# The Need for Compliance Verification in Collaborative Business Processes

John Paul Kasse, Lai Xu, Paul Devrieze, Yuewei Bai

# The Need for Compliance Verification in Collaborative Business Processes

John Paul Kasse[1], Lai Xu[1], Paul deVrieze[1], Yuewei Bai[2],

[1] Computing and Informatics, Faculty of Science and Technology,
Bournemouth University,
Poole BH12 5BB,
Bournemouth, United Kingdom
{jkasse, lxu, pdvrieze}@bournemouth.ac.uk
[2]Industry Engineering of Engineering College, Shanghai Polytehnic University,
Jinhai Road 2360, Pudong, Shanghai, P.R.China
ywbai@sspu.edu.cn

**Abstract.** Compliance constrains processes to adhere to rules, standards, laws and regulations. Non-compliance subjects enterprises to litigation and financial fines. Collaborative business processes cross organizational and regional borders implying that internal and cross regional regulations must be complied with. To protect customs' data, European enterprises must comply with the EU data privacy regulation (general data protection regulation - GDPR) and each member state's data protection laws. An example of non-compliance with GDPR is Facebook, it is accused for breaching subscriber trust. Compliance verification is thus essential to deploy and implement collaborative business process systems. It ensures that processes are checked for conformance to compliance requirements throughout their life cycle. In this paper we take a proactive approach aiming to discuss the need for design time preventative compliance verification as opposed to after effect runtime detective approach. We use a real-world case to show how compliance needs to be analyzed and show the benefits of applying compliance check at the process design stage.

**Keywords:** Compliance, collaborative business process, business process verification, virtual factory, compliance verification.

## 1 Introduction

Compliance is about adherence to regulations, guidelines or predefined legal requirements like norms, laws and standards. In terms of business processes, compliance relates to conformance to different process perspectives [1], [2], namely control flow, resources, data, and time. *Control flow* - strict adherence to the sequential flow of activities and their relationships, *resources* - adherence to policies for allocation and assignment of resources to perform tasks, *data* - adherence to access control and authorization, and *time* - temporal process aspects like delays. The perspectives constrain the process according to the internal organizational policies. Besides, external policies and regulations present compliance demands that must be satisfied especially

for cross organizational business processes i.e. the collaborative business processes [3]–[5], a trend of borderless business processes subject to contractual and international regulations. Moreover, partner organizations vary the core process to suit specific needs of their market or business environment resulting in process variants. Notably, the variants must stay compliant with the core process. Such scenarios justify compliance as a big and relevant topic of various applications.

Current compliance challenges and dynamics have led to new laws and regulations or revision of existing ones, e.g. the GDPR, Sabanese-Oxley Act 2002 (SOX), Base III, ITIL, ISO 2700, and Consumer Protection Act 2015 (CPA) inter alia. An organization's compliance is exhibited by its business processes conforming to the regulations. Non-compliance results in fines, litigations or loss of corporate image. Facebook is striving to rebuild public trust after breach of subscriber trust due non-compliance to data privacy [6].

Compliance provides means to monitor adherence to quality standards for products and services, consumer protection and operational transparence. Also, strict adherence to financial and accounting standards enables firms to maintain sound financial positions to avoid bankruptcy as was the case for Tyco, Global Crossing and Adelphia, Enron, HIH, Société Générale, AOL and Worldcom corporate scandals [7]. Furthermore, where process variants exist and entry to a new market is required, compliant variants can be selected easily for similar environments. For example, a collaborative business process is varied to suit laws and regulations of different countries, the most closely compliant process variant is chosen to save on time.

Compliance in business process management is complex and not automatic to achieve especially where end users are non-experts in modeling. As will be observed (section 2), support for compliance is structured for non-collaborative processes whose interaction is limited to single organizations, and targets control flow and resource perspectives. Employed techniques like process mining are curved upon the detective after-the-effect principle seeking to monitor conformance of observed behavior with modeled behavior. A knowledge gap exists to support checking for compliance of collaborative business processes with policies beyond control flow to external regulations, laws and standards. A review of modelling and verification approaches for collaborative business processes further reveals that compliance has not received deserved attention [8] [9]. The expanded scope of constraints creates complexity and conflicts necessitating their verification e.g. need to ensure that internal and external regulations map and synchronize to avoid any conflicts that can lead to deadlocks in the process. This is preferred at design time.

