

# **FROM LEGISLATION TO POTENTIAL COMPLIANCE VIOLATIONS IN BUSINESS PROCESSES – SIMPLICITY MATTERS**

*Research*

Höhenberger, Steffen, University of Münster, Münster, Germany,  
steffen.hoehenberger@ercis.uni-muenster.de

Riehle, Dennis, University of Münster, Münster, Germany,  
dennis.riehle@ercis.uni-muenster.de

Delfmann, Patrick, University of Koblenz-Landau, Germany,  
delfmann@uni-koblenz.de

## **Abstract**

*The compliance of businesses is a highly relevant topic for companies of all sectors. Compliance comprises all necessities for obeying legal regulations as well as mandatory norms, and violations can entail painful penalties. Since nearly all companies base their daily business on the execution of business processes (consciously or unconsciously), these business processes also have to be compliant. Hence, as part of business process management, business process compliance becomes increasingly important for companies. However, the checking of business processes for their compliance is – against the backdrop of many existing business processes and incessantly changing regulations – not an easy task. Several research approaches have been developed for supporting compliance checking. A common way is to examine the models of the business processes automatically to detect compliance violations with predefined patterns that represent possible violations. Unfortunately, only the fewest approaches are actually applied by companies due to high technical restrictions or difficult handling. The presented graph-theoretic approach strives to overcome this with a high level of simplicity and the applicability to real-world process models. For that, a real legislative passage is gradually transformed into an automatically searchable compliance pattern and applied to real process models.*

*Keywords:* Business Process Compliance, Compliance Patterns, Business Process Modeling, Simplicity.

## 1 Introduction

Business processes get increasingly into the focus of companies. According to a study of PwC (2011), business process management (BPM) is a topic of high importance which contributes to a company's success to a significant extent. However, the field of (business) process management is – due to its huge variety of facets, subdomains and involvements to other business fields – not as easy to delimit, to structure or even to describe as one would think. Indeed, the eponymous and leading element is the process itself, which is defined as “a completely closed, time-logical sequence of activities that are required for working on a process-oriented relevant business object” (Becker and Kahn, 2011, p. 6). Further, a business process is “a special process that is directed by the business objectives of a company and by the business environment” (Becker and Kahn, 2011, p. 6). The definitions imply that the management of business processes is far more than only executing the processes. Business process management is “an integrated system for managing business performance by managing end-to-end business processes” (Hammer, 2010, p. 4–5). This system becomes vivid by, e.g., means of models for BPM as the BPM lifecycle with its phases design, configuration, enactment and diagnosis (van der Aalst et al., 2003; Weske, 2012). The procedure model of Becker et al. (2012b) states the preparation of process modeling, strategy and organizational frame, as-is modeling and process analysis, to-be modeling and process optimization, design of a process-oriented organizational structure, process implementation and continuous process management. Against that, Rosemann and vom Brocke (2010) structure the field of BPM based on six core elements of BPM, namely strategic alignment, governance, methods, information technology, people and culture.

Within this wide field of BPM, a very important part that can be located in any of the models is BPM compliance. Besides high efficiency and minimized costs, business processes have to adhere to a constantly growing number of incessantly changing legal requirements and voluntary regulations. The observance of those regulations is called business process compliance management (BPCM). BPCM has the objective to make and to keep business processes compliant. Sadiq et al. (2007, p. 149) describe compliance as “ensuring that business processes, operations and practice are in accordance with a prescribed and/or agreed set of norms”. In large but even in small companies, the number of business processes that have to be compliant is very high. Hence, “compliance management has become a significant concern for organizations given increasingly onerous legislative and regulatory environments” (Ghose and Koliadis, 2007, p. 169) and the assurance of the compliance of business processes is one of the big challenges companies have to cope with (Rinderle-Ma et al., 2008). Sadiq et al. (2007, p. 149) also claim that “the importance of compliance has dramatically increased over the last few years for businesses in several industry sectors.”

Companies are in demand for high flexibility in their processes. However, business processes are often rigid, and changes can only be implemented slowly and in connection with high costs. In order to address these issues, Becker et al. (2012a) state that “automation support of compliance management is a field garnering increasing attention in Information Systems research.” Automation – for the time being regardless of its concrete embodiment – seems to become inevitable for keeping business processes compliant. There exist several approaches that aim to support the BPCM of companies. While some of the approaches focus the automatic analysis of already performed process executions, others try to prevent compliance violations during the design phase of business processes by analyzing corresponding process models. However, most of the developed approaches stay in the institutes of their developers for mainly two reasons: the approaches are too difficult or laborious to use, or they cannot be applied to real-world process models due to, for example, a lack of generalizability. Additionally, the approaches are often not freely available and thus, are not widely used and tested.

The approach presented in this work aims to provide a remedy for this issue. It was mainly developed during a research project regarding compliance checking in business process models with a focus on the ease of use. In general, we aim to represent legal requirements by so-called compliance patterns and search for these patterns in process models. The compliance pattern is developed in a way that it is

machine-readable in order to examine the process models automatically. From a technical point of view, the approach grounds on graph theory. A graph-theoretic algorithm executes the searches whose implementation is based on a combination of subgraph isomorphism and homeomorphism. By the help of a standardized pattern development procedure, we structurally analyzed several German legislation texts and developed matching patterns within the research project. Subsequently, the patterns were searched in all process models of a German bank service provider (project partner) by using the named algorithm. Finally, the results have been evaluated and filtered by process experts and handed over to domain experts for further assessments.

In order to address the above-mentioned barriers of too difficult and not real-world applicable automation approaches, our work includes the whole procedure of compliance checking from the pattern development to the review of results. Therefore, section 2 gives an introduction and an overview of existing compliance checking approaches and explains this work's approach in more detail. Section 3 shows up existing compliance patterns in the literature and proceeds by developing a compliance pattern gradually. We start with reviewing a German legislation text and end up with a machine-readable compliance pattern. Especially in this section, the simplicity is focused. In addition to the simplicity, section 4 focuses the applicability of the approach to real-world process models by providing returned results for the developed pattern from a real-world application. Eventually, we conclude this work with section 5 by summarizing, stating advantages and limitations and giving an outlook for further research.

## 2 The Checking Approach

Business process management and business process compliance (BPC) are widely discussed topics in the literature. The topic of compliance, general management approaches as well as recommendations for the improvement of business process compliance are widely spread among nearly all steps of the mentioned BPM models in section 1. Besides, a multitude of more operational approaches for detecting compliance violations within particular business processes exists (Becker et al., 2012a; Elgammal et al., 2014). The operational approaches differ in their time of application. While some approaches intend to perform compliance checking after a process is executed (e.g., based on process logs), others focus the checking before a process is executed in order to prevent compliance violations from the beginning (e.g., based on process models) (El Kharbili et al., 2008; Becker et al., 2014). We focus on the latter ones and – according to the distinction of Kharbili et al. (2008) – on the design-time checking in order to be able to apply the compliance checking to real-world process models.

