On the Use of the Conformance and Compliance Keywords During Verification of Business Processes

Heerko Groefsema[1], Nick R. T. P. van Beest[2(✉)], and Guido Governatori[3]

[1] University of Groningen, Groningen, The Netherlands
h.groefsema@rug.nl
[2] Data61, Brisbane, Australia
nick.vanbeest@data61.csiro.au
[3] Brisbane, Australia

Abstract. A wealth of techniques have been developed to help organizations understand their processes, verify correctness against requirements and diagnose potential problems. In general, these verification techniques allow us to check whether a business process *conforms* or *complies* with some specification, and each of them is specifically designed to solve a particular business problem at a stage of the BPM lifecycle. However, the terms conformance and compliance are often used as synonyms and their distinct differences in verification goals is blurring. As a result, the terminology used to describe the techniques or the corresponding verification activity does not always match with the precise meaning of the terms as they are defined in the area of verification. Consequently, confusion of these terms may hamper the application of the different techniques and the correct positioning of research. In this position paper, we aim to provide comprehensive definitions and a unified terminology throughout the BPM lifecycle. Moreover, we explore the consequences when these terms are used incorrectly. In doing so, we aim to improve adoption from research to practical applications by clarifying the relation between techniques and the intended verification goals.

Keywords: Conformance · Compliance · Verification · Review

1 Introduction

Business process management (BPM) has adapted from supporting local rigid and repetitive units of work in factory-based processes to loosely-coupled case based processes in a wide range of different, and often regulated, business contexts. This evolution set in motion an increasing need to assess whether these business processes, supported by business process management tools, are free of error, performed as desired, and follow regulations [14]. To address these distinct—but related—issues, many techniques have been developed over the past decades to help organisations understand their processes, verify correctness and diagnose potential problems [14]. Each of these techniques is very specifically designed and tailored to solving a particular business problem or question, and may be applied at different stages of the BPM lifecycle.

In general, these techniques for verification allow us to check whether a business process *conforms* or *complies* with some specification, and often refer to the popular

C. Di Ciccio et al. (Eds.): BPM 2022, LNBIP 458, pp. 21–37, 2022.
https://doi.org/10.1007/978-3-031-16171-1_2

business process mining technique *conformance checking* and the verification of *regulatory* compliance in BPM. While there are surface similarities among the verification problems and the activities specific to them, the terms have distinct meaning in the area of verification and their use depends on whether only specifications or a specification and implementation is involved in verification [15]. In everyday language, however, the terms conformance and compliance are often used as synonyms, and their distinct differences in verification goals is blurring. As a consequence, the terminology used to describe the techniques or the corresponding verification activity does not always match with the precise meaning of the terms as defined in the area of verification.

Due to the duality of the use of the conformance and compliance terms, several issues have emerged. In science, the confusion of these terms has lead to (i) the wrong motivation being given to justify the work, (ii) a wrong example being used to explain the work, (iii) discussions of related work including irrelevant and excluding relevant work, or (iv) evaluations comparing tools related to different perspectives. Moreover, in practical settings the confusion of these terms may lead to (v) the wrong approach being chosen and answering a question from a different perspective, (vi) the wrong artifact being used for an approach, or (vii) the approach being performed at the wrong stage of the BPM lifecycle. As a result, this inadvertently emerged mismatch between techniques and terminology could harm transfer from research to practical applications, possibly stagnating adoption of relevant approaches and new advances in the field.

In this position paper, we aim to provide comprehensive definitions of the two notions, describe the activities related to them, and the BPM artifacts they apply to.

Method and Structure

To do so, we first define the key artifacts in the BPM lifecycle and introduce the concept of verification in that context and the verification corresponding relations in Sect. 2. Subsequently, we explore the existing goals of verification and the related verification techniques for each goal in Sect. 3, discussing the intent and constraints of each verification goal. Note that, as we define each of the above elements, many definitions refer to the *ISO/IEC/IEEE Systems and software engineering – Vocabulary* standard [17]. Since the vocabulary lists multiple alternative meanings of each term depending on its application domain, throughout this paper we either directly use the variant that relates most to the domains of verification and business process management, or a combination of relevant variants. We do so, because these variants offer the best foundations required for the discussion around the verification of conformance and compliance within the BPM lifecycle. Next, Sect. 4 uses the provided definitions of artifacts and relations to connect verification relations to verification goals and provide a structured overview, highlighting potential areas that may cause confusion and propose a solution. Section 5 discusses the relevance of such a solution by providing examples of terminology and verification goal mismatches and discussing potential consequences. Finally, the findings are summarised in Sect. 6.

2 Verification and the Business Process Management Lifecycle

Validation and verification are well-known evaluation procedures used to investigate whether a software or hardware product fulfills its intended purpose [17]. *Validation*

investigates whether the product fulfills the needs of the user, that is, it tries to answer if the correct product is being made. *Verification*, on the other hand, investigates if the product matches with its specifications, or whether the product is being made correctly.

