Towards Compliance of Cross-Organizational Processes and their Changes*
Research Challenges and State of Research

David Knuplesch¹, Manfred Reichert¹, Jürgen Mangler², Stefanie Rinderle-Ma², and Walid Fdhila²

¹ Institute of Databases and Information Systems, Ulm University, Germany
² Faculty of Computer Science, University of Vienna, Austria
{david.knuplesch,manfred.reichert}@uni-ulm.de,
{juergen.mangler,stefanie.rinderle-ma,walid.fdhila}@univie.ac.at

Abstract. Businesses require the ability to rapidly implement new processes and to quickly adapt existing ones to environmental changes including the optimization of their interactions with partners and customers. However, changes of either intra- or cross-organizational processes must not be done in an uncontrolled manner. In particular, processes are increasingly subject to compliance rules that usually stem from security constraints, corporate guidelines, standards, and laws. These compliance rules have to be considered when modeling business processes and changing existing ones. While change and compliance have been extensively discussed for intra-organizational business processes, albeit only in an isolated manner, their combination in the context of cross-organizational processes remains an open issue. In this paper, we discuss requirements and challenges to be tackled in order to ensure that changes of cross-organizational business processes preserve compliance with imposed regulations, standards and laws.

1 Introduction

Improving the efficiency and quality of their business processes and optimizing their interactions with partners, suppliers and customers have become significant success factors for any enterprise. Hence, enterprises increasingly adopt emerging technologies and standards for business process automation [1]. These enable the definition, execution, and monitoring of the operational processes of an enterprise [2, 3]. In connection with Web service technology, the benefits of business process automation and optimization from within a single enterprise can be transferred to cross-organizational business processes as well [4, 5]. The next step in this evolution will be the emergence of the agile enterprise being able to rapidly implement new processes and to quickly adapt existing ones to

* This work was done within the research project C³Pro funded by the German Research Foundation (DFG), Project number: RE 1402/2-1, and the Austrian Science Fund (FWF), Project number: I743.
environmental changes [3]. While flexibility issues for internal business processes and their implementation (i.e., process and service orchestrations) are well understood [3, 2], the controlled change of the interactions between IT-supported partner processes in a cross-organizational setting (i.e., process and service choreographies) has not been adequately addressed so far. If one partner changes its process in an uncontrolled manner, inconsistencies or errors (e.g., deadlocks) regarding these interactions might occur. This is particularly challenging if there exist running instances (i.e., cases) of these process choreographies. As a consequence, adaptations of cross-organizational business processes turn out to be costly and error-prone.

Generally, business processes cannot be defined or changed without considering business process compliance with imposed compliance rules (e.g., security guidelines). Due to the increasing importance of regulations like SOX and BASEL, compliance has emerged as one of the most urgent challenges for process-aware information systems. So far, it has been addressed by many approaches, which mostly deal with the automation of audits for verifying compliance rules imposed on internal business processes [6, 7, 8, 9].

Compliance of cross-organizational business processes, however, has not been investigated in connection with process changes or with respect to privacy constraints of partner processes. Flexibility on one hand and compliance on the other hand are crucial challenges for collaborative settings. However, the picture will be not complete if we do not consider both in interplay as well. Even though a compliance rule might be fulfilled for a collaborative process before a change, it does not automatically remain satisfied afterwards. Thus, it is indispensable to provide adequate mechanisms to control change effects on the compliance of collaborative business processes in a transparent way.

This paper first provides an overview of basic notations related to compliance and (cross-organizational) business processes in Section 2. Section 3 then discusses different layers of correctness that must be considered when modeling and changing business process models. Section 4 discusses the state of the art in related research areas; i.e., cross-organizational processes, process flexibility, and business process compliance. Section 5 elicits novel requirements and challenges for enabling compliance of cross-organizational business processes and their change. Finally, Section 6 closes the paper with summary and conclusion.

2 Basic Notion and Example

We introduce basic notions related to business compliance in the context of cross-organizational processes and discuss their inter-relations. Fig. 1 gives an overview of the terminology we use. Basic to each cross-organizational process is a collaboration. In turn, the latter contains abstract roles that may be filled by concrete business partners. For each role, the set of related task definitions describes which tasks a role may perform when a cross-organizational process
is executed. The model of a cross-organizational process is denoted as choreography model and consists of public process models connected through message exchanges. In turn, a public process model consist of tasks and refers to a role, whereas a private process model implements a public process model of a business partner enriched with internal (i.e., private) tasks. Both public and private process models may only comprise task definitions related to the role the public process model refers to.