Thus, the pattern-based BPC checking we apply can be positioned within the design phase of the BPM life cycle, as we tend to detect compliance violations right in the design phase before their execution. Within the procedure model of Becker et al. (2012b), our approach is located in the as-is modeling and process analysis phase. Corresponding to this, Schwegmann and Laske (2012) name the identification and documentation of weaknesses and improvement potential, which is similar to the detection of compliance violations. Regarding the core elements of BPM by Rosemann and vom Brocke (2010), the placement is a bit more complex and grainy. Due to the focus on the analysis of process models, our approach can be seen in the element *methods*, yet – due to its character as a semi-automatic approach – it also belongs to the *information technology* element. In both, especially the capability of process design and modeling is affected. However, this is not to say that these placements are closed, and our approach, either in its entirety or only in parts (e.g., only the patterns without technical support) cannot be helpful in other phases as, e.g., the process improvement due to facilitated dissemination of compliance knowledge.

In the context of pattern-based BPC checking, most approaches have in common that they analyze a business process model for specific occurrences (so-called patterns) that represent a legal violation. For doing so, they use a query language that executes this analysis and provides feedback. This procedure is called pattern matching. The feedback can be – dependent on the chosen approach – only

a statement that a process contains a violation or not, or a concrete representation of the detected occurrence. Figure 1 shows a simple example of the latter.

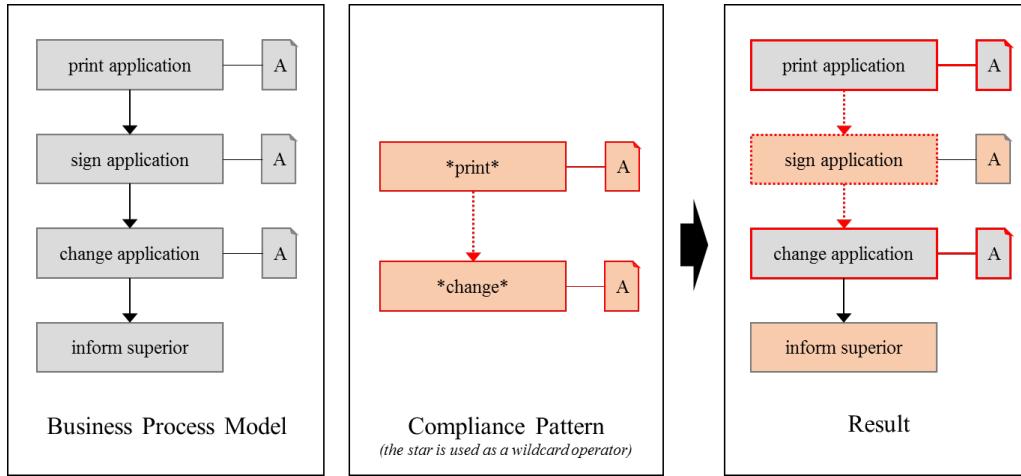


Figure 1. Pattern matching example.

Furthermore, the approaches differ in several additional points. Many approaches only work for a specific modeling language; others only allow dealing with particular structures on which a pattern can be based on. Becker et al. (2012a) classify existing checking approaches according to their generalizability regarding the modeling technique and different compliance rules, Delfmann et al. (2015b) compare different query approaches based on expressiveness, and Ly et al. (2015) elaborate a framework for necessary functionalities of compliance monitoring approaches and check several approaches against these functionalities. While the first two comparisons are directed at analyzing process models, the work of Ly et al. (2015) concentrates on the monitoring of already executed processes based on process logs.

The approach used in this work was chosen due to its possible application to arbitrary modeling language and its flexibility to process various compliance patterns of (partly) high complexity. Since legislation consists of a multitude of different formulations, the approach has to be as less restrictive and as simple and imaginable as possible. A detailed comparison of existing approaches is presented in Delfmann et al. (2015a). In addition, even though the compliance monitoring functionality framework of Ly et al. (2015) is focused on logs and less on process models, the taken approach fulfills or partly fulfills many criteria of the framework (e.g., qualitative time-conditions or explicit non-atomic activities).

Our approach is named Diagrammed Model Query Language (DMQL) and is based on graph theory (Delfmann et al., 2015a). Process models are recognized as graphs consisting of vertices and edges. A vertex stands for an activity or an event of a process model. Additionally, attributes (also symbolized as vertices) are connected to the basic vertices to represent, for instance, a document, an employee or an IT system. Attributes can also be used to enrich an activity by including additional information as, for instance, the combination of a procedure (verb; e.g., “print”) and an object (noun; e.g., “application”) to describe an activity instead of using simple captions without any standardized structure. The kind of an element that a vertex actually represents is fixed by its type (e.g., a basic vertex could be of the types *activity* or *event*, attributes could be of the types *procedure*, *document*, etc.). The modeling language defines which types are possible. Furthermore, the value of a vertex sets its concrete content (e.g., “print” or “A”). An edge connects two vertices and is directed or undirected. Figure 2 provides an insight how a simple process model consisting of activities, edges and attributes is considered from a graph-theoretical point of view.

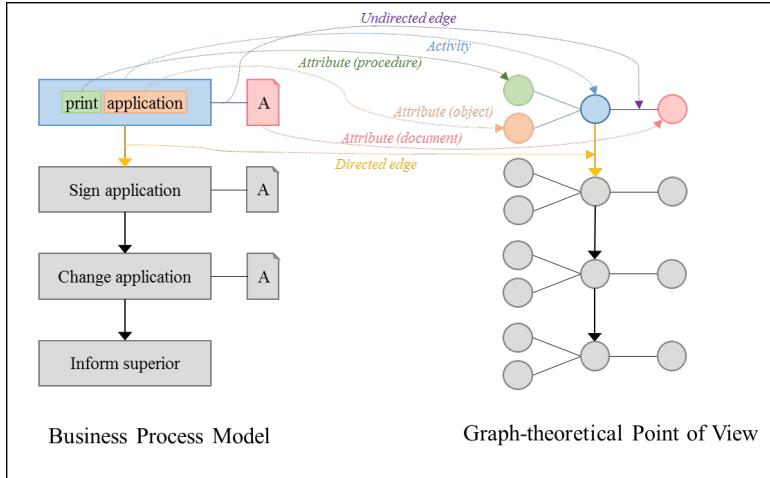


Figure 2. Graph-theoretical point of view.

