

Predictive Compliance Monitoring in Process-Aware Information Systems: State of the Art, Functionalities, Research Directions

Stefanie Rinderle-Ma
 Department of Informatics
 Technical University of Munich, Germany
 stefanie.rinderle-ma@tum.de

Karolin Winter
 Department of Informatics
 Technical University of Munich, Germany
 karolin.winter@tum.de

Abstract

Business process compliance is a key area of business process management and aims at ensuring that processes obey to compliance constraints such as regulatory constraints or business rules imposed on them. Process compliance can be checked during process design time based on verification of process models and at runtime based on monitoring the compliance states of running process instances. For existing compliance monitoring approaches it remains unclear whether and how compliance violations can be predicted, although predictions are crucial in order to prepare and take countermeasures in time. This work, hence, analyzes existing literature from compliance and SLA monitoring as well as predictive process monitoring and provides an updated framework of compliance monitoring functionalities. For each compliance monitoring functionality we elicit prediction requirements and analyze their coverage by existing approaches. Based on this analysis, open challenges and research directions for predictive compliance and process monitoring are elaborated.

Keywords— Predictive Compliance Monitoring, Predictive Process Monitoring, Systematic Literature Review, Research Directions

1 Introduction

The need for online and predictive data analysis capabilities and techniques is immense as, for exam-

ple, “given how COVID-19 has changed the business landscape, historical data may no longer be relevant” [76]. In the area of business process management, Predictive Process Monitoring (PPM) [34, 107, 70] has gained tremendous interest recently and several approaches for predicting, for example, the remaining time of cases, the next activity, or the outcome of a process have been presented. Doing so, PPM is a valuable means for estimating company-relevant key performance indicators such as customer satisfaction. Moreover, many PPM approaches promise to predict violations of Service Level Agreements and process compliance constraints, for example:

- “These predicted values can be metrics or process indicators evaluating the performance of a BP in terms of efficiency and effectiveness, or help to evaluate risks or predict possible service level agreement (SLA) violations.” [70]
- “Other manifestations of process prediction are the prediction of the next activity, the estimation of the completion time or early detection of abnormal process behavior indicating rule violations or compliance breaches” [74]
- “(i) predicting process outcomes, such as prediction of service level objectives (SLOs) values, service level agreement (SLA) violations, or linear temporal logic (LTL) constraint violations, and (ii) proactive process monitoring, such as predicting the next event in a case or its timestamp.” [100]

However, this claim has not been put to the test so far, i.e., it has not been systematically analyzed whether PPM techniques can (fully) support Predictive Compliance Monitoring (PCM). PCM constitutes a vital part of digitalized compliance management and monitoring [64] to especially address online settings, i.e., predicting compliance violations of process instances during runtime.

Example 1 illustrates the complexity of PCM, resulting from compliance constraints referring to multiple process perspectives, stemming from different regulatory documents, and a process event log/stream that might be emitted from multiple, heterogeneous sources/systems:

Example 1 PCM complexity. *The EU has issued several regulations for public health under COVID19 conditions¹, which are refined and implemented by the member states. These regulations have a significant impact on, for example, processes at airports and have changed several times so far. As airports become increasingly digitalized, e.g., by automatic check-in and security checks including sensors for biometric face recognition², more and more process behavior is captured in event streams and must be monitored for compliance with regulations. Moreover, interoperability is a key concern to make test and vaccination certificates acceptable across different countries. Note that not only COVID19 regulations are imposed on passenger transport processes, but also the GDPR and several other regulations must be complied to. The lack of approaches for the continuous monitoring of compliance with prediction of possible violations and explanations of root causes might result in severe threats to passenger safety and high fines.*

Example 1 is still described at a rather high level. Definition 1 captures the PCM problem in a more formal and general way.

Definition 1 PCM Problem. *Let L be a process event log/stream over the set of events \mathcal{E} and C be a set of compliance constraints. We assume the constraints to be provided in a formal notion, e.g., in*

¹<https://eur-lex.europa.eu/homepage.html>, accessed 2021-08-30

²See, for example, Lufthansa: <https://www.lufthansa.com/de/en/star-alliance-biometrics>, accessed 2021-08-30

Linear Temporal Logic or Event Calculus (cf. comparison provided in [27]), but no specific notion is required at this point. Events $e \in \mathcal{E}$ comprise attributes, at minimum a label referring to some process activity, a timestamp indicating when the event occurred, and a case id referring to the corresponding process instance. Further attributes include data elements and their values as well as resources.

Then predictive compliance monitoring (PCM) aims at determining the set of compliance violations $V \subseteq 2^{C \times 2^{\mathcal{E}} \times P \times T \times R}$, i.e., a set of tuples consisting of the constraint $c \in C$ that is or will be violated, the set of events $E \subseteq \mathcal{E}$ by which c is or will be violated, the probability $P \in [0; 1]$ with which the constraint will be violated (in case the violation has already occurred, $P = 1$ holds), the actual or predicted time T the violation will/has occur(red), and the root cause R of the violation.

The PCM problem takes as input a process event log or stream and a set of compliance constraints C . A process event log stores the events in \mathcal{E} that have been emitted during the execution of one or several process instances of one or several process types (ex post). A process event stream refers to the stream of events in \mathcal{E} that are emitted by executing one or several process instances of one or several process types during runtime (online). Compliance constraints stem from regulatory documents such as the General Data Protection Regulation (GDPR), ISO norms, or financial regulations [112]. Regulatory documents are often complex, e.g., “the European Union active legislation, which was estimated to be 170,000 pages long in 2005 and is expected to reach 351,000 pages by 2020” [3]. Consequently, compliance constraints refer to basic or complex control flow patterns and additional process perspectives, i.e., time, data, and resources [64, 108]. By contrast, Service Level Agreements (SLA) refer to “Service Level Objectives (SLOs), numerical QoS objectives, which the service needs to fulfill” [59].