A choreography instance reflects the execution of a cross-organizational process defined by a choreography model. Similar to a choreography model, the choreography instance consists of public process instances, which are based on the private process instances of the related business partner. Such a private process instance is controlled by the respective business partner through executing its task instances. The latter send and receive message instances connecting the public process instance within the choreography instance.

Both the choreography model and the private process models may be subject to compliance rules restricting allowed execution sequences of tasks. In this context, we distinguish between global compliance rules referring to a cross-organizational process (i.e., to a choreography model) and internal compliance rules restricting private process models. Furthermore, warranties can augment public process models. More precisely, they enrich a choreography’s specification of a public process model with additional information about its execution behavior. In turn, a warranty constitutes an internal compliance rule restricting the private process model that implements the augmented public process model.
Example 1. Fig. 2 shows an example of a cross-organizational process from the healthcare domain. The fundamental collaboration comprises three roles: gynecologist, laboratory, and women’s hospital. They are filled by the three business partners John Q. Public, M.D., EgLab Ltd., and University Hospital of Anytown. The choreography model of this cross-organizational process is based on the public process models of the three roles that communicate through message exchanges. For example, the message exchanges sample and results connect the public process of the gynecologist with the one of the laboratory. In turn, the public process of the gynecologist is connected with the one of the women’s hospital based on the two message exchanges patient data and results. The private process models of the business partners, in turn, implement the related public processes and additionally enrich them with internal behavior, i.e. private tasks not contained in public processes models. (In Fig. 2 the private tasks are indented and greyed). The logical execution of this choreography model results in a choreography instance of the cross-organizational healthcare process; it contains related instances of the choreography, i.e. its public as well as private processes, and the messages exchanged. Finally, Fig. 2 shows one warranty and two global compliance rules. Through the warranty, the laboratory ensures the destruction of a blood sample after its analysis. Rule 1 requires a blood sample to be destroyed. In turn, Rule 2 requires the patient to be monitored in the hospital when complications occur.
3 Correctness Levels for Process Models

Business processes models cannot be defined or changed in an arbitrary way. Generally, three correctness levels build on each other and constitute a pyramid of business process model correctness. In particular, each level has to be considered and checked for each definition or change of a process model (cf. Fig. 3):

**Syntactical Correctness** refers to the correct use and composition of the elements of the underlying meta model. Examples of syntactical constraints include the existence of start and end events, as well as the correct use of different kinds of edges; e.g., control flow edges may only connect flow elements (i.e., tasks, gateways, events), while data flow edges connect tasks with data objects. **Syntactical correctness** is a prerequisite for **behavioral correctness**, since the behavior of a syntactically incorrect model is undefined.

**Behavioral Correctness** requires from a process model to be executable and includes properties like the absence of deadlocks and lifelocks. Further, it requires a correct flow of data; e.g., data objects must be written before read the first time. In the context of (dynamic) process changes, **state consistency and data consistency** respectively, must be preserved; i.e., a running process instance must not face deadlocks, lifelocks, and data-flow errors, when dynamically migrating its execution to a new process model version. In the context of cross-organizational processes, **compatibility** between the public processes of the different partners requires their composition is a behaviorally correct process. **Conformance** requires the private process of a partner to implement the behavior of his public process. Finally, **behavioral correctness** is a prerequisite for **semantic correctness**.

**Semantic Correctness** (i.e. business process compliance) requires a process model to comply with imposed compliance rules stemming from regulations, standards, and laws. Hence, each possible execution of the process must not violate any compliance rule. For a set of compliance rules, **consistency** requires the conjunction of the rules to be satisfiable; i.e. consistent compliance rules do not conflict with each other.

4 State of the Art

In this section, we outline the state of research. First, we discuss existing approaches in the area of cross-organizational processes. Then, we deal with process flexibility. Finally, we discuss business process compliance.
4.1 Cross-organizational Processes

Modeling cross-organizational processes has been discussed for many years. There are widespread standards such as BPEL4WS, WS-CDL, and RosettaNet, as well as powerful modeling frameworks and notations (e.g., "Let’s dance" [10], iBPMN [11], and BPMN 2.0). [12] further identified interaction patterns, which describe well-defined patterns for message exchanges between partner processes. For privacy reasons, a choreography definition is usually restricted to those activities relevant for the message exchanges between the partners involved. More precisely, partners publish restricted views on their private processes [13, 14]. Several top-down-approaches exist, which, starting from a choreography of public processes, determine whether or not private processes comply with the corresponding public ones [15, 16, 17]. Furthermore, [18] introduces a set of transformation rules that allow for the inheritance-preserving stepwise enrichment of a public process to obtain the corresponding private one. These rules are also applicable to evolve private processes without changing their public view. To support the opposite direction (i.e., to interconnect existing partner processes), [19] provides a bottom-up-approach for checking whether or not processes can interact with each other successfully. Further, they propose a central as well as distributed architecture, which both allow for the dynamic matching and execution of cross-organizational processes based on a shared registry for public processes. Finally, the scenario of modeling process choreographies and private processes independently is addressed by [20], using the Formal Contract Language to check conformance between choreographies and processes.

