Deep Business Optimization: A Platform for Automated Process Optimization

Florian Niedermann, Sylvia Radeschütz, Bernhard Mitschang

Institute of Parallel and Distributed Systems, Universität Stuttgart
Universitätsstraße 38, D-70569 Stuttgart, Germany
{Florian.Niedermann,Sylvia.Radeschuetz,Bernhard.Mitschang}
@ipvs.uni-stuttgart.de

Abstract: The efficient and effective design, execution and adaption of its core processes is vital for the success of most businesses and a major source of competitive advantage. Despite this critical importance, process optimization today largely depends on manual analytics and the ability of business analysts to spot the "right" designs and areas of improvement. This is because current techniques typically fall short in three areas: they fail to integrate relevant data sources, they do not provide optimal analytical procedures and they leave it up to analysts to identify the best process design. Hence, we propose in this paper a platform that enables (semi-)automated process optimization during the process design, execution and analysis stages, based on insights from specialized analytical procedures running on an integrated warehouse containing both process and operational data. We further detail the analysis stage, as it provides the foundation for all other optimization stages.

1 Introduction

In this section, we first briefly discuss the reasons for and our understanding of process optimization within Business Process Management (BPM) and the role of analytics within this context. Based on this, we briefly analyse weaknesses in today's optimization tools and techniques and introduce our *deep Business Optimization Platform* as a means to address these weaknesses.

1.1 Business Process Optimization

In the past decade, businesses have moved from tweaking individual business functions towards optimizing entire business processes. Originally triggered by the growing significance of Information Technology and increasing globalization [HC93], this trend has - due to the increasing volatility of the economic environment and competition amongst businesses - continued to grow in significance. Hence, achieving superior process performance through BPM is nowadays one of the key sources of competitive advantage for businesses.

Stage	"Traditional" approach	dBOP approach	Benefits
Design	•No data or simulated data used as foundation	•New process is linked to existing data of similar processes	•Allows transfer of experiences from existing processes
	•Process design depends solely on analyst	•Pattern catalogue supports application of best practices	•Speeds up design process and improves process quality
Execution	•Decision making and staff assignment based on static models/roles	•Dynamic decision making based on analytics results	•Improves quality of decisions and staff assignment
	•Decisions only consider process data	•Decisions augmented by operational data	•Improves quality of decisions and staff assignment
Analysis	•Analysis only considers process execution data	•Process data integrated with operational data	•Enables discovery of "deep" insights that are not visible from process data
	•Analytics functions restricted to displaying basic information	•Specialized Data Mining and OLAP procedures used to extract "interesting" process properties	•Speeds up the analysis process and helps with discovering previously unknown optimization potentials

Figure 1: Comparison of traditional and *dBOP* optimization approach

Historically, process optimization in BPM has its roots in Business Process Reengineering (see [HC93], [Cha95]). Within this context, process optimization was often considered to be an exercise, where one static (process) model would be transformed into another static model. Within our work, we also include dynamic optimization - that is, optimization during process execution - so that we (similarly to e.g., [WVdAV04]) arrive at the following three stages of optimization within BPM:

1. **Design**: Design refers to determining the a priori (i.e., before execution) structure of the process. The goal is to design an optimal process based on "best practice" knowledge and experience/data from similar processes.
2. **Execution**: During the process execution, the goal of the optimization is to make optimal choices within the given process structure (e.g., allocation of optimal resources).
3. **Analysis**: The goal of this a posteriori redesign of the process is to change its structure in order to achieve optimal results with respect to the execution results.

1.2 Motivation for a Deep Business Optimization Platform

Despite the importance of process optimization, it is still often conducted in an ad hoc manner. Typically, when designing or analyzing a process, analysts try to get as much data about the process as possible (often in an unstructured way). Then, they try to "find" deficiencies as well as implement appropriate optimizations. There are a number of challenges associated with this approach that are at least partially linked to missing capabilities of current tools (such as [Ora10] and [IBM10]).

