Business Process Compliance Checking: Current State and Future Challenges*

M. El Kharbili, A.K. Alves de Medeiros, S. Stein and W.M.P. van der Aalst
IDS Scheer AG, Altenkesseler Str. 17, D-66115 Saarbrücken, Germany.
E-mails:{marwane.elkharbili, sebastian.stein}@ids-scheer.com
Eindhoven University of Technology, P.O. Box 513, 5600MB, Eindhoven.
E-mails:{a.k.medeiros, w.m.p.v.d.aalst}@tue.nl

Abstract: Regulatory compliance sets new requirements for business process management (BPM). Companies seek to enhance their corporate governance processes and are required to put in place measures for ensuring compliance to regulations. In this sense, this position paper (i) reviews the current work in the context of BPM systems and (ii) suggests future directions to improve the current status. During the literature review, techniques are classified as supporting *forward* or *backward* compliance. The latter is a post-execution compliance (i.e. based on execution histories of systems) and the former takes place at design- or run-time. In a nutshell, this position paper claims that four main aspects need to be incorporated by current compliance checking techniques: (i) an integrated approach able to cover the full BPM life-cycle, (ii) the support for compliance checks beyond control-flow-related aspects, (iii) intuitive graphical notations for business analysts, and (iv) embedding semantic technologies during the definition, deployment and executions of compliance checks.

1 Introduction

Complying to regulations of all sorts is usually needed for various purposes. These range from ensuring that specific norms are met (e.g. quality standards such as ISO9000:2005[1]) to proving correct implementation of internal controls imposed by active legislations (e.g. SOX Sec.404[2]) [SGN07]. BPM is an integrated approach to both managing business and governing underlying IT layers supporting it [Wes07]. Therefore, ensuring the compliance of business processes (BPs) in companies is a crucial feature for a BPM system. This position paper provides an overview of the state-of-the-art efforts in the BPM community for supporting compliance checking of BPs and based on this, discusses aspects that still need to be addressed by future research. The remainder of this paper is organized as follows. Section 2 reviews the related work in for compliance checking of BPs. Section 3 provides an outlook on the current state of research and identifies points that still need to be adressed. Finally, Section 4 concludes this paper and indicates next steps of our work.

*Acknowledgements: **Our thanks go to the EU commission for supporting our research within the SUPER project (*www.ip-super.org*).***

[1] International Organization for Standardization: *www.iso.org*.

[2] The sarbanes-Oxley Act: *www.soxlaw.com*.

2 State-of-the-Art

Our review of related work classifies them into two approaches: *forward compliance checking* and *backward compliance checking*. This classification basically indicates if the techniques have a *pro-active* or *reactive* approach for compliance checking. *Forward* compliance checking techniques target the verification of rules during *design* time or *execution* time. Therefore, these techniques can prevent the actual execution of non-compliant behavior. Subsection 2.1 contains our review for these kind of techniques. *Backward* compliance checking techniques can *detect* that non-compliant behavior has taken place by looking at the history of BP instances' execution and are unable to prevent actual non-compliant behavior. Subsection 2.1 describes the main techniques following this approach.

2.1 Forward Compliance Checking

2.1.1 Design-Time Compliance Checking (DTCC)

These techniques have a preemptive nature and aim at guaranteeing that process instances will be regulatory compliant. In this sense, some approaches guide the user during modelling phase to limit non-compliance of deployed models, while others use techniques like model checking to verify certain properties in already designed (but not yet deployed!) models. Ghose et al. [GK07] propose an approach based on the so-called *compliance patterns* (i.e., pre-defined BP models for which compliance to regulations has been proven). The main idea is to compute the deviation of a given BP model to a certain compliance pattern. Governatori et al. [PR] view BPs as social interaction processes and present a framework for managing the compliance of contractual relationships in BPs. For this, deontic assignments are defined in a multi-modal logical framework in the form of policies/rules. In a following work, Sadiq et al. [GG06, MOS06] present an approach to formalize contract documents and those aspects of BPs that relate to these business contracts. For this purpose, the semantics of business contracts and their violations are described using a specialized logic, the Formal Contract Language (FCL) [GZ05]. Furthermore, the authors have shown how these formal specification of contracts can used to generate compliant processes. Namiri et al. [NS08] presents a formalization for definition BP compliance checking that relies on control patterns. Control patterns constitute a generic and reusable solution to a specific problem and, therefore, can be used to ensure that BP models containing them are regulatory compliant. In [NS07a], the authors add a semantic layer to the BPM stack in which process instances are interpreted according to a defined set of controls. The actual implementation (e.g. database procedures, temporal logic or rules) of the controls is independent from the modelling. In [SN07], the authors show why automation and semantic enactment are necessary for effective BP compliance. They formalize modelling of controls for compliance by motivating the need for logics that are stronger than standard deontic logic. Finally, the work by Schmidt et al. [SBO07] is one of the rare semantic approaches to BP compliance where a compliance ontology is designed and proposed to be integrated in BP models.