To that effect, we adopt concept of compliance-by-design [10] as a paradigm to achieve design time preventive compliance of the business process models with regulatory requirements. Compliance-by-design is a process of developing a software system that implements a business process in such a way that its ability to meet specific compliance requirements is ascertained [11]. To achieve compliant processes at runtime, compliance strategies are built and checked at design time. In this paper we emphasize the need for design time compliance checking through application of formal methods to reason about business processes as system models and compliance requirements as properties to automate compliance verification.

The rest of the paper is organized as follows; section 2 reviews related work and shows how this work differs, section 3 presents an industry based collaborative business

process case to support our analysis, section 4 presents a detailed analysis of the compliance requirements and need for verification. We conclude with section 5.

## 2 Related Work

Compliance, its checking and verification in business process management and workflow management has been widely addressed from different angles; compliancy to control flow aspects of the business process [1], [2], [12], resource allocation using role, task and attribute based approaches [13]–[16], security policy mechanisms [17]–[20] and compliance verification approaches [21], [21]. Categorically, compliance checking is addressed from 2 fronts i.e. at design time or runtime. Some approaches however target both design time and runtime compliance.

*Design time compliance checking* is a preventative approach that addresses compliance of business process models to constraints before execution time i.e. constraints are enforced on models and checked before execution. On contrary, *runtime compliance* checking is a detective after-the-effect approach for monitoring compliance of business processes while in execution [10], [22]. While the runtime approach is considered flexible and declarative being able to capture compliance issues beyond design; the design approach is preferred for being proactive to deal with compliance violations before they arise and permitting early time correction during process design. Following is a discussion of some compliance approaches.

PENELOPE (Process ENtailment from the ELicitation of Obligations and PErmissions) language is based on deontic logic supporting declarative expression of control flow constraints for process events. Permission and obligation constraints to perform events are explicitly expressed as temporal deontic assignments enforced on process models at design time. A compliant control flow non-executable model is generated to support process designers to verify and validate other models by showing decision points and possible violations [12], [23], [24]. The approach's application is limited to control flow and resource related compliance checking.

Relatedly, a process fragment lifecycle technique is proposed to support consistent specification, integration and monitoring of compliance controls in business processes. A process fragment is a connected graph representing part of a business process modified to incorporate compliance requirements, which are later integrated into the original business process by means of the so called 'gluing' and 'weaving' methods to create a compliant process [25]. In this approach, compliance related to control flow and data perspectives is supported. Even then, there is no way to prove lack of deadlocks or livelocks in a constrained process model i.e. no verification is supported which renders it difficult to determine correctness of integrated compliance changes.

In [22], the concept of compliance-by-design is coined to overcome limitations of the after-the-effect approaches like process mining. It provides means to reason about compliance rules by modeling control objectives and applying formal methods to enrich business process models with annotations and visualizations [10]. The concept is supported with a formalism for expressive modeling of compliance specifications i.e. the Formal Contract Language (FCL). FCL is a deontic logic and non-monotonic based

language for design time constraints specification and enforcement on BPMN business process models.

Contract Language (CL)is a also a deontic logic based language for formal automated analysis of electronic contracts. It supports detection of conflicts between service-based contracts and local contracts in SOA environments. Compliance between contract language rules and models is checked via an evaluation algorithm implemented in CLAN tool. The tool also analyses contracts for soundness and completeness [26], [27].

A compliance request language (CRL) is a design time approach for automated contractual constraint enforcement and checking on process models. It uses temporal logic for formal reasoning over formalized compliance patterns to detect violation of compliance constraints [21].

