Knowl Inf Syst (2018) 57:79–133
https://doi.org/10.1007/s10115-017-1142-1

REGULAR PAPER

Are we done with business process compliance: state of the art and challenges ahead

Mustafa Hashmi[1] · **Guido Governatori**[1] · **Ho-Pun Lam**[1] · **Moe Thandar Wynn**[2]

Received: 9 August 2016 / Revised: 2 October 2017 / Accepted: 27 December 2017 /
Published online: 22 January 2018

Abstract Literature on business process compliance (BPC) has predominantly focused on the alignment of the regulatory rules with the design, verification and validation of business processes. Previously, surveys on BPC have been conducted with specific context in mind; however, the literature on BPC management research is largely sparse and does not accumulate a detailed understanding on existing literature and related issues faced by the domain. This survey provides a holistic view of the literature on existing BPC management approaches and categorises them based on different compliance management strategies in the context of formulated research questions. A systematic literature approach is used where search terms pertaining keywords were used to identify literature related to the research questions from scholarly databases. From initially 183 papers, we selected 79 papers related to the themes of this survey published between 2000 and 2015. The survey results reveal that mostly compliance management approaches centre around three distinct categories, namely design-time (28%), run-time (32%) and auditing (10%). Also, organisational and internal control-based compliance management frameworks (21%) and hybrid approaches make (9%) of the surveyed approaches. Furthermore, open research challenges and gaps are identified and discussed with respect to the compliance problem.

Keywords Business processes · Business process compliance · Norms compliance · Normative requirements · Compliance management frameworks

✉ Mustafa Hashmi
mustafa.hashmi@data61.csiro.au

Guido Governatori
guido.governatori@data61.csiro.au

Ho-Pun Lam
brian.lam@data61.csiro.au

Moe Thandar Wynn
m.wynn@qut.edu.au

[1] Data61, CSIRO, 41 Boggo Road, Dutton Park, QLD 4102, Australia

[2] Queensland University of Technology (QUT), 2-George Street, Brisbane, QLD 4000, Australia

1 Introduction

Regulatory compliance aims to ensure that organisation's business operations are in alignment with the governing laws of the organisation or the laws from regulatory bodies. The requirement for *being compliant* has increased over the last two decades due to big corporate scandals such Enron ($74 bn), WorldCom ($180 bn), American International Group ($3.9 bn), Bernnie Madoff ($65 bn), Lehmann Brothers ($50 bn), Petrobras ($2 bn) in the Americas and Parmalat ($11.54 bn) in Australia and Société Générale (€4.9 bn), UBS (€2.3 bn) in Europe, which caused severe depressions in the world's financial markets. Fong and Grillo [56] observed such depressions on the financial markets caused by the bad governance, corruption, bribery, unreliable and false information, asset misappropriation, non-compliance to regulatory laws resulted in $5 trillion in losses to organisations. A survey by KPMG Australia [114] reveals that from 1997 to 2012 the losses due to frauds suffered by the Australian companies have tripled from $105 million to $373 million over that period.

Such huge losses—and in some cases the closures of large companies—resulted in the need to design and implement new regulatory laws to control how businesses should conduct their operations in futures. Thus, several laws such as Sarbanes–Oxley (SOX) Act [184]; BASEL (series of) Acts [21]; Health Insurance Portability and Accountability Act [90]—and Anti-Money Laundering regulations and monetary de facto standards (such as the International Financial Reporting Standard [92]) have emerged. These had a direct impact on the organisations' operations. Failure to comply with these regulatory laws can damage investors' confidence, and result in financial penalties or even criminal prosecution. Hence, adherence to internal controls and regulatory laws, and other sources of compliance has become a *must-to-do* activity for every organisation in the interest of transparency and more efficient operations of their businesses [1].

In today's highly process-oriented business environment, business processes are core of any organisation. Business processes provide an abstract view of the state of the affairs on *how they achieve* business objectives and implement the policies governing their business operations. Organisations may employ different strategies to be compliant [77]: the compliance may be assessed: (i) at *design-time*, that is, at the very early stages of process design, or (ii) at *run-time*, that is, when the processes are running or (iii) *ex post* with audits examining logs of processes.

In response to ever-changing regulations, fear of financial or criminal penalties, and to support these strategies, business process compliance received unprecedented attention from industry and academia alike. As a result of this wider interest, a large body of compliance management frameworks (CMFs), approaches and techniques addressing the compliance problem from a variety of perspectives have been reported in literature over the last decade. However, the literature on business process compliance is largely sparse and does not accumulate a detailed understanding of the domain. In particular, how enterprises are achieving compliance using which compliance management strategies, and which regulatory compliance management approaches are available in literature. A few surveys notably [1,23,106] have been carried out in the past with specialised context in mind, but—to our best knowledge, none provides a holistic view of the state of the affairs of the compliance domain. Also, current research in business process compliance shows a downward trend as the number of approaches is continuing to decrease compared to the number of approaches proposed in the past. This gives rise to the question that should the downward trend in the compliance research be translated, whether we are done with business process compliance and existing approaches are able to handle every aspect of the compliance problem?

Motivated by the observation of downward trend, and to accumulate a detailed understanding on the current research in business process compliance, this survey presents the available works on business process compliance management categorised along compliance management strategies. Also, it highlights shortcomings of the existing approaches, discusses research gaps and challenges faced by the domain, and identifies the areas of improvements or new development to set the agenda for future work.

1.1 Context: business process compliance management

The term *compliance*, in its literal meanings, is the "*ability of an object to yield elastically when a (preferably external) force is applied*" [194]. In other words, given the presence of an external force, the object has to respond flexibly without repelling the force being applied. From a business process compliance, McIntyre [132] defines compliance as:

> A desired outcome, with regard to law and regulations, internal policies and procedures, and commitment to stakeholders that can be consistently achieved through managed investment of time and resources. The compliance management includes the legal and tactical activities in day-to-day business processes.

In contrast, the authors in [70] view compliance as:

> an act or process to ensure that business operations, processes, and practices are in accordance with prescriptive (often legal) documents.

From the above definitions, it is clear that the term *compliance* connects two distinct domains: the *legal domain* and the *business process domain* as illustrated in Fig. 1. The two domains differ from each other in their specificities and their objectives. Essentially, the legal domain (that is, regulatory domain) is *prescriptive* in nature; it ascribes conditions that details which actions can be considered legitimate, and which actions must be refrained while executing a business process. In contrast, the business process domain is more *descriptive* detailing *how* business processes are executed to carry out business objectives. Compliance aims to gain more understanding on *how* enterprises *should* operate in a more sustainable way to continue providing their services without violating the applicable regulations that can significantly effect their business operations [149].

While the business process domain and the legal domain differ in many aspects, there is the possibility of colliding synergies between the two domains and there can be conflicts and inconsistencies. Thus, a careful study of the inter-dependencies of the domains is required [167]. For this reason, the compliance domain has received unprecedented attention from industry and academia alike. This attention is also motivated by recent regulations such as Sarbanes–Oxley (SOX) Act [184], which requires the establishment of stronger and more enforceable strategies to meet the enterprise's compliance reporting requirements. According to O'Neill [152], enterprises initiate compliance-related activities due to a number of factors that affect enterprises processes: (i) imposition of external rules and regulations, (ii) decision to adhere and define its own internal policies and (iii) manage the regulatory processes within the enterprise to fulfil the social requirements. However, the identification of the relevant regulations may cause frustration when regulations are ambiguous and require a great deal of efforts to be understood. Thus, enterprises pay less attention to compliance, even when regulatory bodies put pressure on them to comply with stringent regulations and recommend severe penalties—or even criminal prosecutions for non-compliance [1,155]. To avoid these problems with the regulatory bodies, enterprises are putting more efforts into

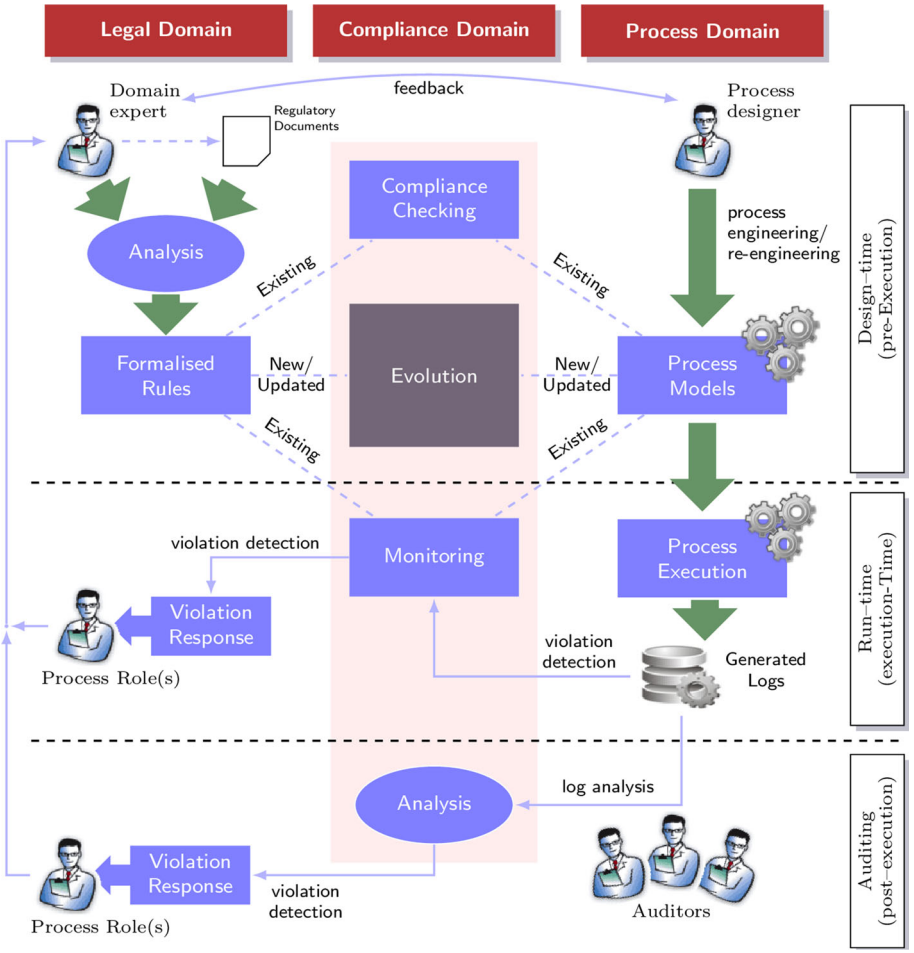

Fig. 1 Compliance management strategies

the compliance-related activities and employ a number of compliance reporting strategies, namely *design-time, run-time* and *auditing* as depicted in Fig. 1.

- **Design-time** (*otherwise, pre-execution time*) is a preventive compliance management strategy where business processes are assessed for any non-compliant patterns at the very early stages of the process design. As such, in this approach, the compliance requirements are captured through a logic-based requirements modelling framework and propagated into business processes. Any non-compliant issues can be detected in the very early stages, thus saving an enterprise's efforts, time and financial resources.
- **Run-time** (*otherwise, execution-time*) compliance checking is a strategy by which enterprises use specialised software products that produce compliance reports, while the processes are being executed. This approach has a limited scope because it still falls in the *after-the-fact category* [77]. Also, it requires human intervention to rectify the detected problems.

– **Auditing** (*otherwise, post-execution*) is a strategy by which specialised compliance consultants manually analyse the logs generated by the processes to detect possible violations. The main drawback of this strategy is the use of manual checks, which requires a great deal of time and resources, and the use of manual checks is thus a costly venture. The increased pressure and threat of possible criminal prosecutions, however, make the auditing method a less attractive compliance reporting strategy.

Each of these strategies might have limitations and yet can be applied in the enterprises where many processes are designed without any knowledge of the applicable regulations opening, thus, the possibilities of violations [167].

1.2 Scope of the survey

The objective of this survey is to deliver a detailed review of current business process compliance approaches at different stages of business processes, from modelling, detection to violation handling and reporting. We consider papers covering topics related to: (i) business process compliance from an organisational perspective, (ii) *design-time* compliance checking, (iii) *design-time* compliance verification, (iv) *run-time* compliance monitoring and detection and (v) *post-execution* time auditing. However, we exclude from the survey those papers that do not exhibit a strong link with the compliance management, or those that interpret compliance in a very generic sense. We also exclude papers that only focus on business process management, and those that merely related to business process dimensions such as control flow, data, resources and time. Although business process management (BPM) has strong link with regulatory compliance, we do not consider the papers that focus on the BPM methodologies related to process design and improvements, process mining techniques and process analytics. Since BPM is a mature domain, we believe that its literature merits to be reviewed separately from the one we present in this paper.

Note that regulatory compliance is generally attributed as a risk factor for enterprises—thus, it is widely linked to risk management. However, regulatory compliance is different from risk management as regulatory compliance aims to address and provide solutions to the alignment between business process and regulations. In contrast, the goal of risk management is to understand, prevent, detect and mitigate various types risks in business processes. Given this separation, we do not consider papers on risk-aware process management, and we would like to point reader's attention to the survey by [179] where a detailed understanding on the risk-aware process management has been accumulated.

1.3 Contribution of the survey

The main contribution of this paper is twofold: *firstly*, it classifies the existing literature on compliance based on the various compliance management strategies (cf. Sect. 1.1 using various techniques or methods to address the compliance problem) and provides a rich discussion on various aspects of the compliance management. *Secondly*, the presented survey identifies the current and future challenges faced by the compliance domain, and it sets the stage for future research agenda. As per Cooper's taxonomy of literature reviews [38], this survey concentrates on how existing literature tackles the compliance problem with the aim to specifically look at the research outcomes of existing compliance frameworks, identify the main issues and highlight the voids through a methodological comparison of the compliance frameworks that should lead the future research agenda in the business process compliance domain.

Outline: The rest of the paper is structured as follows: next (in Sect. 2) we briefly introduce the research methodology of the survey. Sections 3–7 are dedicated to the evaluation of the state of the art. We review in Sect. 3 the compliance frameworks from an organisational perspective. On this basis, we discuss design-time, run-time and auditing compliance approaches in Sects. 4, 5 and 6, respectively. The features of hybrid approaches are discussed in Sect. 7. We highlight the research gaps and future challenges in Sect. 8. Comparison with other surveys, key findings, potential impact and limitations of the presented survey are discussed in Sect. 9 before concluding this paper in Sect. 10.

2 Research methodology

Literature on business process compliance has predominantly focused on the alignment of regulatory rules with business processes, modelling, verification and validation of business processes against compliance rules. Previous studies have been conducted mostly with a specialised context in mind. For example, some studies merely focused on the monitoring frameworks, while others have looked at design-time compliance approaches and some from the organisational perspective. However, the literature of business process compliance is mostly sparse as none of the above-discussed survey accumulates a detailed understanding on the state of the art whether existing compliance approaches can cover the full spectrum of business process compliance. Following the structured guidelines from [107,108], this survey presents the available works on business process compliance management, categorised along compliance management strategies, highlights existing compliance approaches, discusses challenges faced by the domain and sets the agenda for future work.

2.1 Survey questions

The objective of this survey is to accumulate understanding on the current state of the affairs in the compliance domain, summarise the weaknesses of existing techniques and to identify the areas of further work. To achieve this, we have formulated the following questions:

RQ-1 *which are the dimensions of regulatory compliance problem?*
RQ-2 *which are generic strategies for addressing the compliance problem?*
RQ-3 *whether existing approaches fully cover dimensions of the compliance problem?*
RQ-4 *what are the challenges that need to be addressed?*

These questions which establish the requirements for this survey are closely linked and they are simultaneously investigated.

2.2 Literature search

The literature collection process started by querying prominent scholarly databases, e.g. Google Scholar, Science Direct, Scoupus, Web of Science, with the keywords related to the scope of the survey. For this purpose, we first identified a list of subject terms, concepts and keywords, and alternative spellings pertaining research questions in the compliance domain. These terms were then combined using boolean operators to construct the key search terms such as

- compliance AND ("*management*" AND "*frameworks*"),
- compliance AND ("*formal*" AND "*methods*"),
- compliance OR ("*techniques*" OR "*methods*" OR "*approaches*")

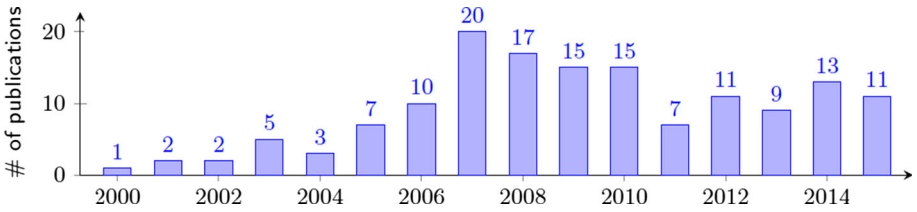

Fig. 2 Distribution of publications per year (2000–2015)

The list of key search terms extracted about the information domain as follows:

(a) *compliance frameworks*: compliance management frameworks, business process compliance management, formal compliance
(b) *compliance strategies*: design-time compliance, run-time compliance, auditing compliance, compliance by design.
(c) *process life cycle aspects*: data-aware, resource-aware compliance, control-flow verification
(d) *compliance approaches*: semantics and ontology compliance management, pattern-based, graph, policy-driven compliance

While constructing the list of key search terms, we noted that there were several subject terms/concepts that are used in literature quasi-synonymously to represent the same concepts such as "conformance" is used for "compliance", "backward compliance" for "auditing" and "retrospective compliance" for "design-time compliance". We also included those terms in the queries.

The keyword queries from the scholarly databases resulted in more than 100 hits each time. We applied various filtering strategies to remove the results that were not relevant to the scope of this paper—in particular, we used proximity operators and lemmatisation to narrow down the search results. In addition, we also collected papers from academic forums such as journals, conferences and workshops using SpringerLink,[1] ScienceDirect,[2] ACM Digital Library,[3] Web of Science,[4] EbscoHost,[5] IEEEXplore[6] and free search database DBLP[7] to name but a few. We only considered premium conferences and journals to ensure the quality and reliability of the collected sources. We applied the same systematic search strategy and collected papers published between 2000 and 2015 (last searched December 2015) as shown in Fig. 2.

During our searches, we came across a number of existing literature studies in business process compliance management [1,46], and compliance modelling languages [51], which provided a rich inventory of publications from both design-time and run-time compliance perspective.

We ran a backward verification search on the publications inventory, and our search validated that most of the publications from these studies were already found in our search. All duplicates were immediately removed. The backward search also revealed that there were a

[1] SpringerLink http://www.springerlink.com.

[2] ScienceDirect http://www.sciencedirect.com/.

[3] ACM Digital Library http://dl.acm.org.

[4] Web of Science http://www.webofscience.com/.

[5] EbscoHost https://www.ebscohost.com/.

[6] IEEEXplore http://ieeexplore.ieee.org/.

[7] Free Search Database DBLP http://dblp.uni-trier.de/.

Table 1 Study selection and quality assessment criteria

Selection criteria	
1.	Papers published between 2000 and 2015
2.	Papers that fall in business process compliance domain
3.	Papers that are able to answer the research questions
4.	Papers with more than five citations
5.	Papers that are published in English language
Rejection criteria	
6.	Papers not published between 2000 and 2015
7.	Papers that have no link or do not fall in business process compliance domain
8.	Any duplicate papers, industrial bulletins, industry case studies
9.	Papers with less than five citations
10.	Papers that have no bibliographical information, such as paper with no date/year information
11.	Papers published in any language other than English
Quality assessment criteria	
12.	If the study objectives are clearly stated
13.	If proposed method/technique is clearly described
14.	If the methodology used in the study is adequate
15.	If the study has a high citation count

number of papers that were not included these studies. Our literature search resulted in more than 2000 papers.

Next, the collected papers were then checked for their relevance to the survey by checking their titles, if they fully (or closely) related to the topics of business process compliance, design-time compliance, run-time compliance, etc. For each paper that seemed relevant, the abstract was read and its contents were quickly scanned to determine its relevance. If the paper passed these preliminary checks, it was included into the pool of our literature. Those papers whose title was not relevant were immediately removed from the analysis. With this process, 183 potentially relevant papers were selected for further analysis and synthesis from the pool of more than 2000 papers.

