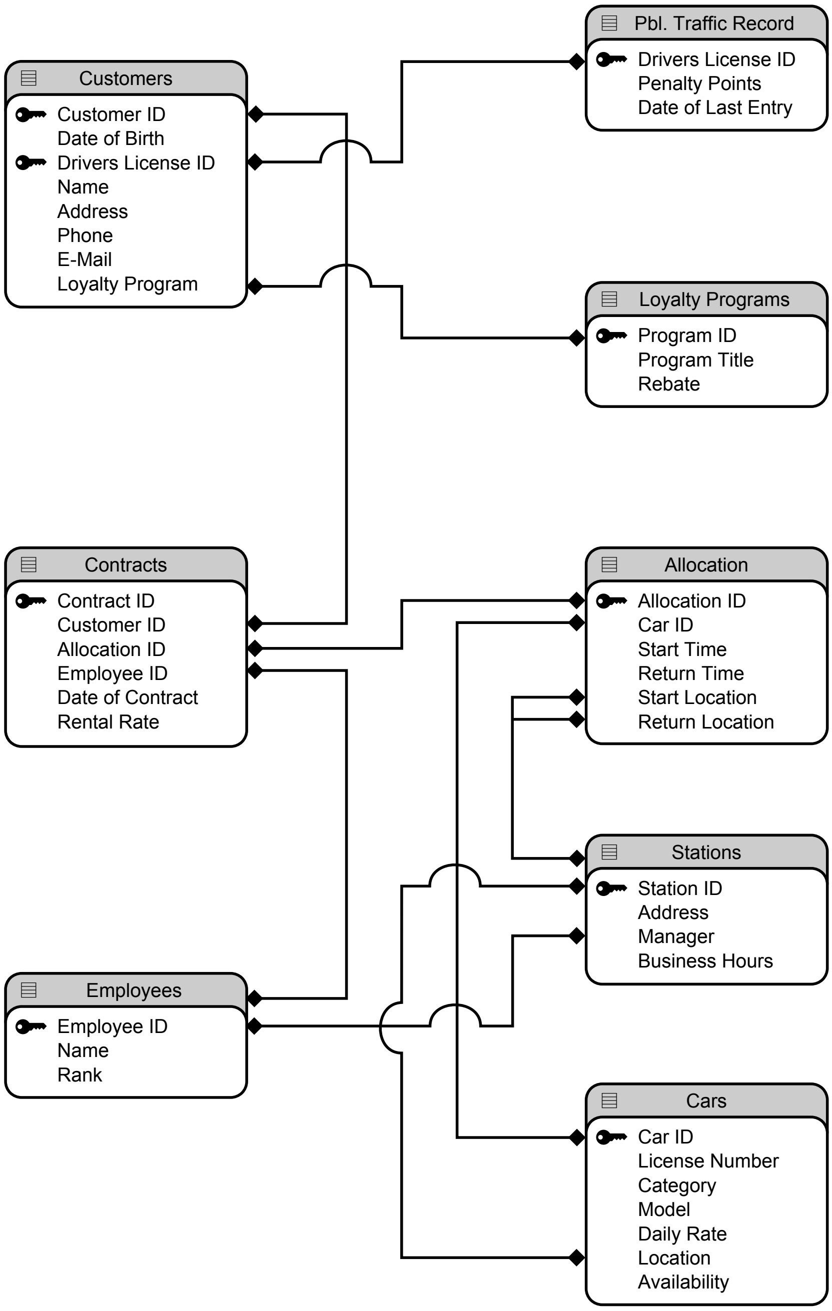


Figure 24: Data warehouse: Car rental data at *Car Rental GmbH*. The illustrated data model displays only process relevant entities. The cardinalities of entity relationships were intentionally left out for simplicity.

e.g., [Radeschütz 10] and [Niedermann 11a]), with various modifications depending on the research focus. The following sections expand the data model previously introduced in Section 4.2.2.

5.1.1 *Data Repository for Rental Contracts*

At the end of any successful rental process, there should ideally be a fully specified and signed rental contract, which is documented and archived in the data warehouse, at the latest, following its conclusion. The central entity in the database is, therefore, the table of contracts. Figure 24 shows a simplified conceptual database model, on which the sample data warehouse for process simulation is based.

Every concluded rental contract is a legal agreement between two contractors: The rental company and the client. The sample rental company has a client base of 10'000 customers, all of which are registered in the customer database. See the following section for details on how every single attribute relates to operational business process data, and how values were generated to fill the sample database.

Car Rental GbmH offers several loyalty programs for its customers. The rules by which customers can participate in these programs is not of further consideration. However, being assigned to loyalty program or participating in one of several promotional schemes can involve regular rebates, leading to lower rental rates. Conversely, an unfavorable public driving record may in some cases result in higher insurance fees, as described further below.

5.1.2 *Customer Attributes*

Each registered customer has a unique identification number *(Customer ID)* that simultaneously serves as a primary key in the customer table. Assuming that every eligible customer has a unique drivers license number, it is possible to use its identification code as an additional primary key. Gender, name and contact information are not regarded as process relevant in the present context.

Several other attributes are generated randomly for every record in the customer database, aiming to resemble real-life distribution of relevant data as closely as possible. The statistical grounds for these features are as follows:

traffic records Traffic violations as tracked by the German Federal Motor Transport Authority, *Kraftfahrt-Bundesamt* (KBA), were taken as an exemplary basis for the corresponding database entity. Table 5 lists the distribution of accumulated penalty points for registered traffic violations.[3] Drivers who reach a penalty score of 18 or more have their driving privileges suspended, i.e., they must turn in their drivers license for a certain period of time, and can be obliged to attend special courses, e.g., for safe and defensive driving (see [KBA 11]).