We illustrate the PCM problem based on the following (abstract) example.

Example 2 PCM problem. Consider an event stream L containing events e_i with attributes label, time, data, and resource. Assume further label $\in \{A, B, C\}$, time being any valid timestamp, data $\in [-100; 100]$, and resource $\in \{r_1, r_2\}$. Moreover, consider that the following constraint c is imposed on L : B may only directly follow A only if data > 0 , otherwise C must directly follow A and C must be executed by resource r_1 . Assume that event stream L evolves in the following steps:

1. $L = \langle e_1 = (A, t_1, 30), \dots \rangle$
2. $L = \langle e_1, e_2 = (B, t_2), \dots \rangle$
3. $L = \langle e_1, e_2, e_3 = (A, t_3, -10), \dots \rangle$

The aim of PCM is to predict possible violations of constraint c based on event stream L . Consider, the prediction model was trained based on historical data and we therefore know that labels A, B, C can occur. When just considering attribute labels, we could say that the probability of a violation of c at Step 1, i.e., A is observed with data > 0 , accounts to $P = \frac{2}{3}$ because we can just have B in Step 2 causing no violation of c . At Step 2 the probability of c being violated by the next step would account to 0 since we have not observed the precondition A . At Step 3, we again observe A but this time data < 0 , i.e., we need to predict whether B or A occurs since in this case the constraint would be violated resulting in a violation probability of again $\frac{2}{3}$. Note that we consider the prediction model to be trained on historical data, i.e., we assume that all behaviour is known and we do not observe new labels. In addition, we do not consider to update the prediction model as soon as new observations arrive, i.e., the probabilities of violations remain the same as the event stream evolves.

This work addresses the question to which extent the PCM problem in its entirety is addressed and solved by existing approaches and which challenges and research directions remain still open. This question can be divided into and investigated along the following research questions:

RQ1 Which PCM approaches exist?

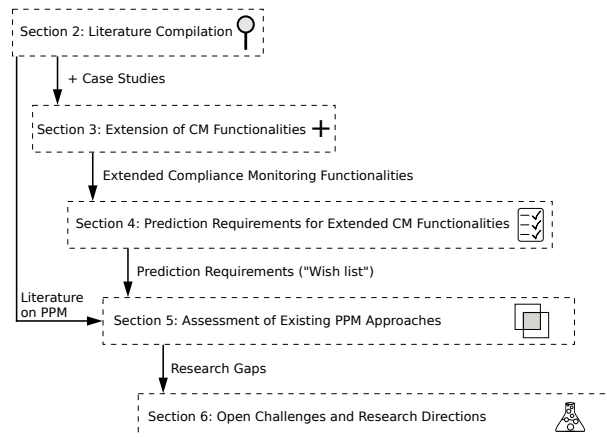


Figure 1: Method Overview

RQ2 Which functionalities must PCM approaches address?

RQ3 Which PCM functionalities are covered by existing approaches? How are they covered?

RQ4 Which open challenges and research directions remain for full PCM support?

Figure 1 depicts the overall method to tackle RQ1 – RQ4. We start with a compilation and analysis of PCM literature (\leftrightarrow RQ1). Then, we analyze existing literature on a) Compliance Monitoring (CM) as closely related research area and because PCM has been mentioned in an existing framework of Compliance Monitoring Functionalities (CMFs) by “*CMF8: Ability to pro-actively detect and manage violations*” [64]. Based on the analysis of PCM and CM literature, b) Predictive Process Monitoring, and c) SLA prediction can be identified as related topics, as well. The literature compilations are analyzed along the existing framework on CMFs [64] for an update since 2015 and the consideration of CMFs (\leftrightarrow RQ2). For PPM literature, specifically, we identify PPM surveys and analyze them plus the PPM literature for papers with possible PCM focus (\leftrightarrow RQ1). Moreover, analogously to [64] we also consider CM case studies as source for further CM functionalities. The analysis of the literature compilation combined with findings

from case studies results in an extended CMF framework (\mapsto RQ2). This framework is analyzed for coverage by existing PPM approaches, categorized along their prediction goals, e.g., next activity or outcome (\mapsto RQ3). We provide suggestions how each CM functionality can be addressed and provide a set of open challenges and research directions (\mapsto RQ4).

The contributions based on RQ1 – RQ4 and the method shown in Fig. 1 comprise literature compilations on PCM, CM, PPM, and SLA prediction (cf. Sect. 2), an extended CM Functionality framework (cf. Sect. 3), a “wish list” on how to address the CM Functionalities in terms of prediction (cf. Sect. 4), a systematic analysis of existing (mostly PPM) approaches along this wish list (cf. Sect. 5), and finally open challenges and research directions for PCM (cf. Sect. 6). Limitations of this survey are discussed in Sect. 7 and results concluded in Sect. 8.

The extended CMF framework as well as the research directions provide several open research topics from a data, algorithmic, and application perspective for PPM and PCM. Overall, this work aims at bridging the gap between online and predictive process analysis techniques and real-world compliance management.