After converting a process model into a graph, the pattern search is executed by the algorithm. DMQL combines subgraph isomorphism with configurable subgraph homeomorphism, which enables the user to search for patterns that include undirected edges, specific types of edges and nodes, labels of edges and nodes, homeomorphic edges (paths) and multiple edges between two nodes as well as loops. Furthermore, algebraic and Boolean expressions can be evaluated. In simple terms, the basic steps are as follows: after selecting a random pattern node and getting all of its already mapped neighbors, their connecting edges are regarded additionally to their representation in the process model graph. In case an edge is isomorphic, the unmapped neighbors of the representation node are evaluated with their connections, and their local and global mapping feasibility is checked. In case an edge is homeomorphic, a path deep search is executed to evaluate the path for its mapping feasibility. Afterward, double results are eliminated, and loops for the randomly chosen node are detected. A more detailed explanation of the functionality is provided by Delfmann et al., (2015a).

The subsequent section explains how compliance patterns can be developed based on legislation texts in order to analyze process models grounded on this graph-theoretical basis.

### 3 The Legislation and its Compliance Patterns

Besides the variety of approaches for checking business process models regarding their compliance, less work exists regarding concrete compliance patterns. Furthermore, one can distinguish between literature that elaborates (structural) patterns with only a vague relation to concrete legislation texts and literature that actually defines patterns based on particular legislation texts.

In terms of the rather structural definition of patterns, the literature is well positioned. Ghose and Koliadis (2007) present an adaptive framework for the compliance checking of *Business Process Model and Notation* (BPMN) models by using comparisons between defined rules and the actual business process model. Awad and Weske (2010) provide structural patterns for BPMN and Ramezani et al. (2012) present a set of structural patterns for Petri nets with 15 categories filled with several patterns based on different structures. Further specifications are provided by, for instance, Namiri and Stojanovic (2007), Sadiq et al. (2007), Kumar and Liu (2008), Elgammal et al. (2010) and Ly et al. (2011, 2015).

Against that, little work is published which elaborates patterns that refer to concrete legislation texts. Schumm et al. (2010) present three patterns in Linear Temporal Logic (LTL) based on the Sarbanes-Oxley Act Sec. 404 (107th Congress, 2002) and ISO 17799-10.1.3 (ISO, 2005). Elgammal et al., (2014) elaborate a further collection by also taking the Sarbanes-Oxley Act Sec. 404 as one source. Beyond this, ISO 27002-10.1.3 (ISO, 2013), the data protection directive 95/46/EC (European Parliament, 1995) and an internal bank policy are analyzed in order to develop seven compliance

patterns in LTL. The German legislation is focused by Becker et al. (2014). The authors describe 13 compliance patterns based on different German laws like the *German data privacy law* (BDSG) (BMJV, 1990), the *money laundering law* (GWG) (BMJV, 2008) or the *securities trading act* (WpHG) (BMJV, 1994). The patterns are formalized by using the *Generic Model Query Language* (GMQL).

However, the named patterns are all presented in a way that necessitates a certain degree of previous knowledge, and they do not provide the reader with a graphical, easily comprehensible representation for a better understanding. Due to this fact, the barrier increases for really applying compliance checking in a real-world setting. Often, only the people who invented an approach are able to apply it and make use of developed patterns. This is precisely the problem we aim to address with this work. Therefore, we gradually depict the procedure for developing a compliance pattern based on a concrete legislation text in the following. The easily understandable approach strives for motivating people without extensive previous knowledge in process and pattern modeling to apply pattern-based compliance checking more extensively.

Indeed, legislation texts often leave room for interpretation and thus, there is no sole correct solution for creating a compliance pattern. Due to this fact, there is to find a balance between accuracy of the depiction and generalizability respectively simplicity for identifying a potential compliance violation. It is valuable to find some more results than to overlook a violation. Therefore, the approach shown below tries to integrate a well-balanced ratio of accuracy and generalizability respectively simplicity. Roughly speaking, the development of a compliance pattern comprises two steps. While they are strongly intertwined during the mental considerations of the pattern creation, they are rather sequential during the modeling of a pattern. In the following, the latter part is focused.

- **Pattern structure.** The first step deals with the general structure of the pattern. One has to decide what exactly is to be searched and how this would look like. The main question to ask is “What is the pattern supposed to detect in terms of structure?” Usually, we create a compliance violation pattern (referred to as compliance pattern in this work) which detects parts of a process model that do not fulfill a legal requirement. An exemplary situation would be, for example, a necessary activity at some time after a special activity has been performed. The underlying structure of the compliance pattern would consist of two activities. One depicts the special activity and the second is the last activity in this process. The necessary activity must not lay between those two. Depending on the granularity of a process model, one has to decide if, for example, different checks like credit worthiness or risk score can be consolidated to one checking activity or if they have to be distinguished.
- **Pattern content.** The second step comprises the selection of an appropriate wording for the description of the elements of the pattern and, if necessary, the selection of translations. Here, one has to pay attention to choose neither too less nor too many different terms as the mentioned balance between accuracy and generalizability is mainly fixed in this step. The questions to answer are “What is stated in the legislation text?” and “How can the legislation text be represented in an appropriate way?” Afterward, the wording is assigned to the developed structure. The question “What part of a process model is the pattern supposed to detect in terms of content?” is to be answered. Hereby, activities can be associated with concrete descriptions like “check invoice”, other attributes like a person responsible or an IT system can be assigned, and activities can be defined as a start or an end activity of a process model.

According to these steps, an exemplary pattern is developed in the following. The exemplary legislation text that is to be transformed into a compliance pattern stems from the *Circular 10/2012 regarding Minimum Requirements for Risk Management (MaRisk BA)* set by the *Federal Financial Supervisory Authority* (BaFin) in Germany. Among other things, the content deals with the loan risk evaluation before a loan is granted. The passage *BTO 1.2.1 – Granting of loans* states the following:

1. “The process of granting loans encompasses the necessary operational steps up to the loan payout. All factors which are material for assessing the risk shall be analyzed and assessed, taking particular account of the debt-servicing capacity of the borrower or the property/project, with the intensity of the assessment depending on the riskiness of the exposures (eg creditworthiness assessment, risk score in the risk classification procedure or an assessment based on a simplified procedure).”
2. As a general rule, the value and legal validity of collateral shall be reviewed prior to granting the loan. When reviewing the value of collateral, available collateral values may be relied on if there are no indications of any change in value.
3. If the collateral value depends largely on the situation of a third party (eg a guarantee), the third party’s counterparty and credit risk shall be appropriately reviewed.
4. The institution shall define the eligible types of collateral and the procedures for determining the value of this collateral.” (BaFin, 2012)

Firstly, we analyze the text regarding a possible **pattern structure**. The described steps (a), (b), (c) and (d) are graphically shown in Figure 3.

(a) The first marginal number of the legislation text states in its first sentence that “operational steps [have to be performed] up to the loan payout.” Due to this statement, a loan payout could be a subsequent activity (blue circle) within the pattern. The directed edge (solid arrow) represents this succession.