PENALTY POINTS	INDIVIDUALS (IN 1'000)	%
none	1'754	19.6
1 – 7	6'626	74.0
8 – 13	456	5.1
> 14	66	0.7
Total	8'951	100.0

Table 5: German traffic register tracked by the Federal Motor Transport Authority. The table shows the distribution of penalty points for traffic offenses, last updated in January 2010. The scores are grouped in intervals. Approximately 0.5% of all violations are not registered electronically, hence do not appear in the statistics.

At *Car Rental GbmH*, like at most other car rental companies, drivers who stand out with an exceedingly offensive traffic behavior are excluded from services, or at least denied access to certain vehicle classes in the premium range. A further consequence of a negative driving record is, that fully comprehensive insurance coverage usually becomes mandatory for a rental contract, hence higher rental costs for the client. Customers without a valid drivers license, or those likely to lose their license due to recurrent traffic violations, are rejected altogether.

public records

Regarding the fact that not all active drivers have a violation record in the central register, the total population of potential rental customers is assumed to be the number of individuals with a valid drivers license, estimated 54 million in Germany, according to the German Federal Highway Research Institute, *Bundesanstalt für Straßenwesen* (BASt), as reported in [Kalinowska 07]. The public traffic record also contains entries with zero penalty points, due to negative prescription of former violations, typically after two consecutive offense-free years. These records can be completely cleared after a certain period of time, depending on the scope of the violations.

3 A more detailed representation of the data, e.g., with gender specific statistics, is available online at
http://www.kba.de/cln_005/nn_191656/DE/Statistik/Kraftfahrer/Verkehrsauffaelligkeiten/BestandVZR/2010__aktuellerpersonenstand__tabelle.html

POPULATION RECORDS In order to achieve a realistic age distribution in the customer base, the data generator was based on demographical statistics for Germany,[4] as released by the Federal Statistical Office, *Statistisches Bundesamt* (see [DESTATIS 11]). Although this approach is not as accurate as the statistical evaluation of a real rental customer database, it is assumed that the age distribution in the general population is an acceptable approximation.

AGE GROUP	% OF POPULATION
under 20	18.8
20 – 39	24.3
40 – 59	31.0
60 – 79	20.8
80 and above	5.1
Total Population	81'802'257

Table 6: German population in the year 2009, grouped by age, as released by the Federal Statistical Office, *Statistisches Bundesamt* (see [DESTATIS 11]).

The age of a customer is the principal criterion for a group of decision tasks in the car rental process. Since it is common practice among German car rental providers to require that a client be in possession of a valid drivers license, only available at 18 (not including escorted driving with 17 years), customers under 20 years of age are generally rejected. Furthermore, the upper limit for rental age is set to 80 years in the present process model. As already discussed in Section 4.2.1, some vehicle types may require a higher minimum age, e.g. 23 instead of 20.

5.1.3 *Company Attributes*

The sample rental company is represented by its employees in rental offices. The corresponding database entity is a simple table of unique identifiers. As with the customers, names and other personal data are not considered as process-relevant data, therefore neglected during sample generation. The rank of an employee is, nevertheless, critical with regard to automated decisions which may be overridden by managers or by a certain class of clerks with adequate authorization.

The products being offered are, obviously, rental cars, i.e., regular motor vehicles. These are classified in two main price cat-

4 Further details available from the online statistical database at
https://www-genesis.destatis.de/genesis/online

egories: *economy* and *premium*. The vehicles are located in several stations, which are all associated with a front-office. Details of vehicle disposition and internal services, such as maintenance and transfer, do not need to be modeled explicitly. These subprocesses only appear as random duration parameters implicitly in the overall business process.

5.2 MEDICAL CLAIMS DATA WAREHOUSE

It is in the nature of the insurance business, that excessive amounts of operational data are accumulated, not only for administrative purposes, but also for comprehensive cost control, in order to devise appropriate coverage schemes for the insurance products offered.

For the purposes of the present study, it was essential to reduce the data complexity and still preserve a reasonable degree of resemblance to the original structures. A generalized ERM for the data dimension of the claims process was already introduced in Section 4.3.2. The corresponding data warehouse, based on this model, is illustrated in Figure 25, and characterized in detail in the following sections.

5.2.1 *Claims Archive*

All medical insurance claims ever filed at the *Health Insurance Corp.* are registered in an archive, which has evolved through several different forms in the course of the company's history. The current data warehouse incorporates all process data and operational knowledge, including the claims database. This section describes the reduced model of the data warehouse and the entity relationships therein, with proper consideration of the claims process described in Section 4.3.

Each claim is assigned to an insurance clerk, called the *case manager*. The clerks have varying levels of authority, and can steer the claims process by accepting or overriding automated decisions, depending on *rank*. Claim files include elementary attributes on the *type* and *volume* of the payout request, which are linked to associated entities that contain further particulars. The *customer* base enlists personal information of the insured members, and accommodates insurance policy details, e.g., benefits and coverage, and is linked to *medical records*.

The predominant key figure in the claims process is *service cost*. Most approval tasks in the process are concerned with the consistency of service types, and must determine whether the requested compensation is reasonable. Apart from obvious discrepancies, such as maternity related expenses associated with an adult male patient, deviations from common charging scales must also be