2.3 Evidence assessment

Once the search process was completed, evidence assessment for each selected paper following the guidelines from [107,108] was carried out. The aim of the evidence assessment was to ensure the quality, credibility, completeness and relevance of the selected papers and to remove any disagreement(s) on the relevance of the contents to objectives of the survey. The evidence assessment was carried out in two phases. In the *first* phase, relevance of the collected papers was scrutinised, and then, the quality of the papers was assessed in the *second* phase using the criteria illustrated in Table 1. We selected papers that fall in the business process compliance domain, able to answer the research questions, and only published in English language[8] between 2000 and 2015.

[8] It might be possible that there are papers on compliance management written in other languages such as German and French, and we exclude such papers from this survey, see Sect. 9.3.

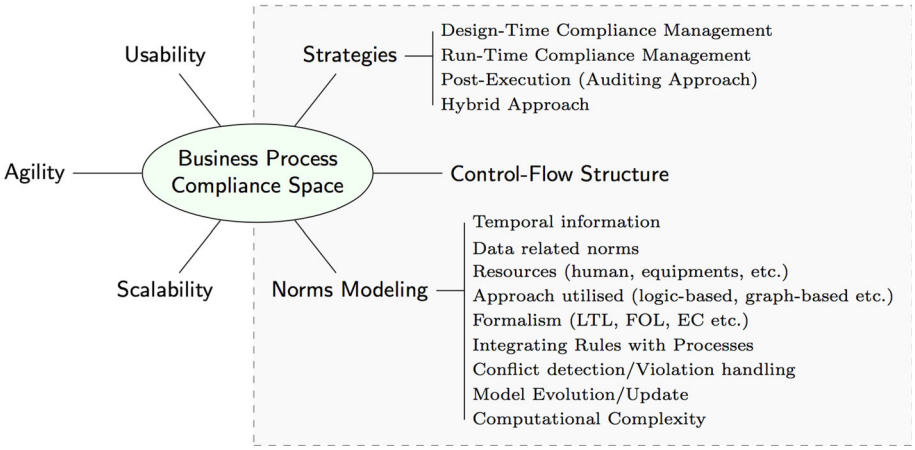

Fig. 3 Dimensions of the compliance problem

The quality of a paper was assessed on whether the paper provides a compliance framework from an organisation perspective or whether the paper reports a compliance approach oriented to design-time, run-time or post-execution strategy or a mix of these strategies. We also considered whether the selected study achieved its objectives and employed an adequate methodology. The differences between the collected papers were identified, and an explanation of the differences was also recorded to ensure the quality of inferences and interpretations of the findings in each paper.

As citation indexing allows to ascertain the scientific relevance and impact of the scholarly contributions [88,133], we also considered citation counts to further assess the quality of the selected papers. For this purpose, we used citation counts from Google Scholar[9] which covers a range of scholarly databases, repositories, online sources and document types for citation indexing. The papers with less than five citations were removed from the list. The evidence assessment phase resulted in a more refined list of 79 papers that were relevant to the research questions, satisfying the selection and quality assessment criteria, and were consequently selected for evaluation in this survey.

2.4 Evaluation criteria

The approaches in this survey are evaluated against a number of criteria derived from both *business processes* and *legal norms* sides of compliance as illustrated in Fig. 3. These criteria have been derived while keeping simplicity in mind and refer to the requirements that a compliance approach should be able to provide support for. Essentially, these requirements aim to extend a compliance product addressing the compliance problem beyond Yes/No type answers. Also, these criteria can be used to specify various features of a compliance approach—thus, it can be used to evaluate the suitability of a compliance framework for compliance reporting.

[9] There are other sources of citation measurements and academic search engines, e.g. Microsoft Academic Search, Scopus, Semantic Scholars, see Wouters and Costas [195] for detailed listing.

2.4.1 Compliance strategy

This criterion refers to various strategies that can be used to verify the compliance at different phases, namely design-time, run-time or post-execution (see Sect. 1.1 for details). Each of these strategies has its benefits and shortcomings. Under this criterion, we evaluate and classify the approaches based on the compliance checking strategy employed in the selected study.

2.4.2 Control-flow structure

The control-flow aspect of business process specifies the order of the activities of the business process, and other structural information (such as events, connectors, triggers for subprocesses, using specific data types and the data flow conforms to a specific schema [164]). The compliance rules might be concerned with some process activities to be executed in a pre-defined order. Business processes, in turn, may require further information that can be extracted from regulations. Control-flow- based compliance rules can be verified at design-time, and in this case, it is highly unlikely that they can be violated.

2.4.3 Temporal information

Many of the norms in the area of business process compliance are concerned with the temporal aspect of the norms. For example, a rule may specify that an invoice must be paid within 7 days from the order date, while other may specify conditions that must be fulfilled by a deadline or persist over time [83,85]. Hence, when a business process is subject to norms, it is particularly important that this process complies with the norms for the whole duration of the validity of norms, or meets the deadlines, or follows the constraints for maintaining or delaying the actions. The temporal aspect of a business process is concerned that a particular task in a process is completed within t unit of time [13]. This criterion evaluates whether existing approaches deal with the temporal information.

2.4.4 Data related norms

The execution of business process tasks may involve managing a large amount of data; for example, information stored in databases may change, new data may be produced, and tasks may need specific data to complete. This information can flow along process in the form of data objects—such as a form. A rule concerning data management can be defined, so data objects must be also represented in the model and could be subject to compliance checking.

2.4.5 Resource related norms

Some norms might be concerned with the resource aspect that *who will execute the task*? For example, a norm may specify that a specific task should be executed by a specific person or by two separate persons, while other norms might enforce a layered hierarchy to perform a specific operation. This criterion evaluates how approaches provide support for compliance checking of norms related to resources.

2.4.6 Approach utilised

This criterion evaluates whether the approach proposed in a selected study is either logic-based, pattern/graph-based, or query-based or process mining-based compliance checking, or whether it uses a mix of these methods.

2.4.7 Formalism

Compliance is about legal norms written in natural language. One of the major requirements for a compliance framework is that it is able to provide support for all types of norms. However, this ability largely depends on the expressiveness of the chosen formalism to accurately capture the norms expressed in natural language. The lack of full modelling support for any type of norm can severely impede the reliability of the compliance approach. This criterion evaluates whether the formalism used in the framework is expressive enough to provide full reasoning support for all types of norms for compliance checking.

2.4.8 Integrating rules with processes

One of the desired features of a compliance approach is that it provides an automated solution to decide which rules are applicable to a process. Attaching the rules automatically can help modellers to implement changes in the processes whenever the business rules are changed [13]. This criterion examines *how existing compliance approaches* integrate compliance rules with business processes.

2.4.9 Conflict resolution and violation handling

Compliance requirements can come from a variety of sources and can be differently interpreted by the domain experts. This could lead to inconsistencies, and implementing inconsistent compliance requirements would lead to wrong results. Inconsistencies might cause situations where business processes might not be able to fulfil rules, thus increasing the risk of rules violations. On the same note, rules can be violated for a variety of reasons, e.g. a data item required for performing an activity is not present or some activity is not executed. It is highly desirable that a compliance approach is not only able to timely detect the violations but also it is able to provide information on possible causes that the modellers can react quickly to fix the violations. In addition, the approach should be able to provide explanations and potential remedies [13]. This criterion evaluates how existing approaches handle the inconsistencies and violations of the compliance rules.

2.4.10 Model evolution

Legal documents change frequently. These changes might also affect business processes because new changes may require introducing new tasks or dependencies between the tasks; thus, processes might grow exponentially and might cause maintainability problems [25, 81, 165]. This criterion examines how effectively existing studies handle the changes in the business rules, and how efficiently these changes can be propagated into processes without affecting the whole system.

2.4.11 Computational complexity

This criterion evaluates whether existing approaches addressing the compliance problem are sustainable given the presence of dynamically changing environments in which hard coded check repository can grow exponentially making it difficult to evolve and maintain the compliance requirements [167]. It is imperative that a compliance approach is computationally efficient and does not suffer from the size of the compliance rules, expressiveness of modelling languages, and computational budget concerning available time and computing infrastructure.

2.4.12 Scalability and agility

In today's ever-changing business environment, in which enterprises operate, is highly scruitnised. In such highly scrutinised environment enterprises are subject to thousands of regulations—especially those operating across geographies, jurisdictions and cultures. Given this, a compliance framework must be not only scalable to accommodate any changes in the regulations, but also it should not suffer from the size of the rules it can accommodate. Also, it should provide enterprises with a timely and demonstrable information enabling them to not only proactively address non-compliance issues but also an outlook to future trends. Thus, making enterprises more effective ensuring corporate integrity and more agile in responding to ever-changing regulatory and business risks.

2.4.13 Usability

Compliance rules might include legal jargons, which are ambiguous and might cause problems for business people. It is paramount that all the stakeholders fully understand the compliance requirements before implementing them into business processes. Hence, it is mandatory to represent compliance rules into business like intermediary terms more understandable to non-technical business people. A maximum balance between the formal representation of compliance rules and their readability would ensure the highest degree of uniform understanding, reduce or eliminate possible conflicts and involvement of non-technical people.

The above-mentioned criteria reflect upon the vital features that a compliance approach should be able to provide support for. Essentially, from an organisational point of view each of these criteria is important with its own niches. Hence, to ensure simplicity in this survey, we only evaluate approaches against *strategies* and *norms modelling* criteria.

3 Evaluation of state of the art

Apart from many challenges that come from different sources of compliance requirements—in general, the most prevalent challenges when considering how technology might help enterprises to deal with the compliance problem are, namely (a) how to verify the compliance of business processes against the governing rules; (b) how to handle ever-changing regulatory requirements; and (c) how to maintain the business agility in dynamic business environments governed by the regulations.

To deal with these challenges and to meet their compliance reporting requirements, enterprises employ different compliance management strategies, namely design-time, run-time

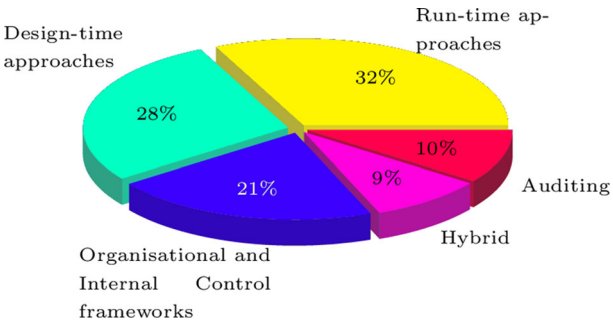

Fig. 4 Distribution of collated studies on compliance strategies

and auditing time compliance management. These strategies lead the emergence of several compliance management frameworks, methods, approaches and systems. Figure 4 shows the distribution of the compliance approaches based on the compliance management strategies where design-time make 28%, run-time 32% and auditing 10% of the surveyed approaches. In contrast, the percentage of organisational and internal control-based frameworks and hybrid approaches is 21% and 9%, respectively. The reported approaches address the compliance problem from a variety of aspects and offer different functionalities.

Next, we review the existing literature on compliance management and highlight their strengths and weaknesses based on the evaluation criteria discussed in Sect. 2.4. Table 2 categorises some existing compliance management frameworks reported in business process compliance literature.[10]

3.1 Organisational compliance requirements management frameworks

More and more enterprises are both venturing globally and even incorporating new technologies to provide a wide range of, and better services to their customers. On the one hand, such new business ventures and the use of new technologies increase the customer base of an enterprise; they bring new challenges from an internal management and regulatory perspective because of the increased role of compliance in their processes on the other. Several research efforts have focused on the ever-increasing compliance requirements of enterprises. The COSO standard [39] provides the guidelines for establishing business objectives and for integrating compliance requirements into business processes for effective operations. However, the standard neither proposes a compliance model nor it describes any compliance controls.

The OCEG's[11] governance and risk compliance (GRC) [147] and CoBIT [35] initiatives provide governance models for enterprises operating in specific domains. For example, CoBIT provides the governance models for establishing, refining and concreting the control objectives for effective and efficient management of IT resources and operations in large enterprises. However, these initiatives are not meant to suggest the ways *to define and correlate the compliance concepts and to integrate them into their business processes* (see [49]); they simply provide guidelines for managing and refining the compliance requirements.

The major risk of non-compliance is the financial loss and the loss of trust, which can lead to drastic consequences for enterprises. Effective risk management is one of the key

[10] This is not an exhaustive list of all represented frameworks in their respective category.

[11] OCEG: Open Compliance Ethics Group, available at: http://www.oceg.org/ (retrieved: 8 May 2017).

Table 2 Snapshot of research in business process compliance

Organisational and internal controls compliance frameworks
Namiri and Stojanovic [139], Evans [54], Rosemann and zur Muehlen [166], Namiri and Stojanovic [141], El Kharbili et al. [48], Karagiannis et al. [103], Hoffmann et al. [91], Schumm et al. [174]
Design-time compliance management
Logic-based approaches
Governatori and Milosevic [72], Governatori et al. [79], Milosevic et al. [135], Milosevic et al. [136], Goedertier and Vanthienen [65], Governatori and Rotolo [76], Governatori and Rotolo [73], Governatori and Rotolo [75], Scannapieco et al. [170], Letia and Groza [122], Lomuscio et al. [124]
Pattern-based approaches
Han et al. [80], Yu et al. [198], Schmidt et al. [172], Yu et al. [197], Namiri and Stojanovic [139], Förster et al. [57,58], Wang et al. [193]
Run-time compliance management
Run-time monitoring-based approaches
Keller and Ludwig [105], Milosevic et al. [134], Kabilan et al. [101], Maggi et al. [129,130], Gómez-López et al. [68], Teresa et al. [180], Knuplesch et al. [112]
Run-time logic-based approaches
Giblin et al. [62], Governatori and Rotolo [74], Alberti et al. [8]
Run-time model-checking approaches
Bai et al. [18], de Moura Araujo et al. [44], Kazmierczak et al. [104], D'Aprile et al. [42], Hassan and Logrippo [86], Vázquez-Salceda et al. [191], Birukou et al. [30], Gómez-López et al. [69]
Compliance auditing approaches
van der Aalst et al. [187], Doganata and Curbera [45], van der Aalst et al. [188], Arya et al. [11], Agrawal et al. [6], Johnson and Grandison [98]
Hybrid approaches
Ghanavati et al. [60], Sapkota et al. [169], Cunningham et al. [40], Rifaut and Dubois [162], Kähmer et al. [102]

determinants of compliance, and minimising the operational risks has been highly emphasised in the COSO framework. Ashby [12] suggests that the adoption of a *process-based comprehensive* approach to effective risk management. However, Ashby also suggests the thoughtful management of the risk not only encompasses the adoption of a comprehensiveness of a process-based approach—a carefully selected GRC framework is also inevitable to ensure that all business processes are fully integrated in order to manage the risk efficiently. Evans [54] discusses the adoption of an end-to-end process-based approach to the management of risk. To improve business agility, Evans also signifies the importance of choosing a right GRC framework for the successful management of potential risks. The study lists eight determinants from the OCEG's capability model—such as *context, organisation, assessment of threats* and *opportunities*, for aligning the business strategy with the business processes for effective management of compliance-related risks at various levels of an enterprise. A rather similar compliance model based on GRC framework can be found in [192].

A few studies conceptualise the risks and business processes and proposed conceptual models for managing and connecting the compliance controls into business processes; for example, conceptual models proposed by [139,166,177] and [141] to name but a few. These

studies identify several business artefacts that represent varying segments of business operations, processes, accounts, control objectives, and risks, and their relationship to business process compliance. The authors in [141] listed a set of properties of the internal control systems of an enterprise that can be used to minimise or even remove the risk of non-compliance. A goal-oriented approach assessing the regulatory requirements using BASAEL-II [181] and a case from financial sector for managing the operational risks is proposed in [162].

3.2 Policy-based frameworks

As we previously mentioned, regulatory documents are not the only source of compliance requirements. Organisations can also implement their own policies for transparency and effective management of their business operations. Namiri and Stojanovic [141] provided a taxonomy of properties that organisations can use to verify their internal control systems. Organisation's internal control systems are generally responsible for implementing the external compliance requirements. If these control systems are compliant, it is relatively easy for enterprises to satisfy the compliance requirements. Some key internal policy-based compliance management approaches are [47,48,72,142,167,176].

The framework reported in [48] defines and integrates compliance requirements by means of policies within an enterprise. Due to the vertical nature of the compliance problem, the authors defined the semantics of several enterprise models and enriched them with compliance requirements modelled as elements of the policy ontology. While enterprise models and compliance management models are two distinct notions, a synergistic relation between these two notions is mandatory to achieve compliance. The framework proposes the integration of compliance requirements into the enterprise goals and strategies to provide a better understanding of compliance at different levels, for example operational processes and business objectives levels. The multi-layered approach introduces mandatory transformations of the different components defined in each layer. The use of business rules as a source to realize and monitor the compliance requirements on a process model has also been proposed. In contrast, the work of [103] sees compliance as more of an enterprise-wide problem than a project-based issue. The authors propose a method to link regulatory laws to business processes supported by the ADONIS platform and the SOX portal. The usability of proposed method was verified by implementing the regulatory rules from SOX Act and claimed that a significant degree of compliance at run-time.

A holistic layered approach dealing with the compliance management problem to achieve agreement among all related regulations, policy controls and stakeholders has been proposed in [32]. The layered approach first identifies all pertinent regulations and conflicts between the involved parties and then the level of the parties' compliance with the rules requirements. The authors used IT solutions and proposed a model to track down the compliance requirements. All business units are required to divulge compliance based on the established relationship in the previous step. The proposed model establishes control among all involved activities to achieve compliance. This framework provides an effective solution to the compliance problem by defining the relationship between all stakeholders at top management level and the relevant regulations; however, the framework does not appear to cover the whole business process model of an enterprise. In addition, no compliance requirements have been specifically generated. This gives the rise to question: *how will the processes of each business unit comply with the regulatory laws?*

3.3 Internal control-based frameworks

Internal controls are policies that limit the way how organisations should operate. For example, an internal control might prescribe to report large transactions by the financial institutions. These controls ensure effective implementation of compliance requirements and might affect different aspects of an organisation. The effective management of the organisations aligned with internal controls allows enterprises to mitigate the risk of non-compliance.

A formal framework to define and relate an enterprise's internal controls to ensure that business operations are in alignment with the regulatory requirements is presented in [142]. The proposed framework first identifies internal policies and controls and validates their interdependency against governing rules. These controls are then formally defined in first-order logic. The framework provides the necessary support to verify whether a system implements the required set of rules, establishes the relationship among the business processes and remains consistent during its evolution. The formal model introduced in this framework provides a rich formal representation of risks, involved entities and their semantic relationship with business processes and controls; however, it only captures the entities involved in the process, not the semantics of each entity. Furthermore, the capturing of the interdependence and contradiction of semantic relations is not possible; that is the formal model is not capable of automatically detecting any contradicting and interdependent controls. This identification of contradicting controls with respect to every entity is highly desirable. It is necessary for the gathering of all information relating to the semantic relationship between entities and/or processes in order to determine whether a process is compliant with a set of rules. Another problem with this framework is that it does not provide a fully automated solution to the compliance problem as some tasks are carried out manually while defining relationship between processes and internal controls. There is also no evidence suggesting that this framework identifies and proposes remedial actions when a rule is violated, and this restricts its scope.

While compliance aims to align the business process specifications and business rules specifications which are two distinct worlds, the idea of maintaining a separate controls directory in order to align the business practices with controls objectives was coined in [167]. The proposed framework allows a formal representation of control objectives in formal compliance language FCL (cf. [72]) and links these control objectives to processes in the form of control tags. These control tags can be derived from the FCL to analyse and visually annotate graph-based process models. The analysis of the process models enriched with these controls tags allows the redesign of compliant business processes. The control objectives are concerned with the data related to the entities involved in a process and impose constraints on the data. A limitation of this approach is that there is *no evidence to show where the contents of the control tags will come from* (especially with respect to the data for the data tags); nor is there any evidence of *the way in which the data constraints can be implemented* on a business process. The same is true for other control tags for resources, temporal and control-flow tags. In addition, [167] primarily focused on the preventive compliance measures to check control-related violations at design-time. However, in some situations, not all details related to a process might be available; thus, compliance measures can be checked only at run-time; however, no such support is provided.