(b) The following sentences describe the “operational steps” in more detail. Despite the detailed content, these steps have to be performed before the loan payout is conducted. Since the pattern strives for detecting a violation of such construct, it has to detect (and thus, to describe) process parts in which the loan payout is conducted, but the necessary previous steps are not carried out. For describing this, we add an activity to the pattern that must not exist on the way to the subsequent activity (which represents the loan payout). Indeed, the text states several activities to be performed (“check debt-servicing capacity,” “check value and legal validity of collateral” and “check third party’s counterparty”) instead of only one activity. However, one has to decide depending on the granularity of the process models whether several activities in favor of accuracy or only one activity in favor of simplicity should be forbidden on the way. Here, we choose one activity as all necessary “operational steps” are referring to a risk evaluation. For the later search, the prohibition is set technically by just selecting a single option.

(c) Furthermore, there is no hint that a particular activity is necessary before the “operational steps” are conducted. Thus, we add an activity to the pattern structure which will be later defined as a start activity of the process. In this way, we assure that all relevant process parts are detected from the start of the process. In the course of this, the directed edge is transformed to a directed path (dashed line) which allows one or more activities to be between the start and the loan payout.

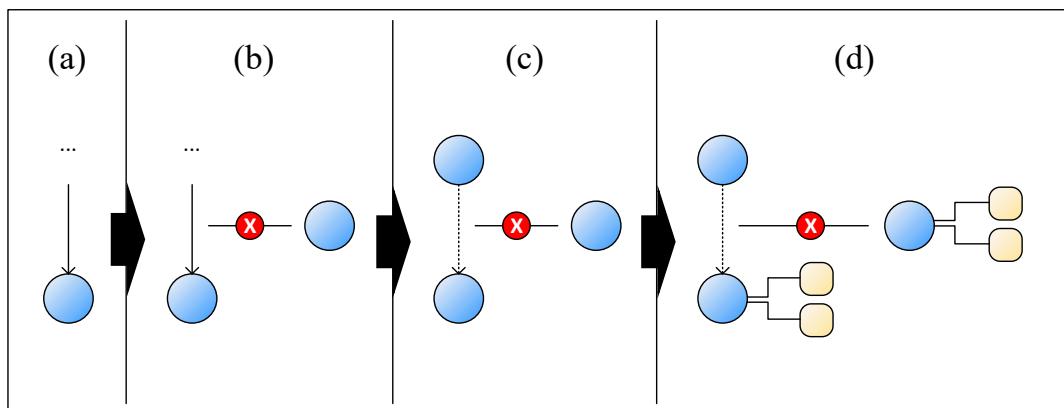


Figure 3. Pattern development (structure).

(d) Finally, we enclose attributes to the activities. They are connected by undirected edges (solid lines). In this case, two attributes are connected to the last activity and the forbidden activity. Both couples represent a procedure (for example, “check”) and a noun (for example, “loan risk”) each. This assignment is conducted later on. Here, these attributes are assigned since the elements of the analyzed process models always consist of a combination of a procedure and a noun. In general, arbitrary attributes (which are present in a process model) can be assigned. The first activity is defined as first process activity later on and thus, does not need any attributes.

In addition to the pattern structure, the **pattern content** has to be specified. For that, it is necessary to define appropriate terms according to the granularity of the process models. If available, one should use glossaries that contain all terms used within the process models in order to determine the appropriate terms. The process models analyzed in our study contain two of such glossaries, one for objects and one for procedures. Based on connections of elements of both glossaries, an activity is described. If a glossary is not available, the granularity can be determined by having an exemplary look at the process models, and the appropriate terms can be obtained by, for example, using a domain thesaurus. Techniques for terminologically standardize conceptual models have, *inter alia*, been published by Kugeler and Rosemann (1998). The necessary steps (a), (b) and (c) for defining the pattern content are described in the following and depicted in Figure 4 and Figure 5.

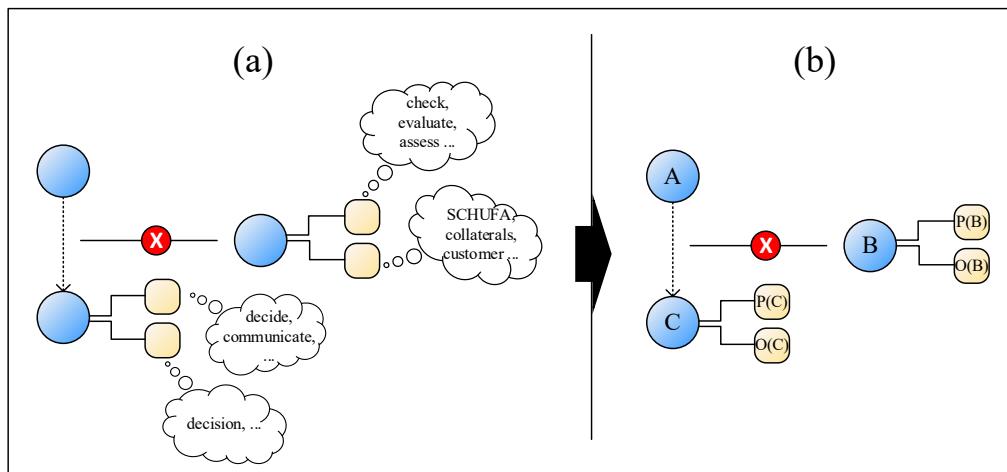


Figure 4. Pattern development (content I).

(a) As already suggested during the structure development, the granularity of the process models in our case is rather on a high level and thus, fine-granular check activities like “check debt-servicing capacity,” “check value and legal validity of collateral” and “check third party’s counterparty” are consolidated into a single activity. All terms of the fine-granular activities are included. Another reason for this consolidation is the higher probability that potential violations are detected due to less specialization and higher simplicity. Following this, couples like “check SCHUFA<sup>1</sup>”, “evaluate collaterals” or “assess customer” are assigned. The second activity which is based on the text passages “granting the loan” or “loan payout” is depicted by the terms “decide loan granting” or “communicate decision.” The latter is also included, as the communication of a decision also has to be preceded by a risk evaluation. Since the process models are in German, the German terms are listed in appendix A.

(b) Afterward, elementIDs are assigned to the structure in order to enable a further content specification of the elements. The IDs can be chosen freely. Here, the activities are named with A, B and C, the corresponding attributes with P(B) and P(C) for the procedures and O(B) and O(C) for the objects of the activities B and C.

<sup>1</sup> The SCHUFA Holding AG is a German credit reporting agency which provides credit information about credit applicants.