2 Literature Compilation

According to the research method depicted in Fig. 1, we started with the literature compilation for PCM which was carried out in January and February 2022 using GoogleScholar³ based on title and abstract, i.e., we use ‘‘allintitle + keyword[s]’’ based on selected keywords and selection criteria, including for all compilations the focus on (business) processes, English language, and identifiable/known publication outlet. The literature lists are available via <https://www.in.tum.de/i17/data/>.

The literature compilation starts with searching for literature covering PCM (cf. Sect. 2.1). Based on the findings from this search, combined with expert knowledge on the topic, we identify three research areas that coincide with PCM, i.e., a) CM, b) PPM, and c) SLA prediction. For each of these research areas,

we conduct a separate literature compilation in Sect. 2.2–2.4. Overall, the compilation and analysis of PCM, CM, PPM, and SLA prediction results in 2619 hits (without citations and patents) in GoogleScholar and a final selection of 167 papers (direct search + snowballing + expert knowledge – duplicates). For different areas, specific analysis procedures are specified and applied, especially including updates and differences to existing systematic literature reviews in the particular area. For PPM, for example, we found 13 surveys. In Sect. 7, we will discuss why research areas such as online process or data mining have not been further investigated in this work.

2.1 PCM Literature Compilation

Table 1 depicts the results for the literature compilation on PCM. The keywords are ordered from specific, e.g., *predictive business process compliance monitoring* to generic, e.g., *predictive compliance* and contain variations of the term *predictive*, i.e., *prediction* and *predicting*.

The first selection based on title includes papers from the following domains: compliance and business process management, manufacturing, information security, cloud computing, and service compositions. We exclude papers from the medical domain. In particular, papers are classified as out of scope if they refer to, e.g., predicting whether a medical treatment would result in the desired effects or whether patients are likely to follow the medical advice. This narrows the initial selection containing 461 hits for all identified keywords down to 38 potentially relevant papers selected based on their title. The main inclusion criterion for papers at this stage is whether they have a connection to business processes. Using this criterion we finally select 6 papers. Excluding theses and duplicates results in 3 relevant papers. Out of these, [49] can be classified as PPM approach. [18, 88] do not refer to compliance constraints, but rather to SLAs.

As a conclusion, the compilation does not include any PCM approaches. The selected papers can be classified as PPM or SLA prediction approaches.

³<https://scholar.google.com/>

keywords	# hits	# title	# final (- dupl., theses)
predictive business process compliance monitoring	0	0	0
predictive process compliance monitoring	2	0	0
predictive compliance monitoring	9	4	0
predictive compliance	92	12	0
business process compliance monitoring prediction	0	0	0
process compliance monitoring prediction	1	0	0
compliance monitoring prediction	6	2	1
compliance prediction	141	12	5
predicting business process compliance monitoring	0	0	0
predicting process compliance monitoring	1	0	0
predicting compliance monitoring	6	0	0
predicting compliance	203	8	0
\sum PCM	461	38	6 (3)

Table 1: PCM: Keywords and # of Selected Papers

2.2 CM Literature Compilation

Table 2 depicts the results for the literature compilation on CM, taking the CM functionality framework and systematic literature survey from 2015 [64] as a yardstick, i.e., we assume that CM literature up to 2015 has been covered already in [64]. Hence, the 25 selected papers are distinguished into 16 papers after 2015 (without SLA), 8 SLA papers (before and after 2015 as [64] explicitly excluded works in SLAs), and 2 compliance monitoring functionality surveys, including [64]. The “+” indicates that only entries that have not been found in previous search runs and summarized in rows above are added. From the 8 SLA-related papers, 7 are considered as out of scope as they do not have any connection to processes. 1 paper is also found in the PPM literature compilation (cf. Sect. 2.3) and further investigated.

keyword(s)	# hits	# selection (cmp. [64])
business process compliance monitoring	20	15
process compliance monitoring	44	+5
compliance monitoring	1520	+17
process compliance auditing	13	3
compliance monitoring [survey / comparison / benchmark]	28	0
\sum Compliance monitoring	1625	40 (25)

Table 2: CM: Keywords and # of Selected Papers

2.3 PPM Literature Compilation

Table 3 presents the results of the literature search for PPM. As for Table 2, the “+” indicates that only entries that have not been found in previous searches yet are added. The search results contain 13 survey papers. Their publication dates include 2 surveys in 2018 [34, 70], 3 surveys in 2019 [107, 104, 101], 3 surveys in 2020 [96, 37, 75], and 4 surveys in 2021 [115, 47, 98, 82]. We add the survey in [74] due to snowballing. The surveys are then analyzed with respect to their coverage of the selected papers summarized in Table 3, i.e., i) which of the papers are analyzed and contribute to the conclusions of a survey, ii) topics “compliance” or “SLA”, and iii) solutions for PCM.

From the 121 selected papers, 61 papers fulfill i)–iii), including the surveys themselves, resulting in 60 uncovered papers. From these 60 papers, the papers are selected that are published in 2019 or later in order to catch recent approaches that could not have been covered by surveys yet. Excluding technical reports results in 35 papers.

The analysis of the surveys and the recent 35 papers identifies 9 papers that are mentioned to address compliance in connection with PPM. Out of these 9 approaches, 4 do not address any constraint definition, 3 address constraint definition through SLA [15, 60, 21], and 2 address constraint definition in the form of predicates [67, 33], e.g., based on LTL constraints. The papers on SLAs will be merged and

keyword(s)	# hits	# selection (after clean up)	# surveys
predictive business process monitoring	52	24	
predictive process monitoring	135	+48	
business process prediction	62	+29	
predictive business processes monitoring	18	+8	
next activity prediction	21	+3	
remaining time prediction	110	+12	
process outcome prediction	29	+1	
concept drift prediction	32	+1	
\sum PPM	459	126 (121)	13

Table 3: PPM: Keywords and # of Selected Papers

analyzed with literature on SLA prediction, the two papers on predicates are retained within the literature on PPM.