2.1.2 Run-Time Compliance Checking (RTCC)

These techniques target executable BP models and, consequently, depend on the BP execution architecture and mechanisms. They typically work by annotating BP models or steps with atomic compliance assertions that are destined to be either used by compliance checking engines for verification or at later stages during execution. In this sense, regulations can either be defined into BP models (e.g. control flow properties such as BP anti-patterns [JK07] which seek to achieve better quality of processes or organizational properties such as Separation of Duty (SoD)), or they can require run-time information (e.g. quality assertion enforcement while executing a BPEL process such as in [DF06]). The authors of [LMX07] identify the need for separate modeling of compliance and processes. Process models are transformed from BPEL into the Pi-Calculus [Mil93] (an algebra for modelling concurrent communicating processes) and compliance rules are modeled in temporal logic using a special graphical notation. Model checking techniques are then used to formally check a process pool. The work in [NS07b] (cf. DTCC) also encompasses execution aspects of BP compliance by proposing a solution for ensuring effectiveness of controls during BP execution and a reaction strategy in case of violation. The approach in [PSSvdA07] introduces a framework in which process models are defined in a declarative way. The authors argue that constraint-based workflow models are more expressive and more flexible than procedural ones. [PFLN04] explains how technical Service Level Agreements (SLA) can be leveraged to the business level. Contracts are specified between entities participating in inter-organizational BP cooperations, and a language for use as part of a business contract architecture is defined. The work builds on an underlying policy framework for description of behavioral constraints. In [Mil05a], policy definitions are integrated into BPs and rely on BP events and transactions for run-time compliance monitoring. In fact, this work poses initial questions about architectures for process compliance monitoring integrating events and policies such as the need for a formal definition of events, event triggers and related resources, event patterns, message handling as well as state management. Additionally, business rule management systems are widely used in the industry for production rule execution. Some compliance measures can be modeled as business rules and be coupled to BP definitions(as is done in the ARIS business rules designer[3]).

2.2 Backward Compliance Checking

Backwards compliance checking (BCC) techniques verify if executions of BP (i.e., process instances) are in accordance with certain constraints or rules. The works in [RA08, ABD05, CMMS07, ea07] are good representatives of current approach for BCC. Rozinat et at. [RA08] has developed *conformance checking* techniques that quantify how much the behavior of a given control-flow process model matches the behavior registered in process instances of a given history log. Whenever differences (or non-compliance) are detected, the developed conformance checking techniques provide an exact indication of where the differences are. This approach has the advantage that a graphical notation is used for specifying the models. However, the provided compliance is restricted to control-flow-related constraints. Thus, no rules involving data fields or performers information can be checked.

[3] www.ids-scheer.com/en/Software/ARIS$_{Software}/ARIS_{Business_{Rules_{Designer}}}/3747.html$

Aalst et al. [ABD05] have created the *LTL Checker*, a technique based on Linear Temporal Logic [GH01]. The approach verifies if a given LTL rule holds for a set of process instances. The result is a partition of this set into two other subsets: one containing the compliant process instances and another containing the non-compliant ones. Although the rules can also contain references to data fields, performers and temporal aspects, no graphical notation is supported, making this techniques less suitable for direct use by business analysts. The works by Alberti et al. [ea07, CMMS07] combine the power of computational logics with graphical notations to specify rules. Models can be made in GOSpeL, a graphical language created by the authors. Afterwards, these graphical models are translated into SCIFF, a declarative language based on computational logic, and applied over proces instances. Although mainly targeted at BCC, the approach can also be used for forward compliance checking by translating the models into the G-SCIFF logic language.