First, there is a high chance that even a capable business analyst is not able to identify all improvement levers, especially in complex processes. This is partially due to the fact that BPM tools (and for that matter, most books on optimization, as discussed in [SM08]) offer no guidance as to how to actually change the process to achieve optimal results. Second, as current tools typically fail to provide data integration capabilities, the analysis does not take into account all relevant data sources [RML08]. This might mean that some improvement opportunities are missed, since they can't be inferred from a single data source. Third, during the design and execution stage, optimization decisions typically rely on either static models or artificially generated data from simulation. Hence, experiences and improvements already made in existing processes might not be applied to new processes. Finally, since analysts have to "find" all the improvement areas themselves, the analysis requires significant time and resources. This incurs both costs for the analysis itself and opportunity costs due to delays in the implementation of the optimized process [Hac02].

To address these challenges, a *deep*[1] *Business Optimization Platform (dBOP)* is required, that supports optimization within BPM during process design, execution and analysis. Figure 1 gives a brief comparison of this approach to "traditional" optimization approaches.

In this paper, we first sketch the architecture and the requirements for such a platform (Section 2). Then, we move on to provide the details about the analysis stage of the optimization cycle: in Section 3, we discuss methods and techniques used for integrating and analyzing the data pertaining to the process. Next, we take an in-depth view in Section 4 on how the actual process optimization is conducted during the analysis stage. In Section 5 we discuss this paper in the context of related work before providing the conclusion and the outlook on future work in Section 6.

2 The Deep Business Optimization Platform

In this section, we first provide a business scenario that is used as a guiding example for the *dBOP* and as an example throughout the paper. Then we briefly show how *dBOP* supports all three optimization stages before we take a closer look at the analysis stage.

2.1 Guiding Example

Throughout the paper, we use the sales process of the *Car Rental Company Ltd.* as schematically shown in Figure 2 as our example.

The company is using BPEL to orchestrate the process and data flow between its backend systems. Now, that the focus of the company's optimization efforts starts to shift towards

[1] Note that throughout the paper, the term "deep" refers to the fact that we base our optimization on an integrated view of all operational data (i.e., data that is generated by some application, such as an ERP system, outside the processs engine itself) and process data that pertains to the process. In contrast to optimization approaches relying exclusively on (actual or simulated) process data, this enables us to find a broader range of optimization levers.

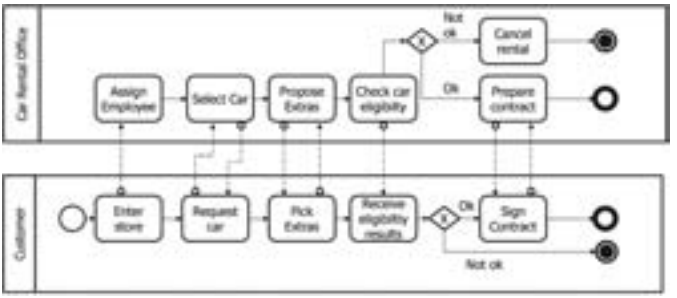

Figure 2: Schematic Car Rental sales process in BPMN

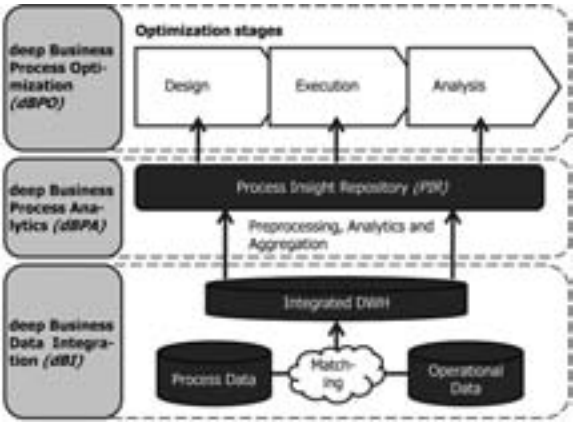

Figure 3: *deep Business Optimization Platform* overview

entire business processes, there are some challenges that cannot be adequately addressed by either the company's Data Warehouse or the process data of the BPEL engine employed.

Fueled by above-benchmark process durations and costs, the company suspects that there are problems in the process flow as well as the staff allocation during the process execution. Further*Car Rental Company Ltd.* is considering moving into a new business, that is, the rental of motorcycles. For that purpose, they have designed a new sales process. Now, they would like to leverage experiences from their car rental process, but so far have not been able to link the two.

In the remainder of this paper, we show how our approach helps to address these challenges, with a specific focus on the process analysis.