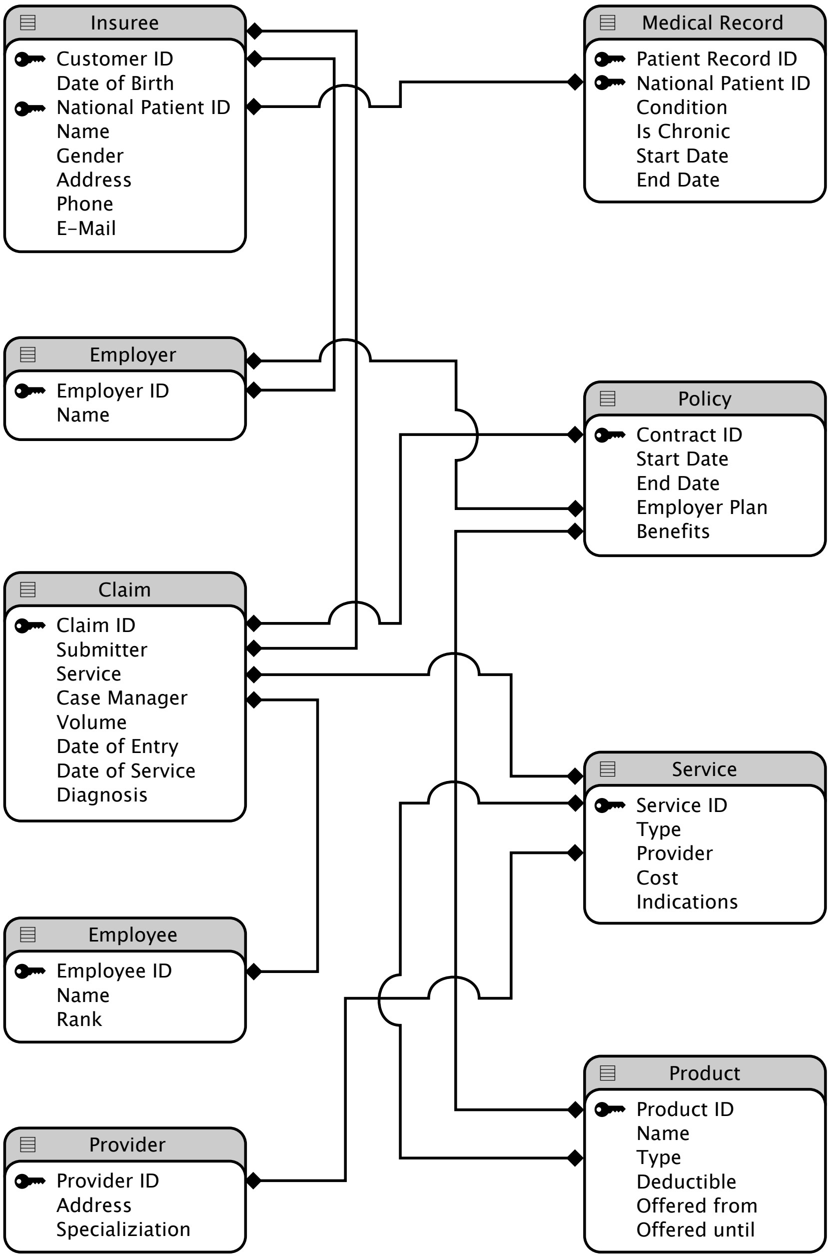


Figure 25: Data warehouse: Medical insurance claims at the *Health Insurance Corporation*. The illustrated data model displays only process relevant entities. Category tables (such as medical conditions and health care categories) and cardinalities of entity relationships were intentionally left out for simplicity.

tracked carefully. The following sections depict the statistical and categorical specifics on which these approval activities are based.

5.2.2 *Medical Statistics*

Despite the vast amount of publicly available medical data, most statistics provide only scarce and, in several cases, unreliable information on the *frequency* of disease types and indications, or contain outdated information. In order to obtain a realistic data distribution of the various medical conditions to be considered in the medical claims process, it would be appropriate to evaluate a comprehensive statistical survey on the incidence of common disease categories.

The International Classification of Diseases (ICD) issued by the World Health Organization (WHO) is a convenient starting point for categorizing diseases and health problems for usage in an insurance data warehouse. The ICD[5] is the international standard diagnostic classification for all general epidemiological, for many health management purposes, and for clinical use, as described in [WHO 11]).

The European Commission (EC) keeps track of data sets on health status, determinants and care in European Union (EU) member countries, categorized in a list of 88 indicators known as European Community Health Indicators (ECHI).[6] Unfortunately, the frequency data provided on disease categories and diagnosis types, e.g., indicators Nr.67 – Nr.70 (see [Kilpeläinen 08] for details), involve relative small sample populations, and most important of all, the data is available for a selected subset of disease categories.

The German Association of Private Health Insurance PKV has also conducted a survey among its members in the year 2009, in order to collect frequency information concerning the incidence of health problems and diseases, published in [PKV 11]. Although only 16 from a total of 45 private insurance companies participated in the latter survey, thus rendering its results non-representative, the patterns in *gender dependency* and the general trends in *frequency distribution* can be regarded as adequate for the purposes of business process simulation.

5 *The International Statistical Classification of Diseases and Related Health Problems* is reviewed regularly by the World Health Organization (WHO), and there are several valid versions of the document. The present work adopts the 10^{th} revision of the classification list (Version 2007).

6 Introductory information on European Community Health Indicators (ECHI) can be found online at http://ec.europa.eu/health/indicators/echi

5.2.3 *Health Service Types and Costs*

According to the data model on which the data warehouse for the claims process is based, claim records include the overall amount requested for payout, as soon as the case is filed. The separate positions making up the sum are generally attributed to particular services or service combinations. The provider bills these accounts on the basis of a predefined list of services, in which the corresponding rates are framed in a certain range.

service fee scales

These service lists are reviewed and updated regularly by authorized medical associations or, in some cases, by government organs. Two examples of such service catalogues can be found in the contemporary German public health care system. The statutory health insurance system is based on a tariff catalogue called the *uniform assessment scale* (Einheitlicher Bewertungsmaßstab, EBM [KBV 11]).[7] Private health insurance companies, on the other hand, are obliged to apply a deviating standard of charges listed in the *scale of fees for physicians* (Gebührenordnung für Ärzte, GOÄ [BMG 11]) issued by the German Federal Ministry of Health (Bundesministerium für Gesundheit, BGM).[8]

Although these catalogues are much too comprehensive for the process simulations conducted in the present work, they do, nevertheless, provide an appropriate groundwork for the service categories depicted in the claims database. Combined with the service categories enlisted in healthcare related legislation in the United States, for instance in Washington State legislature on healthcare coverage (see [WAC 11]), a representative framework is attained for modeling the corresponding aspects of the insurance data warehouse.