(c) However, the elementIDs do technically not specify the type of the attributes; they are only named according to their type due to a better understanding. The type is set by applying type rules. These rules for setting a type can be set for each element by using the syntax: `[elementID].Type == "Type"`. Hereby, the types “Procedure” respective “Object” are assigned to the attributes. The rules are also used for setting possible values of elements. A possible value for an element can be defined by using the syntax: `[elementID] == "Value"`. It is also possible to require several values for an element by using an AND-operator. This is used for combining specific combinations of procedures and objects. In this way, misleading combinations are excluded. Furthermore, it is possible to search several combinations by using the OR-operator.

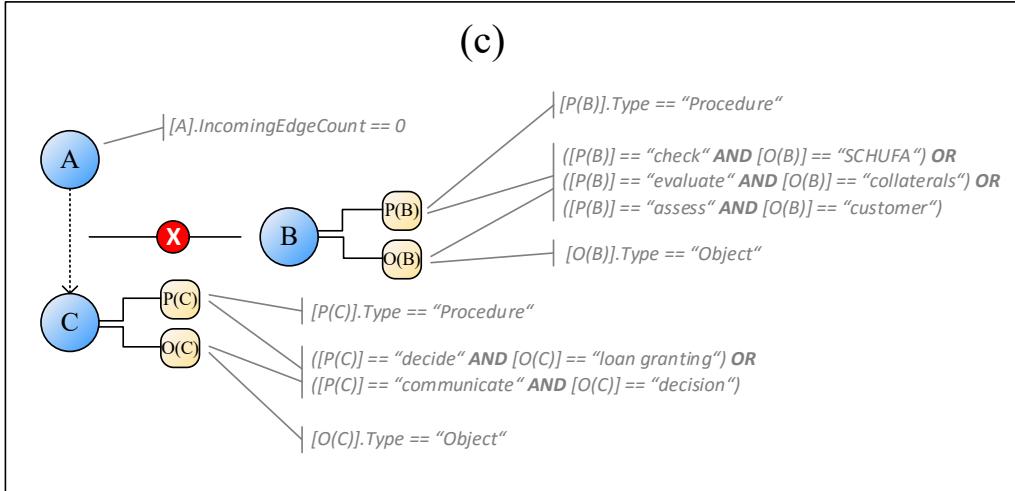


Figure 5. Pattern development (content II).

Finally, the compliance pattern is created. There are many further options available within the approach for refining a pattern. It is, for example, possible to use a wildcard operator (\*) for an arbitrary number of arbitrary characters within a value. In addition, different adaptions of the rules are possible. For detailed information, Delfmann et al. (2015a) introduce the approach technically in detail. Our approach is prototypically implemented as presented. For automatically analyzing business process models, the compliance pattern has to be created as shown, selected and the respective process model has to be chosen. An insight into the prototypical implementation is given in Figure 6. Afterward, section 4 describes the analysis of real-world business process models.

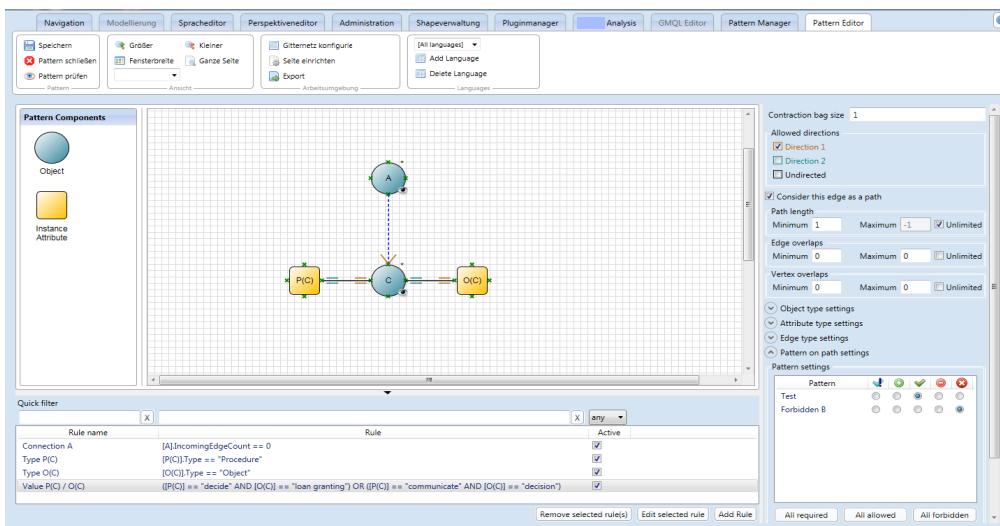


Figure 6. Prototypical implementation.

## 4 Compliance Checking of Real-World Process Models

In order to make use of the developed compliance pattern, it has to be searched within process models. However, regardless of the process model, a manual search is far too costly, and the cost-benefit ratio does not empower the people's willingness to develop patterns for model-based compliance checking. The cost-benefit ratio becomes even worse the larger the process model and the more complex the pattern are. Thus, it is necessary to perform an automatic detection of the developed pattern in order to improve the cost-benefit ratio. For that reason, the approach includes an integrated algorithm based on graph theory (cf. section 2) which is able to detect the pattern within process models automatically. As an input, the algorithm utilizes the pattern as it is developed above. For starting the search, the process model(s) and the pattern(s) must be selected. The details of the used algorithm are explained by Delfmann et al. (2015a).

After executing one or several searches, the analyst is provided with detailed results. Besides the time taken, the number of detected results (possible violations) is stated per analyzed process model. In the case that results are found, they can be regarded in more detail. In this way, the analyst is not only informed that a process may not be compliant, s/he can also have a deeper look at the potentially violating process part in order to assess directly if a real violation exists. Furthermore, all results are stored in terms of documentation of the compliance checking and for a later re-assessment.

However, an approach is useless as far as it does not provide any benefit for real-world companies. Many approaches are tested and verified by exemplary process models that are often simplified and abstracted from reality. Therefore, we applied this approach to real-world process models provided by a large German bank service provider. The context of the company fits well as especially for banks exist a high number of legal regulations. For such companies, it is inevitable to adhere to these compliance rules in their business processes.

The used modeling language of the process models consists only of activities, which – as already stated – are described by combinations of procedures and objects. The modeling language is based on a four-layer architecture. For reasons of the compliance checking, this architecture is integrated into one layer, and different layers are represented by different colors. Furthermore, a process can be split and merged. A brief overview of the elements is provided in Figure 7.