Compliancy has as well been addressed from a privacy and security angle where relevant policies are specified, enforced and checked on process models to comply with security and privacy requirements. Key in the category are the role based access control models (RBAC) [13], [20], [28]–[31] used for access control and authorization based on roles. Users are grouped into roles and permissions are assigned to groups e.g. Auditors assigned access to some resources in the process. In addition, task based access control models (Task-based Access Control) [14], [32], [33] provide a dynamic approach to enforcement of compliance to access and authorization policies based on the tasks executed in the process. Compared to RBAC, TBAC offers simplified, automated and self-admissible models where access to tasks is authorized following the context and progress of the process. On another hand, attribute-based access control models (ABAC) regulate access and authorization through a combination of attributes of both the subject (requester) and the object (e.g. file), and the environment [15], [16], [29], [34]. The proposed models under this category guide constraint specification, enforcement and monitoring to ensure compliance to policies related to resource allocation, authorization and access control for tasks, resources and data in workflow systems. Key constraints are based on requirements to express segregation of duty, binding of duty, need to know among others which prevent or detect fraud, errors of commission or omission. However, these proposals do not provide mechanisms for design time verification. Besides, their application to collaborative environments can be noticed so far.

Moreover, in [17] a framework for enforcement and monitoring of compliance to security policies in large autonomous information systems is proposed and implemented. SecBPMN is used to design process models while security policies are expressed using SecBPMN-Q after which the SecBPMN-Q are verified against SecBPMN specifications via an implemented query engine. A socio-technical security modeling language (STS-ml) is extended to support privacy by design i.e. to model privacy as a requirement and support verification of privacy properties for models through formal reasoning [18]. Little support is provided to address verification among the compliancy constraints. A compliance approach based on Petri-net semantics and syntax is proposed to check compliance on two fronts, i.e. checking rules restricting data attributes and rules restricting activities when a certain data condition holds. Process mining technique is employed to extract logs from the process execution and observe behavior. The approach is an after-the-effect theory tracing already executed

processes, this way it differs from our proactive compliance approach [1], [35]. A formalized constrained workflow involving local and global constraints, separation of duty and binding of duty constraints is proposed to enhance administration of security information in workflow systems. The rationale is to establish necessary conditions for a set of constraints that ensure a sound constrained workflow authorization schema where, for any authorized role or user, there is at least one complete workflow instance when the user can execute the role. Constraints are checked for consistence to avoid deadlocks or security lapses at runtime [33].

Table 1 summarizes the discussed approaches categorized according to how they support compliance enforcement and checking. The categories are; structural, contractual obligations and security and privacy. Other attributes in relation to formalism, application and process perspectives supported are also summarized.

**Table 1.** Summary of Compliance Methods

| Approach | Formalism | Application | Methods | Control flow | Resource | Data | Time |
|---|---|---|---|---|---|---|---|
| *Approaches based on compliance to structural behavior* | | | | | | | |
| Process Mining | Petri nets | Run time | Imperative | √ | √ | | |
| Process fragment lifecycle | - | Run time | Imperative | √ | | √ | |
| Compliance checking approach | Petri nets | Run time | Imperative | √ | | √ | |
| *Approaches based on compliance to contractual obligations behavior* | | | | | | | |
| PENELOPE | Deontic logic | Design time | Declarative | √ | | | |
| FCL | Deontic logic | Design time | Imperative | √ | √ | √ | |
| Contract Language | Deontic logic, temporal logic | Design time | Imperative | √ | | √ | |
| Compliance Request language | Temporal logic | Design time | Imperative | √ | √ | | |
| *Approaches based on compliance to security and privacy policies* | | | | | | | |
| RBAC | Temporal logic | Design time | | | | | |
| TBAC | Temporal logic | Design time | | | √ | √ | √ |
| ABAC | Temporal logic | Design time | | | √ | √ | √ |
| SecBPMN | Temporal logic | Design time Runtime | Imperative | √ | √ | | |
| STS-ml | - | Design time Runtime | Imperative | √ | √ | | |
| Formal constrained workflow | Temporal logic | Design time | Imperative | | √ | √ | |

## 3 Motivating Case Study

This section presents a description of an industry collaborative business process that serves as a motivating case study. 'Pick and Pack' is a process from a big supermarket with a chain of stores across Europe and some parts of Asia.