Furthermore, we recognize an increasing number of papers over the years, overall and for the papers after the surveys confirming that PPM has become an actively researched field.

2.4 Literature Compilation on SLA Prediction

The keywords and results for the literature search and compilation on SLA prediction are summarized in Table 4. Out of 74 initial hits and 49 papers are selected based on their title and focus on computer science, i.e., we exclude papers from the medical and biology domain. After reading we narrow the list down to 11 papers by only including papers with a business process or service composition context. Excluding theses and duplicates results in 5 papers [20, 44, 61, 18, 60]. These 5 papers are merged with results from the literature compilation on PCM ([18, 88]) as well as from the literature compilations on CM and PPM ([15, 60, 21]) resulting in overall 8 relevant papers (10 minus 2 duplicates).

keyword(s)	# hits	# title	# final (- dupl., theses)
predictive SLA	4	4	1
SLA prediction	42	23	5
predicting SLA	8	3	0
predictive Service Level Agreement(s)	6	6	1
Service Level Agreement(s) prediction	10	9	4
predicting Service Level Agreement(s)	4	4	0
\sum SLA Prediction	74	49	11 (5)

Table 4: SLA: Keywords and # of Selected Papers

3 Extension of Compliance Monitoring Functionalities

The overall goal is to assess whether and to which extent PCM is addressed by existing approaches and to set out a research agenda for PCM. This necessitates building a basis for the assessment, i.e., a set of PCM requirements based on which existing approaches can be evaluated and potential research gaps can be identified. We opt for the well-established Compliance Monitoring Functionality framework [64] and update and extend the framework with a focus on predictive compliance monitoring requirements. Ly et al. [64] define the following Compliance Monitoring Functionalities (CMFs):

- *Modeling requirements:* CMF1 (time), CMF2 (data), CMF3 (resources)
CMF1–3 refer to the modeling capabilities of the compliance constraints. The underlying assumption is that all compliance constraints refer to the control flow of a process, e.g., by referring to the existence of an activity plus a maximal duration of this activity (\mapsto CMF1).
- *Execution requirements:* CMF4 (non-atomic activities), CMF5 (life cycles), CMF6 (multiple instances constraints)
CMF4–6 refer to instantiation and execution of

the process instances, more precisely the event and life cycle information that is stored in the process event streams during runtime, and the instantiation of the compliance constraints.

- *User requirements:* CMF7 (reactive management), CMF8 (proactive management), CMF9 (explain root cause of violation), CMF10 (quantify compliance degree)

CMF7–10 refer to support that approaches offer for users to understand and handle compliance violations. CMF8 refers to PCM as proactive management of compliance violations requires the prediction of such violations.

In the following, we analyze the papers from the literature compilation described in Sect. 2 regarding two aspects: i) are the CMFs as outlined in [64] still valid, and ii) is an extension of the CMF framework necessary for PCM. As papers from the literature compilation on PCM were merged into the literature compilation on PPM and SLA prediction respectively, we directly start with describing the findings based on the retrieved literature for CM.

3.1 Findings for CM

As discussed in Sect. 2.2, the CM literature compilation contains 16 papers after 2015 which (at least partly) address one or several of the CMFs of the original framework in [64].

[4] explicitly mentions [63] and the coverage of CMF1 – CMF3 (“*process compliance holism*”). Moreover, CMF8 (“*Proactive response to violation possibility*”) is highlighted in the context of PCM as the ability to avoid violations by providing “*compliance actions*” such as changing the process. In general, the approach works with abstraction based on events. CMF9 is partly addressed with respect to visualization. As additional requirements, [4] emphasize the resolution of ambiguities and inconsistencies in the compliance constraint base as well as the efficiency/performance of the CM approach in order to deal with a large volume of events.

[6, 5, 90] cover CMF1 and CMF3 based on process rewriting, anti-patterns, and complex event processing. In addition, the approach supports life cycles

(\mapsto CMF5). By providing “*violation actions*”, i.e., recovery actions such as “alert” or “suspend” in case of compliance violations, CMF7 is addressed. By providing “*predictive actions*”, the approach covers CMF8. [6, 5] also mention additional requirements, i.e., the reuse of compliance knowledge and the resolution of ambiguities and inconsistencies in constraint base.

[19] addresses compliance and change in process collaborations, for example, compliance in connection with the dynamic replacement of partners leaving a process collaboration or new partners joining. Addressing compliance in distributed settings is hence found an additional requirement.

bpCMon [36] addresses “*multiple process perspectives*”, i.e., CMF1 – CMF3. The approach defines an event-based compliance language (ECL) and designs an event reaction system (ERS). [36] advocate the aggregation of values of multiple events and event correlation for addressing multiple data sources. Moreover, the efficiency of the approaches is put into the spotlight for dealing with a large volume of events.

[39] extend ECA rules with Time (TECA) and covers CMF1 – CMF3. As the description of the proposed framework lacks details, the assessment of further CMFs is difficult.

[55] addresses CMF1 – CMF3, CMF6, CMF7, CMF8, and CMF9. As additional requirements, [55] emphasize the reliable determination of compliance violations. In addition, preventive compliance management is supposed to help users to design processes in a compliant manner. Also the support of compliance constraints might be necessary for constraints the span different partners.