When applying formal methods of mathematics to verification, the procedure is called *formal verification*. Formal verification entails proving or disproving the correctness of a model with respect to a *specification* using formal methods of mathematics. In this case, the model is a representation of the actual system (e.g., based on a specification), just like a business process model is a representation and specification of the actual business process that is being performed. Note, however, that given different verification approaches the model is not necessarily always represented by a business process model. In fact—as we will observe later—sometimes the business process model represents the specification of the verification approach instead. A specification is defined as follows:

Definition 1 (Specification). *A collection of statements that specify in a complete, precise, and verifiable manner, the requirements, design, behavior, or other characteristics of a system or component, and—often—the procedures for determining whether these provisions have been satisfied [17].*

The procedure of verification is an important aspect of the BPM lifecycle [6]. An overview is given in Fig. 1, where we map the business process artifacts of the lifecycle—represented by the circles—with the verification techniques, represented by the arrows connecting different artifacts. For each verification technique, the artifact used as the specification is connected to the artifact used to represent the model using an arrow. For example, the design properties (specification) are verified against the business process model (model) when checking process correctness. For completeness, two dashed arrows representing the validation relations have also been included. These relations are outside of the scope of this paper, which has a focus on verification within the BPM lifecycle alone.

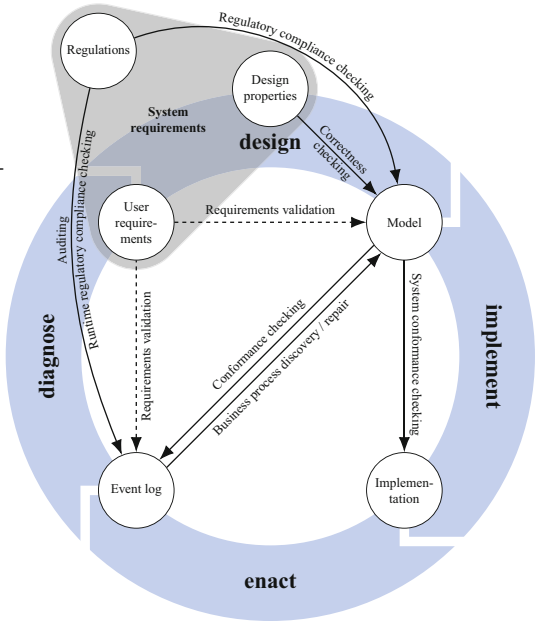

Fig. 1. Verification techniques applied during the BPM lifecycle.

2.1 Business Process Management Artifacts Used for Verification

The BPM lifecycle uses and produces a number of artifacts that can be applied as the model or specification during a number of useful verification techniques. We define and discuss the relevant artifacts depicted in Fig. 1 as circles.

Before defining the relevant artifacts, however, we must first define the business process itself. Informally, a business processes is a collaborations between actors that achieve a specific value-added goal. Within a business process, actors perform activities based on available data and using available resources. When referring to a business process, we refer to the real life process—which may or may not be supported by software systems or be described by a model. More formally:

Definition 2 (Business process). *A partially ordered set of activities, performed by actors using available resources and data, that achieve some desired objective of an organization [17,19,20].*

Within the BPM lifecycle, a business process is first described by a number of specifications (Definition 1) that describe individual sets of requirements. A requirement is defined as follows:

Definition 3 (Requirement). *Provision that contains criteria to be fulfilled [17].*

These individual sets of requirements together define the system requirements. The system requirements are depicted as the gray area in Fig. 1 and include the user requirements, design properties, and regulations. The system requirements are defined as follows:

Definition 4 (System requirements). *A structured collection of requirements—comprising functions, performance, design constraints, and attributes—of the system and its operational environment and external interfaces [17].*

The system requirements can consist of different sets of specifications, including (i) the user requirements, (ii) the design properties of the chosen modeling method, and (iii) the regulations as imposed by external authorities. The user requirements, design properties, and regulations are defined as follows:

Definition 5 (User requirements). *The requirements for use that provide the basis for the design and evaluation of interactive systems to meet identified user needs [17].*

Definition 6 (Design properties). *The context-independent behavioral requirements of the created model given the chosen modelling method [2,18].*

Definition 7 (Regulations). *Requirements, imposed by an authority, that establish the legal and illegal behaviors and states for a specific domain and jurisdiction [17].*

Given the system requirements (Definition 4), a model of the business process (Definition 2) can be derived through the process of refinement. Such business process models can describe the business process along a number of different perspectives, including the control flow, data, and resource perspectives. Moreover, business process models

can be *descriptive* or *prescriptive*. A descriptive model describes the business process as it is performed in the real world, while a prescriptive model describes the business process as it should be performed [10]. The distinction is important since descriptive and prescriptive models fulfill very different roles during verification, roles that should be considered carefully. Another distinction can be made between *procedural* (or imperative) process models and *declarative* process models. Procedural process models use an imperative specification that describe step by step *how* a business process is performed, while declarative process models describe *what* is performed using, often, a logical representation. Note that declarative process models in many cases should be seen as declarative process *specifications* instead, while the actual model obtained from such a specification (sometimes also referred to as the declarative process model) is, in fact, imperative in nature. Although a process model is a specification in itself, the terms model and specification have distinct meaning in the area of verification and one should be careful when referring to logical representations as *models* when applying verification within the area of business process management. Sometimes, however, such a paradigm shift is correct, but should always be treated with extreme care. A business process model is defined as follows:

Definition 8 (Business process model). *A (graphical) representation of a business process that describes the typical business process instance in isolation by specifying the elements of the business process and their relationships along the control flow, data, and/or resource perspectives [5].*

Software systems can support business processes in many different ways. Given a business process model, software support may range from deployment of large information systems, such as business process management systems or case management systems, to individual software packages being used as each task is being performed manually in an ad-hoc way. We refer to the collection of hardware and software systems that support the business process as the implementation, which is defined as follows:

Definition 9 (Implementation). *Result of translating a design into hardware components, software components, or both, whose validity can be subject to test [15, 17].*

These software systems record information observed during execution of each process instance, or case, of a business process in a so-called event log [3]. The information captured in such event logs can be used to not only discover, monitor, and improve processes as supported by the software systems, but also to verify their correct execution against the requirements and regulations.