To create orders, customers must register via the store's online system. Upon submission of customer order, notifications are sent to both the store and the customer as confirmation. Store staff check order details, pick and pack items. Before packing items are verified by picking staff for conformity with order, and after by handover staff. One or more staff may be assigned to an order depending on its size. For items that may be out of stock, the order is put on suspense for a period until stock is availed or staff is permitted to contact customer to seek opinion either to wait, change or cancel order. Item substitution is permissible, for instance changing a fresh vegetable item to tinned one. A customer can cancel an order delayed beyond acceptable waiting time. Ready orders are either picked by the customers, delivered by store or via a preferred courier. Figure 1 is the pick and pack process model.

The case study serves as an example of different perspectives and compliance rules specific to a collaborative process e.g. *control flow*; order confirmation is subject to stock availability, *process data*; orders can only be handed over if they pass the verification check, *process resource*; final order verification must be done by different staff, *process time*; orders can be rejected or cancelled if beyond a specific time. Moreover, there are different stakeholders with differing interests that must be matched and satisfied. Customers buy items and they expect them to be of acceptable quality, non-defective, in right quantities and delivered on time. The store staff and managers work on customer orders; they are expected to meet customer expectations, item availability and timelines. Also, there are different companies in the supply chain like suppliers and couriers. In the background are shareholders whose interest is profit making. They expect financial fluency non-solvency of the company. Unverified compliance issues could lead to process flaws. E.g. packing non-ordered items, wrong item quantities, running out of stock, defect items etc. The business process accesses customer data during execution which raises data privacy concerns in terms of legality and legitimization i.e. who has access to data when and for what purpose.

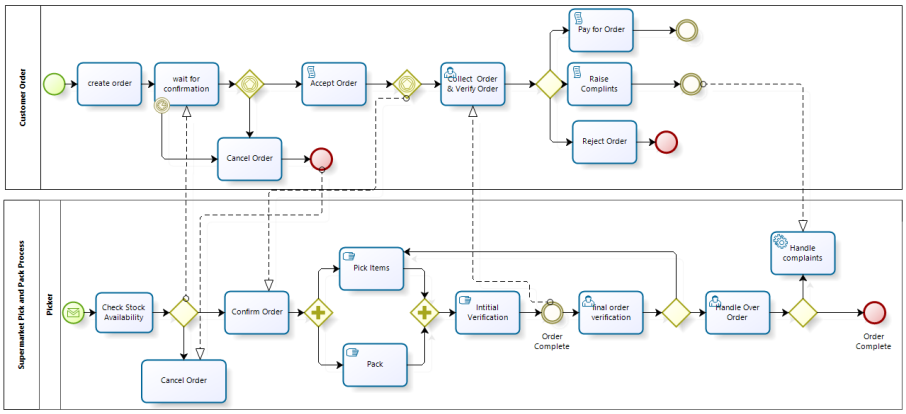


**Fig. 1.** Pick and pack business process

## 4 Need for Compliance Checking in Collaborative Processes

Traditionally, business process design is based on business objectives expressed in the terms of control flow, resources, data and time perspectives. Similarly, checks followed the same line verifying adherence the to these perspectives for single organization processes. The new regulatory requirements coupled with changing contractual obligations in business collaborations renders existing processes non-compliant to the new rules. This creates a need to formalize the regulatory requirements and verify for conformity between them as well as the existing process models. The new rules may lead to structural changes in the control flow, e.g. when tasks are removed or added, which also affects resource allocation, data access or activity time schedules for the business. (For example, Brexit once actualized will affect many business processes across Europe and the relevant regulations). Besides, new data is created that maps into existing resources and task necessitating new forms of access control and authorization. Verification for compliance is therefore needed to support conformance checks for the existing processes to avoid reinventing the wheel, wastage of resources and time to create new processes each time regulations change.

### 4.1 Compliancy with Data Privacy

As described in section 3, we use the case to illustrate the need for compliance verification and how it can be achieved. The process should comply with internal policies and external regulations like GDPR, SOX and BASE III, national fiscal policy, customer protection act. For space limitations, illustration is based on the GDPR to demonstrate the proposed process driven authorization as a mechanism to achieve access control. We however briefly describe all the regulations.