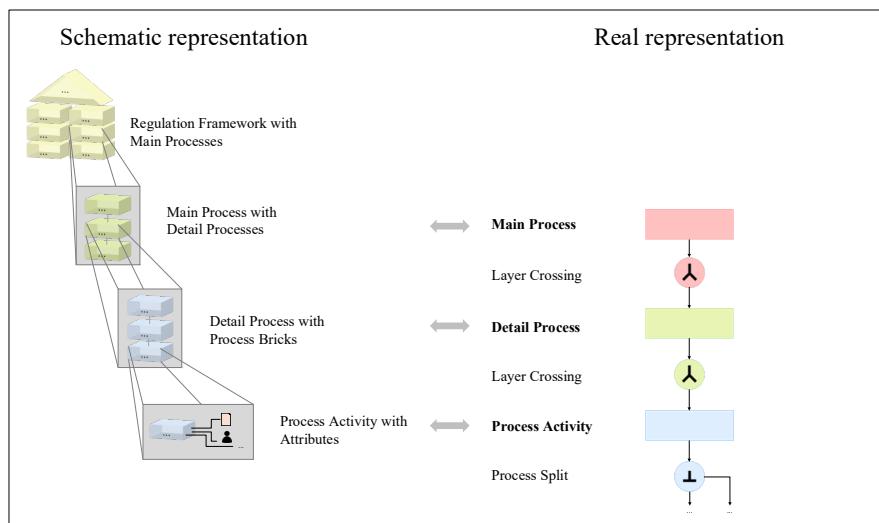


Figure 7. Process modeling language.

The German bank service provider made all of its process models accessible to us. The analyzed process models consist of 25 main processes, 216 detail processes and 3427 process activities, which have 17616 attributes assigned. In total, there are 21284 vertices to be analyzed. We do not omit any process model provided to us but rather incorporate all process models of the bank service provider for

not distorting the applicability of the approach. Finally, the search of the developed pattern provided 49 potential compliance violations. Two exemplary potential violations are briefly shown and explained in the following.

The first result shows a process part concerning a consultation for a private loan (Figure 8). The path starts at the beginning of the process (as requested by the pattern) and ends with the communication of a decision. The path contains activities for a customer contact, a check of the wish of the customer and the receipt of documents. After that, the further processing is checked, and a decision is told to the customer. However, it is questionable how this decision is made, as no risk evaluation activity is located within this path. Thus, a compliance violation may occur according to *MaRisk 2010 – BTO 1.2.1 – Granting of loans*. The legislation stipulates several evaluations that have to be performed before making a decision regarding a loan granting, however, none is present within this process.

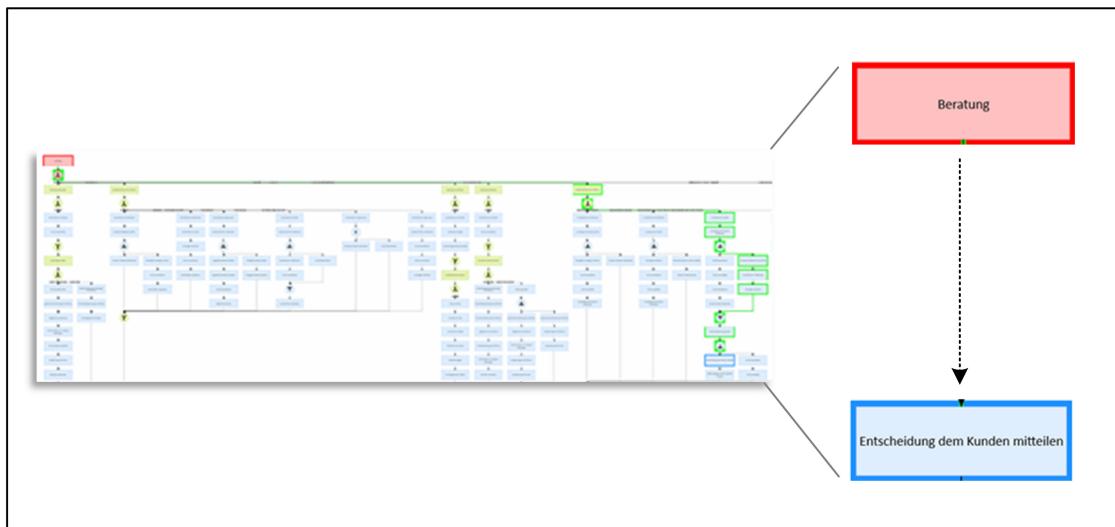


Figure 8. Potential compliance violation (I).

A second result is shown in Figure 9. The found result shows a process part of a customer contact. The part includes the contact via e-mail, a personal talk is conducted including a discussion about the customer's situation, and the further processing is checked. Finally, a decision is told to the customer. However, no risk evaluation is performed on the whole path of this loan granting process. Again, a potential compliance violation is detected.

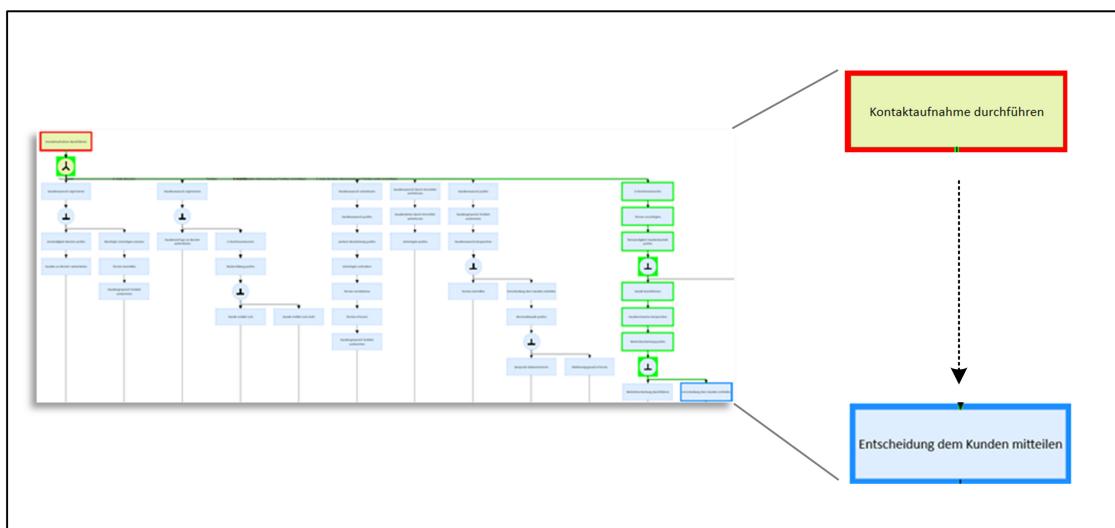


Figure 9. Potential compliance violation (II).

After executing the searches, a banking analyst of the respective company would have to examine the found results if a real violation is detected. Since an automatic compliance checking approach based on patterns can only work as a heuristic, there also may be a number of wrongly detected results. However, by using this approach, the number of vertices to check manually can largely be reduced and especially for larger process models and complex patterns, a compliance checking is enabled.