Definition 10 (Event log). *A collection of traces, where each trace is an ordered sequence of events observed and recorded during the execution of an instance/case of a business process. Each event refers to an action performed by an actor or the supporting implementation at a particular time, for a particular case, and possibly includes relevant data concerning that case [3].*

2.2 Verification Relations

Given the process of verification, between the described artifacts two possible relations can be proven: (i) relations that establish *conformance*, and (ii) relations that establish

compliance. The first defines a relation between a specification and an implementation, while the latter defines a relation between two specifications. More formally:

Definition 11 (Conformance). *A relation between a specification and an implementation that holds when (observed behavior of) the implementation fulfills all requirements of the specification (when the implementation conforms to the specification) [15, 20].*

Definition 12 (Compliance). *A relation between two specifications, A and B, that holds when specification A makes requirements which are all fulfilled by specification B (when B complies with A) [15].*

3 The Goals of Verification Within Business Process Management

Business processes are verified towards a number of different goals. Existing verification techniques can be classified into those that have the goal of system conformance, process conformance, model conformance, model compliance, or regulatory compliance. Note that the strict definition of compliance (Definition 12) describes a relation between two specifications and not a relation between a specification and implementation. As a result, the goals of system and process compliance are included under regulatory compliance. Each of these goals may have multiple supporting techniques. Such techniques have the same goal, but often use different artifacts at different stages of the BPM lifecycle. We discuss these goals and each related technique.

3.1 System Conformance

The verification of a system's implementation against its specification in a process model is referred to as *system conformance*. In this definition, the word conformance refers only to the conformance relation of Definition 11 and not to the collection of popular process mining techniques. In general, conformance is restricted to a limited set of requirements to check against particular aspects and elements, or so-called *conformance points*. Accordingly, the implementation is verified against said conformance points [20]. The technique is depicted in Fig. 1 as the arrow from model to implementation, and is defined as follows:

Definition 13 (System conformance checking). *The process of verifying conformance of the implementation towards the business process model.*

System conformance checking is possible when the implementation is fully supported and automatized by a workflow engine. This type of verification can be applied during different stages of the BPM lifecycle. During design time, the operation of checking can either be reduced to the formal verification of the implementation, or employs testing to ensure that the behavior of the implementation reflects the expected behavior described by the process model. During runtime, typically the event log is used as a proxy for the implementation. However, in general we cannot fully depend on event log data for this purpose, as some computations can produce the same result for some instances, but a model may require a particular type of implementation or calculation.

3.2 Process Conformance

When verifying the behavior of an implementation (as observed in e.g. an event log) against a process model, this is commonly referred to as *process conformance checking* and describes the collection of popular process mining techniques that are either applied online, at runtime, or after-the-fact. During runtime, there is no clear difference between system conformance checking and process conformance checking; in general, process conformance checking is a subcase of system conformance checking. The technique is depicted in Fig. 1 by the arrow from model to event log, and is defined as follows:

Definition 14 (Process conformance checking). *The process of verifying the conformance of the observed behavior of the implementation, as recorded in the event log, towards the business process model.*

In this definition, the word conformance may refer to both the conformance relation (Definition 11) and the collection of popular mining techniques. The specification is represented by a prescriptive normative process model that describes the intended behavior based on best practices, business rules, company policies, legal requirements, etc. The event log is again used as a proxy for the implementation, which implies that the conformance points are limited to the tasks in the process and their contents. Process conformance checking verifies whether the actual behavior of the system matches the prescribed behavior of the normative model, identifies (un)common behavior and new behavior that is not specified or allowed in the model, and reports on deviations.

One of the central concepts in process conformance checking is a so-called *alignment*, which describes a relation between a trace and an execution of a process model as a sequence of moves, relating events in the event log to activities in the model [4,10]. The moves in an alignment can be either a move on log, a move on model, or a synchronous move. An asynchronous move (i.e. a move on log or a move on model) incurs a cost, so that the *optimal* alignment (i.e. the closest match possible between the event log and the model) is defined as the alignment with the lowest total cost.

Another well-known approach uses a unified model of concurrent behavior called *event structures* [11]. In this approach, the event log and process model are each converted into an event structure, which are subsequently aligned via an error-correcting synchronized product. This is specifically suitable in cases where compact context-dependent feedback is required on deviations between the event log and process model.

3.3 Model Conformance

Event logs can be used as a specification to determine whether the process model provides an accurate depiction of the actual behavior, process or implementation. The verification technique used is still conformance checking, but we will refer to it as 'conformance checking for repair' to highlight the difference. The technique is depicted in Fig. 1 as the arrow from event log to model, and is defined as follows:

Definition 15 (Conformance checking for repair). *The process of verifying the conformance of the normative behavior of the business process model towards the observed behavior of the implementation, as recorded in an event log.*

In this definition, the word conformance refers to that of the relation defined in Definition 11. The relation of the process described here to the term conformance checking (Definition 14) is also relevant, as it effectively reverses the artifacts to be verified: the specification artifact is represented by the event log, whereas the model artifact is represented by the (descriptive) process model. That is, conformance checking for repair aims to identify scenarios where the model does not accurately describe the actual behavior as observed in the event log, to subsequently alter, or 'repair', the model by trying to incorporate the additional behavior observed from the event log. The idea is to alter the model such that it improves the correspondence between the model and the log as much as possible, usually by allowing inserting or skipping of activities. As such, the approach searches for models that are optimal in terms of fitness. That is, the fraction of behavior that is in the log but not possible according to the model is minimized.