The fee structures in EBM and GOÄ are likewise highly complex. Therefore, merely a short subset of service categories was adopted for an abridged list of charges to be incorporated in the data warehouse. For calculating the overall service expenses for particular health conditions, the cost structures in the database model were simply based on the average days spent in stationary care, as listed in the A, Section A.4. This method further reduces the complexity of the relational model, and concurrently decreases the data volume down to a practical magnitude.

7 Extended catalogues for special health care domains, such as dental care and alternative medicine, have been neglected for convenience.

8 http://www.bmg.bund.de

Part III

CONCLUSION

6 RESULTS

This thesis demonstrates, that the deep Business Optimization Platform (dBOP) is capable of assisting a process analyst in carrying out optimization tasks and evaluating various improvement approaches, typically at a faster rate than he could without automated optimization assistance. Employing the optimization tools provided by the platform, particularly the dBOP-Modeler, does not require a high degree of process expertise.

dBOP delivers reliable optimization results for the class of business processes evaluated in this study, especially with regard to process segments prone to improvement by parallelization. Elimination of redundancy by detecting semantic similarity among activities likewise attained the expected results, which were predicted in advance by manual evaluation from a human analyst's perspective.

6.1 OUTLOOK

One obvious question that comes into mind is, whether and how the results produced by automated process optimization correlate to the underlying data structure containing process and operational data. An interesting approach would be to run several instances of the same process with varying data distributions, i.e., changing the constitution of the data warehouse for every simulation run.

Furthermore, investigating the effects of the topologic complexity of a process model on the outcomes and general performance of fully automated or semi-automated process optimization using dBOP. A graph-theoretical categorization of the available scenarios into complexity classes would allow the comparison of these categories with regard to their optimization results. One could naturally prefer a statistical approach in which a large set of models with varying structure and complexity are analyzed, and clustered according to the degree of improvement achieved, both in quantitative and qualitative terms. The experiment can also be expanded to include comparing the automated optimization results for similar complexity classes, to those of a human process analyst.

Another possibility is to start with basic process models, i.e., exemplars from the simplest possible complexity class, of which the elements are still complex enough to be structured in a non-optimal manner, and successively increase the complexity — either by simply adding a fixed number of descriptive elements, or by some other measure of graph-theoretic complexity. Conducting an extensive statistical analysis on as many distinct complexity classes as possible might provide insights, both on simulation predictability, as well as on dBOPs general performance, helping to improve future investigations using this platform.

Finally, a useful contribution to business process simulation research — and of course to its applications as well — would be the development of appropriate standards, not only for the description or modeling of processes, but also for their evaluation with respect to predefined performance indicators.

Part IV

APPENDIX

A APPENDIX

A.1 CONNECTING TO A DATA SOURCE USING JDBC

See Listing 1 for a sample data connection using JDBC.

A.2 GENERATING PUBLIC TRAFFIC DATA SAMPLES

Refer to Listing 2 for an example of how the data distribution statistics from Section 5.1 are utilized.

A.3 INTERNATIONAL CLASSIFICATION OF DISEASES (ICD)

The International Statistical Classification of Diseases and Related Health Problems (10th Revision, Version 2007 [WHO 11]) lists the following primary categories:

I. Certain infectious and parasitic diseases

II. Neoplasms

III. Diseases of the blood and blood-forming organs and certain disorders involving the immune mechanism

IV. Endocrine, nutritional and metabolic diseases

V. Mental and behavioural disorders

VI. Diseases of the nervous system

VII. Diseases of the eye and adnexa

VIII. Diseases of the ear and mastoid process

IX. Diseases of the circulatory system

X. Diseases of the respiratory system

XI. Diseases of the digestive system

XII. Diseases of the skin and subcutaneous tissue

XIII. Diseases of the musculoskeletal system and connective tissue

XIV. Diseases of the genitourinary system

XV. Pregnancy, childbirth and the puerperium

Listing 1: A JDBC example

```
// Load the DB2 Universal JDBC Driver
Class.forName("com.ibm.db2.jcc.DB2Driver");
System.out.println("Loaded the JDBC driver");

// Create the connection using the DB2 Universal JDBC Driver
conn = DriverManager.getConnection ("jdbc:db2:CARRNTL",
                                "user", "password");

// Commit changes manually
conn.setAutoCommit(false);
System.out.println("Created a JDBC connection to the data
    source");

// Create the Statement
stmt = conn.createStatement();
System.out.println("Created JDBC Statement object");

// Execute a query and generate a ResultSet instance
rs = stmt.executeQuery("SELECT * FROM T_EMPLOYEES");
System.out.println("Created JDBC ResultSet object");

// Print all of the employee numbers to standard output
while (rs.next()) {
  System.out.println(rs.getString(1));
}
System.out.println("Fetched all rows from JDBC ResultSet");

// Close the ResultSet
rs.close();
System.out.println("Closed JDBC ResultSet");

// Close the Statement
stmt.close();
System.out.println("Closed JDBC Statement");

// Commit transaction
conn.commit();
System.out.println ("Transaction committed");

// Close the connection
conn.close();
System.out.println("Disconnected from data source");
```