4.2 Process Flexibility

Issues related to process flexibility have been discussed for more than a decade [21, 22, 23, 5]. However, existing approaches consider flexibility mainly for processes orchestrations; i.e., workflows implementing a private process and being executed by a single process engine. In approaches like Pockets of Flexibility [24] or Worklets [25], for example, processes are executed on the basis of a loosely or partially specified model, which is then fully specified at runtime. Relevant techniques include late modeling and late composition of process fragments as well as declarative modeling styles [26].

Dynamic process adaptations, in turn, represent the ability of the implemented processes to cope with exceptional situations. On one hand, this includes the handling of expected exceptions, which can be anticipated and thus be captured in the process model [27]. On the other hand, it covers non-anticipated exceptions, which are usually handled through structural adaptations of single process instances based on well-defined change patterns (e.g., to add, delete or move activities) [28]. A particular challenge is to ensure the behavioral correctness (i.e., state and data consistency) of a process instance in this context [21]. Approaches like ADEPT [29] guarantee for the behavioral correctness of the modified process model.
Besides this, there exists support for assisting end-users in reusing ad-hoc changes [30] and for restricting changes to authorized users [31]. Another fundamental aspect concerns process schema evolution [32, 33]; i.e., the ability of the implemented process to change when the business process evolves. Relevant problems in this context concern the handling of running process instances, which were initiated based on the old model version, but are required to use the new specification from now on [33, 32]. Since thousands of active instances may be affected by a given process change the issue of behavioral correctness is rather critical. Traceability of changes and mining of dynamic processes are other relevant issues, closely related to process evolution, which are considered in existing frameworks [34, 35].

Only few approaches exist that address changes of distributed processes and choreographies. [36] shows how partitioned workflows can be changed in a controlled way. [37] distinguishes between shallow and deep service changes in the context of a choreography. While the effects of shallow changes (e.g., changes of service versions, interfaces, and operations) are restricted to a service, deep changes have cascading and disseminating effects on the whole choreography. [38] describes how the version of stateful service instances can be changed efficiently, if the behavior of the new service version covers the behavior of the replaced one. Constructing such new service versions is addressed by [39]. However, no comprehensive solution approach is provided.

4.3 Business Process Compliance

In many domains, process execution is subject to compliance rules and restrictions that stem from laws, regulations, and guidelines (e.g. Basel or Sarbanes-Oxley-Act) [40]. Existing approaches that allow ensuring compliance of business processes with imposed compliance rules differ with respect to the lifecycle phase in which compliance is verified as well as the strategy applied. Moreover, different paradigms and formalisms are used to define compliance rules and process models [41]. Compliance rules are often considered as restrictions to the order in which process activities may be executed. In this context, there exist approaches that formalize compliance rules in temporal logic (e.g., LTL [8] and CTL [6]). Other ones emphasize the modalities of compliance rules (e.g., obligations or permissions) by applying deontic logic [42, 43]. Since these approaches are not easy to comprehend, [44] suggests a pattern-based approach to encapsulate logic. There also exist graphical notations for modeling compliance rules [7, 8, 45]. The integration of compliance rules throughout the process lifecycle has been discussed in [46, 47, 48].

To verify whether compliance rules are fulfilled by process models at design time, many approaches apply model checking techniques [6, 7, 8, 9]. Since these depend on the exploration of the state space of process models, state space explosion constitutes a big obstacle for practical applications. Graph reduction and sequentialization of parallel flows as well as predicate abstraction are proposed to deal with this challenge [8, 7, 49]. Among those approaches,
there exist few that do not only consider the control flow perspective. [50] introduces state-based data objects. [49] enables data-aware compliance checking for larger data-domains and [9] additionally considers temporal constraints. For cycle-free processes, there exist algorithms that allow for more efficient design time compliance verification than model checking [20, 51].

Runtime checking and monitoring (i.e. continuous auditing [52]) of business process compliance are addressed by several approaches: [53] enriches a process models with a semantic layer of internal controls. Another compliance monitoring framework based on common event standards and middleware is presented in [54]. [55] discusses the monitoring and enforcement of compliance within process collaborations. [56] uses Compliance Rule Graphs and [57] colored automata to enable fine-grain compliance diagnostics at runtime.