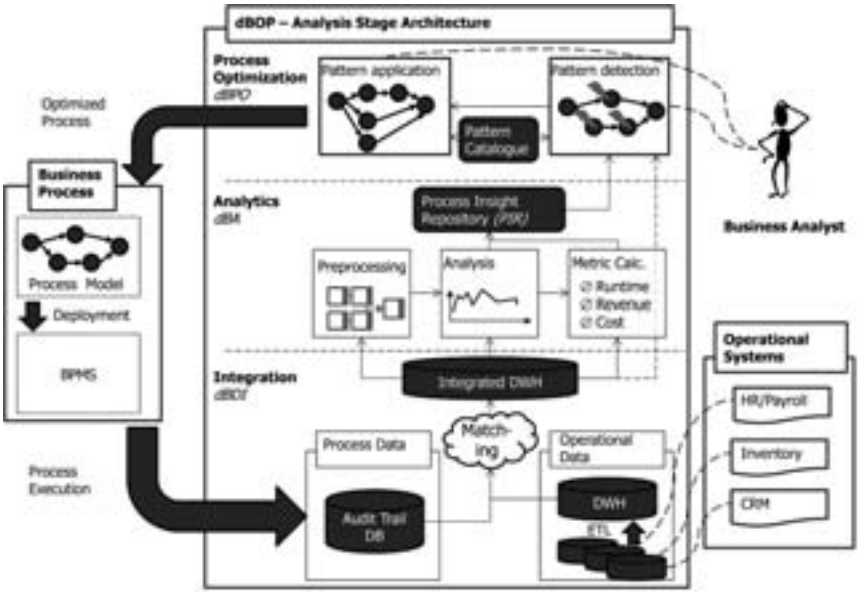

Figure 4: Overview of the *dBOP* analysis stage architecture

2.2 Platform capabilities and architecture

As Figure 3 shows, the *dBOP* is made up of three main architectural layers: The *deep Business Data Integration (dBDI)* layer matches heterogeneous data sources so that the process analysis can extend to all relevant data (for details, see Section 3.1). Building on this integrated data, the *deep Business Analytics (dBA)* layer extracts process optimization-relevant "insights" (see Section 3.2). The results are stored in the *Process Insight Repository (PIR)*. Finally, the *deep Business Process Optimization (dBPO)* layer conducts the actual process optimization based on insights from the *PIR*. The mode, steps and prerequisites of the *dBPO* depend on the stage (design, execution or analysis) during which the optimization is conducted. The focus of this paper is on the analysis stage, in which the *dBPO* uses existing execution data to determine the most feasible optimizations. During the design stage, a similar approach is employed after prior process matching (see [NRM10] for details). During the execution stage, we use the *PIR* to dynamically solve decision and selection problems, e.g., by allowing for attribute-based instead of role-based resource allocation.

2.3 Analysis stage architecture overview

As mentioned before, the focus of this paper is on the analysis stage of the *dBOP*. This is why in this section, we briefly introduce the *dBOP* analysis stage components that provide an implementation of the concepts discussed in Section 2.2.

The analysis stage architecture as shown in Figure 4 consists of three main components:

1. It provides a uniform view on all relevant data pertaining to the process through the *dBDI* layer. This includes process and operational data, as well as the process model (Section 3.1).

2. Building on the results of *dBDI*, optimization-focused insights are generated and stored in the *PIR* using the preprocessing and analytics techniques that are part of the *dBA* layer (Section 3.2).

3. Leveraging the insights gained by integrating and analyzing the process data, the *dBPO* layer is tasked with performing the actual optimization. For that purpose it uses a set of "optimization patterns" stored in the *Pattern Catalogue* that help analysts with detecting and addressing deficiencies in the process (Section 4).

3 Deep Business Data Integration and Analysis

This section introduces the methods and technologies used for the integration and the analysis of the process-relevant data. Both the integration and the analysis layers are necessary in order to provide the optimization with the required insights.

3.1 Deep Business Data Integration

As discussed in Section 2.3, the goal of the *dBDI* layer is to provide a standardized, uniform way of accessing process and operational data together. Since this layer has been discussed intensively (e.g., [RML08] and [RM09]) and implemented (see for instance [RNB10]) in our previous work, we only give a brief overview of the steps required:

1. **Annotation and matching of the process and operational data models:** As the first step of the *dBDI* layer, we need to integrate the data models of the process data and the operational data. For that purpose, we have developed a matching method and an editor that allows for manual, annotation-based/semi-automatic and automatic matching of the data models.