3 Outlook

None of the approaches introduced above tackles compliance checking as having the whole BPM lifecycle as a scope. Concretely this means that that compliance is a vertical concern and starts already at a strategy level and goes down BPM layers to include the design, implementation, configuration, execution and analysis layers taking into account the orthogonal organizational, both design-time (e.g. EPC) and run-time (e.g. BPEL) control and event flow, functional and data aspects. Such an observation is emphasized when seeing natural-language regulations (e.g. laws, standards, norms etc.) as the initial input of every compliance checking effort. These regulations have then different representations depending on the layer where they have to be enforced (i.e contracts language in [GZ05] or temporal logic constraints on BPEL processes in [LMX07]). A *Compliance checking framework shall cover all phases of the BPM life-cycle* because this will lead to a fully integrated approach, where the complexity of compliance architectures, algorithms and coverage is reduced.
Besides, declarative means of modeling compliance present an appealing alternative to "hard-coded" checking algorithms because they scale with change in both the compliance requirement and the BP dimensions. While relying on the definition of compliance as SWRL rules, Yip et al. [F.Y07] recognize the limited expressiveness of the language and propose the use of extensions. As the logic necessary to model compliance varies depending on the application case (e.g. temporal, deontic, horn logic, etc.) a *holistic approach should take this into consideration and offer the possibility to use formalisms of different degrees of expressiveness and computational complexity*. Although the need for formal modeling is clearly identified by many of the approaches listed, the complexity of the developed tools and the pre-required knowledge (e.g. about formal logics) is frequently underestimated as an adoption barrier. This observation is made in [LMX07] and pushed the authors to develop a graphical language for modeling temporal logic constraints on BPEL processes. *A graphical notation for modeling compliance that is easy to learn and use by business analysts will help endorse existing approaches, as it allows hiding the complexity of defining logical constraints formally*. Such an assertion can of course only be validated by rigorous empirical research. A similar observation can be made on the works by Alberti et al. [ea07, CMMS07]. In this regard, the work by Koehler et al. [JK07]

on anti-patterns for BP control flows deserves mention and could be built on to modeling reference compliant BP patterns in order to measure compliance by comparing BPs against the latter.
To scale with the complexity of continuously changing and ambiguous regulations as well as re-engineered BPs while keeping compliance costs reasonable, there is a need for a declarative environment for formal modelling of BP compliance taking regulations as input. Such an environment should support both inter- and intra-organizational collaborations (e.g. markets), as well as take enterprise models into consideration, and *not be only tailored towards structural or syntactical properties of BPs*. Policies as discussed by Milosevic in [Mil05b] would allow to separate between compliance management and decision solving problems (i.e. when is a state or behaviour non-compliant? which policy should be responsible for ensuring compliance? etc.) from compliance checking or enforcement problems (i.e. policy implementation, e.g. as business rules).
Another aspect of compliance are domains such as quality or security. In fact, compliance needs to be modelled differently depending on the domain it has to be enforced in. Aspects of BP compliance that are of fundamental nature such as control flow are generic problems whereas modeling compliance to the ISO27001 standard requires specific constructs to be regarded for BPs. Yip et al. [F.Y07] study compliance for the security domain and extract specific security rules semi-automatically from a security standard. In such cases, semantics play an important role. Also, an important facet in guaranteeing BP regulatory compliance is *enabling business experts to directly supervise the modelling and implementation of compliance management measures*, because requirements on BPs are best made by experts possessing the necessary knowledge of a company's business. Compliance needs to be enforced in domains of higher abstraction than what elementary BPM languages are able to express. The scope of BPs that is considered expands beyond workflows (to incorporate organizational and resource/data aspects) making it necessary to have a sound understanding of the semantics of the handled assets. *Making these semantics machine processable by means of formal conceptual modeling (e.g. ontologies) allows compliance checking to take these into account*. This observation is also made in [LGRMD08] where requirements for semantic constraint support are analyzed. This helps handling the ambiguity of natural language regulations but semantic modeling of compliance is unfortunately still highly ignored by current approaches.