Similar to process conformance, conformance for repair centralizes around the concept of alignment, where alternatives are provided to amend the model that optimizes the alignment such that the event log fits the repaired model at least as well as it fits the original model (see e.g. [21]). Alternative approaches offer an incremental procedure, where differences between the model and the log are presented to the user, who can subsequently choose whether or not to repair the difference (see e.g. [7]).

3.4 Model Compliance

Business processes are generally modeled following a certain standard such as the Business Process Model and Notation (BPMN) standard [16]. Standards like BPMN specify the elements and relations between elements allowed within its specified graphical notation of a business process model, how each element behaves, and more. Consequently, the used standard directly influences the design properties (Definition 6) of the model. Model compliance aims to verify not only syntactic adherence of the business process model to the used standard, but also semantic adherence to the design properties.

Correctness checking is the technique that verifies whether a process model is compliant with its design properties, and includes well-known techniques such as workflow-net soundness [1]. Note here that the term soundness specifically applies to correctness properties of the Petri-net based workflow-nets and should only be used when an intermediate workflow-net representation of the business process is used when establishing correctness. The correctness technique is depicted in Fig. 1 by the arrow between the business process model and design properties artifacts, and is defined as follows:

Definition 16 (Correctness checking). *The process of verifying compliance of the business process model towards the design properties.*

When using this technique, the act of verification entails using the business process model as the model for verification and checking it against a specification that describes the design properties. In this definition, the word compliance refers directly to the compliance relation of Definition 12 and not to that of regulatory compliance, which is discussed in the next section.

3.5 Regulatory Compliance

Companies are subject to large numbers of regulations (Definition 7) that affect the way they do business. When asked by authorities, companies must be able to prove that they comply with regulations, or be prepared to face large fines. In other words, they must prove regulatory compliance:

Definition 17 (Regulatory compliance). *Doing what has been asked or ordered, as required by rule or law [17].*

Regulatory compliance of business processes can be proven at different stages of the BPM lifecycle, while using different artifacts. At each stage, different techniques are required to verify whether a process model, a running instance of a process, or a process log adheres to a set of relevant regulations. Here we specifically use the word adheres because the different techniques, applied at the different stages of the BPM lifecycle, define different types of relations, i.e., compliance or conformance (Definitions 11–12).

At design time, the implementation does not exist and there are no running instances that generate data. Therefore, all that can be done is to check whether the specification of the process model *complies* (Definition 12) with the specification stating the regulations. In doing so, the technique attempts to prove compliance not only from the control flow perspective, but also other perspectives using semantic annotations [22]. Although it is possible to fully prove compliance of certain sets of regulations at design time, in most cases this process should be considered a preventative measure that attempts to *mitigate the risk of violating the regulations*. To ensure anything further, one must also prove the process was actually followed when performed (e.g., by proving process conformance). Nevertheless, the technique has no access to data from runtime instances and, therefore, can often not cover the full set of regulations. The technique is depicted in Fig. 1 by the arrow from regulations to model, and is defined as follows:

Definition 18 (Regulatory compliance checking). *The process of verifying compliance of the business process model towards the regulations in order to prove or disprove regulatory compliance of the modelled behavior.*

At runtime, data from running process instances can be used to determine whether the enactment satisfies the conditions given by the regulations. The activity can be understood as a *conformance* relation (Definition 11) where the conformance check points fully cover the requirements mandated by the regulations. Even if the conformance points cover the legal requirements, it is only possible to determine breaches against the regulations based on the events observed till the time when *regulatory compliance* (Definition 17) is checked by proving the conformance relation (Definition 11). However, we cannot use conformance to check if the full instance will satisfy the legal requirements, since—for the activities that have not been executed—we can only rely on the specified business process model to prove the *compliance* relation (Definition 12) for the remaining possible execution paths. The technique is illustrated in Fig. 1 by the arrow from the regulations to the event log, and is defined as below. Note that the name of the defined activity refers to *regulatory compliance* (Definition 17) even though the activity defines a *conformance* relation (Definition 11). This observation lies at the core of the discussion in the remainder of this position paper, and will be explored in detail.

Definition 19 (Runtime regulatory compliance checking). *The process of verifying the conformance of the currently observed behavior, as recorded in the event log, towards the regulations in order to prove or disprove regulatory compliance of the currently observed behavior.*

After-the-fact regulatory compliance checking, known as auditing, has access to the full instance data and can, therefore, prove regulatory compliance in its entirety. Using only this approach, however, is a high risk endeavor that companies prefer to mitigate as much as possible, because—at this point—any violation of the regulations that has happened cannot be rolled back anymore. As a result, regulatory compliance verification should occur at multiple stages of the BPM lifecycle to both mitigate risks of violations and prove regulatory compliance. For auditing, we speak of a *conformance* relation (Definition 11) where the set of conformance points cover the legal requirements to prove *regulatory compliance* (Definition 17). The technique is illustrated in Fig. 1 by the arrow from the regulations to the event log, and is defined as follows:

Definition 20 (Auditing). *The process of verifying the conformance of the observed behavior towards the regulations in order to prove or disprove regulatory compliance.*

4 Overview of the Relations and Goals of Verification

Within the area of BPM, the term business process conformance is mostly referred to in the context of the popular mining technique, while the term business process compliance generally refers to the context of regulatory compliance. In the context of verification, however, conformance and compliance are defined in the contexts of their *relations* (i.e., Definitions 11 and 12). When comparing perspectives, the use of the conformance and compliance terms does not match, as the *relation* and the *goal* of verification are used interchangeably. To highlight this mismatch between the verification relations and their goals, Table 1 summarizes the verification techniques described in Sect. 3. The table lists each technique together with the stage of the lifecycle it is applied, the artifacts used as the model and specification (i.e., Definitions 2–10), the type of relation (i.e., Definitions 11 or 12), and the goal of verification (i.e., Sects. 3.1–3.5).

Table 1. Overview of verification techniques in the context of BPM.

Verification technique	Lifecycle stage	Model artifact	Specification artifact	Relation type	Verification goal
System conformance checking	Implement	Implementation	Prescriptive model	Conformance	System conformance
Conformance checking	Enact	Event log	Prescriptive model	Conformance	Process conformance
Conformance checking	Diagnose	Event log	Prescriptive model	Conformance	Process conformance
Conformance checking for repair	Diagnose	Descriptive model	Event log	Conformance	Model conformance
Correctness checking	Design	Model	Design properties	Compliance	Model compliance
Regulatory compliance checking	Design	Model	Regulations	Compliance	Regulatory compliance
Regulatory compliance checking	Enact	Event log	Regulations	Conformance	Regulatory compliance
Auditing	Diagnose	Event log	Regulations	Conformance	Regulatory compliance

From Table 1, it can be observed that, between all verification techniques, only two relations are compliance relations, and both of these techniques use the business process

model as the model for verification. Secondly, out of the other six techniques that have a conformance relation, only four have a conformance related goal. Finally, although three different verification techniques have the goal of regulatory compliance, only one has an actual compliance relation, while the others have conformance relations.

Given these observations, it is clear that there exists a gray area between the use of the conformance and compliance keywords among the verification relations and goals. The main 'offenders' are the techniques of *regulatory compliance checking* during enactment and *auditing*. These techniques both define *conformance* relations with the goal of checking regulatory *compliance*. Both these techniques were naturally developed out of the realization that proving a compliance relation between two specifications (i.e., model and regulations) could only provide so many preventative guarantees, and that runtime data and temporal information is required for definitive and complete results. It is not that these techniques are at fault. They very much prove regulatory compliance while defining a conformance relation. The conformance relation does not, suddenly, become a compliance relation when one has the goal of verifying regulatory compliance, nor does the goal suddenly become verifying regulatory conformance.

Even though the compliance and conformance terms are effectively synonyms in everyday language, it remains especially important that both research and application have clearly defined lines between developed and applied techniques and their related *keywords*. In literature, however, the conformance and compliance keywords are increasingly used interchangeably, which may cause confusion around the positioning and application of the different verification techniques within the research community itself, as well as in their application areas.

To ameliorate the issue, we must establish clear boundaries for the use of the conformance and compliance keywords within the context of verification during the BPM lifecycle. Figure 2 illustrates a step towards our proposed solution, featuring an additional gray area compared to Fig. 1 that represents business process execution. It includes the subset of BPM lifecycle artifacts used and created during enactment.

Given the additional area, we can now see that we can define correct boundaries through the use of three keywords instead of two. These keywords are (i) compliance, (ii), conformance, and (iii) regulatory compliance. That is,

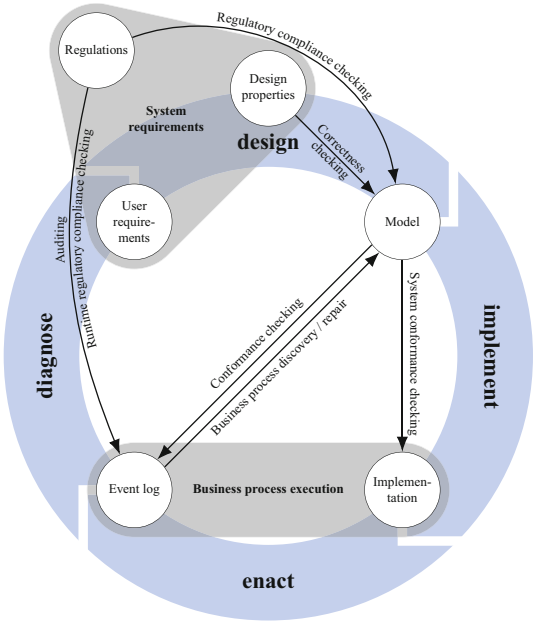

Fig. 2. Verification techniques applied during the BPM lifecycle (continued).

when we speak of *compliance*, we are applying verification using a specification from the system requirements and the business process model as the model for verification. On the other hand, when we speak of *conformance*, we are applying verification using the business process model with artifacts within the business process execution area. Finally, when we speak of *regulatory compliance*, we are applying verification using the regulations as the specification and artifacts within the business process execution area as the model for verification. Note that we use *compliance* (instead of regulatory compliance) to cover the verification of a model against regulations. Although this creates an area of overlap, this is not harmful since it correctly refers to compliance on both the relation and the regulatory goal. Moreover, when verifying (subsets of) the system requirements against a more refined set of such requirements, or a business process model against a more refined business process model, it is also *compliance*.