2. **Audit data standardization and mapping the integrated data model to the instance data:** Since the audit data generated during the process execution is typically stored in proprietary format, some standardization of the data is necessary in order to allow for integration with a number of different platforms. To achieve this task, we use a tailored version of the Business Process Analytics Format (BPAF), a standard published in [WfM09]. This step further includes transferring the matching of the operational and process data models to their concrete instances.

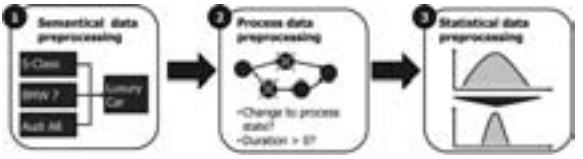

Figure 5: Data preprocessing for the Deep Business Analysis

3.2 Deep Business Analytics

In Section 2, we have already briefly presented the *dBA* layer. The goal of this layer is to provide a concise view on the aspects of the process (such as activity execution times, waiting times, relevant process attributes) that might be relevant for optimization. To achieve this goal, several Data Mining techniques and algorithms are adopted to the specific requirements of integrated process and operational data. First, however, the data needs to preprocessed to provide an optimal input for the analysis algorithms [HK06].

3.2.1 Data preprocessing

Since the field of data preprocessing is already covered extensively in Data Mining literature and research [KKP06], we discuss only the specifics of our data preprocessing approach shown in Figure 5:

1. **Leverage of semantics in the preprocessing of attributes:** A central element of our approach to data integration (see Section 3.1) is the semantic annotation of both the operational and the process data. This annotation is used as domain knowledge during preprocessing, e.g., to cluster types of cars or car rentals.
2. **Preprocessing of process activities and other process elements:** To reduce dimensionality, we remove process elements from the Data Mining input that are not result-relevant (e.g., have no influence on the process result or its execution time)
3. **Statistical data preprocessing:** After the specific steps mentioned above have been concluded, standard (mainly statistical) preprocessing steps such as data cleaning and data normalization are applied.

3.2.2 Conducting the analysis

The key step of the *dBA* is the application of Data Mining algorithms to the integrated, preprocessed data in order to extract the insights required for the optimization. For this purpose, techniques and algorithms from several of the areas listed in [HK06] are used and adapted (see [RM09]). The techniques most widely used are:

- **Concept and class description**, mainly through the calculation of key process metrics such as minimum, maximum, average and median activity duration.

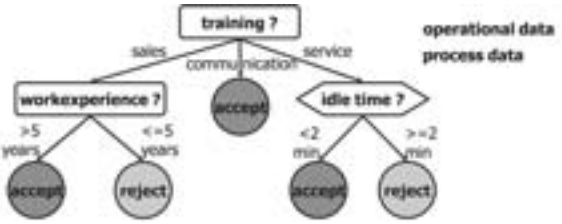

Figure 6: "Predict Flow" pattern: sample usage of classification trees [RM09]

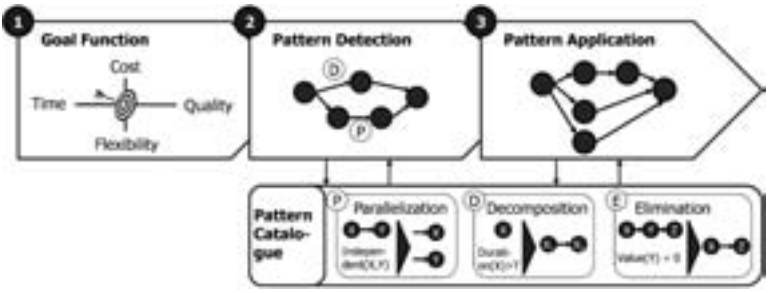

Figure 7: The optimization process

- **Mining frequent patterns**, associations and correlation by association rule mining and multiple linear regressions. This is used to determine the factors that influence process and activity outcomes most. This is e.g. used for the "Resource Allocation" pattern, where Data Mining is used to determine the "optimal" resource (e.g., the best possible salesman in the Car Rental example) to be used for a certain task. For more information on optimization patterns, see Section 4.2.
- **Classification and prediction** to predict the class label of objects, e.g. by classification trees and neural networks. This can be used to predict the further flow of the process at any given time, as is, e.g., required for the "Predict Flow" pattern shown in Figure 6.