4 Conclusions and Future Work

This position paper has focused on answering the following research question: How are current approaches within the BPM research area handling compliance checking? To answer this, a state-of-the-art review of research on this topic has been performed. This review indicates that current works have made a good progress in this direction but no approach is still able to cover *all* phases of the BPM life-cycle. Furthermore, most approaches seem to be yet quite far from their target end users: business analysts. We have provided an outlook explaining the four main factors that need to be incorporated by current compliance checking techniques: (i) an integrated approach able to cover the full BPM life-cycle, (ii) the support for compliance checks beyond control-flow-related aspects, (iii) intuitive graphical notations for business analysts, and (iv) embedding of semantic technologies during the definition, deployment and executions of compliance checks. There-

fore, having chosen policies and business rules as formal means of modeling compliance, our future research agenda will be centered around defining compliance checking techniques that are semantic, declarative, policy-based and cover the full BPM lifecycle.

References

[ABD05] W.M.P. van der Aalst, H.T. de Beer, and B.F. van Dongen. Process Mining and Verification of Properties: An Approach Based on Temporal Logic. In R. Meersman et al., editor, *OTM Conferences (1)*, volume 3760 of *LNCS*, pages 130–147. Springer, 2005.

[CMMS07] F. Chesani, P. Mello, M. Montali, and S. Storari. Testing Careflow Process Execution Conformance by Translating a Graphical Language to Computational Logic. In R. Bellazzi, A. Abu-Hanna, and J. Hunter, editors, *AIME*, volume 4594 of *Lecture Notes in Computer Science*, pages 479–488. Springer, 2007.

[DF06] Wilhelm Rossak Daniel Foetsch, Elke Pulvermueller. Modeling and Verifying Workflow-based Regulations. In *Proceedings of the international workshop on regulations modeling and their validation and verification. REMO2V06.*, pages 825–830. CEUR-WS.org/vol-241, Luxemburg, June 2006.

[ea07] M. Alberti et al. Expressing and Verifying Business Contracts with Abductive Logic Programming. In G. Boella et al., editor, *Normative Multi-agent Systems.*, volume 07122 of *Dagstuhl Seminar Proceedings*. IBFI, Schloss Dagstuhl, Germany, March 2007.

[F.Y07] N. Parameswaran & P. Ray F.Yip. Rules and Ontology in Compliance Management. In *Proceedings of the 11th IEEE International Enterprise Distributed Object Computing Conference*, number 1541-7719, page 435, 2007.

[GG06] Shazia Sadiq Guido Governatori, Zoran Milosevic. Compliance checking between business processes and business contracts. In *Proceedings of the 10th IEEE International Enterprise Distributed Object Computing Conference (EDOC'06)*, pages pp. 221–232, 2006.

[GH01] D. Giannakopoulou and K. Havelund. Automata-Based Verification of Temporal Properties on Running Programs. In *ASE '01: Proceedings of the 16th IEEE international conference on Automated software engineering*, page 412, Washington, DC, USA, 2001. IEEE Computer Society.

[GK07] A.K. Ghose and G. Koliadis. Auditing business process compliance. In *Proceedings of the International Conference on Service-Oriented Computing (ICSOC-2007)*, volume 4749 of *Lecture Notes in Computing Sicence*, pages 169–180, 2007.

[GZ05] G. Governatori and Milosevic. Z. Dealing with contract violations: formalism and domain specific language. In *Proceedings of the Conference on Enterprise Computing EDOC 2005.*, page 4657. IEEE Press., 2005.

[JK07] Jussi Vanhatalo Jana Koehler. Process anti-patterns: How to avoid the common traps of business process modeling. *IBM WebSphere Developer Technical Journal.*, 28 Feb 2007.