Listing 2: Data generation example: Public Traffic Record

```
public static int generateRandomPenaltyPoints(){
        /*
         * Assumptions:
         * 50 percent of all drivers are registered,
         * and among these
         *     20 percent have reduced penalty to 0 points
         *     74 percent have 1-7 points
         *     5 percent have 8-13 pts
         *     1 percent are above 13 pts
         */
        //Classify according to real-life distribution
        Integer penaltyPoints = 0;
        Integer lowerPenaltyLimit = 0;
        Integer upperPenaltyLimit = 0;
        Integer penaltyGroup = (int) (Math.random() * 101);

        if (penaltyGroup == 1) { // 1 percent
                lowerPenaltyLimit = 14;
                upperPenaltyLimit = 20;
        }
        else if (penaltyGroup < 7){ // 5 percent (2..6)
                lowerPenaltyLimit = 8;
                upperPenaltyLimit = 13;
        }
        else if (penaltyGroup < 81){ // 74 percent
                lowerPenaltyLimit = 1;
                upperPenaltyLimit = 7;
        }

        penaltyPoints = lowerPenaltyLimit + (int) (Math.
            random() * (upperPenaltyLimit - lowerPenaltyLimit
            ));

        return penaltyPoints;
}
```

XVI. Certain conditions originating in the perinatal period

XVII. Congenital malformations, deformations and chromosomal abnormalities

XVIII. Symptoms, signs and abnormal clinical and laboratory findings, not elsewhere classified

XIX. Injury, poisoning and certain other consequences of external causes

XX. External causes of morbidity and mortality

XXI. Factors influencing health status and contact with health services

XXII. Codes for special purposes

A.4 MEDICAL INCIDENCE STATISTICS

ICD GROUP	DAYS IN HOSPITAL	FEMALE %	MALE %
I	10.6	1.75	1.85
II	11.0	11.12	11.24
III	10.4	2.95	2.92
IV	12.0	0.57	0.54
V	35.8	7.02	5.75
VI	8.7	7.85	8.33
IX	10.3	10.01	14.88
X	9.5	3.92	4.35
XI	8.1	11.48	12.76
XII	10.7	1.58	1.40
XIII	13.0	17.17	15.51
XIV	7.5	4.51	4.10
XV	7.6	3.26	0.00
XVI	15.6	0.38	0.33
XVII	8.4	0.70	0.62
XVIII	8.3	8.84	8.28
XIX	10.2	6.89	7.14
Total	avg. 11.1	100.00	100.00

Table 7: Medical incidence statistics in private health care. Note that the table of frequencies in the original survey publication is not ordered correctly, with occasional errors in the classification codes (see [PKV 11]).

A.5 HEALTHCARE SERVICE CATEGORIES

The following list of medical service categories is an abridged compilation, based on the uniform assessment scale (Einheitlicher Bewertungsmaßstab, EBM [KBV 11]) and the scale of fees for physicians (Gebührenordnung für Ärzte, GOÄ [BMG 11]) as applied in Germany, and an examplary classification scheme adopted in the United States (as in [WAC 11]).

1. Adult day health
2. Ambulance transport
3. Blood processing/administration
4. Dental services
5. Detoxification
6. Diagnostic services (lab & x-ray)
7. Family planning services
8. Healthcare professional services
9. Hearing care (audiology/hearing exams/aids)
10. Home health services
11. Hospice services
12. Hospital services - inpatient/outpatient
13. Maternity care and delivery services
14. Medical equipment, durable
15. Medical equipment, nondurable
16. Medical nutrition services
17. Nursing facility services
18. Organ transplants
19. Oxygen/respiratory services
20. Personal care services
21. Prescription drugs
22. Prosthetic/orthotic devices
23. Therapy -occupational/physical/speech
24. Vision care (exams/lenses)

[Aagesen 10] Gustav Aagesen & John Krogstie. *Analysis and Design of Business Processes Using BPMN*. In Brocke & Rosemann [vom Brocke 10], pages 213–235. ISBN 978-3-642-00416-2. URL http://dx.doi.org/10.1007/978-3-642-00416-2_10.

[Abramowicz 11] Witold Abramowicz, editor. Proceedings of the 14th International Conference on Business Information Systems (BIS 2011). Springer, June 2011.

[Barjis 08] Joseph Barjis. *The Importance of Business Process Modeling in Software Systems Design*. Science of Computer Programming, Vol. 71, pages 73–87, March 2008. ISSN 0167-6423. URL http://portal.acm.org/citation.cfm?id=1347457.1347664.

[BASt 11] BASt. *Bundesanstalt für Straßenwesen (Federal Highway Research Institute of the German Ministry of Transport, Building and Urban Development)*. Website, 2011. URL http://www.kba.de.

[Bleicher 91] Knut Bleicher. Organisation: Strategien, Strukturen, Kulturen. Dr. Th. Gabler Verlag, Wiesbaden, 2nd edition, 1991. ISBN 9783409315524.

[Blind 03] Knut Blind & Christiane Hipp. *The Role of Quality Standards in Innovative Service Companies: An Empirical Analysis for Germany*. Technological Forecasting and Social Change, Vol. 70, No. 7, pages 653–669, 2003.

[BMG 11] BMG. *Gebührenordnung für Ärzte (GOÄ)*. Bundesministerium für Gesundheit, Regulation, 2011. URL http://www.gesetze-im-internet.de/go__1982/index.html. (In German).

[Brand 95] N. Brand & H. van der Kolk. Workflow Analysis and Design. Kluwer Bedrijfswetenschappen, Deventer, 1995. (In Dutch).

[Daft 09] Richard L. Daft. Organization theory and design. South-Western Cengage Learning, 10th edition, 2009. ISBN 9780324598896. URL http://books.google.de/books?id=CmFjF5tNmuEC.

[Davenport 90] Thomas H. Davenport & James E. Short. *The New Industrial Engineering: Information Technology and Business Process Redesign*. Sloan Management Review, Vol. 31, No. 4, 1990. URL http://www.archive.org/details/newindustrialeng00dave.