COMS [57] covers CMF1 – CMF3 as well as CMF4 – CMF6 due to special focus on the activity life cycle. Moreover, CMF9 is addressed by employing an Match-Condition-Action Rule approach. As additional requirements, [57] address the integration of events from multiple heterogeneous source. The approaches abstracts from activity label equivalence by using semantic activity equivalence instead.

[58] follow a model-driven approach, featuring complex event processing and business rules. The approach distinguishes between functional and non-functional compliance requirements. However, based

on the level of detail, the CMFs and additional requirements cannot be fully assessed.

[62] tackle distributed compliance monitoring, driven by recent IoT developments. The approach employs the SCIFF monitoring framework (abductive logic programming) as well as parallelization of computation through horizontal/vertical partitioning of logs and models and covers CMF1. As can be seen from the consideration on parallel computing, this approach addresses efficiency/performance as additional requirements. Another focus is on handling of “out-of-order events”.

[68] cover CMF1 – CMF3 based on Multi-perspective DECLARE constraints and Integer Linear Programming. The approach differentiates compliance states “possibly satisfied”, “possibly violated”, “permanently satisfied”, “permanently violated” and hence partly contributes to CMF8. As additional requirements, the approach mentions the “early detection of conflicting constraints” where only one of the constraints can be fulfilled at a time.

[73] strive at a “decentralized solution switching from control- to artifact-based monitoring” [28]. The artifact-based approach features data flow and control flow guards. Hence, we conclude that CMF2 is fulfilled and CMF5 and CMF8 at least partly. As additional requirements, [73] feature decentralized process settings, specifically, the exchange of physical objects with compliance requirements.

[99] focuses on more general requirements and challenges, not at a technical solution.

ProMSecCo [106] focuses on security constraints and hence addresses CMF3 (separation of duty and binding of duty constraints). A further assessment is hard due to the provision of too few details.

[116] address CMF5 as well as CMF3 (separation of duty constraints). The approach is implemented with DROOLS. The paper provides too few details on the conceptual model/design of the approach. Hence, further assessment of CMF coverage is hard.

The retrieved survey [102] analyzes process mining and auditing and provides a categorization along domains. Within that survey we could not find any additional requirements. The paper referring to SLAs is discussed in Section 3.3. Overall, the coverage of CMF1–10 can be summarized as follows:

- CMF1: [4, 6, 5, 90, 36, 39, 55, 57, 62, 68]
- CMF2: [4, 6, 5, 90, 36, 39, 55, 57, 68, 73]
- CMF3: [4, 6, 5, 90, 36, 39, 55, 57, 68, 106, 116]
- CMF4: [57]
- CMF5: [5, 90, 57, 73, 116]
- CMF6: [55, 57]
- CMF7: [5, 90, 55]
- CMF8: [4, 5, 90, 55, 68, 73]
- CMF9: [4, 55, 57]
- CMF10: [55]

Conclusion. CMF1 - CMF10 as proposed in [64] are still valid and approaches since its publication in 2015 address several of the outlined CMFs. After 2015, new directions/requirements include:

- Efficiency/performance of compliance monitoring
- Compliance monitoring in distributed processes
- Integration of event streams form multiple data sources
- Consistency of the constraint base

These requirements all refer to *data*. We will extend CMF1–10 with these new *data requirements* and describe and illustrate them in Sect. 3.5.

3.2 Findings for PPM

[23] discuss the PPM perspectives control flow, time, data, and resources. Time, data, and resources correspond to CMF1–3. Control flow, by contrast, is implicitly assumed in the original CMF framework [64], i.e., compliance constraints are considered to always refer to control flow information of the underlying processes or process instances, but no explicit CMF covers control flow. In the context of existing approaches for PPM that mainly refer to SLAs,

we opt for explicate this assumption with by a dedicated CMF as a SLA connection to control flow does not always exist. Considering SLAs without control flow aspects, basically, abstracts from any underlying process, i.e., the SLA could be checked on processes, cloud, or web services. Moreover, [23] add the *conformance perspective*. The latter covers deviations from a normative process model (if such a model exists). This normative model can be expressed by a Petri net or a set of LTL rules and is supposed to address “*questions in the context of compliance management, auditing, security*” [23].

[21, 11] mention the requirement to consider external (process) context data. This can be underpinned by other recent approaches such as [97, 24] showing that context can provide useful information for root cause analysis and explainability in PPM.

[41] combine process simulation with prediction, the focus is on time (CMF1). Here the approach distinguishes *traverse time* (throughput time of an instance), *execution time* (throughput of an instance for one task), *inter-arrival time* (time distance between to starting time of two instances), and *workload burstiness* (time between two instances are started on a specific task).

[87] advocate to update the set of possible/future violations when new events occur during runtime. This can be seen as a refinement of CMF8.

There are several PPM approaches that focus on the explainability of the prediction results [98, 9, 71, 86, 95]. They can be mapped onto CMF9 on explaining root causes for compliance violations as proposed in [64], but CMF9 can be refined into more precise CMFs, i.e., i) root cause analysis and ii) effective communication of root cause as in [64], as well as additionally in iii) explaining and visualizing prediction results, iv) explaining and visualizing the set of future violations, and v) explaining and visualizing the effects of mitigation actions on predicted/future violations. Moreover, we advocate to rename CMF9 into *CMF9’: Explainability*.

Finally, a collection of PPM papers address the properties and quality of the input data, i.e., the event streams, including sparsity, variation, and repetitiveness [40], size of the input data [48, 42], and balanced vs. imbalanced data [46, 50]. This will

be reflected in an additional CMF on data properties and quality.