4 Deep Business Process Optimization

After the analysis of the process and the operational data has been concluded, all the prerequisites for conducting the process optimization are fulfilled. Now, the business analyst can use the *dBPO* to (semi-)automatically detect and apply a set of optimization patterns.

4.1 Optimization methodology

As Figure 7 shows, the optimization methodology consists of three main steps:

1. **Goal function definition:** Most organizations do not have a single goal, but rather have to work in a complex goal system (e.g., cost, quality, revenue, flexibility) together with associated constraints (e.g., utilization needs to be kept below x%). Hence, the analyst first has to pick the optimization goal as well as any constraints that should be considered during the optimization.
2. **Pattern detection:** Identification of potential optimizations and interaction with the business analyst as to if and how to apply them.
3. **Pattern application:** Rewriting of the process to implement the desired optimizations, including any manual process modifications done by the business analyst.

After the optimization process has been concluded, the new, optimized process model is returned.

4.2 Optimization patterns

In this section, we briefly discuss the basic foundations of optimization patterns and the *Pattern Catalogue* before moving on to a concrete example, the "Automated Approval" pattern.

4.2.1 Foundations

Process optimization patterns are a formalization of (typically verbally described) "best practice" techniques for the optimization of processes (as found for instance in [Rei05]) with respect to one or several goal functions. A pattern consists of a detection and an application component.

The detection components is made up by a(formal) description of the process structure and the analysis results that indicate that the application of the pattern might be beneficial. The application component is the specification of the transformation logic that is required to achieve the desired optimization.

The key to a successful application of the pattern is a high degree of automation and formalization. For this reason, the patterns need to be described in formal language that shifts as much work as possible from the business analyst to the optimizer.

4.2.2 Pattern Catalogue

The core of the *dBPO* layer is formed by the *Pattern Catalogue* that prescribes both the pattern detection and application approach for the available optimization patterns.

Pattern	Description	Scope	Automation	Stage	Data required
Parallelization	Parallelization of sequential activities	Structure	Semi-automated	Analysis, Design	Instances, process
Elimination	Removal of a non-value adding activity	Activity	Semi-automated	Analysis, Design	Instances, process and operational
Decomposition	Splitting up of overly lengthy activities	Activity	Detection: Semi-automated, Application: Manual	Analysis	Instances, process and operational
Res. Allocation	Dynamic allocation of optimal resources	Resource	Automated	Execution	Instances, process and operational

Figure 8: *Pattern Catalogue* excerpt

Within this catalogue, patterns are distinguished by several different criteria, such as optimization scope (e.g., (sub-)processes or single activities), degree of automation/required analyst input, time of application (design, execution or analysis time) and data requirements.

Based on these criteria and an extensive literature search, we have defined a set of now more than 20 optimization patterns, some of which are shown in Figure 8.

4.2.3 Example application: "Automated Approval" pattern

In this section, we briefly illustrate the optimization process with the "Automated Approval" pattern. This pattern refers to a frequent situation in business processes, where a human actor has to manually approve whether the prerequisites for a certain process step are met (in our example, this would be the approval that the customer is eligible for the desired car). Such an approval is typically associated with significant human involvement which usually equates high costs and long delays. As we want to optimize process time (without further constraints), the optimization pattern is applied as shown in Figure 9

1. The Pattern "Automated Approval" is selected from the *Pattern Catalogue*.
2. The process is scanned for all decisions whose execution, on average, show a significant impact w.r.t. the chosen process goal function (in our case: time/process duration). In this case, the "Check Car Eligibility" activity increases the process duration by 10 minutes on average.
3. A query to the *PIR* is used to find out the process and operational attributes that statistically influence the decision most. In our example, all recurring customers with > 5 rentals or those customers seeking to rent an economy car are nearly always approved.