## 5 Conclusion

The paper depicts a procedure of developing compliance patterns from legislation texts for business process compliance checking. The used approach supports the whole procedure from the pattern development to the detection of results. A special focus is set on an easy application of the approach while simultaneously enabling the application to real-world process models. Thus, the approach aims at motivating people without large process or pattern modeling knowledge to apply more model-based compliance checking based on compliance patterns. Therefore, the approach tries to resolve the barriers that many approaches entail: “application too difficult” or “application not possible in real-world scenarios.”

After motivating the topic and embedding this work into a contextual frame, we differentiated the compliance checking approach from other existing approaches. Subsequently, the pattern development based on real German legislation was focused. For that, we showed up related work regarding the development of compliance patterns and classified the existing literature. After that, we transformed a concrete legislation passage gradually into an automatically usable compliance pattern by using the provided modeling approach. Finally, we searched the pattern in real-world process models of a German bank service provider and briefly presented exemplary results.

The depicted approach comes around with several advantages. The graphically supported development of compliance patterns supports the user to understand easily what s/he just modeled. Additionally, the rather simple kept rules provide a powerful means to develop not only simple but also more complex compliance patterns. Due to its flexibility, the approach is not restricted to a specific modeling language. In this way, the people are enabled to regard their known process models as they are and model the pattern in a similar way. As also the results can be reviewed within the existing process model style, the approach addresses process (modeling) experts as well as legislation experts who may not have deep process modeling knowledge. Furthermore, we tested the approach in a real-world setting with process models that are not simplified or abstracted for testing the approach. Beyond this, many approaches exist in the literature, but only very few are freely accessible. The approach shown in this paper can be accessed via <https://em.uni-muenster.de> (ERCIS Competence Center Conceptual Modeling, 2015).

However, the whole procedure has some limitations. During the pattern development, the quality of a pattern is very dependent on the modeler’s adaption ability. Different process models may be created on a different granular basis, and thus, the challenge of defining patterns of the same granular basis persists. In combination with the granularity, the approach is very dependent on the correct wording. In many process models, there is no semantic standardization as provided glossaries or even a distinction between procedures and objects. However, this problem is rather adopted from existing process models. Modelers tend to model all entities of a process by stating a textual description instead of using attributes. Hence, it becomes harder to analyze such models. Regarding the pattern search, the limitations are, on the one hand, located at the performance of the approach. The larger and the more complex the process models and the patterns are, the longer the searches last until they are finished. Thus, only process models up to a particular size can be analyzed. On the other hand, the analyzed process models only stem from one company and consequently are limited in terms of their quantity. For really making statements about a general applicability to real-world process models, a larger number of process models that stem from a broad range of industry sectors have to be analyzed. A last rather small limitation of the pattern-based compliance checking, in general, is the necessity that a law specialist has to review all returned results due to the heuristic character of all these approaches.

In summary, the named limitations should be tackled next in order to disseminate the pattern-based compliance checking even more. Its full potential can only be exploited in times of fast-changing legislation if it is applied as often as possible. Therefore, the research is in demand to keep compliance checking simple in terms of usage and collaborate with companies from practice in order to enable an improved pattern development as well as further process model analyses for compliance.

## Appendix A – German Terms

elementID	German terms		German terms
	Object	Procedure	
<b>B</b>	SCHUFA*	anfordern	Risikoeinheit
		prüfen	prüfen
	Schufaauskunft	einholen	Bonitätskriterien
	Sicherheit	erfassen	Bonitätsnote
	Sicherheiten*	anlegen und bewerten	Bonitätsunterlagen
		anzeigen	auswerten
		bewerten	prüfen
		einbeziehen	Berechtigung
		grob bewerten	Kreditbeurteilung
		im System vorhanden*	Kreditmanagement
		*prüfen	Kreditwürdigkeit (Rating)
		überwachen	Kundenbeurteilung
		prüfen	*
		notwendig prüfen	durchführen
	*Rating	anstoßen	KDF
		durchführen	auf Aktualität prüfen
		genehmigen	auf Plausibilität prüfen
			prüfen
			KDF-Abschätzung
<b>C</b>	Risikogruppe des Kunden	prüfen	KDF-Berechnung*
			Prüfung
	Entscheidung	mit Kunden besprechen	Auswertungen
		dem Kunden mitteilen	Auswertungs-ergebnis
	Kredit-entscheidung	durchführen	sichten
		prüfen	
			Kreditentscheidung des Beraters
			prüfen
			Negative Entscheidung
			dem Kunden mitteilen

## References

- 107th Congress (2002). “Sarbanes-Oxley Act.” Retrieved on 2015-11-27, from <http://www.sec.gov/about/laws/soa2002.pdf>
- Awad, A. and M. Weske (2010). “Visualization of Compliance Violation in Business Process Models.” In: S. Rinderle-Ma, S. Sadiq, & F. Leymann (Eds.), *Business Process Management Workshops* (pp. 182–193). Berlin/Heidelberg, Germany: Springer.
- BaFin (2012). “Circular 10/2012: Minimum Requirements for Risk Management (MaRisk BA).” Retrieved on 2015-11-27, from [https://www.bafin.de/SharedDocs/Downloads/EN/Rundschreiben/dl\\_rs\\_1210\\_ba\\_marisk.html](https://www.bafin.de/SharedDocs/Downloads/EN/Rundschreiben/dl_rs_1210_ba_marisk.html)
- Becker, J., P. Delfmann, H.-A. Dietrich, M. Steinhorst and M. Eggert (2014). “Business Process Compliance Checking – Applying and Evaluating a Generic Pattern Matching Approach for Conceptual Models in the Financial Sector.” *Information Systems Frontiers*, 1–47.
- Becker, J., P. Delfmann, M. Eggert and S. Schwittay (2012a). “Generalizability and Applicability of Model-Based Business Process Compliance-Checking Approaches - A State-of-the-Art Analysis and Research Roadmap.” *BuR - Business Research*, 5(2), 221–247.
- Becker, J. and D. Kahn (2011). “The Process in Focus.” In: J. Becker, M. Kugeler, & M. Rosemann (Eds.), *Process Management: A Guide for the Design of Business Processes* (2nd ed., pp. 3–13). Berlin/Heidelberg, Germany: Springer.
- Becker, J., M. Kugeler and M. Rosemann (Eds.) (2012b). *Prozessmanagement* (7th ed.). Berlin/Heidelberg, Germany: Springer.
- BMJV (1990). “Bundesdatenschutzgesetz (BDSG).” Retrieved on 2015-11-27, from [http://www.gesetze-im-internet.de/bdsg\\_1990/](http://www.gesetze-im-internet.de/bdsg_1990/)
- BMJV (1994). “Wertpapierhandelsgesetz (WpHG).” Retrieved on 2015-11-27, from <http://www.gesetze-im-internet.de/wphg/>
- BMJV (2008). “Geldwäschegesetz (GwG).” Retrieved on 2015-11-27, from [http://www.gesetze-im-internet.de/gwg\\_2008/](http://www.gesetze-im-internet.de/gwg_2008/)
- Delfmann, P., D. Breuker, M. Matzner and J. Becker (2015a). “Supporting Information Systems Analysis Through Conceptual Model Query – The Diagrammed Model Query Language (DMQL).” *Communications of the Association for Information Systems*, 37(24).
- Delfmann, P., M. Steinhorst, H.-A. Dietrich and J. Becker (2015b). “The Generic Model Query Language GMQL – Conceptual Specification, Implementation, and Runtime Evaluation.” *Information Systems*, 47(1), 129–177.
- El Kharbili, M., A. K. A. de Medeiros, S. Stein and W. M. P. van der Aalst (2008). “Business Process Compliance Checking: Current State and Future Challenges.” In: *Proceedings of the Conference Modellierung betrieblicher Informationssysteme (MobIS)*. Saarbrücken, Germany.
- Elgammal, A., O. Turetken, W.-J. van den Heuvel and M. Papazoglou (2010). “Root-Cause Analysis of Design-Time Compliance Violations on the Basis of Property Patterns.” In: P. P. Maglio, M. Weske, J. Yang, & M. Fantinato (Eds.), *Service-Oriented Computing* (pp. 17–31). Berlin/Heidelberg, Germany: Springer.
- Elgammal, A., O. Turetken, W.-J. van den Heuvel and M. Papazoglou (2014). “Formalizing and Applying Compliance Patterns for Business Process Compliance.” *Software & Systems Modeling*, 1–28.
- ERCIS Competence Center Conceptual Modeling (2015). “[em] - A Meta Modeling Tool.” Retrieved on 2015-11-27, from <https://em.uni-muenster.de/>
- European Parliament (1995). “Directive 95/46/EC.” Retrieved on 2015-11-27, from <http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:en:HTML>
- Ghose, A. and G. Koliadis (2007). “Auditing Business Process Compliance.” In: *Proceedings of the International Conference on Service-Oriented Computing (ICSOC)* (pp. 169–180). Vienna, Austria.
- Hammer, M. (2010). “What is Business Process Management?” In: J. vom Brocke & M. Rosemann