GDPR regulates data privacy where data controllers are responsible for data protection in the organization. It requires keeping data for individuals private, have their consent to collect and process it, notify them if there is any change, avail it to owners if needed in a required format and seek their consent before it can be transferred to third parties. In the case, customer data is collected and processed. When orders are delivered by delivery companies customer data is exchanged. Within Europe, different countries treat different kinds of customer data differently. Therefore, there are challenges even for specifying same business function process in different ways in different countries. SOX and Base III relate to financial standards to protect shareholders and the public from financial manipulations, intentional errors and fraudulence. The super market is required to maintain a stable financial position to the satisfaction of shareholders. Fiscal policy is a national law that differs per region. It demands openness and transparency of business processes to enable tax assessment, tracking and monitoring to prevent tax fraud. Table 3 lists extracted compliance scenarios, requirements sections of the regulations.

**Table 2.** Compliance requirements generated from the case

| Req | Use case compliance scenario | Compliance requirement | Policy level/ Regulation |
|---|---|---|---|
| Rq.1 | Customers registers on system with private data | Inform data owners which data is collected, processed and use | Data privacy GDPR |
| Rq.2 | Customer submits order(s). The system notifies customer of successful submission immediately | Notify customer of the order details submitted | Internal policy |
| Rq.3 | Notify customer when order(s) will be ready. Orders are ready between 30 and 60 minutes | Notify customer of the waiting times | Internal policy |
| Rq.4 | For delays notify customer. Customer can wait or cancel the order. | Notify customer of delays. Right to cancel order and get a full refund. | Internal policy, Consumer Rights Act 2015 |

Considering the discussed knowledge gaps in the previous section and the analysis from the case study, we illustrate compliance verification using the proposed design time approach.

## 4.2 Supporting Verification for Compliance

### 4.2.1 Compliance Verification Approach

Underpinning the approach is the need to support end users to specify and verify collaborative business process for adherence to regulatory constraints. This is achieved through the approach's components i.e. the rules modeler, rules verifier and rules enforcer.

**i. Compliance Modeler**

Compliance rule modeler supports the extraction of requirements from their sources and translates them into constraints based on defined compliance attributes (fig. 2). Some are adopted and adapted from [21], [36], [37] while other are proposed. For automated application, the attributes are formalized to achieve formal semantics based on temporal logic languages.

**ii. Compliance Rules Verifier**

To our knowledge none of the existing framework supports this capability. It is intended to ensure coherent, accurate, complete and consistent constraints. Conflicts between constraints are likely to exist and thus it is necessary to verify them before enforcement. For example, internal policies may conflict with external regulations. If unchecked, conflicts may create deadlocks or live-locks in the process. Consistency is required between; internal policies and collaboration contractual policies and between internal policies and national regulatory policies. Internal policies are translated into requirements i.e. properties to be satisfied while the external policies into system

models using temporal logic, then apply formal reasoning and model checking techniques to support automatic verification amongst them. The intention is to derive an ideal state where both internally and externally derived constraints can be used to constrain a business process without inbound conflicts, ambiguities and inconsistencies. Some of the targeted error checks relate to resource authorization and access control that would otherwise be a source of flaws and insecurity in the business process; for instance Privilege leakage, locking and conflict [33]. Verification will be achieved by integrating with existing model checkers. Specifically, NuSMv [38] a version of the traditional SMV [39] model checker is preferable for its expressive power in checking models for satisfiability to constraints.