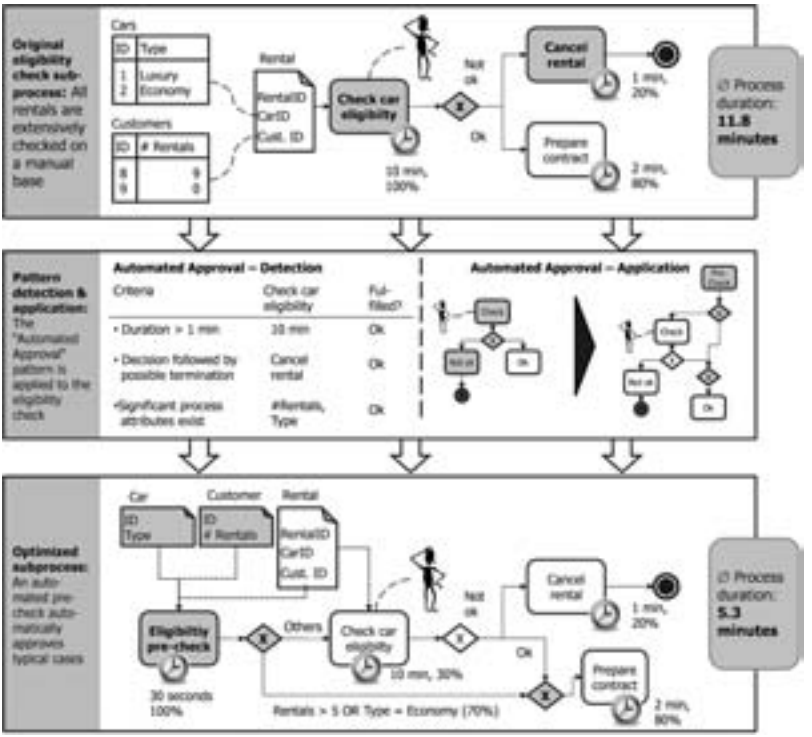

Figure 9: Applying the "Automated Approval" pattern

4. Once these are determined, an (automated) decision is inserted just before the manual decision. Since the automated decision is able to handle 70% of the decisions, the process duration is significantly shortened.

5 Related work

For the discussion of related work, it is helpful to distinguish between work similar to the *dBOP* as a whole and topics related to the tasks of the respective architecture layers.

Looking at the platform as a whole, [CCDS04] take a similar approach in the sense that they are using models similar to our patterns to achieve a higher level of insights when analyzing processes, however, without concrete support for optimization. Other work focuses on providing better optimization and design tools [BKKO06], however, without leveraging execution data.

Considering the integration of heterogeneous data sources as required for the *dBDI*, this is the classical domain of schema matching approaches such as [ADMR05] which, however, sometimes struggle with the specifics of process data. [DH05] explores the possibilities of using semantics for integrating operational data sources, an approach that we have extended to the specific requirements of matching operational and process data. Looking at

commercial products, IBM Information Server supports the integration of heterogeneous data sources, however, the process is not well-suited for process data.

With regards to the specialized data processing and analytics required for the *dBA*, this can be seen as an application of Business Process Analytics [zMS09]. However, this rather new area has in many cases not reached the maturity necessary to provide actual algorithms to be used during the analysis. We hence leverage classical Data Mining [HK06] and data preprocessing approaches [KKP06] and custom-tailor them to the specifics of process data, such as high dimensionality or specific optimization goals.

The optimization itself has so far been considered a task that has to be done largely manually by the business analyst. Hence, related work typically focuses on describing the basic concepts and benefits of business process optimization/reengineering such as [Cha95] or [HC93], providing guidance for the optimization process [SM08] or discussing single optimization techniques [Rei05] (exceptions to this pattern, such as [VdA01] are typically restricted to single, highly specific process types). Looking at the tool landscape, applications such as [IBM10] typically focus on providing somewhat processed information about actual or expected process behavior (i.e., simulation), however, without guidance as to how to apply it to a concrete process optimization.

Overall, the approach of a system that automatically adapts according to feedback from its execution can be conceptually seen as an application of cybernetics [Wie48] to BPM.