[Davenport 93] Thomas H. Davenport. Process innovation: reengineering work through information technology. Harvard Business School Press, 1993. ISBN 9780875843667. URL http://books.google.de/books?id=kLlIOMGaKnsC.

[Davenport 95] Thomas H. Davenport. *The Fad That Forgot People*. Fast Company Magazine, No. 1, 1995. URL http://www.fastcompany.com/magazine/01/reengin.html.

[Davenport 05] Thomas H. Davenport. *The Coming Commoditization of Processes*. Harvard Business Review, No. 83, pages 100–108, June 2005.

[Davenport 10] Thomas H. Davenport. *Process Management for Knowledge Work*. In vom Brocke & Rosemann [vom Brocke 10], pages 17–35. ISBN 978-3-642-00416-2. URL http://dx.doi.org/10.1007/978-3-642-00416-2_2.

[de Vries 97] Henk de Vries. *Standardization – What's in a name?* Terminology, Vol. 4, No. 1, pages 55–83, 1997.

[DESTATIS 11] DESTATIS. *Statistisches Bundesamt Deutschland (German Federal Statistical Office)*. Website, 2011. URL http://www.destatis.de.

[Dumas 05] Marlon Dumas, Wil van der Aalst & Arthur H. M. ter Hofstede, editors. Process Aware Information Systems: Bridging People and Software through Process Technology. Wiley-Interscience, 2005. ISBN 978-0-471-66306-5.

[Gadatsch 10] Andreas Gadatsch. Grundkurs Geschäftsprozess-Management: Methoden und Werkzeuge für die IT-Praxis.

Vieweg + Teubner, Wiesbaden, 6th edition, 2010. ISBN 9783834807625.

[Gaur 06] Harish Gaur & Markus Zirn, editors. BPEL Cookbook: Best Practices for SOA-based integration and composite applications development. Packt Publishing, Birmingham - Mumbai, 2006. ISBN 9781904811336.

[Gladwin 94] Bruce Gladwin & Kerim Tumay. *Modeling Business Processes with Simulation Tools*. In Proceedings of the 26th conference on Winter simulation, WSC '94, pages 114–121, San Diego, CA, USA, 1994. Society for Computer Simulation International. ISBN 0-7803-2109-X. URL http://portal.acm.org/citation.cfm?id=193201.193992.

[Gregoriades 08] Andreas Gregoriades & Alistair Sutcliffe. *A Socio-Technical Approach to Business Process Simulation*. Decision Support Systems, Vol. 45, No. 4, pages 1017 – 1030, 2008. ISSN 0167-9236. URL http://www.sciencedirect.com/science/article/pii/S0167923608000766.

[Hammer 90] Michael Hammer. *Reengineering Work: Don't Automate, Obliterate*. Harvard Business Review, No. July-August, 1990. URL http://www.scribd.com/doc/22596983/.

[Hammer 93] Michael Hammer & James Champy. Reengineering the Corporation: A Manifesto for Business Revolution. HarperBusiness, New York, 1st edition, 1993. ISBN 088730687X.

[Hammer 03] Michael Hammer & James Champy. Reengineering the Corporation: A Manifesto for Business Revolution. HarperBusiness Essentials Series. HarperBusiness Essentials, 2003. ISBN 9780060559533. URL http://books.google.com/books?id=mjvGTXgFl6cC.

[Harrell 95] Charles Harrell & Kerim Tumay. Simulation Made Easy: A Manager's Guide. Industrial Engineering and Management Press, 1995. ISBN 9780898061369. URL http://books.google.com/books?id=siFrQgAACAAJ.

[Kalinowska 07] Dominika Kalinowska, Jutta Kloas & Hartmut Kuhfeld. *Fahrerlaubnisbesitz in*

Deutschland. Berichte der Bundesanstalt für Straßenwesen (see [BASt 11]), Heft M187, August 2007. URL http://www.bast.de/nn_40694/DE/Publikationen/Infos/2007-2006/16-2007.html. (In German).

[KBA 11] KBA. *Kraftfahrt-Bundesamt (Federal Motor Transport Authority)*. Website, 2011. URL http://www.kba.de.

[KBV 11] KBV. *Einheitlicher Bewertungsmaßstab (EBM)*. Kassenärztliche Bundesvereinigung, Regulation, Juli 2011. URL http://www.kbv.de/8156.html. (In German).

[Kilpeläinen 08] Katri Kilpeläinen & Arpo Aromaa. *European Health Indicators: Development and Initial Implementation*. Technical report, European Community Health Indicators Monitoring, 2008. URL http://www.echim.org/docs/ECHIM_final_report.pdf.

[Laguna 11] Manuel Laguna & Johan Marklund. Business Process Modeling, Simulation and Design. Dorling Kindersley, India, 2011. ISBN 9788131761359. URL http://books.google.de/books?id=d-V8c8YRJikC.

[Leymann 00] Frank Leymann & Dieter Roller. Production Workflow: Concepts and Techniques. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2000. ISBN 0-13-021753-0.

[Leymann 10] Frank Leymann, Dimka Karastoyanova & Michael P. Papazoglou. *Business Process Management Standards*. In Brocke & Rosemann [vom Brocke 10], pages 513–542. ISBN 978-3-642-00416-2. URL http://dx.doi.org/10.1007/978-3-642-00416-2_24.

[Modarres 06] Mohsen Modarres. *Predicting and Improving Complex Business Processes: Values and Limitations of Modeling and Simulation Technologies*. In Proceedings of the 38th conference on Winter simulation, WSC '06, pages 598–603. Winter Simulation Conference, 2006. ISBN 1-4244-0501-7. URL http://portal.acm.org/citation.cfm?id=1218112.1218225.