From this, it is clear that when using these three terms, it introduces clear boundaries that should be used to distinguish between verification techniques applied within the BPM lifecycle. The result is illustrated in Fig. 3, and should help both research and application to position work, accurately describe requirements, and interpret results. For example, consider an approach that obtains a business process model from an event log using a process mining technique and checks system requirements (e.g., regulations or user requirements) against the obtained model. That is, it obtains a model that describes the business process as it is performed in the real world (i.e., a descriptive model) from observed behavior of the implementation, and checks it against a specification. In this case, the approach would be a *regulatory compliance* approach when it verifies against regulations, a *compliance* approach when it verifies against design properties, and a *requirements validation* approach when it checks user requirements.

Note that we are not proposing the use of these keywords over more specific terms. Using more specific keywords is always encouraged. That is, using the keyword regulatory compliance over the keyword compliance when verifying regulations against the business process model is entirely correct. Instead, the proposed keywords should always be the highest level keywords used to describe techniques in the relevant areas. For instance, the keyword conformance should never be used to describe regulatory compliance even though, at a higher level, the technique describes a conformance relation. By following these guide-

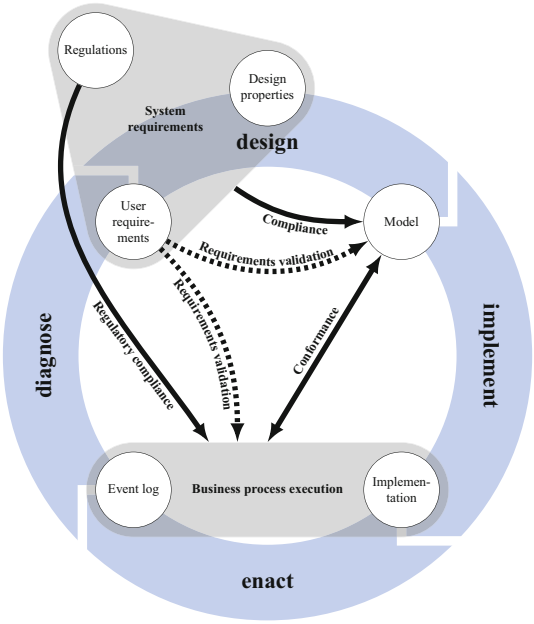

Fig. 3. Business process conformance and compliance during the BPM lifecycle.

lines, the community is ensured of using non-conflicting terminology and the proper positioning and application of techniques.

5 Discussion

The definition of clear boundaries between available techniques and tools is important for both researchers and practitioners. For researchers, it is not only important to ensure that the right terminology is used when describing their techniques and tools, but also to assist practitioners to select the correct tool for its intended purpose. Furthermore, such boundaries allow researchers to properly position their work, including the use of examples, selection of relevant related work, and evaluating against relevant work. For practitioners, on the other hand, it is important to ensure the validity of the results. That is, to ensure that the applied technique or tool verifies what was intended to be verified and be able to rely on the results and draw correct conclusions from those results. Consequently, more precise terminology allows to select the right portfolio of tools to collectively verify each aspect of the design and its implementation against each aspect of the set of system requirements, including user requirements, design properties, and regulations.

The question, however, remains whether some of the discussed techniques are possibly of value to the goals set for the other techniques. That is, we must discuss whether we actually should make the proposed distinction, or whether this is merely an intellectual issue. To do so, we discuss the relevance of some techniques to the goals set for the other techniques. That is, we discuss whether the technique of process conformance checking (Definition 14) is relevant to the goal of regulatory compliance (Sect. 3.5). Similarly, we discuss whether the technique of regulatory compliance checking (Definition 18) is relevant to the goal of process conformance (Sect. 3.2), and finally, we discuss whether the technique of process conformance checking (Definition 14) is always relevant to conformance from a legal point of view. In the remainder of this section, we discuss these questions, highlight any advantages or limitations that such applications yield, and present any analysis gaps that such applications may permit.

5.1 Should Process Conformance Be Used to Prove Regulatory Compliance?

As the popularity of process mining increased within the community, the idea slowly evolved that proving a *conformance* relation between an event log and a business process model can prove *regulatory compliance*. As such, the use of conformance checking techniques has been suggested as valuable to, for instance, agile compliance management [10] and GDPR [9]. Although technically conformance checking can be applied to prove regulatory compliance, it should be made clear that this approach is not ideal and can only prove regulatory compliance up to some point.

When using this approach, several strict conditions must be met, while results often lead to non-obvious inconclusive outcomes. First, a prescriptive business process model is required to check conformance. Second, this prescriptive model must be proven regulatory compliant using design time regulatory compliance checking (Definition 18).

One should be careful to note that, although design time regulatory compliance checking can check prescriptive models, it generally uses descriptive models. Third, the conformance checking must report any unfitting behavior. We must stress here that any unfitting behavior is not necessarily a violation of regulations. It simply means that a deviation was made from the possible executions described by the prescriptive model. As a result, this type of checking effectively denies any form of process flexibility.