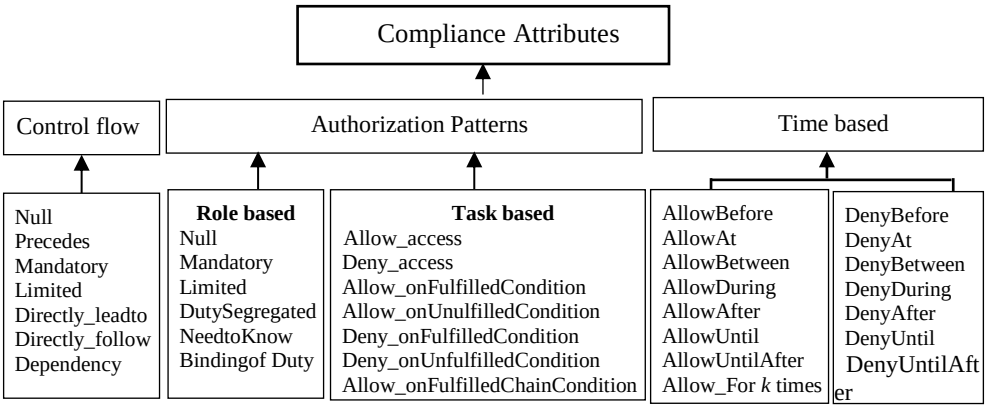


**Fig 2.** Compliancy attributes meta-model

**iii. The Attribute Enforcer**

Verified compliance constraints are enforced on the business process activities constraining them according to requirements. For instance, to achieve privacy, access to data is controlled and authorized based on its need to accomplish a time bound activity in the business process i.e. access is legitimized. In such scenario, during runtime the task will invoke the authorization API seeking access to a specific data item. The authorization engine will then check its access policy repository built according to the access control policy. Based on the request outcome, the task will progress, halt, terminate or be skipped for the business process to progress.

### 4.2.2 Application to the Case Study

In this section we briefly show the application of the compliance attributes that will compose the implementation for the compliancy verification mechanism using the stated compliancy scenarios and requirements. All scenarios and requirements are related to Table 3 in section 4.2.

*Scenario 1:* Customers register on system with private data.

*Requirement*: Inform data owners what data is collected, processed and intended use.

*Regulation*: Internal and external regulations on data management and privacy apply. Relevant regulations are; GDPR - data privacy. National regulation – data privacy. Industry specific best practice on data management principles.
*Enforcement and verification*: a combination of attributes is applied. In terms of control flow, the initial activity is not *preceded* by any other activity, proceeding to the next step in the process can only be allowed upon fulfillment of a condition i.e. upon successful system registration (*Allow_onFulfilledconditon*). If the condition is not fulfilled, access is denied until fulfilled (*Deny access, DenyUntil*).
*Scenario 2:* The customer submits order(s).
*Requirement:* Notify customer of (un)successful submission immediately by SMS or email about the order details submitted.

*Regulation*: Internal policy. In terms of control flow the next task is determined. Regarding resources, access to customer data, at what level in the process and who can communicate with the customer must be authorized. There is also constraint on structure and content of the communicated message

*Enforcement and verification:* Before communication, the preceding activity must have succeeded (*AllowAfter*) and requirement 1 (Rq.1 in Table 3) fulfilled. The task can execute and access (*Allow_access*) customer data (email address or contact number) to initiate the communication. Otherwise access is denied if orders are not submitted. But communication can still happen for incomplete order to establish why never completed purchase (*Allow_onUnulfilledCondition*). This can provide valuable feedback.

*Scenario 3*. Notify customers when orders will be ready.

*Requirement:* Confirm order and possible pick up/ delivery times

*Regulation:* Internal policy. Orders are ready six (6) hours from the order submission time. Customer initiated interactive communication allowed with special category staff e.g. sales support staff only if changes must be made to the order.

*Enforcement and verification:* This requirement is *mandatory* and allowed access to customer data (*Allow_access*) for immediate automated communication (*NeedtoKnow*) to the customer at point when the order is submitted (*AllowAt*). Interactive communication is restricted to specific staff (*Limited*). Otherwise if the condition of successful order submission is not fulfilled access is denied (Deny access) at that moment in time (*DenyAt*).

For automated application especially for the non-expert end-users as illustrated in the preceding section, a declarative approach will be adopted for implementation where all combinations of the attributes and their executions or behavior are implicitly permissible except where explicitly forbidden i.e. by stating what is non-permissible.

## 5 Conclusion

Compliance is a major concern today regardless of the industrial sector to keep pace with changing regulations besides the rising concerns of security, product and service quality and data privacy. With EU revising its GDPR set to commence by May 2018; concerned organizations are working towards meeting its requirements before deadline

by realigning their business processes. To support them in due course is a necessary step. In doing so, other than the detective after-the-effect compliance checking, a proactive preventive approach is preferred to identify and combat compliancy violations before they take place to avoid the costs of fines or litigations. The effort of this research is geared towards a comprehensive approach for modeling, verification and enforcement of compliance constraints on collaborative business processes with an end user perspective.

**Acknowledgements:** This research has been partially sponsored by EU H2020 FIRST project, Grant No. 734599, FIRST: vF Interoperation suppoRting buSiness innovaTion.