[Niedermann 08] Florian Niedermann. *Development of a method for the integrated handling of process variables and operational data sources*. Master's thesis, Universität Stuttgart, Fakultät Informatik, Elektrotechnik und Informationstechnik, Germany, January 2008.

[Niedermann 10] Florian Niedermann, Sylvia Radeschütz & Bernhard Mitschang. *Deep Business Optimization: A Platform for Automated Process Optimization*. In Witold Abramowicz, Rainer Alt, Klaus-Peter Fändrich, Bogdan Franczyk & Leszek A Maciaszek, editors, Business Process and Service Science - Proceedings of ISSS and BPSC: BPSC'10; Leipzig, Germany, September 27th - October 1st, 2010, volume P177 of *Lecture Notes in Informatics*, pages 168–180. Gesellschaft für Informatik e.V. (GI), September 2010. ISBN 978-3-88579-271-0. URL http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2010-89&engl=.

[Niedermann 11a] Florian Niedermann, Bernhard Maier, Sylvia Radeschütz, Holger Schwarz & Bernhard Mitschang. *Automated Process Decision Making based on Integrated Source Data*. In Abramowicz [Abramowicz 11], pages 1–10. URL http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2011-24&engl=1.

[Niedermann 11b] Florian Niedermann, Sylvia Radeschütz & Bernhard Mitschang. *Business Process Optimization using Formalized Optimization Patterns*. In Abramowicz [Abramowicz 11], pages 1–10. URL http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2011-23&engl=1.

[Niedermann 11c] Florian Niedermann, Sylvia Radeschütz & Bernhard Mitschang. *Design-Time Process Optimization through Optimization Patterns and Process Model Matching*. In Proceedings of the 12th IEEE Conference on Commerce and Enterprise

Computing (CEC), pages 48–55. IEEE Computer Society, November 2011. URL http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2011-26&engl=1.

[Niedermann 11d] Florian Niedermann & Holger Schwarz. *Deep Business Optimization: Making Business Process Optimization Theory Work in Practice*. In Proceedings of the Conference on Business Process Modeling, Development and Support (BPMDS 2011), pages 1–10. Springer, June 2011. URL http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2011-25&engl=1.

[Niedermann 12] Florian Niedermann. *Deep Business Optimization: Concepts and Architecture for an Analytical Business Process Optimization Platform*. PhD thesis, Universität Stuttgart, Fakultät Informatik, Elektrotechnik und Informationstechnik, Germany, 2012. (In progress).

[OASIS 07] OASIS. *Web Services Business Process Execution Language (WS-BPEL) Version 2.0*. Technical report, Organization for the Advancement of Structured Information Standards (OASIS), April 2007. URL http://docs.oasis-open.org/wsbpel/2.0/wsbpel-v2.0.pdf.

[OASIS 11] OASIS. *Organization for the Advancement of Structured Information Standards*. Website, 2011. URL http://www.oasis-open.org.

[OMG 08] OMG. *Business Process Maturity Model (BPMN) 1.0*. Technical report, Object Management Group (OMG), 2008. URL http://www.omg.org/spec/BPMM/1.0/.

[OMG 09] OMG. *Business Process Model And Notation (BPMN) 1.2*. Technical report, Object Management Group (OMG), 2009. URL http://www.omg.org/spec/BPMN/1.2/.

[OMG 11] OMG. *Business Process Model And Notation (BPMN) 2.0*. Technical report, Object Management Group (OMG), 2011. URL http://www.omg.org/spec/BPMN/2.0/.

[Peterson 02] Brian E. Peterson, John W. Jr. Kwant, Vaughn C. Cecil & Wayne A. Provost. *Electronic creation, submission, adjudication, and payment of health insurance claims*. US Patent 6343271, January 2002. URL http://www.patentlens.net/patentlens/patent/US_6343271/en/.

[PKV 11] PKV. *Zahlenbericht der privaten Krankenversicherung 2009/2010*. Survey Report, Revised Edition, January 2011. ISSN 0503-8839. URL https://bestellungen.pkv.de/zahlenberichte/zahlenbericht-2009-2010/. (In German).

[Radeschütz 08a] Sylvia Radeschütz & Bernhard Mitschang. *An Annotation Approach for the Matching of Process Variables and Operational Business Data Models*. In Proc. of the 21st International Conference on Computer Applications in Industry and Engineering (CAINE 2008), pages 1–6, Honolulu, USA, November 2008. The International Society for Computers and Their Applications. URL http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2008-86&engl=.

[Radeschütz 08b] Sylvia Radeschütz, Bernhard Mitschang & Frank Leymann. *Matching of Process Data and Operational Data for a Deep Business Analysis*. In Proc. of the 4th International Conference on Interoperability for Enterprise Software and Applications (I-ESA 2008), Berlin, March 26-28, 2008., pages 171–182, London, March 2008. Springer-Verlag. URL http://www2.informatik.uni-stuttgart.de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2008-05&engl=.

[Radeschütz 09] Sylvia Radeschütz & Bernhard Mitschang. *Extended Analysis Techniques For a Comprehensive Business Process Optimization*. In Proc. of the International Conference on Knowledge Management and Information Sharing (KMIS 2009), Portugal, October 6th-8th, 2009., pages 1–6. Springer, October 2009. URL http://www2.informatik.uni-stuttgart.

de/cgi-bin/NCSTRL/NCSTRL_view.pl?id=INPROC-2009-68&engl=.

[Radeschütz 10] Sylvia Radeschütz, Florian Niedermann & Wolfgang Bischoff. *BIAEditor: matching process and operational data for a business impact analysis*. In Proceedings of the 13th International Conference on Extending Database Technology (EDBT 2010), March 22-26, Switzerland, pages 705–708, 2010.

[Reijers 05a] Hajo A. Reijers. *Process Design and Redesign*. In Dumas et al. [Dumas 05], Chapter 9, pages 207–234. ISBN 978-0-471-66306-5.