Therefore, regulatory compliance can be proven through conformance checking by proving there is no unfitting behavior. However, it cannot prove that any unfitting behavior is an actual violation of regulations. One would still require additional regulatory compliance checking or auditing to prove this. In addition, it can only prove regulatory compliance along the control flow perspective, because the design time regulatory compliance checking techniques used to check the prescriptive model only has access to design time information and lacks process enactment information, such as data, resources, multiple instances etc. In this way, the limitations of the preventative measure of design time regulatory compliance checking (Definition 18) is transferred to an approach that in fact has process enactment information.

Although further model annotations of regulations are possible to consider other perspectives than that of the control flow, these approaches edge more towards also doing regulatory compliance checking while conformance checking, than just conformance checking—and would still deny any process flexibility. On the other hand, conformance checking approaches that enable process flexibility by allowing a certain level of unfitting behavior can never prove regulatory compliance without applying some form of actual regulatory compliance checking. As a result, the approach of using conformance to check regulatory compliance will always remain sub-optimal and should ideally be avoided. However, by continuing to use the keywords of conformance and compliance interchangeably, or using regulatory compliance examples to position conformance work, this approach may become common within application areas despite its non-ideal application.

5.2 Should Regulatory Compliance Be Used to Prove Process Conformance?

The application of regulatory compliance (Definition 18) to prove process conformance may, at first sight, seem completely irrelevant. However, it is possible but requires an unconventional approach. Again, it should be made clear that this approach is not ideal and can only prove conformance up to some point. That is, the approach can only obtain a degree of fitness and not a degree of precision. To obtain a degree of fitness of an event log with respect to a process model using regulatory compliance, we must first obtain a declarative specification of the prescriptive business process model. That is, we must obtain a set of declarative rules (e.g., temporal logic expressions) that together describe all possible paths within the business process model.

One example to automatically obtain such a declarative specification includes obtaining an event structure from (sets of) process model(s) and extracting a specification in the form of computation tree logic expressions [8]. Once a declarative specification is obtained, execution traces of the business process (captured by the event log) can be evaluated against the declarative specification using formal regulatory compliance verification techniques such as existing model checking tools and packages [12,13].

To obtain a degree of fitness for an execution trace, or all execution traces within the event log, we can divide the number of satisfied temporal logic expressions by the total number of temporal logic expressions being verified. In this way, the degree of fitness decreases as more temporal logic expressions are violated.

Next to the degree of fitness, results include sets of satisfied and violated temporal logic expressions. Consequently, these results will be difficult to interpret by non-experts. As a result, the approach to use regulatory compliance to check conformance is non-ideal due to partial and difficult to interpret results, and should be avoided. By continuing to use the keywords of conformance and compliance as being interchangeable, or using regulatory compliance examples to position conformance work, this approach may, however, appear within application areas despite its non-ideal application.

5.3 Should Process Conformance Always Be Used to Prove Legal Conformance?

In a previous section, we gave a short outline how to use what we called process conformance to prove regulatory compliance from the process oriented information systems point of view. In this section, we are going to look at the issue from a legal point of view. First of all, in legal documents there is often no real distinction between compliance and conformance (and, sometimes the two english terms are translated to a single term in other languages). The two terms both generically mean to obey to a set of prescriptions. For instance, consider the proposal for the European Union's Artificial Intelligence (AI) Act. According to the current proposal, AI (and more generally) systems operating in specific sectors have to comply with the Act, as the explanatory text recites:

> "Those AI systems will have to *comply* with a set of horizontal mandatory requirements for trustworthy AI and follow *conformity* assessment procedures before those systems can be placed on the Union market."

As we can see, the Act does not differentiate between the model of an AI system and its implementation. Furthermore, the Act seems to indicate that compliance refers to the behavior of day-to-day operations of the implementation; on the contrary, systems have to obtain conformity certificates before the system is placed on the market or operates in the European Union. Accordingly, conformance certificates are based on the evaluation of the systems before the systems are deployed. This poses the question if process and system conformance as understood in the business process community (as discussed in the previous sections) offer suitable techniques for providing conformance certificates for AI systems against the requirements set by the Act. The answer seems to be negative, since the requirements for conformance certificates appears to be closer to what we called regulatory compliance. Thus, while some of the techniques and methodologies developed for business processes appear adequate for the AI Act, the terminology used to describe them might not correspond to the terminology used by the legal and business communities; therefore, there is risk that BPM solutions will not fit for some applications or are evaluated with negative results, and effective techniques not to be adopted, limiting the impact of BPM technology for this important market.

6 Conclusion

The notions of conformance and compliance received substantial attention in the past decade in the BPM community. Often the two terms are used interchangeably, both in the field and in the broader community. However, from a technical point of view, they have been proposed with a different meaning. In general, compliance and conformance are two types of verification of systems, relating two BPM artifacts. In this paper, we provided comprehensive definitions of the two notions and activities related to them throughout the lifecycle of the development and deployment of process aware information systems and the artifacts they apply to (i.e., design specifications and regulatory frameworks, process models, implementations, and event log). While there are surface similarities among the verification problems and the activities specific to one of them, we discuss some of the reasons why, in general, effective methods for one particular type of verification (e.g., conformance) cannot guarantee a successful verification for a different type of relation (e.g., compliance). Accordingly, the discussion pointed out the need for a uniform set of definitions (and this is what we attempted in this contribution), and consequently, a unified terminology to present them. Finally, we addressed the problem whether the notions used in the BPM community have a counterpart in the wider audience, in particular, in the legal domain, where the terms are often used. It turns out that the picture is not so clear, given that the notions are used with their commonly understood meaning (corresponding essentially to what we call regulatory compliance) and not with their technical meaning. The major observation is that when interacting with external partners, first one has to understand what is the verification problem to be addressed, and then to determine what are the technical capabilities to be used. We believe that the discussions about the different techniques (and the shortcomings of using other techniques) offer guidelines to see how to succeed in the tasks based on BPM technology.