Rather similar works [139,140] used a pattern-based approach for managing the control directory of different actors in the compliance management process and present in detail the relationship between a business process and control objectives. Their approach suggests remedial recovery actions that react to the violation of a control objective and can be linked to each control objective. For most part, these approaches provide run-time compliance

monitoring functionalities, but no mechanism is provided for the design-time compliance checking.

Hoffmann et al. [91] presented a formal framework for annotated process models and introduced the notion of clausal compliance constraints. They devised a lower-order polynomial time *I-algorithm* to check the completeness of compliance constraints as *partly exact*, *partly approximate* or *guaranteeing* only. The proposed framework has a number of shortcomings: *first*, the *I-algorithm* does not seem to work in the presence of conflicts between the obligations, because the algorithm operates in polynomial time and can only be used for checking the constraints on basic processes, that is, the processes that have no loops. *Second*, from a constraints-modelling perspective, the formalism used in their work lacks expressiveness for modelling the compliance requirements (for example, modelling preference-based norms such as "*if you cannot do P, then do at least Q*" are clearly an example of permission-based (or CTD) requirements, and such requirements cannot be modelled). *Third*, from a business process modelling perspective, the proposed framework suffers from several difficulties as data contents and temporal aspects of the behaviour of activities cannot be modelled using *I-algorithm*. Similarly, it is not possible to annotate the predicates that represent the qualitative properties of the data. Accordingly, the support for the temporal behaviour is limited only to *what is encoded in the control flow* of the process. Hence, it is not possible to quantitatively measure that how long an activity takes to complete, and also it cannot be formally expressed.

Schumm et al. [174] introduced the notion of reuse of compliant process fragments to embed them into a business process at design-time. They combined the formalism of compliance requirements and automated verifications of a given process in a template that can be reused for another processes. This template-based approach is less advantageous, given the varying nature of compliance requirements, and the need to add, remove and update these requirements. For example, in the process model, when a new task (or a sub-process) is introduced, and has its own requirements—then the previously stored compliant fragment has to be concretised or even fully re-customised. This is due to the fact that previously stored requirements might not be captured in the template fragment. In addition to that, only specific requirements relevant to the control-flow aspect of a business process can be handled with these compliant fragments.

3.4 Ontology and semantics-based frameworks

In the context of SeaFlows[12] [126] reported a compliance management framework to address the challenge of the semantic constraints condition on business processes to comply with regulatory laws. The framework incorporates a graphical modelling language to capture process-related compliance rules which provides primitives to capture complex compliance rules in the form of directed graphs. In addition, it indicates the need for an independent compliance requirements repository that is maintained separately from the business process repository. The framework does not only simply provide *Yes/No* type answer to show compliance with a process; rather, it is capable of validating semantic constraints and of checking compliance and the violations of compliance rules, both at design-time and at run-time. The compliance support is provided in textual description of violations (log files) and is enriched with compliance rules violations and compensation activities that can be used as input to process analysis and evaluations.

While it is claimed that SeaFlows provides so-called, *lifetime compliance*, it does have its drawbacks. For example, there is no indication of how well the semantic constraints can

[12] Semantic Constraints in Process Management Systems, available at: https://www.uni-ulm.de/in/iui-dbis/forschung/abgeschlossene-projekte/seaflows/ (retrieved: 8 May, 2017).

be represented in a process model, or how the implicit constraints are derived. Moreover, the semantic constraints can be often conflicting and redundant; however, there is no indication how the developed framework handles conflict and redundancy issues. Theoretically, a process model, or an instance, that violates semantic constraints, might still be syntactically correct; however, this is not applicable in real situations because it is semantically incorrect. Hence, it is essential to have implicitly derived constraints free from any conflicts and redundancies for effective compliance and the challenging task of balancing the semantic constraints. Another issue with this framework arises from the validation of the consistency of semantic constraints and compliance rules across different processes. However, SeaFlows offers no explicit solution or technique to address this issue. Moreover, no solution for establishing and verifying the relationship between the compliance rules and a business process to achieve full compliance has been proposed.

An ontology-based framework in the scope of an *intelligent compliance management* (iCMP) project to explore the application of semantic web rules and *OWL*[13] ontology to represent business domain and compliance knowledge is discussed in [196]. The authors first extracted the compliance requirements and documented them semi-automatically and then check the business rule constraints modelled using SWRL.[14] This approach facilitates the extraction of compliance requirements from source documents; it also deals with data incompleteness, which is a major deficiency of semantic web technologies. Ideally, the orientation of their framework is diverse as it provides an effective solution to defining model compliance data. For example, the policies and requirements are formally defined using clearly defined data structures. However, the provided support is limited to the extraction of compliance requirements from data only. This is because the semantic constraints imposed on other business process aspects such as control flow, time and resources are not supported.

The complexity of ontological mapping of compliance requirements and reasoning about them is especially a challenging work because compliance rules are continuously added, removed or updated. This can significantly increase the size of compliance rules repositories, and managing huge compliance rules repositories is a difficult task. Accordingly, with added rules, computation complexity might also increase. There is no indication of how the complexity of compliance requirements can be handled, as no support is provided to manage the changes in the compliance rules. These factors limit the effectiveness of the [196]'s framework.

In contrast, [22] presented a rule-based semantic framework for automated regulatory compliance in construction sector. Their framework comprises an *abstract ontology* that provides the core concepts of regulations and semantic mappings; a *core ontology* to define the concepts of the target domain; and a *data format* ontology defining the data format on which the compliance checking will be performed. While in the *regulation extraction* and *semantic data mapping* phase, respectively, the domain experts enrich the regulation documents with metadata and specify the semantic mappings between the semantics of the regulation ontology and semantics of the data format. The compliance checking is performed once the semantics data mapping process is complete; then, a series of SWRL rules enhancing the regulatory documents are generated. In the last step, the compliance results are generated from semantically enhanced documents. The proposed framework allow the users to manually add additional data should they wish to enhance the A-Box ontology. Essentially, their framework is beneficial in the situation where users may have less data available for compliance checking.

[13] Web Ontology Language http://www.w3.org/TR/owl-features/.

[14] Semantics Web Rule Language https://www.w3.org/Submission/SWRL/.

4 Design-time compliance management

The *design-time* compliance management (DT-CM) approaches are efficient ways of verifying the compliance in the early stages of a process design and fall into the category of "*static compliance checking*" methods. The aim of the design-time approaches is to check the compliant behaviour of business processes against all the applicable rules, thus preventing actual execution of non-compliant processes [106]. The design-time approaches can be divided into two sub-categories, namely (i) *design-time compliance checking*, and (ii) *design-time compliance verification*. *Design-time compliance checking* targets the implementation and checking of regulatory rules, while a process is being modelled. This allows the process designers to take corrective measures at very early stages of the process design, thus completely preventing potential violations. However, because of their rigid nature, the design-time compliance checking approaches cannot be fully automated. Thus, they are more suitable for cases such as business contracts where business processes are derived from the defined specifications [14].

In contrast, *design-time verification* is used to verify whether a designed process conforms to the policies before actual execution. Unlike design-time checking, design-time verification approaches are rather flexible in nature and allow a higher degree of automation. A number of design-time compliance *checking* and *verification* approaches have been reported in literature, and these can be categories based on their underlying technique(s), e.g. *logic-based approaches, static compliance checking-based approaches, object life cycle-based approaches, pattern-based and query-based approaches*. The rest of this section gives a comprehensive view of both these sub-categories of design-time compliance management based on their languages, tools, systems and formal approaches in these categories.

4.1 Logic-based approaches

To properly verify that a business process has comply with the norms regulating the process, one has to align the formal specifications of the process and the formal specifications of the norms. This means that the normative specifications should be able to tell us what deontic concepts (obligations, permissions, prohibitions) and violations that a process is subject to, which can be modelled with the help of formal logics.

Governatori and Milosevic [72] propose a normative system to describe contracts in terms of deontic concepts and support the breach or violations of some obligations based on the notion of contrary-to-duty obligations (or reparational obligations) [144,157]. The proposed formalism lays the foundation for contract specification language known as *"business contract language"* (BCL). Later, the same authors extended their formalism and proposed FCL [79], a new business contracts modelling language to check the compliance of business processes and business contracts. Their approach coupled with semantics specifically developed for compliance checking which can help in determining the current state of affairs, i.e. *"ideal"*, *"sub-ideal"* and *"non-ideal"*, when comparing business processes and contract conditions. However, these semantics support relatively simple normative expressions in which deontic constraints are expressed as single events; the support for rather complex events relationships is very limited. In addition, the handling of deadlines in FCL obligations modalities is poorly expressed.

Milosevic et al. [135,136] used FCL to achieve compliance in a progressive manner. Initially, collaborative interaction or contract framing behaviour among all involving parties is identified; then, internal process compliance and the contract behaviour for each party are determined. Different heuristics are applied at this point to reflect different contract conditions

and to specify a set of actions to be taken when a violation occurs. The likelihood of contract violations is then checked at supplementary stages of a process design.

The authors in [64,65] achieved process compliance using a rules set of permissions and obligations (deontic logic). They proposed PENELOPE (process entailment from the elicitation of obligations and permission), a declarative language to elicit business rules imposed either by internal policies or by external regulations in the form of temporal deontic expressions. These expressions are used to generate compliant processes covering control-flow and temporal constraints among activities in a business process. Aiming to achieve compliance at design-time, PENELOPE focuses on the verification and validation of a process model at design-time and does not intend to apply deontic rules at run-time. The proposed language has limitations, however, a major issue arising from its underlying formalism discrete event calculus (EC) [113] used for modelling the obligations and permissions. Furthermore, the language can only model a subset of obligations types. This is because of the problems with the underlying semantics of EC, which fails to capture the effects of the obligations onto the tasks of a process [84].

Governatori and Rotolo [76] proposed process compliance logic (PCL), an extension of FCL, for capturing various types of normative requirements. The proposed logic is based on defeasible logic (cf. [145]), and deontic logic of violations (cf. [73]), which transforms the deontic obligations subject to a business process into normal forms and represents them as PCL expressions. The PCL expressions for deontic systems define a behavioural and state space to identify the differences between the process execution paths and the PCL constraints. To test the effectiveness of the PCL, the authors used a three-step compliance checking algorithm that they have previously proposed [75]. Scannapieco et al. [170] also presented a formal approach to integrate the business policy constraints and the organisational goals in such a way that allows a business process to simultaneously fulfil the policy constraints as is the case with [76]'s work that primarily deals with the control-flow aspect of a business process. The data, resources and temporal aspects are not addressed.

A logic-based model-checking approach for compliance verification of the integrated business process models is reported in [122]. The proposed approach extends the norm temporal logic of Ågotnes et al. [3] and introduces obligation and permission operators into the temporal logic to model the various compliance requirements from HACCP standard[15] in the food safety domain. The compliance checking is performed by a four-step mechanism where, in the first step, the domain knowledge—that is, the normative requirements—is translated into Norms Temporal Logic and Attribute Language with Complement (NTL–ALC) logic. Then, a WF-net using a Kripke structure is generated with states that are labelled with all normative requirements, specified in the form of normative formula f pertaining to the state. Each formula f in the state of a WF-net is verified if the formula f representing the norm holds in the state. If f does not hold, the state violating the norm is added to the set of breached states. The proposed approach allows the integration of subsumption-based reasoning, with the possibility of checking the compliance of various types of norms. The extended NTL–ALC permits to integrate the abstract and concrete business processes, thus making it a more explicit representation of compliance requirements for business process models.

The problem of compatibility checking between business processes and business contracts is addressed in [79]. The compliance checking approach involves the use of logic-based formalism to express business contracts and check their violations. The authors develop a

[15] The Hazard Analysis Critical Control Point System, available at: http://www.standards.org/standards/listing/haccp (retrieved: 8 May 2017).

semantics approach to determine ideal, sub-ideal and non-ideal scenarios for the comparison of business process execution paths and the contract conditions. In contrast, [77] reported an algorithm to check the deontic modalities of a business contract against a business process. To achieve this, activities involved in the process are annotated as having certain effects. A rather similar approach is reported in Lomuscio et al. [124] where the authors used multi-agent systems to verify contract-regulated service compositions. However, their approach only enables the checking of compliance violations and does not suggest any remedies if any violations of rules occur.

Moreover, [78] reported a rule-based compliance verification framework, the Regorous,[16] based on the compliance-by-design methodology proposed in [77]. In the framework, a business process will be modelled using FCL, and verification will be performed by: (i) generating the execution traces corresponding to the business process, and (ii) based on the compliance checking algorithm proposed in [74], the set of traces will be passed to SPINdle [116]— a defeasible logic reasoner, for the evaluation of the FCL rules. In case any non-compliance issues are detected, the compliance checker returns the processes (along with the traces and tasks) and the rules that have been violated. Its objective is to ensure that a business process complies with all pertinent regulations before the actual deployment of that process.

4.2 Static compliance checking frameworks

Static compliance checking is concerned with the techniques associated with a thorough analysis of the behavioural properties of a system to investigate whether a property satisfies applicable requirements is performed. For static compliance checking, it is not necessary for the system to be fully operational. This also implies that such techniques can be applied to the properties that are in an intermediate (or potentially) incomplete state. The static compliance checking method provides several benefits over run-time compliance checking counterpart. This is because static techniques can frequently produce counter-examples from the violations and allow asking *What—If* question [95]. This, in turn, facilitates a greater understanding of behavioural properties and a detailed analysis to rectify the potential problems. Model-checking and the design-time compliance checking approaches fall into the category of static checking methods.

A static compliance checking framework that uses a static method to check business process models against business rules is proposed in [123]. The authors employ a classical model-checking approach and used high-level specifications languages such as BPEL and BPSL. Their approach enables the formalisation of a business process model with π-calculus and transforms them into a finite state machine (FSM) representation. In the case where process modeller discovers a non-compliant process, the counter-examples are automatically created at process design level. This makes the compliance checking process rather easier and less error- prone, thus reducing the risk of non-compliant operations. This framework provides effective support for compliance checking, as the process designers can immediately react to any non-compliant behaviour. However, it is unclear how transparent the compliance checking is. Similarly, it is also unclear whether π-calculus accurately represents the mapping between the process models and compliance rules, and the subsequent transformation into FSM representations.

Nishizaki and Ohata [143] propose a rather similar approach for checking the compliance of business processes for information systems, using the UPPAAL [154] model checker. The

[16] Regorous Compliance Checker: https://www.regorous.com/ (retrieved: 10 Oct 2013).

business processes are defined as timed automata, and the regulatory rules are translated into computational tree logic (CTL) specifications, which are then fed to the model checker. The model checker automatically searches for all execution paths to verify the compliance of the rules. Where some rules prove to be non-compliant, the model checker provides a counter-example against the violated regulations as transitions traces. In this approach, the use of timed automata allows the specification and verification of queries using a real-time clock variable to represent the timed constraints. This makes the model checkers more suitable for verifying the compliance of real-time systems. This approach differs from the work of [123] in terms of the underlying formalism used to model the regulatory rules as the authors employ BPSL for the specifications of rules, while [143] use timed automata and timed CTL. Despite the fact that the use of real-time automata allows for a description of the real-time properties of the system, this latter approach is less practical because of its limited capabilities in defining and verifying the compliance issues. It does, however, have advantage as the model checker generates counter-examples to make corrections in the models, which is not possible with the framework of [123].

4.3 Object life cycle approaches

While business process captures the business activities in terms of tasks, they usually also capture the flow of objects in a process to represent data exchange between tasks. Objects used in process models are generally associated with a set of states that represent their processing status during execution. In order to ensure that business processes correctly represent and manipulate object states, processes can be extended with input/output state of an object. Object life cycles [173,178,185] are main tools for modelling valid states of an object during it life—and compliance with object life cycles ensures consistency within business processes that applicable elements of policy related to the object are fully adhered to.

Küster et al. [115] introduced the notion of object life cycle and coverage to check whether a process model is compliant with the referenced *object life cycle* at design-time. The proposed technique first generates a process model from one or more referenced object life cycles. In the first step, the object life cycle is used to generate a set of actions for the process model to identify transitions in the given object life cycles. This ensures that invalid composite states cannot be reached in the composite object life cycle. The order of the process model is then determined, and actions are combined with process fragments in the second and third step, respectively. The process fragments are connected in the final step. This approach provides support to process designers; however, it is not fully automated as synchronization points among the process object life cycles have to be defined manually in the case of several object life cycles. In addition, in some cases, the number of object life cycles can be very large, and this can increase the size of the process model. The increased size of the process models can make them difficult to handle. In this approach, no mechanism is provided to determine how the compliance checking will be affected if the size of the process models is relatively large; how several referenced objects life cycles are taken as an input; or how compliance will be preserved if a generated process is customised. Also, the dependencies between the compliance rules and the alternatives become a matter of concern (i.e. achieving the correct compliance rules) when a process model is customised.

Contrastingly, Schleicher et al. [171] extended the work of [115] to address the issue of a synchronisation point (variability) and to preserve compliance in customised process models. The authors introduce an approach based on the concept of a business process template that implicitly contains compliance constraints and points of variability to prevent process designers from bothering with compliance constraints at design-time. The reported algorithm

ensures that compliance constraints are not violated when a process model is customised. The problem with the algorithm is that it does not provide any mechanism to handle dependencies between the alternatives and dynamic compliance rules, as mentioned above.

4.4 Patterns/graph-based approaches

Automated compliance checking of the legal requirements for the business processes is highly desirable. These requirements are often written in a natural language and must be translated into a machine-readable format for automated verification. Generally, formal languages (such as event calculus, temporal logic and deontic logic), which provide the reasoning support, are used to translate the legal requirements. However, due to their complexity, the comprehension and usability of these languages are difficult, especially for non-technical users such as process analysts and compliance experts. Thus, the usability of the formal languages is one of the main concerns for non-technical users who possess less knowledge of these languages [49]. To address the usability concern of the formal languages, researchers proposed to embed the formulas into a formal language that translates the compliance requirements into easy-to-understand visual patterns or graphs. This leads to the emergence of many pattern/graph-based compliance verification approaches in the business process compliance domain.

The authors in [80] proposed a pattern-based property specification language, PROPOLS[17] for specifying temporal business rules. The PROPOLS defines a collection of properties for a service composition, with each property being a rule or a logical composition of rules that govern the ordering of the primary services within a service composition. Each rule consists of a pattern element and a scope element. Because each pattern specifies the existence behaviour of a single business activity or temporal relationship among activities, PROPOLS enables process designers to insert, delete or rearrange processes to be compliant, based on temporal business rules. Deviations from the business rules are identified using finite state automata (FSA) to inform process designers about non-compliant behaviour. The automata are derived from a set of business rules and existing process schema. The work of [80] is extended in [198] where the authors propose a synthesis framework to generate a process models from a set of temporal business rules. The proposed approach generates a process model and a requirements model (temporal rules) to achieve intuitive specifications and *correction by design*. This helps the process designers to rectify design-time mistakes. In addition, it also allows automated verification of semi-automatically generated process models.

Schmidt et al. [172] discussed an ontology based approach to representing service processes and their compliance requirements to verify whether the designed service processes are compliant with the relevant compliance requirements. The proposed approach employs two distinct ontologies: a process ontology defining the concepts that are needed to represent service processes and a compliance requirements ontology consisting of the concepts that represent the objectives and requirements of compliance rules. The authors report three distinct categories of compliance requirements in their model: *syntactic, semantic* and *pragmatic*. To verify the compliant process elements on the semantic requirements of service processes, a reasoner is applied. The problem with the proposed approach is that it isolates only processes whose requirements are instantiated as compliant processes; other uninstantiated processes are not included. Moreover, there is no indication of how the proposed approach deals with non-compliant processes as no remedial actions can be taken in the proposed approach.