## References

[1] T. E. Ramezani, V. Gromov, D. Fahland, and W. M. P. van der Aalst, "Compliance Checking of Data-Aware and Resource-Aware Compliance Requirements," *Move to Meaningful Internet Syst. OTM 2014 Conf. OTM 2014. Lect. Notes Comput. Sci. vol 8841*, no. 2, pp. 237–257, 2014.

[2] D. Borrego and I. Barba, "Conformance checking and diagnosis for declarative business process models in data-aware scenarios," *Expert Syst. Appl.*, vol. 41, no. 11, pp. 5340–5352, 2014.

[3] J. Ziemann and T. Matheis, "Modelling of cross-organizational business processes-current methods and standards," *Proc. EMISA '07*, vol. 2, no. 2, pp. 87–100, 2007.

[4] P. R. Telang and M. P. Singh, "Specifying and verifying cross-organizational business models: An agent-oriented approach," *IEEE Trans. Serv. Comput.*, vol. 5, no. 3, pp. 305–318, 2012.

[5] K. A. Schulz and M. E. Oklowska, "Facilitating cross-organisational workflows with a workflow view approach," *Data Knowl. Eng.*, vol. 51, no. 1, pp. 109–147, 2004.

[6] J. Guynn, "Facebook CEO Mark Zuckerberg finally speaks on Cambridge Analytica: We need to fix 'breach of trust,'" *Tech*, 2018. [Online]. Available: https://www.usatoday.com/story/tech/2018/03/21/facebook-ceo-mark-zuckerberg-finally-speaks-cambridge-analytica-we-need-fix-breach-trust/445791002/. [Accessed: 12-Apr-2018].

[7] C. Johnson, "Enron ' s Ethical Collapse : Lessons for Leadership Educators," *J. Leadersh. Educ.*, vol. 2, no. 1, pp. 45–56, 2003.

[8] J. P. Kasse, L. Xu, and P. de Vrieze, *A comparative assessment of collaborative business process verification approaches*, vol. 506. 2017.

[9] J. P. Kasse, J. Nabukenya, Towards adoption of business process analysis and design techniques in transitional countries : design and validation. 2, 248–256 (2012).

[10] S. Sadiq and G. Governatori, "Managing Regulatory Compliance in Business Processes," *Handb. Bus. Process Manag. 2*, vol. 2008, pp. 159–175, 2010.

[11] M. Kochanowski, C. Fehling, F. Koetter, F. Leymann, and A. Weisbecker, "Compliance in BPM today - an insight into experts ' views and industry challenges," *Inform. 2014. Big Data - Komplexität meistern.*, pp. 769–780, 2014.

[12] S. Goedertier and J. Vanthienen, "Designing Compliant Business Processes with Obligations and Permissions," *BPM 2006 Int. Work. BPD, BPI, ENEI, GPWW, DPM, Semant. Vienna, Austria, Sept. 4-7, 2006.*, pp. 5–14, 2006.

[13] P. R. Sandhu, "The RBAC96 Model," 2003.

[14] R. K. Thomas and R. S. Sandhu, "Task-based Authorization Controls ( TBAC ): A Family of Models for Active and Enterprise-oriented Authorization Management," *Database Secur.*, vol. 11, pp. 166–181, 1997.

[15] E. Yuan and J. Tong, "Attributed Based Access Control (ABAC) for Web Services," in *The IEEE International Conference on Web Services*, 2005, pp. 561–569.

[16] M. Gautam, "Poster : Constrained Policy Mining in Attribute Based Access Control," pp. 121–123, 2017.

[17] M. Salnitri, F. Dalpiaz, and P. Giorgini, "Modeling and verifying security policies in business processes," in *Lecture Notes in Business Information Processing*, 2014, vol. 175 LNBIP, pp. 200–214.