References

1. Aalst, W.M.P.: Verification of workflow nets. In: Azéma, P., Balbo, G. (eds.) ICATPN 1997. LNCS, vol. 1248, pp. 407–426. Springer, Heidelberg (1997). https://doi.org/10.1007/3-540-63139-9_48
2. van der Aalst, W.M.P.: The application of petri nets to workflow management. J. Circuits, Syst. Comput. **8**(01), 21–66 (1998)
3. van der Aalst, W.M.P.: Process Mining-Data Science in Action, 2nd edi. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49851-4
4. van der Aalst, W.M.P., Adriansyah, A., van Dongen, B.: Replaying history on process models for conformance checking and performance analysis. Wiley Interdisc. Rev. Data Min. Knowl. Discov. **2**(2), 182–192 (2012)
5. van der Aalst, W.M.P., Weijters, T., Maruster, L.: Workflow mining: discovering process models from event logs. IEEE Trans. Knowl. Data Eng. **16**(9), 1128–1142 (2004)
6. van der Aalst, W.M.P., ter Hofstede, A.H.M., Weske, M.: Business process management: a survey. In: van der Aalst, W.M.P., Weske, M. (eds.) BPM 2003. LNCS, vol. 2678, pp. 1–12. Springer, Heidelberg (2003). https://doi.org/10.1007/3-540-44895-0_1

7. Armas Cervantes, A., van Beest, N.R.T.P., La Rosa, M., Dumas, M., García-Bañuelos, L.: Interactive and incremental business process model repair. In: Panetto, H., et al. (eds.) OTM 2017. LNCS, vol. 10573, pp. 53–74. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-69462-7_5
8. van Beest, N.R.T.P., Groefsema, H., García-Bañuelos, L., Aiello, M.: Variability in business processes: automatically obtaining a generic specification. Inf. Syst. **80**, 36–55 (2019)
9. Burattin, A., van Zelst, S.J., Armas-Cervantes, A., van Dongen, B.F., Carmona, J.: Online conformance checking using behavioural patterns. In: Weske, M., Montali, M., Weber, I., vom Brocke, J. (eds.) BPM 2018. LNCS, vol. 11080, pp. 250–267. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-98648-7_15
10. Carmona, J., van Dongen, B.F., Solti, A., Weidlich, M.: Conformance Checking-Relating Processes and Models. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-99414-7
11. García-Bañuelos, L., van Beest, N.R.T.P., Dumas, M., La Rosa, M., Mertens, W.: Complete and interpretable conformance checking of business processes. IEEE Trans. Software Eng. **44**(3), 262–290 (2017)
12. Groefsema, H., van Beest, N.R.T.P., Aiello, M.: A formal model for compliance verification of service compositions. IEEE Trans. Serv. Comput. **11**(3), 466–479 (2018)
13. Groefsema, H., van Beest, N.R.T.P., Armas-Cervantes, A.: Automated compliance verification of business processes in apromore. In: BPM Demo Track (CEUR), vol. 1920, pp. 1–5 (2017)
14. Groefsema, H., Bucur, D.: A survey of formal business process verification: from soundness to variability. In: Third International Symposium on Business Modeling and Software Design (BMSD 2013), pp. 198–203. SciTePress (2013)
15. International Organization for Standardization: Information technology – open distributed processing, reference model: Overview part 1. Standard ISO/IEC 10746–1:1998, International Organization for Standardization, Geneva, CH, December 1998. https://www.iso.org/standard/20696.html
16. International Organization for Standardization: Information technology – object management group business process model and notation. Standard ISO/IEC 19510:2013, International Organization for Standardization, Geneva, CH, July 2013. https://www.iso.org/standard/62652.html
17. International Organization for Standardization: Systems and software engineering – vocabulary. Standard ISO/IEC/IEEE 24765:2017(E), International Organization for Standardization, Geneva, CH, September 2017. https://www.iso.org/standard/71952.html
18. Kiepuszewski, B., ter Hofstede, A.H.M., Bussler, C.J.: On structured workflow modelling. In: Wangler, B., Bergman, L. (eds.) CAiSE 2000. LNCS, vol. 1789, pp. 431–445. Springer, Heidelberg (2000). https://doi.org/10.1007/3-540-45140-4_29
19. Ko, R.K.L.: A computer scientist's introductory guide to business process management (BPM). Crossroads **15**(4), 4:11-4:18 (2009)
20. Milosevic, Z., Bond, A.: Digital health interoperability frameworks: Use of RM-ODP standards. In: EDOC Workshop 2016, pp. 1–10 (2016)
21. Polyvyanyy, A., van der Aalst, W.M.P., ter Hofstede, A.H.M., Wynn, M.T.: Impact-driven process model repair. ACM Trans. Software Eng. Methodol. **25**(4), 1–60 (2016)
22. Sadiq, S., Governatori, G., Namiri, K.: Modeling control objectives for business process compliance. In: Alonso, G., Dadam, P., Rosemann, M. (eds.) BPM 2007. LNCS, vol. 4714, pp. 149–164. Springer, Heidelberg (2007). https://doi.org/10.1007/978-3-540-75183-0_12