To complement design time and runtime compliance checking, backward compliance checking of process logs has been proposed by checking logs for compliance with LTL-formulas [58]. Finally, declarative approaches [26, 43, 42] ensure compliance in an elegant manner. Since processes are defined by means of a set of rules, imposed compliance rules only have to be added to the process definition to ensure business process compliance.

In summary, there exist many approaches considering aspects of compliance of cross-organizational processes and their changes. However, only few approaches discuss flexibility issues in the context of cross-organizational processes or in the context of business process compliance. Even fewer approaches address business process compliance of cross-organizational processes. Finally, Fig. 4 shows that the interplay of change and compliance in the context of cross-organizational processes has not been addressed yet.

5 Challenges

As mentioned in Section 4, the interplay of change and compliance in the context of cross-organizational processes has not been addressed yet. In this section we outline challenges arising in the context of this interplay:

**Modeling Cross-organizational Compliance Rules.** First of all, the (graphical) modeling of warranties and cross-organizational compliance requires proper support. In addition to intra-organizational compliance rules, messages as well as the role executing a task must be considered (cf. Rule 2 of Example 1).
Change Propagation. Before applying a change, all three correctness levels have to be checked as we will illustrate by means of the scenario from Fig. 2: If the message-sending task forward results is deleted in the gynecologist’s private process, the corresponding public process will be affected. Furthermore, the message results as well as the public and private process of the women’s hospital will be affected by this change. This indicates that a partner must not change his public process in an uncontrolled manner. Otherwise, behavioral correctness of the choreography his public process is involved in cannot be preserved. Thus, the women’s hospital should be supported in adapting its public and private processes to the change. Hence, a major research challenge is how to propagate changes of a partner’s private process to the other partners in order to keep the choreography behaviorally correct.

Efficient Cross-organizational Instance Migration. Changes are particularly challenging if there exist running choreography instances being in different states or partially differing from the original process model (e.g., due to adaptations of single choreographies in the context of exceptions). For any of these hundreds or even thousands of choreography instances it has to be determined whether or not the change or parts of it are applicable, while satisfying all three correctness levels at the same time. Since large numbers of cross-organizational process instances may be affected, efficient algorithms are required.

Ensuring Compliance with Regard to Privacy. Using Example 1 from Fig. 2, we can show that privacy constraints of partner processes aggravate compliance checking at design time. For example, the gynecologist cannot correctly verify Rule 1, since the laboratory hides the relevant task destroy sample from its public process. Nevertheless, when additionally considering the warranty of the laboratory, it becomes clear that Rule 1 is satisfied. Similar difficulties occur when compliance is monitored at runtime. This requires a solution supporting compliance checking with respect to privacy issues.

Efficiently Ensuring Compliance at Change Time. Ensuring compliance of cross-organizational process models and instances at change time is another challenging research issue. Particularly, in the context of compliance, effects of changes cannot be easily traced. For example, assume that task destroy sample is deleted in the laboratory’s private process (cf. Fig. 2). Although this task is not part of the laboratory’s public process, compliance of the gynecologist’s process models with Rule 1 will be affected by this change. However, the task affects the laboratory’s warranty that must be withdrawn. Consequently, warranties, internal compliance rules, and global compliance rules should be re-evaluated. Due to the high complexity of current compliance checking approaches and due to the potentially large number of affected process instances, optimization strategies and efficient algorithms are required.

Adequate User Feedback. Users require intelligible feedback on compliance violations. Sources and effects of compliance violations have to be highlighted.
Furthermore, the available courses of action for healing compliance violations should be offered to selected user roles.

6 Summary and Outlook

This paper emphasized that ensuring compliance for cross-organizational processes and their changes raises several challenges. We first provided an overview about the relevant terms in this context, and introduced an example from the medical domain. Next, we introduced three correctness levels, i.e., syntactical correctness, behavioral correctness, and semantic correctness. Then we discussed the state of the art. We came to the viewpoint that the interplay of change and compliance in the context of cross-organizational processes has not been addressed yet. Finally, we provided unanswered challenges that raise from that interplay.

In our future work, we plan to address the challenges denoted in Section 5, within our research project $C^2$Pro: First, we want to enable graphically modeling of warranties and cross-organizational compliance rules. Second, we will put emphasis on enabling cross-organizational compliance checking and monitoring with respect to privacy issues and warranties. As stated, this requires the checking of syntactical and behavioral correctness before. In addition, we plan to study the efficient application of changes to running choreographies instances first and then look and instances that partially differ from the original process models. Finally, we plan to combine our approaches for ensuring compliance and correct changes to cross-organizational processes and thus enable compliance of cross-organizational changes.

References