Conclusion. An additional requirement reflecting the control flow perspective of compliance constraints will be added to the CMF framework. Moreover, the CMF framework will be extended by a CMF on the ability to exploit external (process) context data and data properties and quality. These additional CMFs can be added to the new group *Data requirements*. Further on, a refinement of CMF8 and CMF9 will reflect the work on explaining and visualizing results of prediction.

3.3 Findings for SLA Prediction

[20] mention prediction across multiple process cases, i.e., instance spanning predictions, but do not provide any solutions. The approach is directed towards explainability by providing a measure for reliability of predictions, but just for individual cases. Those aspects are covered by CMF6 and CMF9.

[44] use an abstract notation for service orchestrations, i.e., “*compositions with a centralized control flow*” and “*predict possible situations of SLA conformance and violation, and to obtain information on the internal parameters of the orchestration (branch conditions, loop iterations) that may occur in these situation*”. These aspects are covered by CMF8 and CMF9.

[61] predict SLAs and adapt service compositions in order to avoid a violation of SLAs. Mitigation actions and adaptations are part of CMF8.

[18] target the problem of state space explosion which addresses the newly added requirement on efficiency of compliance monitoring.

An analysis of non-compliance to prevent compliance violations in the future with only limited prediction capabilities is presented in [88] and addresses CMF8.

[15] present a BPI cockpit and address also CMF8. CMF6, CMF8 and partly CMF9 are addressed by [21]. [60] address CMF1, CMF2, CMF8.

Conclusion. SLA prediction approaches confirm efficiency as requirement for compliance monitoring.

3.4 Findings Based on Case Studies

Analogously to [64], we analyze case studies and real-world compliance constraint collections in order to derive further possible extensions of the CMF framework. Case studies can be found in various domains including data protection [111], finance [108], and manufacturing [35, 114]. As discussed in [108], real-world compliance constraints refer to the modeling requirements CMF1-3 plus control flow patterns such as existence, absence, and ordering. A collection of real-world constraints that span multiple process instances and processes can be found in [84]. Constraints spanning multiple instances are referred to by CMF6 in the original CMF framework [64]. When looking into literature and the real-world constraints, CMF6 can be refined into constraints that reflect i) the simultaneous execution of events, ii) constrained execution, iii) order of events, iv) non-concurrent execution of events, and v) constrained start of following instances [114, 113].

Conclusion. Case studies and collections of real-world compliance constraints confirm modeling requirements CMF1-3. Additional requirements can be specifically identified in real-world constraints that span multiple process instances or processes. These additional requirements will be included as refinement of CMF6.

3.5 Extended Compliance Monitoring Functionality Framework

Figure 2 depicts the extended CMF framework. Following [64], each of the CMFs is partitioned into sub CMFs reflecting specific requirements on the expressiveness of the CMF, e.g., quantitative vs. qualitative time requirements. The proposed extensions based on the findings from literature and case studies (cf. Sect. 3.1 — 3.4) are marked in bold font. Each of the CMFs in the extended framework depicted in Fig. 2 is illustrated by an example in Sect. 4.

The first extension refers to the modeling requirements by explication of CMF0 on control flow. Following control flow patterns for compliance constraints [64], we opt for the basic building blocks **existence (CMF0.1)**, **absence (CMF0.2)**, and **or-**

dering (CMF0.3).

For the execution requirements, the extensions comprise the refinement of CMF6 on multiple instance constraints, following the categorization for instance-spanning constraints proposed in [114], i.e., constraints on **simultaneous (CMF6.2)**, **constrained (CMF6.3)**, **order (CMF6.4)**, and **non-concurrent (CMF6.5)** execution of tasks across process instances/processes as well as **constrained start of following instances (CMF6.6)**.

The user requirements are extended by refinement of CMF8 and CMF9. For CMF8, the **update of the set of possible and future violations (CMF8.3)** of compliance is added. CMF9 is renamed to **CMF9': Explainability** and refined by **explain and visualize prediction results (CMF9.3)**, **explain and visualize the set of possible and future violations (CMF9.4)**, and **explain and visualize effects of mitigation actions (CMF9.5)**.

Finally, the group of **Data requirements** on process event data such as logs and streams as PCM input is added. The consistency of the constraint base as also mentioned in literature is considered beyond the scope of this work. In detail, the extensions comprise **efficiency/performance of CM (CMF11)**, **integration of data from multiple sources (CMF12)**, **distributed processes (CMF13)**, **context data (CMF14)**, and **data properties and quality (CMF15)**.

4 Prediction Requirements for Extended Compliance Monitoring Functionalities

We illustrate the extended framework depicted in Fig. 2 by examples. Moreover, each CMF is analyzed on how it could be verified through PPM, i.e., we provide the prediction requirements that are necessary to verify each of the CMF, resulting in a **PCM “wish list”**. These prediction requirements serve then as input for the assessment of existing PPM approaches in covering PCM (cf. Sect. 5).