[Reijers 05b] Hajo A. Reijers & S. Limam Mansar. *Best practices in business process redesign: an overview and qualitative evaluation of successful redesign heuristics*. Omega, The International Journal of Management Science, Vol. 33, No. 4, pages 283–306, 2005.

[Rosenkranz 10] Christoph Rosenkranz, Stefan Seidel, Jan Mendling, Markus Schaefermeyer & Jan Recker. *Towards a Framework for Business Process Standardization*. In van der Aalst et al. [van der Aalst 10b], pages 53–63. ISBN 978-3-642-12186-9. URL http://dx.doi.org/10.1007/978-3-642-12186-9_6.

[Slob 02] Florens J.C. Slob & Henk J. De Vries. *Best Practice in Company Standardisation*. Erasmus Research Institute of Management (ERIM) Report Series, No. ERS-2002-81-ORG, August 2002. URL http://ssrn.com/abstract=371029.

[Tumay 96] Kerim Tumay. *Business Process Simulation*. In Proceedings of the 28th conference on Winter simulation, WSC '96, pages 93–98, Washington, DC, USA, 1996. IEEE Computer Society. ISBN 0-7803-3383-7. URL http://dx.doi.org/10.1145/256562.256581.

[van der Aalst 04] W.M.P. van der Aalst & Kees Max van Hee. Workflow Management: Models, Methods, and Systems. Cooperative information systems. MIT Press, 2004. ISBN 9780262720465. URL http://books.google.com/books?id=01xW1_Za-I0C.

[van der Aalst 10a] Wil M. P. van der Aalst, Joyce Nakatumba, Anne Rozinat & Nick Russell. *Business Process Simulation*. In Brocke & Rosemann [vom Brocke 10], pages 313–338. ISBN 978-3-642-00416-2. URL http://dx.doi.org/10.1007/978-3-642-00416-2_15.

[van der Aalst 10b] Will van der Aalst, John Mylopoulos, Norman M. Sadeh, Michael J. Shaw, Clemens Szyperski, Stefanie Rinderle-Ma, Shazia Sadiq & Frank Leymann, editors. Business Process Management Workshops, volume 43 of *Lecture Notes in Business Information Processing*. Springer, Berlin / Heidelberg, 2010. ISBN 978-3-642-12186-9.

[Venkatesh 08] Viswanath Venkatesh & Hillol Bala. *Technology Acceptance Model 3 and a Research Agenda on Interventions*. Decision Sciences, Vol. 39, No. 2, pages 273–315, May 2008. ISSN 0011-7315. URL http://dx.doi.org/10.1111/j.1540-5915.2008.00192.x.

[vom Brocke 10] Jan vom Brocke & Michael Rosemann, editors. Handbook on business process management. International Handbooks on Information Systems. Springer, 2010. ISBN 9783642004155. URL http://books.google.com/books?id=UycqSWE_jngC.

[W3C 04] W3C. *Web Services Glossary*. Technical report, World Wide Web Consortium (W3C), 2004. URL http://www.w3.org/TR/ws-gloss/.

[WAC 11] WAC. *Washington Administrative Code WAC 388-501-0065*. Washington State Legislature, 2011. URL http://apps.leg.wa.gov/wac/default.aspx?cite=388-501-0065.

[Weske 01] Mathias Weske, Thomas Goesmann & Rüdiger Striemer. *Analysing, Modelling and Improving Workflow Application Development Processes*. In Software Process Improvement And Practice, pages 35–46, 2001.

[Weske 04] M. Weske, W.M.P. van der Aalst & H.M.W. Verbeek. *Advances in Business Process Management*. Data & Knowledge Engineering, Vol. 50, No. 1, pages 1–8, 2004. ISSN 0169-023X. URL http://www.sciencedirect.

com/science/article/B6TYX-4BJ5XYV-1/2/3e56e6f379c572e74a9f27501bb18730.

[Weske 07] Mathias Weske. Business Process Management: Concepts, Languages, Architectures. Springer Verlag, 1st edition, November 2007. ISBN 3540735216. URL http://www.worldcat.org/isbn/3540735216.

[WfMC 99] WfMC. *Terminology and Glossary, Document Number WFMC-TC-1011 - Issue 3.0*. Technical report, Workflow Management Coalition, 1999. URL http://www.wfmc.org/standards/docs/TC-1011_term_glossary_v3.pdf/.

[WHO 11] WHO. *Classification of Diseases (ICD)*. World Health Organization (WHO) Website, 2011. URL http://www.who.int/classifications/icd.

[Yin 02] Robert K. Yin. Case Study Research: Design and Methods, Third Edition, Applied Social Research Methods Series, Vol 5. Sage Publications, Inc, 3rd edition, December 2002. ISBN 0761925538. URL http://www.worldcat.org/isbn/0761925538.

[zur Muehlen 08] Michael zur Muehlen & Jan Recker. *How Much Language Is Enough? Theoretical and Practical Use of the Business Process Modeling Notation*. In Zohra Bellahsene & Michel Léonard, editors, CAiSE, volume 5074 of *Lecture Notes in Computer Science*, pages 465–479. Springer, 2008. ISBN 978-3-540-69533-2.

All referenced URLs last accessed on August 1, 2011.

ERKLÄRUNG / DECLARATION

Hiermit erkläre ich, dass ich die vorliegende Arbeit selbstständig verfasst habe und dabei keine andere als die angegebene Literatur verwendet habe.

Alle Zitate und sinngemäßen Entlehnungen sind als solche unter genauer Angabe der Quelle gekennzeichnet.

I hereby declare that this thesis is from my own work and effort, and all other sources of information used have been acknowledged.

Stuttgart, Juli 2011

Kemal Tolga Ergin