Yu et al. [197] introduced a compliance verification approach to BPEL schema, which employs an ontology language for property specifications. The verification process starts

[17] PROPOLS is an ontology-based property specification language based on PPS to specify service composition properties.

with a high-level description of a BPEL schema to implement in the process. Then, semantic mapping between the operations is defined in the ontology language, and a finite and deterministic *labelled transition system* (LTS) model is generated. From this LTS model, a *total and deterministic finite automata* (TDFA) is built. This includes the set of final states and error states to collect a list of all the unwanted events of each state. In the last step, verification of the compliance BPEL schema determines whether all acceptable event sequences of the BPEL schema are present in the list of acceptable sequences generated in the form of TDFA.

Förster et al. [57,58] presented a *pattern-driven process* approach to visually express the compliance constraints on the process behaviour. The authors use PPSL, an extension of UML activity diagrams [151]. The activity diagrams are used to specify possible patterns that need to be applied in the business process models. This enables the process designers to have an abstract view of a possible behaviour of a business process. For example, UML activity diagram patterns that extend the edges with the stereotype $\ll after \gg$ show that it is not necessary that two activities be strictly sequentially executed. These patterns are then used to check whether business process conform, by transforming them into temporal logic systems, while business processes are transformed into a labelled transition system defined by a semantic domain meta-model that enables the application of model checking to ensure the conformance of the business processes to patterns. Although the proposed approach provides a flexible way for the process designers to check the quality of conformance, the approach is not free of issues. The definition of process behaviour as visual patterns at design-time is one such issue, because these patterns might depend on each other, or even might reflect contradictory behaviour. Currently, the approach does not provide any mechanism to gain (potentially prior) knowledge of the inter-dependencies among different patterns. Furthermore, these patterns are not able to expressively model negative obligations; that is, a rule might stipulate conditions that prevent some activities from ever happening, while others have already been executed. Essentially, from a business process compliance perspective, negation is an important aspect of modelling prohibitions; however, no explicit support is provided in this framework for modelling the negations. In addition, this approach only focuses on the control-flow aspect of business processes and does not provide any support for modelling and checking their compliance with the data, resources and control-flow aspects of a business process.

The authors of [139] employed a pattern-based approach to modelling an enterprise's internal policy controls. They build their model on the de facto internal control standard (*otherwise known as* COSO[18]). In the process execution phase, a bidirectional interaction between BPM and internal control management is established. Later, all information about the current instance of the business process is enacted. In case of any violations, a recovery action (defined in the controls) is executed. The major benefit of their approach is its ability to define different controls beyond workflow, and in different environments, to reuse of the process models. However, the proposed model is not fully automated because it requires manual selection of a control pattern and its design on a business process corresponding to the domain specific compliance requirements. Furthermore, there is no support for handling inter-control dependencies; for example, different controls can contradict, subsume or even block the execution of other controls in a business process interaction. This signifies the need to establish a stronger correlation between processes and controls. Moreover, this approach does not support compliance verification beyond the design-time, nor does it support resource and temporal aspects of the business process.

[18] Internal control, an integrated framework: the committee of sponsoring organisations of the treadway commission [39].

In the context of REO tool kit project, Arbab et al. [10] presented a channel-based coordination language for the design-time verification of business process models. The language uses model-checking and bi-simulation techniques to formally analyse the correctness of the business processes against imposed constraints. For the verification of compliant behaviour, in this approach, business process are first modelled either in BPMN [150] or in UML [151] activity diagrams, which can be mapped into constraints automata. The compliance requirements are represented using linear temporal logic (LTL), and then, the model-checking techniques embedded into REO tool kit are used to verify the compliance of business processes. The work reported in [174] is grounded in the REO tool kit, where the REO is used for the automated compliance verification of business process fragments against the business constraints mapped in LTL.

Contrastingly, Elgammal et al. [51] proposed COMPAS—a comprehensive compliance governance framework that provides an all-round compliance support for service-oriented-architecture (SOA)-based systems. The framework adopts a model-driven development approach for designing compliant processes/services, using a view-based modelling framework and domain-specific languages to model the compliance concerns in process models. For compliance checking, business processes are annotated with compliance constraints in the form of (reusable) process fragments. These fragments underline the required behaviour of the control flow of a process model, and they are formalised using LTL. Then, the annotated process fragments are then assessed to validate the compliant behaviour of the process models at run-time, using event logs. A protocol component evaluates the generated event logs to check whether the process model complies with the behaviour described in the attached compliance constraints process fragment. If the monitoring protocol detects any non-compliant behaviour, it reports a violation and publishes it as a violation event.

As far as the modelling of compliance requirements is concerned, the framework uses a compliance request language (CRL) [49,52] for modelling normative requirements. The core of the CRL is LTL-based graphical compliance patterns, which are high-level compliance templates to model the compliance constraints—predominantly, compliance requirements from the control flow (structural) perspective of business processes. In addition, most of these patterns are used by other frameworks, and more, recently additional patterns representing features specific to normative reasoning, such as exceptions to rules and compensation of violations, have been included [52].

In [193], the authors proposed a formal approach that addresses the issue of determining the compliance of *product recycle management* (PLM) [160] systems and workflow management systems by using the data of the design objects which may evolve over the various versions of the product life cycle in the PLMs. This compliance-by-design checking approach employs the workflow nets, which are annotated by defining the version-annotated processes. In the annotated processes, the version annotations are specified with the certain tasks, as per the specifications of the access control privileges. The aim of the access control privileges is to control some operations at a particular state of the product life cycle, which may be subject to some restrictions. Later, the semantic and syntactical properties of the annotated process are defined, and these are then used to verify the behavioural and syntactical compliance of the annotated processes by merging the version-annotated process and transformed WF-nets.

A version-annotated process is considered compliant only if its compliance properties correspond to the soundness properties of the WF-net. If the soundness properties of both nets do not match, it means that the data design object's life cycle is not compliant. The existence of non-live tasks in the process can be one of the reasons for a non-compliant version-annotated process. Remedial action(s) can be taken to correct the problem by modifying the process model or access control specification from the task. The proposed approach provides

technical foundations for merging the two types of process models to create a new type of compliant WF-net. The proposed approach has fundamental issues with the annotating process. This process is semi-automated, which means that some of annotation task have to be manually performed by the domain experts. Annotating the hundreds of tasks in a process model—each task having (possibly) several related compliance rules—is a tedious task and potentially error-prone.

4.5 Query-based approaches

The idea of query-based approaches is to query the execution traces of running process instances for patterns of compliance violations. If the query returns at least one pattern of the compliance rule, a violation is triggered. In order to query the process instances, the possible patterns of rule violations have to be identified, which can be done either manually or automatically [125].

Awad et al. [15] discussed a BPMN-Q, a *query-based* approach to compliance checking. The approach is capable of answering Yes/No questions to verify whether a process is compliant. The authors use a graph reduction technique to gain the Yes/No answer. As an execution of a query graph, the graph reduction approach splits a process graph into a set of execution paths from the first to last nodes in the graph. Then, the order of execution is determined with respect to an execution path by finding the precedence between the occurrences of nodes. In the final step, a process graph is matched to a query graph. If it satisfies all sequence flow and path edges, the BPMN-Q returns a YES to a rule representing a complaint process. In the case where BPMN-Q does not find a match, a NO is returned to convey a violation of a rule. Their approach provides an answer to the rule query effectively and enables order checking between activities involved in a process. However, one problem with the graph reduction approach is that it might remove some activities that, at first glance, might not be pertinent to a query. This might include those activities which have to play a significant role in the completion of a process.

The work is later extended with the authors introducing the ways to visualise the violations of control flow ordering in the compliance rules [7,14]. Again, they use structural BPMN-Q queries to express the compliance rules, which are called ‘patterns’. These queries are used to find the set of process models that are subject to compliance checking in a process repository. Temporal formulas are then derived from the queries to check against the process model. In the final step, anti-patterns are derived automatically from BPMN-Q queries to report any rule violations in the process models. Because [15,19] use a graph reduction and model-checking approach, the proposed solution to derive anti-patterns queries has some limitations. The generated anti-patterns depend on the input state transition system of the process model. If the transition system is generated from a reduced process model, the resulting anti-pattern would not be usable on the original process model. Similarly, as the generated anti-patterns are given as a disproof of rule violation, it is possible that some violations are not reported by the model checker. Moreover, re-implementation of a translation software will be required in the case where some changes are made in the model checker software.

5 Run-time compliance management

Once a process model has been designed and the actual execution of the process instance is initiated, continuous monitoring of the running process instances is pivotal to detect any divergent behaviour, while processes are still running. Run-time compliance management

approaches can be broadly categorised into: (i) *run-time compliance monitoring* and (ii) *run-time compliance detection.* The aim of run-time compliance monitoring approaches is to continuously monitor the processes to check whether they violate internal controls or policies. For this purpose, a dedicated monitoring observer (or process engine) keeps track of the system's behaviour and occurrences of specific events during the process execution.

In contrast, run-time detection aims to identify the undesired systems behaviour by comparing the actual behaviour with the expected behaviour and alert the process designers to any violations. The process designers can then take appropriate actions to rectify the violations. Run-time monitoring and run-time detection approaches can be further classified into *monitoring-based approaches, logic-based approaches and model-based approaches*. The rest of this section gives a short overview of some approaches from the run-time compliance checking domain.

5.1 Run-time compliance monitoring approaches

An architecture to monitor service-level agreements (SLAs) for dynamic electronic services, in particular, the web services is introduced [105]. In the blocked architecture, the SLA requirements are first automatically generated by the SLA-driven system administration block, for further interaction with the web service-level agreement (WSLA) monitoring environment. In the WSLA monitoring phase, the monitoring is divided into two sub-phases: (a) *measurement service*, which measures all subsets of the SLA parameters generated by system administration block; and (b) *the condition evaluation*, which obtains measured values of SLA parameters from the measurement service and verifies these parameters against guarantees specified in the SLA. During the testing, if a breach is detected, a violation trigger is invoked to alert parties involved in the SLA. This verification of the SLA parameters can be done periodically, or when a new SLA parameter is available.

Milosevic et al. [134] discussed a compliance monitoring mechanism for electronic contracts. In their role-based architecture, the authors introduce a discretionary enforcement mediator (DEM) to measure the performance of a contract. The DEM can signal non-conformance of a contract event if it detects any deviating behaviour of the event. The DEM maintains a separate notary block in which it collects information about each violation, and this used to endorse the execution of corrective measures. This approach provides an effective means of monitoring the adherence to clauses of a contract; however, the approach is not fully automated. In an extension to their work on contract management, [100,101] use a multi-tier contract ontology for business contracts monitoring. They deduced a contract workflow model (CWM) from the multi-tier contract ontology consisting of different types of obligations written into a contract and different spaces from which each obligation passes. These obligations are monitored for potential breaches of clauses stipulated in the contract with respect to the actual execution of identified events.

While the proposed model provides an automated monitoring and tracking of obligation fulfilment, some components of this model are semi-automated and so do not support full automated compliance management at run-time. Moreover, this work focuses on control-flow compliance monitoring only; the data, resources and temporal aspects of a business process are considered. Similar to [105]'s work, there are other run-time approaches for monitoring the violations of SLAs (see [120,121] for more details).

Mobucom, a run-time compliance verification framework using ProM OS to dynamically check the compliance of running process instances with business constraints is proposed in [129,130]. Business constraints are modelled using Declare, a framework for declarative models formalised in LTL. The constraints reference model and a partial trace characterising

the process instances are then fed to Mobucom, which infers the status of each business constraint. It continuously produces the overview of the running instances reporting whether each instance is currently complying with the reference models.

In [138], the same authors extended Mobucom with formal semantics based on the EC to represent complex knowledge bases to handle events and properties that evolve over time. The use of lightweight EC as formalism provides the reasoning on how the process execution affects the "*state*" of the declarative models. However, these studies, due to the use of LTL and EC, have shortcomings in addressing various types of normative requirements [71,84]. In addition, the framework is not able to handle the business constraints related to data and resources aspects. Also, the framework is not able to reason about the violations and possible recovery actions, should a constraint have been violated. Consequently, the framework is not able to handle the compensatory constraints.

A model-based monitoring framework for diagnosing non-compliance during the execution of business processes is introduced in [68]. The framework uses the diagnostic theory that permits identifying the events that cause the non-compliance. To discover any non-compliance issues, business constraints are modelled by means of numerical constraints to create the dynamic diagnostic models (DDMs) and observational models (OMs). The created DDMs and OMs determine the compliance rules to describe the instance executions and then transformed into computable models for further diagnosis. Constraints programming (CP) techniques are then applied to automatically validate the constraint satisfaction problems. The CP analysis uses the numerical and boolean variables to identify incorrect event occurrences and correct time interval, where event should have been occurred to satisfy the all the compliance rules. Their approach can be used both run-time when the process instances are running, or after execution to detect and prevent failures in future instances.

Later, Teresa et al. [180] proposed compliance validation and diagnostic approach for business data constraints using constraint programming paradigm at run-time. In their approach, the validation and diagnostic of any inconsistencies in the data constraints are performed in various steps when an instance is running. In the first step, the data constraints are obtained based on the referential integrity between the database tables. This defines the relations derived from the functional dependency between the constraints by means of primary and foreign keys. Then, functional dependency graph defining the instance of data flow variables and tuples that represent the observational model is derived. In the last step, the observational model is instantiated to the business process model to discover any inconsistencies in the process model. The compliance checking in this approach is carried out by means of *constraint satisfaction problem*. Although the proposed approach efficiently audits the data constraints compliance, however, the scope of their approach is limited only to data constraints—temporal and resource constraints are not consider in the said approach. In contrast, Knuplesch et al. [112] proposed a run-time framework for monitoring the compliance of rules based on the visual *extended compliance rule graphs* (eCRG) language [111,175]. Their framework covers data, resources and temporal constraints as well as interactions between the business partners. Moreover, it also proactively detects the violations of compliance rules. However, their framework has fundamental problems as it is based on the first-order-logic-based compliance rules graphs [126]. In literature, it has been argued that first-order logic is not suitable for representing legal constraints [87]. Hence, the compliance results produced by Knuplesch et al.'s framework cannot be relied upon.

5.2 Logic-based formal run-time approaches

Giblin et al. [62] employed a formal approach to introducing the REALM model, a model-driven compliance automation method for regulatory policy and event monitoring. The meta-model of REALM supports the expression of temporal ordering and time periods as temporal logic modalities in real time. The domain discourse of a regulation, on the other hand, is represented by the UML model. This approach only considers the temporal aspect of a process life cycle and neglects control-flow, data and resources aspects.

The authors in [8] introduced a declarative programming language, SCIFF, an abductive logic programming for business contracts specification and monitoring. The run-time verification of contracts is performed by means of an abductive proof procedure which supports the dynamic occurrence of events, that is, the insertion of new facts during computation, and violation monitoring.

For run-time compliance checking, Governatori and Rotolo [74] used formal contract language (FCL) to propose their algorithm. The FCL constraints are used to define the state space and behaviour of contract policies that are used to compare the behaviour of execution paths of a business process. The algorithm operates in a stepwise fashion, where it first collects a set of all tasks involved in a business interaction. In the second step, these tasks are used to determine the norms that are triggered at run-time. Finally, compliant or non-compliant behaviour of a task is declared after comparing all tasks with normative constraints. Essentially, the compliance checking reported in [74] is an automated monitoring of the business processes to suggest remedies and/or mitigation of the control-flow deficiencies. Thus, *after-the-fact* detection does not have a preventive focus. In addition, data, resources and temporal aspects of a process life cycle have not been considered in this work.

5.3 Model-checking-based approaches

Model checking is a state-of-the-art technique where the system specifications are verified against certain properties. For a system to be compliant, all the properties must be satisfied over all possible states of the system. To verify the compliant behaviour, a model and the properties are fed into the model checker such as SPIN,[19] NuSMV and [20] UPAAL.[21] The model checker then thoroughly searches the model against the properties and generates counter-examples if any of the properties do not apply [26,131]. Since model checking is a well researched area, it is widely used in a multitude of domains, including the business process compliance domain. There is a wide body of proposals grounded on model checking for the verification of process models.

Bai et al. [18] adopted a model-based approach for policy enforcement and monitoring of the dynamic behaviour of web services at run-time. Their approach defines a policy model based on the WS-policy framework and includes the definition of the policies and policy-sensor correlation matrix adopted from the W3C standard[22] for specifying services of policy requirements. Policy consistency support is reported in this work; however, there is no indication of how policy violations can be handled, and no remedial actions are suggested to address these violations. Gilliot and Accorsi [63] presented a lightweight *violation anticipation monitor* (VAM) architecture for a priori run-time anticipation obligation violations.

[19] SPIN model checker, available at: http://spinroot.com/spin/whatispin.html.

[20] NuSMV: symbolic model verification, available at: http://nusmv.fbk.eu/.

[21] UPAAL: Uppsala—Aalborg model checker, available at:http://www.uppaal.org/.

[22] The W3C standard: http://www.w3.org/standards/.

Based on run-time verification (verifier module), statistical reasoning, and (linear temporal logic) LTL-based model-checking technique. VAM can answer as *"true"*, *"false"*, *"presumably true"* or *"presumably false"* to represent compliance at run-time. Remedial decisions are taken on the basis of true or false predictions (that is, where *true* means a process is compliant with all the regulations, and *false* means that the process is not compliant). However, when VAM answers *"presumably true"* or *"presumably false"*, it is up to the process owner to grant or revoke rights, or even to stop the execution of the process.

A run-time compliance checking (RTCC) technique to validate the business process with respect to the business rules is presented in [44]. They use UML[23] to model processes and OCL[24] expressions to represent the business rules. The model validation is based on the simulation of the execution of process instances based on case studies. Their simulation algorithm steps through the process model executing the actions associated with the activities with the help of a UML-based specifications environment, the USE tool [67], and checking the violations of any associated business rules. Their technique can precisely detect the situations in which the compliance rules are violated, and provide feedback to the analysts about the adequacy of a business process with respect to the business policies. However, the evaluation criteria used in this technique do not guarantee compliance; they simply provide some assurance that the process will not fail in the most elementary situations. Another issue is that the detected errors are not automatically corrected; in the case of violation, a business analyst's intervention would be necessary.

Kazmierczak et al. [104] introduced a state-based norms compliance model checker, called NoRMC. The proposed approach is based on norms' compliance CTL (NCCT; see [4]), and aims to verify which agents in the process interaction have to comply with the norms of an object to hold. The normative system is modelled as Kripke structure, and the constraints are defined to verify the agent behaviour on every state during the interaction. The prohibitions, represented as forbidden transitions, are modelled as a serial relation over Kripke structure, and all the forbidden transitions are removed from the structure after its implementation. The norms' compliance checker takes a model, a normative system and the CTL formulas, models the obligations and returns the states where the formulas are satisfied, so that countermeasures can be taken to repair the violation. Currently, the norms checker's usage is only limited to modelling obligations and prohibitions.

An annotation-based compliance verification framework for checking the compliance of business processes with legal norms is reported in [42]. The authors extend the business processes with semantic annotations through the specification of the effects of the atomic tasks and the obligations generated from their execution. The framework borrows AI techniques for reasoning about *actions* and *commitments*, and for the verification purpose model, checking techniques are employed using Answer Set Programming (ASP) and Coloured Petri Nets (CPNs). For the purpose of (semi)-automated verification, the norms are translated into LTL specifications, and these specifications are then fused onto business processes. The annotated business processes are then fed into a model checker, which returns a positive answer as its output if there is no violation, or a negative answer if a process model violates any specification. The main issue with their framework is that it provides structural compliance only; however, compliance is not about only how the activities are performed to achieve the enterprise goals but also about the tasks and the effects of the tasks on the execution of the business process. In addition, it is not clear whether the framework is able to capture all the obligation modalities of the norms.

[23] Unified modelling language: http://www.omg.org/spec/UML/.

[24] Object constraint language: http://www.omg.org/spec/OCL/.