<p>Modeling requirements</p> <p>CMF0.1 existence</p> <p>CMF0.2 absence</p> <p>CMF0.3 ordering</p> <p>CMF1.1 time qualitative</p> <p>CMF1.2 time quantitative</p> <p>CMF2.1 activity data</p> <p>CMF2.2 case data</p> <p>CMF3.1 unary resource condition</p> <p>CMF3.2 extended resource condition</p>	<p>User requirements</p> <p>CMF7 reactive management</p> <p>CMF8.1 early detection of conflicting rules</p> <p>CMF8.2 possible/future violations</p> <p>CMF8.3 update set of possible and future violations</p> <p>CMF8.4 recommendations for users to avoid violations</p> <p>CMF9.1 root cause analysis</p> <p>CMF9.2 effective communication of root cause</p> <p>CMF9.3 explain and visualize prediction results</p> <p>CMF9.4 explain and visualize set of possible and future violations</p> <p>CMF9.5 explain and visualize effects of mitigation actions</p> <p>CMF10.1 compliance degree of single traces</p> <p>CMF10.2 compliance degree of entire process/system</p>
<p>Execution requirements</p> <p>CMF4 non-atomic activities</p> <p>CMF5 activity life cycles</p> <p>CMF6.1 multiple instantiation of compliance constraints</p> <p>CMF6.2 simultaneous</p> <p>CMF6.3 constrained execution</p> <p>CMF6.4 order</p> <p>CMF6.5 non-concurrent</p> <p>CMF6.6 constrained start of following instances</p>	<p>Data requirements</p> <p>CMF11 efficiency/performance of CM</p> <p>CMF12 integration of data from multiple sources</p> <p>CMF13 distributed processes</p> <p>CMF14 context data</p> <p>CMF15 data properties and quality</p>

Figure 2: Extended Compliance Monitoring Functionalities (Extensions in bold)

4.1 Modeling requirements

Modeling requirements refer to the expressiveness of the compliance constraints reflected by control flow, time, data, and resources. CMF0.1, CMF0.2, and CMF0.3 relate to basic control flow patterns in compliance constraints such as occurrence and absence of activities as well as ordering. CMF1 refers to the time perspective which can be either qualitative or quantitative. CMF2 refers to the data perspective, which can be either activity data, case data/extended data conditions, unary data conditions or a comparison of multiple data objects. CMF3 captures resource conditions which can be either unary or extended.

CMF 0.1 existence.

Example: “Activity “bill” must be executed at least once” [79]

Prediction Requirements: predict set of next activities / events, ranked by probability of occurrence; distinction between immediately/eventually occurs

CMF 0.2 absence.

Example: “If activity “check-out” is ever executed, then activity “charge” must never be executed” [79]

Prediction Requirements: predict absence of activities (implicitly) via prediction of set of next activities / events, ranked by probability of occurrence; distinction between immediately/eventually absent

CMF 0.3 ordering.

Example: “When the client “checks-out” the bill must be “charged”.” [79]

Prediction Requirements: predict set of next activities / events, ranked by probability of occurrence; distinction between immediately/eventually follows

CMF 1.1 time qualitative.

Example: “For payment runs with amounts beyond € 10000, the payment list has to be signed before being transferred to the bank and has to be filed afterwards for later audits.” [66]

Prediction Requirements: predict set of next activi-

ties / events sequences, ranked by probability of occurrence; distinction between immediately/eventually follows

CMF 1.2 time quantitative.

Example: “A passenger ship leaving Amsterdam has to moor in Newcastle within 16 h.” [64]

Prediction Requirements: predict set of next activities / events, ranked by probability of occurrence; distinction between immediately/eventually follows plus remaining time to either complete the process or the activity or until next event happens plus data (status)

CMF 2.1 activity data (unary+extended).

Example: “If the PainScore of patient p is greater than 7 and the status is uninitialized then the status must be changed to initialized and a timer event is generated to treat patient p within 1 h.” [64]

Prediction Requirements: predict next event/activity in combination with its associated value for one or multiple event attributes

CMF 2.2 case data.

Example: A passenger ship may never be used for fishing.

Prediction Requirements: predict next activity / event depending on case data prediction

CMF 3.1 unary resource condition.

Example: “Orders of more than 1000€ can only be approved by a senior manager.” [64]

Prediction Requirements: predict next activity and its associated resource plus eventually data attributes

CMF 3.2 extended resource condition.

Example: “Final approval of the assessment can only be granted by the manager that requested the assessment.” [64]

Prediction Requirements: predict next activity and its associated resource

Wish list for modeling requirement prediction: Compliance prediction with respect to modeling CMFs requires next activity / event prediction, including fine-granular probabilities. Especially interesting is the prediction of activity absence. Moreover, temporal and resource prediction as well as the prediction of data values is required.

4.2 Execution requirements

CMF4 relates to the support of non-atomic activities, i.e., activities that have a duration, typically expressed by at least the occurrence of start and end/completion events in the process event log/stream. CMF5 also relates to the support of activity life cycles [57] including activation, suspension, completion and a balance between start and complete events. CMF6 refers to support for multiple instance constraints. Note that no prediction requirements are formulated for CMF6.1 as it solely refers to the multiple instantiation of compliance constraints and is hence independent of any process predictions. CMF6.2–6.5 refer to requirements for compliance constraints spanning multiple process instances / processes.

CMF 4 non-atomic activities (explicit).

Example: “An order creation cannot be completed until the customer registration is completed.” [64]

Prediction Requirements: predict and distinguish different event types

CMF 4 non-atomic activities (implicit).

Example: “Activity check project can be executed only while the project is under preparation.” [64]

Prediction Requirements: predict and distinguish different event types

CMF 5 activity life cycles (activation).

Example: “A start event creates an activity instance and puts it into the active state” [64]

Prediction Requirements: predict and distinguish life cycle states/transitions of next event/activity

CMF 5 activity life cycles (completion).

Example: “Each completion event moves its associated activity instance to the completed state, provided that the instance is currently active.” [64]

Prediction Requirements: predict and distinguish life cycle states/transitions of next event/activity

CMF 5 activity life cycles (balance start/complete events).