- (Eds.), *Handbook on Business Process Management 1* (2nd ed., pp. 3–16). Berlin/Heidelberg, Germany: Springer.
- ISO (2005). “ISO/IEC 17799:2005.” Retrieved on 2015-11-27, from [http://www.iso.org/iso/catalogue\\_detail?csnumber=39612](http://www.iso.org/iso/catalogue_detail?csnumber=39612)
- ISO (2013). “ISO 27002:2013.” Retrieved on 2015-11-27, from [http://www.iso.org/iso/catalogue\\_detail?csnumber=54533](http://www.iso.org/iso/catalogue_detail?csnumber=54533)
- Kugeler, M. and M. Rosemann (1998). “Fachbegriffsmodellierung für betriebliche Informationssysteme und zur Unterstützung der Unternehmenskommunikation.” *Rundbriefe Desw GI-Fachausschusses*, 5(2), 8–15.
- Kumar, A. and R. Liu (2008). “A Rule-Based Framework Using Role Patterns for Business Process Compliance.” In: N. Bassiliades, G. Governatori, & A. Paschke (Eds.), *Rule Representation, Interchange and Reasoning on the Web* (pp. 58–72). Berlin/Heidelberg, Germany: Springer.
- Ly, L. T., F. M. Maggi, M. Montali, S. Rinderle-Ma and W. M. P. van der Aalst (2015). “Compliance Monitoring in Business Processes: Functionalities, Application, and Tool-Support.” *Information Systems*, 54, 209–234.
- Ly, L. T., S. Rinderle-Ma, D. Knuplesch and P. Dadam (2011). “Monitoring Business Process Compliance Using Compliance Rule Graphs.” In: R. Meersman, T. Dillon, P. Herrero, A. Kumar, ... M. Mohania (Eds.), *On the Move to Meaningful Internet Systems: OTM 2011* (pp. 82–99). Berlin/Heidelberg, Germany: Springer.
- Namiri, K. and N. Stojanovic (2007). “Pattern-Based Design and Validation of Business Process Compliance.” In: R. Meersman & Z. Tari (Eds.), *On the Move to Meaningful Internet Systems 2007: CoopIS, DOA, ODBASE, GADA, and IS* (pp. 59–76). Berlin/Heidelberg, Germany: Springer.
- PwC (2011). *Zukunftsthema Geschäftsprozessmanagement*. Retrieved on 2015-11-27, from [http://www.pwc.de/de\\_DE/de/prozessoptimierung/assets/PwC-GPM-Studie.pdf](http://www.pwc.de/de_DE/de/prozessoptimierung/assets/PwC-GPM-Studie.pdf)
- Ramezani, E., D. Fahland and W. M. P. van der Aalst (2012). “Where Did I Misbehave? Diagnostic Information in Compliance Checking.” In: A. Barros, A. Gal, & E. Kindler (Eds.), *Business Process Management* (pp. 262–278). Berlin/Heidelberg, Germany: Springer.
- Rinderle-Ma, S., L. T. Ly and P. Dadam (2008). “Business Process Compliance (Aktuelles Schlagwort).” In: *Proceedings of the Fachtagung für Entwicklungsmethoden für Informationssysteme und deren Anwendung (EMISA)*. Sankt Augustin, Germany.
- Rosemann, M. and J. vom Brocke (2010). “The Six Core Elements of Business Process Management.” In: M. Rosemann & J. vom Brocke (Eds.), *Handbook on Business Process Management 1* (2nd ed., pp. 107–122). Berlin/Heidelberg, Germany: Springer.
- Sadiq, S., G. Governatori and K. Namiri (2007). “Modeling Control Objectives for Business Process Compliance.” In: *Proceedings of the 5th International Conference on Business Process Management (BPM)* (pp. 149–164). Brisbane, Australia.
- Schumm, D., O. Turetken, N. Kokash, A. Elgammal, ... W.-J. van den Heuvel (2010). “Business Process Compliance through Reusable Units of Compliant Processes.” In: F. Daniel & F. M. Facca (Eds.), *Current Trends in Web Engineering* (pp. 325–337). Berlin/Heidelberg, Germany: Springer.
- Schwegmann, A. and M. Laske (2012). “Istmodellierung und Istanalyse.” In: J. Becker, M. Kugeler, & M. Rosemann (Eds.), *Prozessmanagement* (pp. 165–194). Berlin/Heidelberg, Germany: Springer.
- van der Aalst, W. M. P., A. H. M. ter Hofstede and M. Weske (2003). “Business Process Management: A Survey.” In: *Business Process Management* (pp. 1–12). Berlin/Heidelberg, Germany: Springer.
- Weske, M. (2012). *Business Process Management* (2nd ed.). Berlin/Heidelberg, Germany: Springer.