In contrast, Hassan and Logrippo [86] proposed a compliance validation approach for representing and combining legal requirements. Their semi-automated approach detects the inconsistencies and violations using a three-step mechanism. In the first step, normative requirements are manually represented using first-order logic. Then, an autoanalyser generates the model, which is understood as an ontology representing structural and logic requirements. The generated model is then parsed to a logic analyser *Alloy* [94], which creates metafiles (also called *theme filters*) that are able to filter output based on the entity type and relations. Finally, the alloy analyser detects the compliance or produces a counter-example. This approach is similar to the ones in [9,191]. The major problem with this approach is the use of first-order logic, and it has been argued that first-order logic is not suitable to reason about the legal knowledge [87].

Gómez-López et al. [69] proposed a framework for the diagnostic and validation of business data constraints where the data flow variables and the stored data are analysed. The authors make use of constraints programming paradigm permitting the framework to automatically validate and diagnose the non-compliance issues even if the data constraints are not fully instantiated. Although the framework provides a good solution to address the compliance checking of data constraints, however, the scope is limited to constraints related to data only.

A run-time compliance governance approach in the service-oriented architecture (SOA) domain is discussed in [30]. In the first step of the approach, business process models and activities relevant to the monitoring and checking of the compliance requirements are identified by extended process engine (EPE) and passed to process engine output (PEO). As input, an Apache ODE engine then releases these process models for further compliance checking. In the second step, the business level events (policies) are identified and sent to the PEO engine. Once the processes and business level events have been identified, they are mapped and analysed by the business intelligence components to detect possible violations. The results of offline compliance monitoring and compliance checking are made available to the compliance dashboard. The problem with this approach is that the framework does not provide fully automated support for compliance governance by attaching the events and generating rules for compliance monitoring. Moreover, the compliance checking is done manually. In addition, there is no indication of how the system will deal with a detected policy violation, and no remedial steps are suggested. Furthermore, only the data aspect of service processes is considered in [30]'s work.

In the context of security compliance, [161] presented *Predictive Security Analysis at Run-Time* (PSA@R), a model-based approach for evaluating the security status of business processes at run-time. This approach integrates the formal process modelling with the simulation of process behaviour, to identify and predict violations of the security policies at run-time. The proposed modular approach operates with the control flow and security properties of the business processes as formalised views. Each view in the PSA@R system is formalised for the evaluation of security status of critical processes in the near future. For example, critical processes are formalised by a process view using asynchronous product automata (APA) [148], and the security requirements are formalised by a security view. The compliance of the security requirements is then monitored, and potential violations (in the near future) are predicted by comparing the predicted states with the security requirements using an on-the-fly prediction method. The prediction method employs an algorithm that computes accepting states referring to the *security critical states*. These critical states are then used to check the violations of the security requirements for computed states. If any deviations from the expected behaviour are detected, an alarm is raised for a decision support or reaction.

Moreover, Rieke et al. [161] have validated the effectiveness of their approach by using the security policies from the hydro-power generation domain, and this approach seems promising in terms of checking the compliance of security requirements. However, it is not clear how the security module models these security requirements or what types of security policies can be modelled. Furthermore, the proposed approach does not elaborate how the compliance of interconnected requirements can be verified. This is because the sensitive nature of the security domain means that most of the requirements have a complex interrelationship in order to ensure high degree of safety of the critical systems. More recently, Salnitri et al. [168] reported a run-time verification approach for checking the compliance of security policies in business processes using a SecBPMN-Q language an extension of BPM-Q [15]. Since BPMN-Q is LTL-based visual language, their proposed approach suffers from the same limitations as in [14,15,19,49].

6 Compliance auditing approaches

Compliance auditing is a retrospective reporting method that enterprises use to divulge their compliance. Usually, auditing is conducted by specially hired compliance auditors, who manually audit the huge trails of system-generated log files. Auditing the large amount of log files is a time-consuming task and prone to errors. The increased pressure from the regulatory bodies and possible penalties (for non-compliance) make this approach rather less attractive. However, with detailed information about processes increasingly available in high-quality event logs, auditors no longer have to rely on a small set of samples offline. A number of automated systems use process mining techniques and database technologies and can scan system logs to collect evidences to determine whether business processes are executed within the given set of rules.

6.1 Process mining-based approaches

Process mining [186] is a technique for extracting the process models from execution logs. This enables the enterprises to assess whether their processes were correctly executed by following prescribed set of rules. The extracted processes can be then audited to collect evidence of compliance.

A property formulation language and process mining tool that enable the verification of business process properties based on event logs are presented in [187]. The language is based on LTL and is tailored to event logs stored in the MXML format. The format used is tool independent of logged events and can be generated from audit trails, the transaction logs and other data sets. The language provides the support for the control-flow aspects of the business processes only, and other aspects such as resources, data and temporal aspects are not included. At a later date, applied process mining techniques in security and introduced an α-algorithm. In the first step, the proposed approach detects anomalous process executions in the mined workflow nets (WF-nets) for concrete cases. Then, the process conformance is checked by comparing process fragments with the identified WF-net. The α-algorithm discovers a net that models all acceptable behaviour of a process, using a given complete event log. A token game is then played to verify the conformance of the identified WF-net. In the token game, anomalous audit trails do not correspond to the possible firing sequences of identified WF-nets. Moreover, the token game also detects the point at which the audit trails diverges from the normal behaviour that allows a real-time verification of the audit trails.

Doganata and Curbera [45] discussed a semi-automatic auditing method for unmanaged processes. The method is based on the business provenance that sequentially records the collection of events for unmanaged processes. Similarly, van der Aalst et al. [188] introduced an automated auditing tool "*Auditing 2.0*" to provide support for compliance auditors using process mining techniques. The auditing framework provides support for considering the running process instances and compares them with models based on historic data or business rules. Arya et al. [11], on the other hand, used a similar approach to gain insights into the conformance of an operational process of a given process model. The authors implemented their approach in the Prom[25] framework. The approach uses current event logs (collected in real time) that carrying information about the activities being performed and the order in which they are performed. Later, they compared these simulated event logs against existing conformance technique based on Petri Nets.

The authors in [61] proposed an auditing framework to verify the compliance status of business processes against regulations. The verification is performed by the analysts who manually define the local context description of the accumulated effects. The effect accumulation process involves the derivation of a set of scenario labels at a point in the process [89]. Once all the effects are accumulated, the processes are annotated with the effects in the form of parsimonious effect annotations. Since two types of annotations effects can be derived, i.e. formal and informal, the proposed framework incorporates formal annotations. These annotations are modelled and parsed using CTL. Once the processes are annotated, they are transformed into directed graphs called *semantics process networks* (SPNs). These networks are used to evaluate the properties of the processes using an algorithm that exhaustively traverses all paths of the annotated processes to check any rule violations. Finally, the compliance results are reported whether the SPN satisfies the applicable compliance requirements or not.

While the framework uses annotations to validate the compliant behaviour of the business processes, an unlike semantic annotations, the formal representation of the effects in this framework cannot make any distinction between different types of obligations; it is not clear how the annotations of different types of obligation can be validated. Also, as the violation of obligation largely depends on the temporal conditions, i.e. deadlines, it is not possible to evaluate when an obligation is violated; it is only possible to determine whether an activity annotated with the rules description exists or it is absent from the graph. This restricts the feasibility of their framework to reason about and handle the violation conditions. A rather similar work based on the above framework is proposed in [89]; however, it is different from the effects annotation that are not only used to verify the compliance, but they are also used to reason about the process outcomes.

Ramezani et al. [158,159] reported a conformance checking approach based on Petri Net patterns and alignments. They created a repository of 55 control-flow-based compliance rules spanning over 15 distinct categories, including compliance rules for data, resources and organisational rules. The collected rules were formalised in terms of Petri Nets rather than logics. For conformance checking, they employed alignment techniques from [190] to analyse process compliance with the formalised Petri Net patterns. If the patterns are consistent with the compliance rules, the execution behaviour is consistent. However, if any deviant behaviour is observed, a violation of the rule is reported and the alignment shows the reason(s) for the deviations. The approach offers the solution for compliance checking of control-flow rules; however, this only provides the structural compliance of the rules. In addition, conformance checking of business processes against the business rules has different

[25] process mining: http://www.processmining.org.

specifications and properties than those in the legal domain. Thus, the proposed approach is not suitable for compliance checking of the normative requirements.

6.2 Database technology-based formal approaches

The database technology to assist compliance with the internal controls of SOX Act is used in [6]. The authors employs workflows and discovery-driven OLAP to verify compliance with internal controls and irregularities in the financial data, respectively. Initially, the internal processes are first modelled as workflows containing the required control activities, and the log of each workflow is stored in database tables. Policies are later enforced at run-time. This ensures that only routine transactions comply with the prescribed workflows, which serve as on-the-shelf compliant workflows. During the compliance auditing, these on-the-shelf workflows are reconstructed using correlation rules from the activity logs and are compared with the required workflows to determine whether transactions are compliant with internal controls.

Another enabling database technology for compliance auditing reported in literature is *Hippocratic database* (HDB). A HDB is an active enforcement system that controls the access and disclosure of private information in accordance with privacy policies and applicable laws. It enables the compliance auditing by back tracking past disclosures of information to detect suspicious information disclosures [118].

Johnson and Grandison [98] used HDB for auditing the compliance of data protection laws. The approach uses an HDB active enforcement architecture that operates as a middleware layer on the top of the database to enforce fine-grained policies concerning the disclosure of information. In the first step, the policy creation (HDB control) centre allows the creation of policies and then negotiates the preferences based on an input/output mechanism. Once policies and preference negotiations are formally defined, they are stored in an HDB logging system. Upon receiving an automatic audit query, the HDB logging system performs a statistical analysis of the query logs and generates a list of suspicious transactions, which are then combined into a single audit query. To confirm compliance, the output audit query contains the user identity, time, purpose, recipient and exact information about the policy and pertinent disclosure information. Some similar works using HDB technology for compliance auditing can be found in [5,99].

7 Hybrid approaches

Apart from the above-mentioned classification in compliance management, some hybrid approaches can also be found in the literature. These hybrid approaches claim to provide a full spectrum of compliance support. Moreover, some hybrid approaches, apart from the usual components of compliance checking and monitoring, incorporate new artefacts from a business strategy point of view. The rest of this section discusses some identified hybrid methods.

Ghanavati et al. [60] introduced a framework for tracking legal compliance in the health care domain. The framework demonstrates compliance tracking by defining and maintaining the correlation between the health care information custodian's policy models and business process models using goal-oriented language (GRL) and use case map (UCM) notations. The custodian policy models consists of a *source links* and *responsibility links*. The source links are relationship links between the legislative policy definitions and hospital UCM model elements. While responsibility links, on the other hand, establish relationships between the

UCM elements and GRL elements. These links are later checked for potential differences to see whether compliance requirements have been met. Any difference between what is implemented in the business process model and what is required by the privacy legislation (policy custodian model) is reported as rules violations.

Rifaut and Dubois [162] used goal-oriented techniques to present their compliance assessment framework for quality improvement based on ISO/IEC 15504 standard. However, the framework is in its evolutionary stage; the authors report future work in methodology and in tool support for the management of compliance requirements and their traceability to the process assess model (PAM) for assurance purposes. In contrast, [102] introduced a formal technique to elicit the regulatory requirements. The proposed technique represents the context of the policy rules, with case frames to semantically verify the regulations against the requirements. The technique uses words matched with a dictionary of policy regulation to detect the regulation sentences relevant to the requirements, such as structural similarity. Any dis-resemblance in the words format is detected and notified to the analyst as a violation.

In contrast, Sapkota et al. [169] discussed several semantic methodologies for automated regulatory compliance support using Semantic Web technologies. The proposed framework, RegCMatic, addresses the problem of automatic extraction, representation and reasoning upon regulatory information, and the generation of links between the internal compliance tasks and applicable regulations. Using various document formats such as PDF and HTML, the authors first extracted the regulations so that they could be converted into a machine-readable format. The list of extracted regulatory obligations was then processed using GATE [40], a text engineering platform, and the executable semantic rules were generated from the regulatory ontology. The authors implemented their proposed work using an industry case study that used Eudralex EU regulations.[26] Despite the nature of the regulatory requirements used in [169], the work seems to provide an effective method for addressing the compliance problem where the business rules are frequently changed. However, the extraction of regulatory information is not fully automated due to the document format used. In practice, regulatory bodies use different document formats, and extracting information from a variety of document formats is a challenging task and requires human intervention to adjust the document format as required.

The consistency of the regulatory rules is one of the issues (as reported in [13]) that cause frustration for the analysts. Inconsistency in the rules can lead to their misinterpretation and the incorrect modelling of the regulations. Jiang et al. [96,97] proposed a consistency and compliance checking framework (CCCF), using the Norm Nets (NNs) and Coloured Petri Nets (CPNs). The NN are used to formalise the regulatory rules and their relationship, whereas the CPN semantics implement the compliance-checker toolbox. The CCCF framework provides information on whether a set of regulations is consistent, and whether the business processes comply with the imposed regulations. Although the [97]'s framework is able to provide a reasonable degree of automated support for verifying the compliance to regulation, the transformation of the legal rules into NNs is primarily manually interpreted. In addition, from a business process perspective, the transformation of the model event sequences that model the behaviour of the agent (that is, trace generation) is also manual; this renders the proposed framework to be less effective. In contrast, the compliance checker proposed in [78] performs these tasks automatically. Another downside of this framework is that there is no mechanism for modelling the temporal constraints in CPNs; thus, the compliance to regulation with temporal modality cannot be verified.

[26] Eudralex, available at: http://ec.europa.eu/health/documents/eudralex/index_en.htm (retrieved: 25th October 2012).

8 Gaps and research challenges ahead

This section discusses and identifies the gaps and current research challenges faced by the business process compliance domain based on the classification and evaluation criteria (cf. Sect. 2.4) and briefly describe each challenge.

Formalisation of Norms: The research on business process compliance, in general, appreciates the significance of formalisation of the legal requirements as we observed that existing approaches (e.g. [13,65,75,126]) incorporate a varied set of formalisms for modelling and representing the legal norms for their automated compliance checking. Essentially, the effectiveness of a compliance approach—apart from other aspects (e.g. complexity, level of formality), largely depends on the expressive power of the formal language whether it can properly represent different types of norms. Failure to represent the meanings of norms in an intuitive way can severely impede the reliability of the compliance results although a great deal of formal languages (e.g. [14,52,93]) to represent legal norms exists; however, we noted that these languages have their own shortcomings, e.g. some formalisms are unable to get the effects of the tasks (event calculus), while others are not conceptual relative to the legal domain (temporal logics), and some do not provide temporal operators (first-order logic). Accordingly, some do not accurately represent the legal norms and produce results that lead to paradoxes giving inaccurate reasoning results for the formalisation of specific types of the legal norms.

Table 3 depicts the strengths and weaknesses of some existing compliance approaches to represent different types of norms attributed to paradoxes. The symbol "+" indicates the framework is able to provide modelling support for a specific type of a legal norm, and the symbol "−" indicates the framework is unable to model the norm type.

From Table 3, it is clear that majority of the compliance frameworks using a varied set of formalisms are able to provide modelling and reasoning support only for a fraction of norms types. For example, PENELOPE language [65] is only able to support obligations and permissions, while other norm types are not supported. PCL [76], on the other hand, is able to represent all types of legal norms using modal defeasible logic and deontic logic of violations. However, this does not mean PCL can solve all the formalisation problems that we have in reality. For example, PCL is not able to represent "*nested norms*" (norms within a norm) as it is not a feature semantically supported in the language [117].

In contrast, DECLARE [130], BPMN-Q [15] and COMPAS [51] are temporal logic-based frameworks and only able to address norms related to structural compliance. These frameworks cannot represent various types of obligations, violations and their compensations. DECLARE framework, on the other hand, can represent achievement obligations and prohibitions, while BPMN-Q is able to represent achievement and prohibitions onlyand COMPAS is not able to represent permissions and other types of norms. In the same vein, other frameworks (e.g. [61,93,104,126]) also lack modelling support for all types of legal norms resulting in paradoxes that present a huge gap. Resolving the paradoxes of representing legal norms is one of the major challenges without which is would not be possible get the desired compliance checking results.

Accordingly, regulations continuously change, i.e. new rules might be introduced, while older might be removed or updated—the changes in the regulations might prescribe new types of norms that might introduce new complexities. Then, the question would be *how to represent the new types of norms when there need be?* This introduces the researchers with a continuous dilemma of designing new modelling languages that are expressive, scalable and flexible to faithfully represents ever-changing compliance requirements.

Table 3 Snapshot of shortcomings of existing approaches to formalisation of norm types

Framework	Types of obligations									
	Achievement	Preemptive	Non-preemptive	Maintenance	Punctual	Perdurant	Compensation	Permission	Prohibition	Violation
PENELOPE [65]	+	–	–	–	–	–	–	+	–	–
PCL [76]	+	+	+	+	+	+	+	+	+	+
DECLARE [130]	+	–	–	–	–	–	–	–	+	–
BPMN-Q [15]	+	–	–	–	–	–	–	–	+	+
SEAFLOWS [126]	+	–	–	–	–	–	–	–	+	+
COMPAS [51]	+	–	–	+	–	–	+	–	+	+
AUDITING BPC [61]	+	–	–	–	–	–	–	–	–	–

Norms Extraction and Elicitation: Generally, the source of legal norms that organisations have to comply with is normative documents which are generally written in natural language. It is rare to have some kind of structured representation of rules in such documents. Moreover, these also include complex sentences, legal jargons and technical terms. For an accurate and effective formalisation of legal norms, it is imperative to properly extract rules from the legal texts. However, mostly this task is manual leaving thus, the high chances of errors, misinterpretations and conflicts/redundancies due to human involvement. This is because analysts might interpret technical terms differently, important conditions in the rules might be overlooked or wrongly confer the rights or obligations to agents [81,199]. This can adversely affect the formalisation of norms as wrong extraction might lead to wrong representations. We observed that current research (e.g. [20,41,109]) has exploited *natural language processing* (NLP) and *machine learning*-based approaches (e.g. [33,43]) to automate the norms extraction task from a variety of perspectives. For example, some to extract the document structure, while others classifying law paragraph according to the regulatory contents (e.g. [29]) and distinguishing terms to be part of the rules (e.g. [59])—each claiming varied degree of success. However, the experimental comparison with performances claim made in these studies is difficult due to the fact that no data sets nor systems exist to evaluate them. In addition, in our view, norms extraction process is far deeper than just extracting the document structure, and classifying the terms but identify and extract deontic component of rules, and correctly assign the terms to the antecedent and the consequent of the rules. Also, extract the co-reference links that are present in the legal documents, align the terms that are used in the legal text and the terms that we want to use in the rule providing, thus, a unified representation of the norms for further formalisation. We strongly believe that the proper extraction of norms is an ongoing challenge and does not seem to be fully automated in near future. However, we also believe that—due to the complexity of the legal texts and time required to manually extract norms, (even partially) automating this task would be beneficial.

Multi-Jurisdictional Requirements: As today's organisations operate across jurisdictions such as regional, national or international locations, legal documents can be interpreted differently across geographies. Meaning that at one location, a rule might be interpreted in one way, while at the other location, it is interpreted in another way. This can lead to overlaps and inconsistencies in the interpretations of norms. Although this has received some attention (e.g. [146]) to automate the analysis of multi-jurisdictional norms—existing approaches are only partially automated. Further work is needed, if we want to fully automate the analysis in order to detect overlaps and conflicts among different interpretations for a unified interpretation of norms for further formalisation.

Control-Flow Structure: Compliance is not only about the tasks that an organisation has to perform to achieve its business goals, it is also on their effects (i.e. how activities in the tasks change the environment in which they operate), and the artefacts produced by the tasks (e.g. the data resulting executing a task or modified by the tasks) and which resources are used. To the best of our knowledge, most of the existing approaches centre around structural (control-flow) compliance of business processes defining the control-flow patterns (e.g. [52,110], checking compliance of control-flow-based rules including parallel process control flow (e.g. [159]), dependency between data and control flow (e.g. [182]). However, from a compliance perspective, handling loops in business processes presents a challenge. Essentially, the presence of loops might cause reachability, possibility of live locks and redundancy problems making, thus, the compliance checking of even a small fragment of a business process very hard. Albeit attempts have been made deal with the loops (e.g. [91,137]) and self-loops (e.g. [53]), the results are far from being satisfactory. This is because mostly handling the loops is undecidable, but with some restrictions it can be decidable yet complete

in PSPACE [28], coNP and NP [36]. This is an open and ongoing challenge; further research is needed to design approaches that are able to handle control-flow rules with loops.