[LGRMD08] L.T. Ly, K. Göser, S. Rinderle-Ma, and P. Dadam. Compliance of Semantic Constraints A Requirements Analysis for Process Management Systems. In Sadiq S., Indulska M., and zur Muehlen M., editors, *Proceedings of the International Workshop on Governance, Risk and Compliance - Applications in Information Systems (GRCIS08).*, June 2008.

[LMX07] Y. Liu, S. Müller, and K. Xu. A static compliance-checking framework for business process models. *IBM Syst. J.*, 46(2):335–361, 2007.

[Mil93] R. Milner. The polyadic pi-calculus: a tutorial. In F. L. Bauer, W. Brauer, and H. Schwichtenberg, editors, *Logic and Algebra of Specification*, pages 203–246. Springer-Verlag, 1993.

[Mil05a] Z. Milosevic. Towards Integrating Business Policies with Business Processes. In W.M.P. van der Aalst, B. Benatallah, F. Casati, and F. Curbera, editors, *Business Process Management*, volume 3649, pages 404–409, 2005.

[Mil05b] Zoran Milosevic. *Towards Integrating Business Policies with Business Processes*, pages 404–409. 2005.

[MOS06] Sadiq S. W. Milosevic, Z., J. L. Fiadeiro Orlowska, M. E. In: S. Dustdar, and A. P. Sheth. Towards a methodology for deriving contract-compliant business processes. In *Proceedings of the 4th International Conference on Business Process Management (BPM06)*, Vienna, Austria., 2006. 5-7 September, 2006.

[NS07a] K. Namiri and N. Stojanovic. Pattern-Based Design and Validation of Business Process Compliance. In R. Meersman and Z. Tari, editors, *OTM Conferences (1)*, volume 4803 of *Lecture Notes in Computer Science*, pages 59–76. Springer, 2007.

[NS07b] Kioumars Namiri and Nenad Stojanovic. *Pattern-Based Design and Validation of Business Process Compliance*, pages 59–76. 2007.

[NS08] K. Namiri and N. Stojanovic. Towards A Formal Framework for Business Process Compliance. In M. Bichler, T. Hess, H. Krcmar, U. Lechner, F. Matthes, A. Picot, B. Speitkamp, and P. Wolf, editors, *Multikonferenz Wirtschaftsinformatik*. GITO-Verlag, Berlin, 2008.

[PFLN04] J. Cole S. Gibson S. Kulkarni P. F. Linington, Z. Milosevic and S. Neal. A unified behavioural model and a contract language for extended enterprise. *Data & Knowledge Engineering.*, Volume 51, Issue:5–29, October 2004.

[PR] Governatori G. Sadiq S. Colomb R. M. Padmanabhan, V. and A. Rotolo. Process Modelling: The Deontic Way. In *The Third Asia Pacific Conference on Conceptual Modelling (APCCM 2006)*.

[PSSvdA07] M. Pesic, M. Schonenberg, N. Sidorova, and W. van der Aalst. *Constraint-Based Workflow Models: Change Made Easy*, pages 77–94. 2007.

[RA08] A. Rozinat and W.M.P. van der Aalst. Conformance Checking of Processes Based on Monitoring Real Behavior. *Information Systems*, 33(1):64–95, 2008.

[SBO07] R. Schmidt, C. Bartsch, and R. Oberhauser. Ontology-Based Representation of Ccompliance Requirements for Service Processes. In *Proceedings of the Workshop on Semantic Business Process and Product Lifecycle Management*, pages 28–39, 2007.

[SGN07] S.W. Sadiq, G. Governatori, and K. Namiri. Modeling Control Objectives for Business Process Compliance. In G. Alonso, P. Dadam, and M. Rosemann, editors, *BPM*, volume 4714 of *Lecture Notes in Computer Science*, pages 149–164. Springer, 2007.

[SN07] Governatori G. Sadiq, S. and K. Namiri. *Modeling Control Objectives for Business Process Compliance*, pages 149–164. Lecture Notes in Computer Science. Springer, 2007.

[Wes07] M. Weske. *Business Process Management: Concepts, Languages, Architectures*. Springer-Verlag, Berlin, 2007.