6 Conclusion and future work

This paper demonstrated 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. Our *dBOP* platform helps to address this challenge by providing an environment for the (semi-) automated optimization of processes:

- The *dBOP* integrates process and operational data, as well as any other required data source. Hence, our analysis of the process is able to uncover "deep" insights that are not readily visible when looking at single data sources.
- Instead of just querying the process database, we use customized Data Mining techniques and algorithms to extract insights from the data available to us. Instead of just looking at the exhibited process behavior (*What* is happening?), we are able to examine the factors that might be causal for certain behaviors (*Why* is it happening?).
- Through the pattern catalogue, aan analyst using the platform is not just presented with information, but also given concrete guidance as to how to use it to improve the process and increase business value. Hence, the platform presented in this paper allows for a fast and cost-efficient process optimization that ensures high process quality while taking into account different business goals and constraints.

In our future work, we will refine the algorithms and tools used by the *dBOP*. Further, we will extend our approach to the design and execution stages as outlined in Section 2.2 and conduct a thorough empirical validation of the various aspects of the platform.

References

[ADMR05] D. Aumüller, H.H. Do, S. Massmann, and E. Rahm. Schema and ontology matching with COMA++. In *Proceedings SIGMOD*, 2005.

[BKKO06] S. Betz, S. Klink, A. Koschmider, and A. Oberweis. Automatic user support for business process modeling. In *Proceedings SBPM*, pages 1–12, 2006.

[CCDS04] M. Castellanos, F. Casati, U. Dayal, and M.C. Shan. A comprehensive and automated approach to intelligent business processes execution analysis. *Distributed and Parallel Databases*, 16(3):239–273, 2004.

[Cha95] J. Champy. *Reengineering Management*. HarperCollins, 1995.

[DH05] A.H. Doan and A.Y. Halevy. Semantic integration research in the database community: A brief survey. *AI magazine*, 26(1):83, 2005.

[Hac02] R. Hackathorn. Minimizing Action Distance. *DM REVIEW*, 12:22–23, 2002.

[HC93] M. Hammer and J. Champy. *Reengineering the corporation: a manifesto for business revolution*. Brealey, London, 1993.

[HK06] J. Han and M. Kamber. *Data mining: concepts and techniques*. Morgan Kaufmann, 2006.

[IBM10] IBM. IBM WebSphere Business Modeler Advanced,, 2010. `http://www-01.ibm.com/software/integration/wbimodeler/advanced/`.

[KKP06] SB Kotsiantis, D. Kanellopoulos, and PE Pintelas. Data preprocessing for supervised learning. *International Journal of Computer Science*, 1(2):111–117, 2006.

[NRM10] F. Niedermann, S. Radeschütz, and B. Mitschang. Design-time Process Optimization through Optimization Patterns and Process Model Matching. In *Proceedings of the 12th IEEE Conference on Commerce and Enterprise Computing (to appear)*, 2010.

[Ora10] Oracle. Oracle Business Process Analysis Suite, 2010. `www.oracle.com`.

[Rei05] H. Reijers. Process Design and Redesign. *Process-Aware Information Systems. Bridging People and Software through Process Technology*, pages 363–395, 2005.

[RM09] S. Radeschütz and B. Mitschang. Extended Analysis Techniques For a Comprehensive Business Process Optimization. In *Proceedings KIMS*, 2009.

[RML08] S. Radeschütz, B. Mitschang, and F. Leymann. Matching of Process Data and Operational Data for a Deep Business Analysis. *Enterprise Interoperability III*, 2008.

[RNB10] S. Radeschütz, F. Niedermann, and W. Bischoff. BIAEditor - Matching Process and Operational Data for a Business Impact Analysis. *Proceedings EDBT*, 2010.

[SM08] A. Sharp and P. McDermott. *Workflow Modeling: Tools for Process Improvement and Applications Development*. Artech House Publishers, 2008.

[VdA01] WMP Van der Aalst. Re-engineering knock-out processes. *Decision Support Systems*, 30(4):451–468, 2001.

[WfM09] WfMC. Business Process Analytics Format - Draft Specification. 1.0, Document Number WFMC-TC-1025, 2009. `http://www.wfmc.org`.

[Wie48] N. Wiener. *Cybernetics: Control and communication in the animal and the machine*. Cambridge, MA, MIT Press, 1948.

[WVdAV04] M. Weske, WMP Van der Aalst, and HMW Verbeek. Advances in Business Process Management. *Data & Knowledge Engineering*, 50(1):1–8, 2004.

[zMS09] M. zur Mühlen and R. Shapiro. Business Process Analytics. *Handbook on Business Process Management*, 2, 2009.