Integrating Rules with Processes: In order to verify the compliant status of business process, existing studies (e.g. [42,86,167]) proposed to enrich processes with the control objectives by means of semantic annotations (cf. Sect. 2.4.8). Enriching processes with semantic annotations increases the understanding of the interaction between business process specifications and compliance controls specifications for the involved stakeholders (i.e. compliance officers and process owners). On the one hand, embedding the compliance rules into business processes makes the compliance checking more transparent to the stakeholders. However, it makes the management of large compliance repositories a very difficult task, on the other hand. This is explained by the fact that compliance rules frequently changed, removed or updated. The changes in the compliance rules increase the risk of size and complexity of the process models and might cause maintainability problems.

In contrast, there are some proposals (e.g. [158,159,189]) favouring to separate the compliance concern from business processes by capturing each aspect of a compliance requirement in a separate rule based on a process vocabulary. For this, the use of a common business vocabulary based upon the control flow, agent and data- related primitives to specify the compliance rules is discussed in [189]. These primitives cover a full spectrum of business processes aspects and can be used to formulate the compliance rules. However, separating the compliance concern from business processes, and the use of common process vocabulary raises the question *how to enforce the compliance requirements into the tasks of a business process* even if a common process vocabulary is used. This would make it difficult to trace the enforcement of compliance requirements into business operations. Hence, the question is: *how feasible is to use a common vocabulary to integrate the compliance rules* because processes and legal rules are two separate concerns. Further research may empirical analyse the effectiveness of these approaches to find consensus as to which approach is better because both the approaches present their own challenges to link the legal rules with business processes for their compliance checking.

Handling Violations: Our survey also points to the need for fully automated approaches for effective management of the compliance violations (cf. Sect. 2.4.9). Several proposals aiming at classifying and characterising (e.g. [16]), detecting (e.g. [50,159]), analysis root causes (e.g. [50]) and explaining (e.g. [17]) the violations have been made. Mostly, they require human intervention. Also, they are only able to (semi-)automatically resolve the violations of structured process with simple compliance rules. However, the violation handling is a very complex problem and ongoing challenge to provide fully automated and dynamic approaches to detect and remedy the compliance violations—especially for unstructured processes and complex compliance rules.

Dealing with Model Evolution: While business process and legal documents continuously change, this can significantly increase the size and complexity of the business processes and compliance repositories—thus, maintainability becomes an apparent concern (cf. Sect. 2.4.10). A very limited research exist in literature on *how to effectively manage the maintainability problem?* Essentially, the changes in the compliance rules give rise to the questions: *should the whole system be affected* when only cosmetic changes occurs in the fragment of the business process or the rules document? Another question is *how the change in the rules can be propagated into business process whenever a legal document changes?* Small changes to the system should remain small and must be treated locally without affecting the whole system. In such situations where business processes and compliance rules continuously change, *isomorphic approach* [24] can be benefited for effective management of the model evolution and of the legal rules. A true isomorphic approach can provide the

stakeholders confidence to not only trace back the changes from the source of change through the final representation, but also it allows to keep the changes localised, and no further parts of the system will be affected [81]. There is huge room for further investigations on *how to effectively manage the changes should the sources of legal documents and business processes evolve?*

Complexity and Performance: Technical and structural complexity of the compliance rules is another factor as ascertaining the meanings of compliance rules is less than straightforward. With legal jargons, unintended and inconsistent interaction between different provisions, and different interpretations of the compliance rules makes the representation and subsequent compliance checking rather a challenging task. The complexity of the compliance checking is NP-complete [36][27]—that is, even with a small piece of legislation checking the compliance of structured business processes might not be possible. Thus, checking the compliance of complex business processes or large sets of legal rules can be unachievable. In addition, expressiveness of the formalism, the number of compliance rules is other factors that add up to computational complexity. However, some limited evidence from literature [85] suggests that the compliance checking problem may be intractable due the computational complexity of the formal language. But from the application perspective, the problem seems feasible as the compliance checking of reasonably large processes can still be checked within an acceptable amount of time (potentially in minutes). Yet, there is still a huge vacuum in this area, and addressing the issues related to computational complexity—not straightforward though, is inevitable.

In addition, the complexity of the compliance checking can significantly affect the system performance—current research interestingly lacks this aspect. Since, compliance has already been proved to be NP-complete, further empirical research is needed to analyse the effects of implementing the control objectives and their consequent compliance checking.

Usability and Generalisability: Mostly logic-based formalisms are complex and often frustrate non-technical users. For business analysts and legal experts who possess less technical knowledge of formalisms—the use, comprehensions and understanding the technical aspects of a formal language is a major concern. We observed that several attempts have been made (e.g. [14,15,49]) offering user-friendly graphical/pattern-based languages surmounting the need to understand the complex logic formulas for the non-technical users. However, this raises another problem *whether the proposed pattern (or notation) can really enhance the understandability and usability of the formalism?* Indeed, a visual notation that refers a modelling pattern can be easily understood—however, what about the pattern itself? Essentially, there is a trade-off between the visual notation and the graphical pattern. The fundamental question is *does the pattern provides a good balance between the visual notation and the semantic concept it refers in order to give a clear semiotic representation?* Meaning that, is there a balance between the pattern, the graphical symbol and the corresponding concepts? Moreover, this also raises a question on whether a pattern referring complex logic formulas written in one logic can effectively represent the formulas written in other logics? Current literature lacks empirical evidence on the usability and generalisability of the graphical languages leaving thus, the need for further investigations.

To sum up, the above-discussed gaps show that business process compliance is still a challenging area of research. We do not claim, however, this set of challenges to be exhaustive and was extracted from the surveyed literature. Since process compliance is an interdisciplinary

[27] Colombo Tosatto and colleagues [36] formally proved that checking whether a business process is partially compliant is an NP-complete, and the complexity of checking whether a business process is either fully compliant or not compliant is coNP-complete problem.

research—it is expected that new challenges will continue to emerge from an interdisciplinary perspective.

9 Discussion

Further on the previous section, in what follows we are going to discuss the key findings compared with existing surveys, potential impact, outlook on the future work and some limitations of this survey.

9.1 Comparison and key findings

Existing literature on business process compliance management points that it is an interdisciplinary and challenging topic having deep roots in the business process domain and the compliance domain. We identified 13 (non-)functional dimensions such as control flow, data, resources and time, formalisation and integration of rules, and model evolution and divide the literature along 3 compliance management strategies (e.g. design-time, run-time and auditing) and hybrid approaches. Despite the fact the state of the art provides a rich set of approaches and frameworks to address the compliance problem, we also identified the limitations of these approaches and further research challenges.

To best of our knowledge, the presented survey is comprehensive and provides a holistic view of the state of the affairs on business process compliance domain compared to previous similar surveys which are centred-around to specific aspects of the compliance problem limiting their scope. For example, [31] surveys the rule-based system specifications modelling approaches in the context of semantic web, while [163] reports on the results of research project to accumulate understanding on the relationship between risk management and internal controls to guide the research agenda in business process risk management, compliance and internal controls. Otto and Anton [153] studied existing compliance approaches for extracting the required information for modelling compliance requirements. In the context of COMPAS project [37], the authors provide an overview of the state of the art in the compliance languages with emphasis on languages for regulatory and legislative provisions. Their survey identifies various aspects of compliance, discovery, modelling and reporting of compliance.

In contrast, Abdullah et al. [1] studied the challenges faced by the industry and available solutions for addressing the compliance problem. Their survey focuses on the understanding and lack of passion to address the compliance problem in the industry sectors and shortcomings of the available compliance solutions as well as the complexity of the compliance problem. A rather similar literature survey on the practice of regulations analysis and the approaches that aims to achieve and maintain regulations compliance from an Information System and eGovernment Services perspective has been reported in [183]. The survey in [34], on the other hand, focuses on how modelling languages are used to align the compliance requirements on business processes. Their work is somewhat similar to the survey presented in [31]. However, this work focuses on the security policies, trust management in the context of privacy and inter-organisational compliance requirements modelling. In contrast, Elgammal et al. [51] survey formal languages for modelling business process compliance requirements with the focus on design-time compliance and highlights the capabilities and limitations of the surveyed languages chosen from temporal and deontic families of logics. Their survey is somewhat similar to the work of [153] and [37] where authors survey existing compliance approaches for extracting information to representing normative requirements.

An evaluation of functional and non-functional capabilities of compliance management frameworks in the context of business process compliance has been reported in [46]. Their evaluation is based on three-point evaluation criteria, namely compliance management solutions, methodology and architecture of the evaluated compliance solutions. The authors evaluate various functional areas of the regulatory compliance from a business process management perspective, e.g. the strategy model and the business process model. Becker et al. [23], on the other hand, present a literature study based on the generalisability and applicability of the business process compliance frameworks and only cover the aspect of the implementation results of the surveyed frameworks.

Fellmann and Zasada [55] survey the dominating trends and issues in business process compliance over four dimensions, namely variables of general business process modelling (for example, information, location, resources), temporal aspects of process modelling, distinction between the approaches based on the formality, that is, whether the approach is a verification or a validation approach.

Suriadi et al. [179], on the other hand, analyse business process management literature from a risk management perspective. These include the risk such as regulatory non-compliance, financial frauds, natural disasters and data leakages to name but a few, within the business processes. Contrastingly, [119] systematically investigates the holistic view of the security in process-aware information systems along process life cycle and the type of actions. However, both these surveys fully exclude business process regulatory compliance—in particular, approaches to representing and checking the compliance of regulatory frameworks thus have a different scope.

More recently, [82] examined whether existing CMFs are able to provide modelling and reasoning support for various types of normative requirements. They primarily examined the conceptual foundations of the selected CMFs under pre-defined evaluation criteria and the obligation modalities representing various classes of the normative requirements. However, their work is heavily concentrated onto one aspect (i.e. norms modelling and reasoning support) of the CMFs restricting, thus, the focus of their study.

A state of the compliance report by Pricewaterhousecoopers (PWC) [27] surveys the corporate compliance officers to give the benchmarking data to understand the common industry practices. Their survey also discusses the current and future challenges faced by the compliance officers, and how they can expand their roles to actively contribute to the enterprise strategy to align the compliance requirements. In particular, the survey discusses the growing need for continuous monitor and testing of the system around compliance-related issues and underpins the significance of increased understanding of the technology solutions in order to demonstrate the ongoing effectiveness of compliance activities. In addition, the challenges regarding reacting in real-time to changes or possible threats have been highlighted. Goedertier et al. [66] evaluate different approaches, principles to declarative process modelling ranging from imperative models to representing declarative modelling approaches. These approaches differ from the business concern, state space and the constraints types they are able to model and the modelling and reasoning framework they use—yet their objective remains the same. This survey is restricted to only declarative process modelling paradigms and does not consider the compliance concerns as is done in [156].

Each of these studies have been conducted with a specific focus in mind as illustrated in Table 4—however; the presented survey provides a detailed understanding on the state of the affairs covering a range of compliance dimensions, which have not been left out in existing surveys. For example, comparative evaluation of [46] uses a 3-point evaluation criteria to evaluate (non-)functional capabilities of solution components of 32 compliance frame-

Table 4 Summary of existing surveys and their focus

Survey By	Focus and scope
Hashmi and Governatori [82]	Examine norms modelling constructs of the selected CMFs from a modelling and reasoning support perspective
Goedertier et al. [66]	Survey the principles, techniques and languages for modelling declarative business process modelling
Bernstein and Falcione [27]	Survey of current and future challenges faced by the compliance officers for compliance-related activities within organisations
Ly et al. [127,128]	Survey the compliance monitoring approaches using monitoring functionalities of selected frameworks
Fellmann and Zasada [55]	Survey on recent trends in business process compliance research
Suriadi et al. [179]	Analyse risk-aware business process management from risk manifestation within business processes
Leitner and Rinderle-Ma [119]	Holistically survey the security in process-aware information systems including security control in the phases of process life cycle
Abdullah et al. [1]	Survey the issues and industry challenges faced by the current IS research for regulatory compliance
Becker et al. [23]	Evaluate the generalisability and applicability of model-based compliance checking approaches
El Kharbili [46]	Survey functional and non-functional capabilities and solution components of compliance frameworks
Elgammal et al. [51]	Survey of the modelling languages for modelling the legal norms for design-time compliance verification
Turki and Marija [183]	Survey approaches that aim to achieve and maintain regulatory compliance from IS perspective
COMPAS project [37]	Survey on compliance modelling languages for regulatory and legislative provisions
Otto and Anton [153]	Survey approaches to modelling norms and key legal concepts from regulatory documents
Rikhardsson et al. [163]	Accumulates understanding on the relationship between risk management, compliance and internal controls
Bonatti et al. [31]	Survey rule-based system specifications modelling approaches in the context of semantic web

works and approaches. In contrast, we review 79 approaches scattered around compliance management strategies.

The review by [127,128] surveys the compliance monitoring approaches for business processes including process patterns, enabling technologies and related techniques (e.g. conformance checking) and domain-specific approaches only; thus, it has a limited scope. We go beyond the compliance monitoring approaches and include approaches related to design-time and auditing compliance management strategies. The review in [55] examined the trends regarding the scope and life cycle phases of 84 compliance approaches, whereas we examined 79 approaches in this survey. The reason for the difference in number of approaches could be attributed to the scope of the survey as well as the data collection and synthesis methodology to extract the literature. We manually extracted the papers—however, ours includes

the papers that proposed approaches using a mix of compliance management strategies (cf. Sect. 7).

In a nutshell, our survey provides a holistic view of the business process compliance domain, as none of these studies accumulate a detailed understanding on the state of the affairs of the compliance domain and whether existing approaches cover a full spectrum of the compliance problem, and what the challenges are that need to be investigated?

9.2 Potential impact

The main result of the survey is a classification of existing literature along the dimensions of the compliance problem (see Fig. 3), and it shows that existing approaches mostly focus on the run-time and design-time compliance management with a varied degree of concentration on the compliance checking strategy, process aspects and formalisation of norms. Also, it highlights several topics (cf. Sect. 8) that need exigent attention in the ongoing as well as future research development across the life cycle of the compliance problem. Essentially, a diverse range of research community can be benefited from the findings of this survey—especially, to address the most challenging and continuous dilemma of developing new modelling languages that are expressive, scalable and flexible to faithfully represent compliance requirements and approaches to checking their compliance.

9.3 Limitations of the survey

We made best efforts to make this survey rigorous and complete within the scope of the research questions on business process compliance management. A large amount of papers published between 2000 and 2015 were collected, and consequently, 79 papers were selected in this survey. However, because of interdisciplinary nature of the compliance problem, we do not claim that we have included all studies in this survey. It is possible that some papers published in other related domains that must have not been included. Also, it is possible that there are highly relevant papers published in other languages that must have been left out, since we only considered papers published in English language.

The publication biasness refers to reporting of positive studies than the negative ones in the survey [2,107]. This is considered as a risk, and it could effect the reliability of the survey results. A proper care was taken while reporting on the strengths and weakness of the selected papers. However, grey literature, e.g. working papers, unpublished tech reports or non-peer-reviewed papers that might have the relevance with this survey, was not considered. Hence, it is highly likely that some approaches or methods might have been left out even in the presence of pre-defined papers selection criteria.

10 Conclusions

In this paper, we presented a holistic view of the available literature on business process compliance management. The quintessence of this survey was to identify the areas of improvements or new developments in the business process compliance management through a systematic analysis of the literature. For this purpose, this paper formulated four research questions: (i) *which are the dimensions of regulatory compliance problem*; (ii) *which are generic strategies for addressing the compliance problem*; (iii) *whether existing approaches fully cover dimensions of the compliance problem*; and (iv) *what are the challenges that need to be addressed*. To address these questions, we adopted a structured methodology through

which we collected and assessed a large corpus of literature published between 2000 and 2015 of which 79 papers related to the research questions and derived criteria covering a wide range of features from the spectrum of compliance management were reviewed. Each selected source was also scrutinised for its relevance and quality of contents to ensure the reliability of the presented review.

RQ-1 was addressed in Sect. 2.4 when we discussed various compliance dimensions identified from literature establishing the requirements that a compliance product should be able to provide support for. Moreover, these dimensions provided the basis for evaluating various features of a compliance management approach. Section 3 addresses RQ-2 by investigating what strategies have been proposed to create compliance management frameworks considering how technology might help enterprises to deal with the compliance problem in order to verify the compliance of business processes against governing regulations. In order to address RQ-3, Sects. 4 through 7 study the strengths and weaknesses of design-time, run-time, auditing and hybrid approaches, respectively, whether these approaches fully cover all dimensions of the compliance problem (identified in Sect. 2.4). Finally, for RQ-4, Section 8 identifies the limitations of the surveyed frameworks and outlines the challenges that require urgent attention.

The results of our survey reveal that mostly compliance management approaches centre around three distinct categories of approaches, i.e. design-time 28%, run-time 32%, auditing 10% with a varied degree of concentration on the compliance checking strategy, process aspects and formalisation of norms. Besides, organisational and internal control objective-based frameworks (21%) and hybrid approaches (9%) also make a good portion of the all surveyed approaches.

In addition, the survey also reveals that although a great deal of work addressing the compliance problem from a variety of perspectives exist, none of the surveyed frameworks fully cover all dimensions of the compliance problem as a number of challenges still remain (cf. Sect. 8). This observation, however, should not be attributed to the shortcomings of the surveyed approaches. Rather, they might be due to the issues that might have not been addressed by the current approaches or technically complex or human-related challenges that need to be addressed. In particular, the need to develop automated techniques for extracting norms from legal documents, and formal languages expressive enough to model various types of legal norms, and norms pertaining data and resources. Also, there is need for fully automated techniques that are scalable and computationally efficient. Finally, there is a huge room for investigation on the issue related to handling maintainability of ever-changing business processes and legal norms.

In future, we plan to address some of the issues highlighted in this paper—in particular, to extend the reasoning capabilities of formal languages such as linear temporal logic with the aspect of normativity for representing various types of normative requirements. Further on, we plan to investigate technical and structural complexity of compliance rules especially the complexity of different interpretations, inconsistencies between the rules, and the effects of evolution on legal rules as well as process models.

Acknowledgements We thank Régis Riveret for his valuable discussions and suggestions and anonymous reviewers for their many valuable comments and suggestions.