[18] M. Robol, M. Salnitri, and P. Giorgini, "Toward GDPR-compliant socio-technical systems: Modeling language and reasoning framework," *Lect. Notes Bus. Inf. Process.*, vol. 305, pp. 236–250, 2017.
[19] J. Müller, "Security Mechanisms for Workflows in Service-Oriented Architectures," 2015.
[20] C. Combi, L. Viganò, and M. Zavatteri, "Security Constraints in Temporal Role-Based," *Codaspy*, pp. 207–218, 2016.
[21] A. Elgammal, O. Turetken, W. J. van den Heuvel, and M. Papazoglou, "Formalizing and appling compliance patterns for business process compliance," *Softw. Syst. Model.*, vol. 15, no. 1, pp. 119–146, 2016.
[22] S. Sadiq, G. Governatori, and K. Namiri, "Modeling Control Objectives for Business Process Compliance," *5th Int. Conf. BPM 2007, Brisbane, Aust. Sept. 24-28, 2007.*, pp. 149–164, 2007.
[23] S. Goedertier, "Declarative Techniques for Modeling and Mining Business Processes.," no. 284, p. 248, 2008.
[24] Goedertier and Vanthienen, "Compliant and Flexible Business Processes with\nBusiness Rules," *Bpmds*, no. January, pp. 94–103, 2007.
[25] D. Schumm, F. Leymann, Z. Ma, T. Scheibler, and S. Strauch, "Integrating Compliance into Business Processes Process Fragments as Reusable Compliance Controls," pp. 2125–2137, 2010.
[26] S. Fenech, G. J. Pace, and G. Schneider, "Automatic conflict detection on contracts," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2009, vol. 5684 LNCS, pp. 200–214.
[27] S. Fenech, G. J. Pace, and G. Schneider, "CLAN: A tool for contract analysis and conflict discovery," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 5799 LNCS, no. October, pp. 90–96, 2009.
[28] A. M. Ertugrul and O. Demirors, "An exploratory study on role-based collaborative business process modeling approaches," in *Proceedings of the 7th International Conference on Subject-Oriented Business Process Management - S-BPM ONE '15*, 2015, pp. 1–5.
[29] A. R. Khan, "Access control in cloud computing environment," *ARPN J. Eng. Appl. Sci.*, vol. 7, no. 5, pp. 613–615, 2012.
[30] R. Sandhu, "Rationale for the RBAC96 family of access control models," *Proc. first ACM Work. Role-based access Control - RBAC '95*, no. 1, p. 9–es, 1996.
[31] A. Alshehri and R. Sandhu, "Access Control Models for Virtual Object Communication in Cloud-Enabled IoT," in *2017 IEEE International Conference on Information Reuse and Integration (IRI)*, 2017.
[32] M. Wu, "Role and Task Based Authorization Management for Process-View," *Proc. Second Int. Conf. Secur. Cryptogr.*, no. 707, pp. 85–90, 2007.
[33] K. Tan, J. Crampton, and C. A. Gunter, "The Consistency of Task-Based Authorization Constraints in Workflow Systems," *Proc. 17th IEEE Comput. Secur. Found. Work.*, pp. 155–169, 2004.
[34] Axiomatics, "Attribute Based Access Control (ABAC)," 2018. [Online]. Available: https://www.axiomatics.com/attribute-based-access-control/. [Accessed: 09-Apr-2018].
[35] E. Ramezani, D. Fahland, and W. M. P. van der Aalst, "Diagnostic Information in Temporal Compliance Checking," *ech. rep., BPM Cent. Rep.*, no. 2, 2012.
[36] E. Gammal, "Towards a comprehensive framework for business process compliance FRAMEWORK FOR BUSINESS PROCESS," 2014.
[37] N. Hall, M. B. Dwyer, G. S. Avrunin, and J. C. Corbett, "Property Speci cation Patterns for Finite-State Veri cation 1 INTRODUCTION 2 DESIGN AND OTHER PATTERNS," *n Proc. Second Work. Form. methods Softw. Pract.*, vol. 2, pp. 7–15, 1998.
[38] A. Cimatti, E. Clarke, and E. Giunchiglia, "Nusmv 2: An opensource tool for symbolic model checking," *Comput. Aided Verif.*, vol. 2404, pp. 359–364, 2002.
[39] A. Biere, A. Cimatti, E. Clarke, and Y. Zhu, "Symbolic Model Checking without BDDs ?," *Tools Algorithms Constr. Anal. Syst.*, no. 97, pp. 193–207, 1999.