References

1. Abdullah NS, Sadiq S, Indulska M (2010) Emerging challenges in information systems research for regulatory compliance management. In: Proceedings of CAiSE'10. Springer, pp 251–265
2. Achimugu P, Selamat A, Ibrahim R, Mahrin MN (2014) A systematic literature review of software requirements prioritization research. Inf Softw Technol 56(6):568–585
3. Ågotnes T, van der Hoek W, Rodríguez-Aguilar JA, Sierra C, Wooldridge M (2007) On the Logic of Normative Systems. In: Proceedings of the 20th international joint conference on artificial intelligence. AAAI Press, Menlo Park, pp 1175–1180
4. Ågotnes T, Van der Hoek W, Wooldridge M (2010) Robust normative systems and a logic of norm compliance. J Log 18(1):4–30
5. Agrawal R, Bayardo R, Faloutsos C, Kiernan J, Rantzau R, Srikant R (2004) Auditing compliance with a hippocratic database. In: Proceedings of the thirtieth international conference on very large data bases, vol 30, VLDB Endowment, VLDB '04, pp 516–527
6. Agrawal R, Johnson C, Kiernan J, Leymann F (2006) Taming compliance with Sarbanes–Oxley internal controls using database technology. In: Proceedings of the 22nd international IEEE conference on data engineering, p 92
7. Ahmed A, Sakr S (2010) Querying graph-based repositories of business process models. In: DASFAA workshops, pp 33–44
8. Alberti M, Chesani F, Gavanelli M, Lamma E, Mello P, Montali M, Torroni P (2007) Expressing and verifying business contracts with abductive. In: Boella G, van der Torre L, Verhagen H (eds) Normative multi-agent systems, Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI), Schloss Dagstuhl, Germany, Dagstuhl, Germany, No. 07122 in Dagstuhl seminar proceedings
9. Antón AI, Bertino E, Li N, Yu T (2007) A roadmap for comprehensive online privacy policy management. Commun ACM 50(7):109–116
10. Arbab F, Kokash N, Meng S (2008) Towards using REO for compliance-aware business process modeling. In: Margaria T, Steffen B (eds) ISoLA'08, vol 17. Springer, Berlin, pp 108–123
11. Arya A, van Dongen B, van der Aalst W (2010) Towards robust conformance checking. In: BPM workshops'10, pp 122–133
12. Ashby S (2008) Operational risk: lessons from non-financial organisations. J Risk Manag Financ Inst 1:406–415
13. Awad A (2010) A compliance management framework for business process models. Ph.D. thesis, Hasso Plattner Institut, Potsdam University, Germany
14. Awad A, Weske M (2009) Visualisation of compliance violations in business process models. In: 5th Workshop on business process intelligence, vol 9, pp 182–193
15. Awad A, Decker G, Weske M (2008) Efficient compliance checking using BPMN-Q and temporal logic. In: Proceedings of the 6th international conference on business process management (BPM 2008). Springer, Milano, pp 326–341
16. Awad A, Smirnov S, Weske M (2009) Towards resolving compliance violations in business process models. In: Sadiq S, Indulska M, zur Muehlen M, Dubois E, Johannesson P (eds) Proceedings of the 2nd international workshop on governance risk and compliance GRCIS, pp 18–33
17. Awad A, Weidlich M, Weske M (2009) Specification, verification and explanation of violation for data aware compliance rules. In: Baresi L, Chi CH, Suzuki J (eds) Proceedings of the 7th international joint conference on service-oriented computing (ICSOC-Service Wave 2009). Springer, Stockholm, pp 500–515
18. Bai X, Liu Y, Wang L, Tsai WT, Zhong P (2009) Model-based monitoring and policy enforcement of services. In: Proceedings of the 2009 world conference on services, vol I, pp 789 –796
19. Barnawi A, Awad A, Elgammal A, Elshawi R, Almalaise A, Sakr S (2016) An anti-pattern-based runtime business process compliance monitoring framework. Int J Adv Comput Sci Appl (IJACSA) 7(2):551–572
20. Bartolini R, Lenci A, Montemagni S, Pirrelli V, Soria C (2004) Semantic mark-up of Italian legal texts through NLP-based techniques. In: Proceedings of the fourth international conference on language resources and evaluation (LREC 2004), Lisbon, Portugal
21. BCBS (2013) Basel III: The liquidity coverage ratio and liquidity risk monitoring tools. http://www.bis.org/publ/bcbs238.pdf
22. Beach T, Rezgui Y, Li H, Kasim T (2015) A rule-based semantic approach for automated regulatory compliance in the construction sector. Expert Syst Appl 42(12):5219–5231
23. Becker J, Delfmann P, Eggert M, Schwittay S (2012) Generalizability and applicability of model-based business process compliance-checking approaches–a state-of-the-art analysis and research roadmap. BuR Bus Res J 5(2):221–247

24. Bench-Capon T, Gordon TF (2009) Isomorphism and argumentation. In: Proceedings of the 12th international conference on artificial intelligence and law, ACM, NY, USA. ICAIL'09, pp 11–20
25. Bench-Capon TJM, Coenen FP (1992) Isomorphism and legal knowledge based systems. Artif Intell Law 1(1):65–86
26. Bérard B, Bidoit M, Finkel A, Laroussinie F, Petit A, Petrucci L, Schnoebelen P (2001) System and software verification–model checking techniques and tools. Springer, Berlin
27. Bernstein S, Falcione A (2015) Moving beyond the baseline Leveraging the compliance function to gain a competitive edge: state of compliance survey 2015. Survey report, Pricewaterhousecoopers
28. Bhattacharya K, Gerede C, Hull R, Liu R, Su J (2007) Towards formal analysis of artifact-centric business process models. In: Alonso G, Dadam P, Rosemann M (eds) Proceedings of the 5th international conference on business process management (BPM 2007). Springer, Berlin, pp 288–304
29. Biagioli C, Francesconi E, Passerini A, Montemagni S, Soria C (2005) Automatic semantics extraction in law documents. In: Proceedings of the 10th international conference on artificial intelligence and law, ACM, New York, NY, USA, ICAIL'05, pp 133–140
30. Birukou A, D'Andrea V, Leymann F, Serafinski J, Silveira P, Strauch S, Tluczek M (2010) An integrated solution for runtime compliance Governance in SOA. In: Proceeding of international conference on service-oriented computing (ICSOC), pp 122–136
31. Bonatti PA, Shahmehri N, Duma C, Olmedilla D, Nejdl W, Baldoni M, Baroglio C, Martelli A, Coraggio P, Antoniou G, Peer J, Fuchs NE (2004) Rule-based policy specification: state of the art and future work. Rewerse project report-i2-d1, Universitá di Napoli Fedrecio II
32. Bonazzi R, Pigneur Y (2009) Compliance management in multi-actor contexts. In: Proceedings of international workshop on governance, risk and compliance (GRCIS), An ancillary meeting of CAISE
33. Brighi R, Palmirani M (2009) Legal text analysis of the modification provisions: a pattern oriented approach. In: Proceedings of the 12th international conference on artificial intelligence and law (ICAIL'09), ACM, New York, NY, USA, pp 238–239
34. Cabanillas C, Resinas M, Ruiz-Cortés A (2010) On the identification of data-related compliance problems in business processes. In: Jornadas Científico-Técnicas En Servicios Web Y SOA (JSWEB'10), Valencia, España, vol 1, pp 89–102
35. COBIT (2007) Control objectives for information related technology—COBIT 4.1. http://www.isaca.org/Knowledge-Center/cobit/Pages/Downloads.aspx
36. Colombo Tosatto S, Governatori G, Kelsen P (2015) Business process regulatory compliance is hard. IEEE Trans Serv Comput 8(6):958–970
37. COMPAS-Project (2008) D2.1 state-of-the-art in the field of compliance languages—compliance-driven models, languages, and architectures for services. Deliverable D2.1v1.0, Tilburg University, The Netherlands
38. Cooper HM (1988) Organizing knowledge syntheses: a taxonomy of literature reviews. Knowl Soc 1(1):104–126
39. COSO (1994) Internal control–integrated framework. http://www.coso.org/
40. Cunningham H, Maynard D, Tablan V, Ursu C, Bontcheva K (2001) Developing language processing components with GATE: a user guide. https://gate.ac.uk/sale/tao/tao.pdf
41. d Araujo DA, Rigo SJ, Muller C, Chishman R (2013) Automatic information extraction from texts with inference and linguistic knowledge acquisition rules. In: 2013 IEEE/WIC/ACM international joint conferences on web intelligence (WI) and intelligent agent technologies (IAT), vol 3, pp 151–154
42. D'Aprile D, Giordano L, Gliozzi V, Martelli A, Pozzato G, Theseider Dupré D (2010) Verifying business process compliance by reasoning about actions. In: Dix J, Leite Ja, Governatori G, Jamroga W (eds) Proceeding of the 11th international workshop on computational logic in multi-agent systems (CLIMA XI). Springer, Berlin, pp 99–116
43. de Maat E, Winkels R (2010) Suggesting model fragments for sentences in Dutch Laws. In: Proceedings of legal ontologies and artificial intelligence techniques, pp 19–28
44. de Moura Araujo B, Schmitz EA, Correa AL, Alencar AJ (2010) A method for validating the compliance of business processes to business rules. In: Proceedings of SAC'10, ACM, pp 145–149
45. Doganata Y, Curbera F (2009) Effect of using automated auditing tools on detecting compliance failures in unmanaged processes. In: Proceedings of the 7th international conference on business process management (BPM 2009), Ulm, Germany, pp 310–326
46. El Kharbili M (2012) Business process regulatory compliance management solution frameworks: a comparative evaluation. In: Ghose A, Ferrarotti F (eds) Proceedings of the 8th Asia-Pacific Conference on Conceptual Modelling (APCCM 2012). ACS, Inc., Melbourne, Australia, pp 23–32
47. El Kharbili M, Stein S (2008) Policy-based semantic compliance checking for business process management. MobIS workshops, CEUR workshops 420:178–192

48. El Kharbili M, Stein S, Markovic I, Pulvermüller E (2008) Towards a framework for semantic business process compliance management. Banking 08(i):1–15
49. Elgammal A (2012) Towards a comprehensive framewcompliacbusiness process compliance. Ph.D. thesis, Tiburg Universtity
50. Elgammal A, Türetken O, van den Heuvel WJ, Papazoglou MP (2010) Root-cause analysis of design-time compliance violations on the basis of property patterns. In: Proceedings of the 8th international conference on service-oriented computing (ICSOC 2010), San Francisco, CA, USA, pp 17–31
51. Elgammal A, Turetken O, van den Heuvel WJ, Papazoglou M (2011) On the formal specification of regulatory compliance: a comparative analysis. In: Proceedings of ICSOC'10, pp 27–38
52. Elgammal A, Turetken O, van den Heuvel WJ, Papazoglou M (2016) Formalizing and applying compliance patterns for business process compliance. Softw Syst Model 15(1):119–146
53. Eshuis R (2006) Symbolic model checking of UML activity diagrams. ACM Trans Softw Eng Methodol 15(1):1–38
54. Evans GP (2014) Managing risk with an end-to-end process view: adopting a process-based approach to risk management. BPTrends article. https://www.bptrends.com/managing-risks-with-an-end-to-end-processview/
55. Fellmann M, Zasada A (2014) state-of-the-art of business process compliance approaches. In: Proceedings of European conference on information system (ECIS'14), Tel Aviv, Israel
56. Fongon P, Grillo K (2004) Corporate implications of Sarbanes–Oxley Act: a public policy. http://www.global-trade.law.com/ITRN711
57. Förster A, Engels G, Schattkowsky T (2005) Activity diagram patterns for modeling quality constraints in business processes. In: Proceedings of MoDELS'05, pp 2–16
58. Förster A, Engels G, Schattkowsky T, Straeten RVD (2006) A pattern-driven development process for quality standard-conforming business process models. Proceedings of VL/HCC 2006:135–142
59. Francesconi E (2010) legal rules learning based on a semantic model for legislation. In: Proceedings of SPLeT workshop
60. Ghanavati S, Amyot D, Peyton L (2007) Towards a framework for tracking legal compliance in healthcare. In: Proceedings of CAiSE'07, pp 218–232
61. Ghose A, Koliadis G (2007) Auditing business process compliance. In: Krämer B, Lin KJ, Narasimhan P (eds) Collection of ICSOC 2007. Springer, Berlin, pp 169–180
62. Giblin C, Liu AY, Müller S, Pfitzmann B, Zhou X (2005) Regulations expressed as logical models (REALM). In: Proceeding of JURIX 2005, IOS Press, pp 37–48
63. Gilliot M, Accorsi R (2009) Runtime predictions of policy violations in automated buisness processes. Extended abstract: presented at prime life/IFIP Summer School Program, Sept 7–11, Nice/France
64. Goedertier S, Vanthienen J (2006) Business rules for compliant business process models. In: Proceeding of BIS 2006, Gesellschaft für Informatik, pp 558–579
65. Goedertier S, Vanthienen J (2006) Designing compliant business processes with obligations and permissions. In: Eder J, Dustdar S (eds) Business process management workshops 2006. Springer, Berlin, pp 5–14
66. Goedertier S, Vanthienen J, Caron F (2015) Declarative business process modelling: principles and modelling languages. Enterp Inf Syst 9(9):161–185
67. Gogolla M, Bttner F, Richters M (2007) USE: a UML-based specification environment for validating UML and OCL. Sci Comput Program 69(1–3):27–34 **(special issue on experimental software and toolkits)**
68. Gómez-López M, Gasca R, Rinderle-Ma S (2013) Explaining the incorrect temporal events during business process monitoring by means of compliance rules and model-based diagnosis. In: Proceeding of EDOCW'13, pp 163–172
69. Gómez-López MT, Gasca RM, Pérez-Álvarez JM (2015) Compliance validation and diagnosis of business data constraints in business process at runtime. Inf Syst 48:26–43
70. Governatori G (2005) Representing business contracts in RuleML. Int J Coop Inf Syst 14(2–3):181–216
71. Governatori G, Hashmi M (2015) No time for compliance. In: Proceedings of EDOC15, Adelaide, Australia, pp 9–18
72. Governatori G, Milosevic Z (2005) Dealing with contract violations: formalism and domain specific language. In: Proceedings of EDOC 2005. IEEE Computer Society, pp 46–57
73. Governatori G, Rotolo A (2006) Logic of violations: a Gentzen system for reasoning with contrary-to-duty obligation. Aust J Log 4:193–215
74. Governatori G, Rotolo A (2008) An algorithm for business process compliance. In: Proceedings Jurix 2008. IOS Press, pp 186–191
75. Governatori G, Rotolo A (2010) A conceptually rich model of business process compliance. In: Proceedings of APCCM'10, vol 110, pp 3–12

76. Governatori G, Rotolo A (2010) Norm compliance in business process modeling. In: Proceedings of RuleML 2010. Springer, pp 194–209
77. Governatori G, Sadiq S (2009) The journey to business process compliance. In: Handbook of research on BPM, IGI Global, pp 426–454
78. Governatori G, Shek S (2013) Regorous: a business process compliance checker. In: Proceedings of ICAIL'13, ACM, Rome, pp 245–246
79. Governatori G, Milosevic Z, Sadiq S (2006) Compliance checking between business processes and business contracts. In: Proceeding of EDOC'06, pp 221–232
80. Han J, Jin Y, Li Z, Phan T, Yu J (2007) Guiding the service composition process with temporal business rules. In: Web Services 2007
81. Hashmi M (2015) A methodology for extracting legal norms from regulatory documents. In: Proceedings of EDOCW'15. IEEE Computer Society, pp 41–50
82. Hashmi M, Governatori G (2017) Norms modeling constructs of business process compliance management frameworks: a conceptual evaluation. Artif Intell Law. https://doi.org/10.1007/s10506-017-9215-8
83. Hashmi M, Governatori G, Wynn MT (2013) Normative requirements for business process compliance. In: Service research and innovation–third Australian symposium, ASSRI 2013, Sydney, NSW, Australia, Nov 27–29, 2013. Revised selected papers, pp 100–116. https://doi.org/10.1007/978-3-319-07950-9_8
84. Hashmi M, Governatori G, Wynn MT (2014) Modeling obligations with event-calculus. In: Proceedings of RuleML'14, Czech Republic, pp 296–310
85. Hashmi M, Governatori G, Wynn M (2015) Normative requirements for regulatory compliance: an abstract formal framework. Inf Syst Front 18(3):429–455
86. Hassan W, Logrippo L (2008) Requirements and compliance in legal systems: a logic approach. In: Proceedings of RELAW'08, Barcelona, Spain, pp 40–44
87. Herrestad H (1991) Norms and formalization. In: ICAIL'91, ACM, pp 175–184
88. Herther NK (2009) Research evaluation and citation analysis: key issues and implications. Electron Libr 27(3):361–375
89. Hinge K, Ghose A, Koliadis G (2009) Process SEER: a tool for semantic effect annotation of business process models. In: Proceedings of EDOC '09, pp 54–63
90. HIPAA TUG (1996) The US Health Insurance Portability and Accountability Act of 1996
91. Hoffmann J, Weber I, Governatori G (2009) On compliance checking for clausal constraints in annotated process models. Inf Syst Front 14(2):155–177
92. IFRS (2014) IFRS 7 international financial reporting standards: financial instruments disclosures. http://www.ifrs.org/IFRSs/Pages/IFRS.aspx
93. Ingolfo S, Jureta I, Siena A, Perini A, Susi A (2014) Nómos 3: legal compliance of roles and requirements. In: Yu E, Dobbie G, Jarke M, Purao S (eds) Conceptual modeling, vol 8824. lecture notes in computer science. Springer, Berlin, pp 275–288
94. Jackson D (2006) Software abstractions: logic, language, and analysis. The MIT Press, Cambridge
95. James E, Jonathan S (2011) The benefits of static compliance testing for SCA next. In: Proceedings of the SDR'11, The Wireless Innovation Forum, Inc
96. Jiang J, Virginia D, Huib A, Frank D, Yao-Hua T (2013) Norm compliance checking. In: Proceedings of AAMAS'13, Saint Paul, USA, pp 1121–1122
97. Jiang J, Aldewereld H, Dignum V, Wang S, Baida Z (2014) Regulatory compliance of business processes. AI & Society, Heidelberg, pp 1–10
98. Johnson C, Grandison T (2007) Compliance with data protection laws using Hippocratic Database active enforcement and auditing. IBM Syst J 46(2):255–264
99. Johnson CM, Grandison TWA (2007) Compliance with data protection laws using Hippocratic Database active enforcement and auditing. IBM Syst J 46(2):255–264
100. Kabilan V, Johannesson P, Rugaimukamu D (2003) Business contract obligation monitoring through use of multi-tier contract ontology. In: Meersman R, Tari Z (eds) On The Move (OTM) workshops to meaningful internet systems. Springer, Berlin, pp 690–702
101. Kabilan V, Johannesson P, Rugaimukamu DM (2003) An ontological approach to unified contract management. In: Proceedings of 13th European Japanese conference on information modelling and knowlege bases, pp 106–110
102. Kähmer M, Gilliot M, Müller G (2008) Automating privacy compliance with ExPDT. In: Proceedings of the 10th IEEE conference on e-commerce technology and 5th conference on enterprise computing, pp 87–94
103. Karagiannis D, Mylopoulos J, Schwab M (2007) Business process-based regulation compliance: the case of the Sarbanes–Oxley Act. In: 15th IEEE international requirements engineering conference (RE 2007) pp 315–321

104. Kazmierczak P, Pedersen T, Ågotnes T (2012) NORMC: a norm compliance temporal logic model checker. STAIRS, frontiers in artificial intelligence and applications 241:168–179
105. Keller A, Ludwig K (2002) Defining and monitoring service-level agreements for dynamic e-business. In: Proceedings of the 16th USENIX conference on system administration, USENIX Association, Berkeley, USA, pp 189–204
106. Kharbili ME, Medeiros AKAD, Stein S, van der Aalst W (2008) Business process compliance checking: current state and future challenges. In: Modellierung Betrieblicher Informationssyteme, MobIS, pp 107–113
107. Kitchenham B (2004) Procedure for performing systematic reviews. Technical Report TR/SE-0401, Software Engineering Group, Department of Computer Science, Keele University, Keele, UK
108. Kitchenham B, Charters S (2007) Guidelines for performing systematic literature reviews in software engineering. Technical Report EBSE 2007-001, Keele University and Durham University Joint Report
109. Kiyavitskaya N, Zeni N, Breaux TD, Antón AI, Cordy JR, Mich L, Mylopoulos J (2008) Automating the extraction of rights and obligations for regulatory compliance. In: Li Q, Spaccapietra S, Yu E, Olivé A (eds) Proceedings of the 27th international conference on conceptual modeling (ER 2008). Springer, Berlin, pp 154–168
110. Knuplesch D, Ly L, Rinderle-Ma S, Pfeifer H, Dadam P (2010) On enabling data-aware compliance checking of business process models. In: Parsons J, Saeki M, Shoval P, Woo C, Wand Y (eds) Proceedings of the 29th international conference on conceptual modeling (ER 2010). Springer, Berlin, pp 332–346
111. Knuplesch D, Reichert M, Ly LT, Kumar A, Rinderle-Ma S (2013) Visual modeling of business process compliance rules with the support of multiple perspectives. In: Proceedings of the 32th international conference on conceptual modeling (ER 2013), Hong-Kong, pp 106–120
112. Knuplesch D, Reichert M, Kumar A (2015) Visually monitoring multiple perspectives of business process compliance. In: Proceedings of the 13th international conference on business process management (BPM 2015), Innsbruck, Austria, pp 263–279
113. Kowalski R, Sergot M (1989) A logic-based calculus of events. In: Schmidt J, Thanos C (eds) Foundations of knowledge base management, topics in information systems. Springer, Berlin, pp 23–55
114. KPMG (2013) A survey of fraud, bribery, and corruption in Australia and New Zealand. Survey series: issues and insights, KPMG Forensic. https://www.kpmg.com/AU/IssuesAndInsights/ArticlesPublications/Fraud-Survey/FDocuments/fraud-bribery-corruption-survey-2012v2.pdf
115. Küster JM, Ryndina K, Gall H (2007) Generation of business process models for object life cycle compliance. In: Proceedings of the 5th international conference on business process management (BPM 2007), Brisbane, Australia, pp 165–181
116. Lam HP, Governatori G (2009) The making of SPINdle. In: Governatori G, Hall J, Paschke A (eds) Proceedings of the 2009 international symposium on rule interchange and applications (RuleML 2009). Springer, Las Vegas, pp 315–322
117. Lam HP, Hashmi M, Scofield B (2016) Enabling reasoning with LegalRuleML. In: Alferes JJ, Bertossi L, Governatori G, Fodor P, Roman D (eds) Proceedings of the 10th international web rule symposium (RuleML 2016). Springer, Stony Brook, pp 241–257
118. LeFevre K, Agrawal R, Ercegovac V, Ramakrishnan R, Xu Y, DeWitt D (2004) Limiting disclosure in hippocratic databases. In: Proceedings of the thirtieth international conference on very large data bases, vol 30, VLDB endowment, VLDB '04, pp 108–119
119. Leitner M, Rinderle-Ma S (2014) A systematic review on security in process-aware information systems? Constitution, challenges, and future directions. Inf Softw Technol 56(3):273–293
120. Leitner P, Wetzstein B, Rosenberg F, Michlmayr A, Dustdar S, Leymann F (2009) Runtime prediction of service level agreement violations for composite services. In: Proceedings of the 3rd workshop on non-functional properties and SLA management in service oriented computing. Springer, Heidelberg, pp 176–186
121. Leitner P, Michlmayr A, Rosenberg F, Dustdar S (2010) Monitoring, prediction and prevention of SLA violations in composite services. In: Proceedings of ICWS'10, pp 369–376
122. Letia IA, Groza A (2013) Compliance checking of integrated business processes. Data Knowl Eng 87:1–18
123. Liu Y, Müller S, Xu K (2007) A static compliance-checking framework for business process models. IBM Syst J 46(2):335–361
124. Lomuscio A, Qu H, Solanki M (2008) Towards verifying contract regulated service composition. In: Proceedings of ICWS'08, pp 254 –261
125. Ly LT (2012) SeaFlows—a compliance checking framework for supporting the process lifecycle. Ph.D. Thesis, University of Ulm, Osnabrck, Germany
126. Ly LT, Rinderle-Ma S, Göser K, Dadam P (2012) On enabling integrated process compliance with semantic constraints in process management systems. Inf Syst Front 14(2):195–219

127. Ly LT, Maggi FM, Montali M, Rinderle S, van der Aalst W (2013) A framework for the systematic comparison and evaluation of compliance monitoring approaches. In: Proceeding of EDOC'13. IEEE Computer Society
128. Ly LT, Maggi FM, Montali M, Rinderle-Ma S, van der Aalst WM (2015) Compliance monitoring in business processes: functionalities, application, and tool-support. Inf Syst 54:209–234
129. Maggi F, Montali M, Westergaard M, van der Aalst W (2011) Monitoring business constraints with linear temporal logic: an approach based on colored automata. In: Proceedings of the 9th international conference on business process management (BPM 2011). Springer, pp 132–147
130. Maggi F, Montali M, van der Aalst W (2012) An operational decision support framework for monitoring business constraints. In: de Lara J, Zisman A (eds) Fundamental approaches to software engineering. Springer, Berlin, pp 146–162
131. Mateescu R, Sighireanu M (2003) Efficient on-the-fly model-checking ror regular alternation-free Mu-calculus. Sci Comput Program 46(3):255–281 **(special issue on formal methods for industrial critical systems)**
132. McIntyre SR (2008) Integrated governance, risk and compliance: improve performance and enhance productivity in federal agencies. Technical reports, PricewaterhouseCoopers
133. Meho LI, Tibbo HR (2003) Modeling the information-seeking behavior of social scientists: Ellis's study revisited. J Am Soc Inf Sci Technol 54(6):570–587
134. Milosevic Z, Jösang A, Dimitrakos T, Patton MA (2002) Discretionary enforcement of electronic contracts. In: Proceedings of EDOC'02. IEEE Computer Society, Washington, DC, USA, pp 39–50
135. Milosevic Z, Sadiq S, Orlowska M (2006) Towards a methodology for deriving contract-compliant business processes. In: Dustdar S, Fiadeiro J, Sheth A (eds) Proceedings of the 4th international conference on business process management (BPM 2006). Springer, Vienna, pp 395–400
136. Milosevic Z, Sadiq S, Orlowska M (2006) Translating business contract into compliant business processes. In: Proceedings of EDOC'06. IEEE Computer Society, pp 211–220
137. Monakova G, Kopp O, Leymann F, Moser S, Schäfers K (2009) Verifying business rules using an SMT solver for BPEL processes. In: Business process, services computing and intelligent service management, Leipzig, Germany, pp 81–94. http://subs.emis.de/LNI/Proceedings/Proceedings147/article2475.html
138. Montali M, Maggi FM, Chesani F, Mello P, Aalst WMPvd (2014) Monitoring business constraints with the event calculus. ACM Trans Intell Syst Technol 5(1):17:1–17:30
139. Namiri K, Stojanovic N (2007) Pattern-based design and validation of business process compliance. In: Proceedings of CoopIS'07. Springer, Berlin, pp 59–76
140. Namiri K, Stojanovic N (2007) Using control patterns in business processes compliance. In: Proceedings of WISE'07, Springer, pp 178–190
141. Namiri K, Stojanovic N (2008) Towards a formal framework for business process compliance. In: Proceedings of MKWI'08, München
142. Namiri K, Stojanovic N (2008) Towards a formal framework for business process compliance. In: Multikonferenz Wirtschaftsinformatik (MKWI 2008), Germany, pp 1185–1196
143. Nishizaki S, Ohata T (2013) Real-time model checking for regulatory compliance. In: Das V, Chaba Y (eds) Mobile communication and power engineering, communications in computer and information science, vol 296. Springer, Berlin, pp 70–77
144. Nute D (ed) (1997) Defeasible deontic logic, synthese library, vol 263. Academic Publishers, Dordrecht
145. Nute D (2003) Defeasible logic. In: Bartenstein O, Geske U, Hannebauer M, Yoshie O (eds) Web knowledge management and decision support. Springer, Berlin, pp 151–169
146. OASIS LegalRuleML Technical Committee (2015) LegalRuleML technical committee specifications. https://www.oasis-open.org/committees/legalruleml/charter.php, Retrieved 12 March 2016
147. OCEG (2012) Governance, Risk and Compliance Capability Model. https://www.oceg.org/about/what-is-grc/
148. Ochsenschläger P, Repp J, Rieke R, Nitsche U (1998) The SH-verification tool–abstraction-based verification of co-operating systems. J Form Asp Comput 10(4):381–404
149. Olivieri F (2014) Compliance by design. Synthesis of business processes by declarative specifications. Ph.D. Thesis, Dipartimento di Informatica, Università digli Studi di Verona, Italy and Institute for Integrated and Intelligent Systems, Griffith University, Australia
150. OMG (2010) Business Process Model Notation (BPMN). Standard. http://www.omg.org/spec/BPMN/2.0/
151. OMG (2011) Unified Modeling Language (UML 2.0). http://www.omg.org/spec/UML/2.0/
152. O'Neill A (2014) An Action framework for compliance and governance. Int J Clin Gov 19(4):342–359
153. Otto PN, Anton AI (2007) Addressing legal requirements in requirements engineering. In: Proceedings of the 15th IEEE international requirements engineering conference (RE 2007). IEEE Computer Society, pp 5–14

154. Pattersson P, Larson K (2000) UPPAAL 2K. Bull Eur Assoc Theor Comput Sci 70:40–44
155. Pershkow BI (2002) Sarbanes-Oxley: investment company compliance. J Invest Compliance 3(4):16–30
156. Pesic M, Schonenberg H, van der Aalst W (2007) DECLARE: full support for loosely-structured processes. In: Proceedings of 11th IEEE international conference on enterprise distributed object computing (EDOC'07), pp 287–287
157. Prakken H, Sergot M (1997) Dyadic denontic logic and contrary-to-duty obligations. In: [144], pp 223–262
158. Ramezani E, Fahland D, van der Aalst W (2012) Where did i misbehave? Diagnostic information in compliance checking. In: Proceedings of the 10th international conference on Business Process Management (BPM 2012), Tallinn, Estonia, pp 262–278
159. Ramezani E, Fahland D, van Dongen BF, van der Aalst W (2013) Diagnostic information for compliance checking of temporal compliance requirements. In: Proceedings of the 25th international conference on advanced information systems engineering (CAiSE 2013), Valencia, Spain, pp 304–320
160. Rangan RM, Rohde SM, Peak R, Chadha B, Bliznakov P (2005) Streamlining product lifecycle processes: a survey of product lifecycle management implementations, directions, and challenges. J Comput Inf Sci Eng 5(3):227–237
161. Rieke R, Repp J, Zhdanova M, Eichler J (2014) Monitoring security compliance of critical processes. 2014 22nd Euromicro international conference on parallel, distributed, and network-based processing (PDP 2014). Italy, Torino, pp 552–560
162. Rifaut A, Dubois E (2008) Using goal-oriented requirements engineering for improving the quality of ISO/IEC 15504 based compliance assessment frameworks. In: Proceedings of the 16th IEEE international requirements engineering conference (RE 2008), pp 33–42
163. Rikhardsson P, Best PJ, Green P, Rosemann M (2006) Business process risk management and internal control: a proposed research agenda in the context of compliance and ERP systems. In: Second Asia/Pacific research symposium on accounting information systems, Melbourne
164. Rinderle-Ma S, Mangler J (2011) Integration of process constraints from heterogeneous sources in process-aware information systems. International workshop on enterprise modelling and information systems architectures (EMISA 20110). Hamburg, Germany, pp 51–64
165. Roddick JF, Al-Jadir L, Bertossi L, Dumas M, Estrella F, Gregersen H, Hornsby K, Lufter J, Mandreoli F, Männistö T, Mayol E, Wedemeijer L (2000) Evolution and change in data management–issues and directions. SIGMOD Rec 29(1):21–25
166. Rosemann M, zur Muehlen M (2005) Integrating risks in business process models. In: Proceedings of ACIS'05
167. Sadiq S, Governatori G, Namiri K (2007) Modeling control objectives for business process compliance. In: Proceedings of BPM'07. Springer, pp 149–164
168. Salnitri M, Dalpiaz F, Giorgini P (2014) Modeling and verifying security policies in business processes. In: Bider I, Gaaloul K, Krogstie J, Nurcan S, Proper HA, Schmidt R, Soffer P (eds) Proceedings of the 15th international conference on business process modeling, development and support (BPMDS 2014). Springer, Berlin, pp 232–249
169. Sapkota K, Aldea A, Duce DA, Younas M, Bañares Alcántara R (2011) Towards semantic methodologies for automatic regulatory compliance support. In: Proceedings of PIKM'11, pp 83–86
170. Scannapieco S, Governatori G, Olivieri F, Cristani M (2011) Designing for compliance: norms and goal. In: The 5th international symposium on rules: research based and industry focused (RuleML 2011), Ft Lauderdale
171. Schleicher D, Anstett T, Leymann F, Mietzner R (2009) Maintaining compliance in customizable process models. In: Meersman R, Dillon T, Herrero P (eds) On the move to meaningful internet systems: OTM 2009. Springer, Heidelberg, pp 60–75
172. Schmidt R, Bartsch C, Oberhauser R (2007) Ontology-based representation of compliance requirements for service processes. In: Proceedings of the workshop on semantic business process and product lifecycle management, pp 28–39
173. Schrefl M, Stumptner M (2002) Behavior-consistent specialization of object life cycles. ACM Trans Softw Eng Methodol 11(1):92–148
174. Schumm D, Turetken O, Kokash N, Elgammal A, Leymann F, Heuvel WJVD (2010) Business process compliance through reusable units of compliant processes. In: Proceedings of the 10th international conference on current trends in web engineering. Springer, Vienna, Austria, pp 325–337
175. Semmelrodt F, Knuplesch D, Reichert M (2014) Modeling the resource perspective of business process compliance rules with the extended compliance rule graph. Proceeding of the 15th international conference on enterprise. Business-process and information systems modeling, Thessaloniki, Greece, pp 48–63

176. Spira LF, Page M (2003) Risk management: the reinvention of internal control and the changing role of internal audit. Account Audit Account J 16(4):640–661
177. Strecker S, Heise D, Frank U (2011) RiskM: a multi-perspective modeling method for IT risk assessment. Inf Syst Front 13(4):595–611
178. Stumptner M, Schrefl M (2000) Behavior consistent inheritance in UML. In: Laender AHF, Liddle SW, Storey VC (eds) Proceedings of the 19th international conference on conceptual modeling (ER 2000). Springer, Berlin, pp 527–542
179. Suriadi S, Weiß B, Winkelmann A, ter Hofstede AHM, Adams M, Conforti R, Fidge C, La Rosa M, Ouyang C, Pika A, Rosemann M, Wynn M (2014) Current research in risk-aware business process management–overview, comparison, and gap analysis. Commun Assoc Inf Syst 34(1):933–984
180. Teresa M, Gómez-López Gasca RM, Pérez-Álvarez JM (2015) Compliance validation and diagnosis of business data constraints in business processes at runtime. Inf Syst 48:26–43
181. The Basel Committee on Banking Supervision (2004) BASEL II accord - the international convergence of capital measurement and capital standards: a revised framework. https://www.bis.org/publ/bcbsca.htm
182. Trčka N, van der Aalst WMP, Sidorova N (2009) Data-flow anti-patterns: discovering data-flow errors in workflows. In: van Eck P, Gordijn J, Wieringa R (eds) Proceedings of the 21st international conference on advanced information systems engineering (CAiSE 2009). Springer, Berlin, pp 425–439
183. Turki S, Marija BO (2010) Compliance in e-government service engineering: state-of-the-art. 1st International conference on exploring services science (IESS (2010) Springer. Switzerland, Geneva, pp 270–275
184. US-Government (2002) Public Company Accounting Reforms and Investor Protection Act (Sarbanes-Oxley Act), Public Law 107–204, 116 Stat. 745
185. van der Aalst WM, Basten T (2001) Identifying commonalities and differences in object life cycles using behavioral inheritance. In: Colom JM, Koutny M (eds) Proceedings of the 22nd international conference on application and theory of Petri nets (ICATPN 2001). Springer, Berlin, pp 32–52
186. van der Aalst WMP, de Medeiros AKA (2005) Process mining and security: detecting anomalous process executions and checking process conformance. Electron Notes Theor Comput Sci 121(Suppl C):3–21. https://doi.org/10.1016/j.entcs.2004.10.013
187. van der Aalst W, de Beer HT, van Dongen BT (2005) Process mining and verification of properties: an approach based on temporal logic. In: Robert Meersman ZT (ed) CoopIS'05. Springer, Berlin, pp 130–147
188. van der Aalst W, van Hee KM, van Werf JM, Verdonk M, (2010) Auditing 2.0: using process mining to support tomorrow's auditor. Computer 43(3):90–93
189. van der Aalst W, van Hee K, van der Werf JM, Kumar A, Verdonk M (2011) Conceptual model for online auditing. Decis Support Syst 50(3):636–647
190. van der Aalst W, Adriansyah A, van Dongen B (2012) Replaying history on process models for conformance checking and performance analysis. Wiley Interdiscip Rev Data Min Knowl Discov 2(2):182–192
191. Vázquez-Salceda J, Aldewereld H, Grossi D, Dignum F (2008) From human regulations to regulated software agents' behavior. Artif Intell Law 16(1):73–87
192. Vicente P, Mira da Silva M (2011) A conceptual model for integrated governance, risk and compliance. In: Mouratidis H, Rolland C (eds) Advanced information systems engineering. Springer, Berlin, pp 199–213
193. Wang Z, ter Hofstede AH, Ouyang C, Wynn M, Wang J, Zhu X (2014) How to guarantee compliance between workflows and product lifecycles? Inf Syst 42:195–215
194. Ward M (1995) Principles and applications of electrochemical quartz crystal microbalance. Physical electrochemistry: principles, methods and applications. Marcel Dekker Inc, New York, pp 293–338
195. Wouters P, Costas R (2012) Users, narcissism and control ? tracking the impact of scholarly publications in the 21st century. Technical reports, SURFfoundation, Utrecht, The Netherland
196. Yip F, Parameswaran N, Ray P (2007) Rules and ontology in compliance management. In: Proceedings of EDOC'07, Washington, DC, USA, p 435
197. Yu J, Manh T, Han J, Jin Y, Han Y, Wang J (2006) Pattern based property specification and verification for service composition. In: Proceedings of WISE 2006. Springer, pp 156–168
198. Yu J, Han YB, Han J, Jin Y, Falcarin P, Morisio M (2008) Synthesizing service composition models on the basis of temporal business rules. J Comput Sci Technol 23:885–894
199. Zeni N, Kiyavitskaya N, Mich L, Cordy JR, Mylopoulos J (2013) GaiusT: supporting the extraction of rights and obligations for regulatory compliance. Requir Eng 20(1):1–22

Mustafa Hashmi received his Ph.D. in Computer Science in 2015 from Queensland University of Technology, Australia, and Masters degree in IT in 2004 from Universiti Utara Malaysia (UUM). He joined NICTA (now Data61, CSIRO) as a graduate researcher to work on Business Process Compliance. Before that he has been working in Germany in various capacities primarily in ICT domain. Mustafa's research interests include, but not limited to, in the areas of automation and analysis of business processes, regulatory compliance management, non-monotonic reasoning, defeasible and model logics and their applications to solve complex problems in large scale enterprises.

Guido Governatori received his Ph.D. in Legal Informatics in 1997 from CIRSFID, University of Bologna. Since them he held academic positions at Imperial College London, Griffith University, Queensland University of Technology, and The University of Queensland. He joined NICTA (now Data61, CSIRO), in 2008, where he leads the research activities on Business Process Compliance. Guido's research interests include non-classical logics, non-monotonic reasoning, formal models of normative reasoning, and their applications to business process modelling. A basic guideline of his research is to investigate conceptually sound formal models and methods grounded on understood principles of the underlying (application) phenomena, and with the aim of providing logic-based computationally oriented solutions.

Ho-Pun Lam is currently senior research scientist at Data61, CSIRO in Australia. He received his Ph.D. from the University of Queensland in 2012 and has been a researcher at the Software System Research Group at NICTA from 2011 to 2016. His primary research interests include non-classical logics, non-monotonic reasoning, formal models of normative reasoning, especially on devising algorithms to improve the computational complexity of defeasible reasoning (under different variants) and their applications in different domains, such as legal informatics, business process compliance and UAV navigation, etc.

Moe Thandar Wynn is currently senior lecturer within the Business Process Management discipline at Queensland University of Technology. She holds a Ph.D. in the area of workflow management from Queensland University of Technology (2007). Moe is an active researcher and educator on the topic of business process analytics. She has contributed a number of novel algorithms and software tools to help organisations pinpoint inefficiencies and derive concrete performance improvements, starting from common IT logs. Her current research interests are in the areas of Big Process Data Analytics and Comparative Organisational Mining. She is a member of IEEE Task force on Process Mining and a working group member for IEEE standardisation of XES log standard.