Example “For every activity instance, each start event has a single corresponding completion or cancelation event” [64]

Prediction Requirements: predict and distinguish life cycle states/transitions of next event/activity

CMF 6.2 simultaneous execution.

Example: “Finished orders of one day are delivered to the post office simultaneously in the evening.” [114]
Prediction Requirements: predict occurrence of events across multiple instances/processes plus temporal prediction

CMF 6.3 constrained execution.

Example “All print jobs must be completed within 10 min in at least 95% of all cases.” [114]
Prediction Requirements: aggregated prediction of data values for specific events across multiple instances/processes

CMF 6.4 order.

Example: “If a flyer or poster order is received $\mathcal{P}2$ is started” [114]
Prediction Requirements: predict next activity/event across multiple processes

CMF 6.5 non-concurrent.

Example: “Flyers and posters as well as bills and posters cannot be printed concurrently on one printer since they require a different paper format.” [114]
Prediction Requirements: predict durations of events/activities across multiple processes

CMF 6.6 constrained start of following instances.

Example: “Printer 1 may only print 10 times per day.” [114]
Prediction Requirements: predict next event/activity together with predicting time, resource, and data

Wish list for execution requirement prediction: Compliance prediction with respect to execution CMFs requires approaches to distinguish the semantics of different event types and life cycle states/transitions and to predict different event types and life cycle states/transitions. Moreover, predictions of next activity/event plus prediction of time, data, and resources should be possible across multiple process instances and processes.

4.3 User requirements

CMF7 refers to the ability to reactively detect and manage compliance violations and is hence not relevant in the context or predictive process and compliance monitoring. CMF8 addresses the pro-active detection and management of compliance violations. CMF9 refers to providing explanations of root causes of compliance violations. CMF10 captures the ability to quantify the degree of compliance.

CMF 8.1 early detection of conflicting rules.

Example: “Every time an order is delivered, the warehouse must be replenished. If the replenishment truck is broken, the warehouse cannot be replenished. Consider an execution where the truck is broken and the order delivered. Approaches able to detect conflicts among rules would in this case point out an (implicit) violation: the first constraint requires a replenishment and the second forbids it.” [64]
Prediction Requirements: detect conflicting rules as soon as possible with precise probability/likelihood; continuously update set of conflicting rules as event stream evolves

CMF 8.2 possible/future violations.

Example: “Conducting a payment run creates a payment list containing multiple items that must be transferred to the bank. Then, the bank statement must be checked for payment of the corresponding items. For payment runs with amount beyond 10,000€, the payment list has to be signed before being transferred to the bank and has to be filed afterwards for later audits. For a concrete payment run with an amount beyond 10,000€, the monitoring system can deduce from the constraints that two activities (namely sign the payment list and file the payment list) are pending and need to be executed to comply. This can be exploited for ensuring that the pending tasks are scheduled and for preventing the transfer of the payment list to the bank unless it has been signed.” [64]
Prediction Requirements: detect and predict set of compliance violations (cf. Def. 1) as soon as possible and as complete as possible with precise probability/likelihood (connected with CMF8.1);

CMF 8.3 update set of possible/future violations.

Example: “when $p > 0$, for each pending activation, an ILP problem is instantiated using the correlation condition. When the activation becomes fulfilled, the corresponding ILP problem is deleted.” [68]

Prediction Requirements: *continuously update compliance prediction for all compliance constraints and events as event stream evolves;*

CMF 8.4 mitigation actions for users to avoid violations.

Example: “Requests for building permits need to be handled within 3 months. Based on historic information, i.e., comparing a request currently being handled with earlier requests, one can predict the remaining processing time. A counter measure is taken if the predicted remaining processing time is too long.” [64]

Prediction Requirements: *determine and provide mitigation actions based on compliance predictions as soon as possible, with precise assessment of risk and impact of the mitigation actions; continuously update mitigation actions based on updates of compliance predictions;*

CMF 9.1 root cause analysis.

Example: “When a patient is diagnosed with cryptorchidism, an operation must be performed either through laparoscopy or with an open surgery but not both. This rule can be violated in two different ways (can have two different root-causes), i.e., no operation is performed or both laparoscopy and open surgery are performed in the same case.” [64]

Prediction Requirements: *precisely determine root causes for predicted compliance violations as soon as possible; provide root cause analysis for single and multiple instances (the latter also in an aggregated manner);*

CMF 9.2 effective communication of root cause.

Example: “when a name laboratoriumonderzoek occurs some of the constraints move to a conflict state since some of them require the execution of vervolgconsult poliklinisch to be satisfied and for others the execution of this activity is forbidden.” [64]

Prediction Requirements: *continuously visualize root causes for predicted compliance violations to users, for single and multiple instances (also in an aggregated manner) for multiple process perspectives and views;*

CMF 9.3 explain and visualize prediction results.

Example: “According to the obtained Shapley values, the high value of OEE in the examined instance (0.95) is strongly associated with a high prediction score in favor of class “Passed”.” [72]

Prediction Requirements: *continuously provide explanations for compliance predictions at algorithmic level (i.e., which input leads to which output) and continuously visualize prediction results in their context, possibly together with providing post hoc explanations (together with CMF9.1);*

CMF 9.4 explain and visualize set of possible/future violations.

Example: Anomaly detection: “integrate root cause representation into anomaly detection, i.e., the results of the anomaly detection should already provide information on root causes; representation by visualization” and “use representation inspired by Ishikawa or “fishbone” diagrams as they have proven useful for root cause identification” [9]

Prediction Requirements *continuously visualize predicted compliance violations together with their root causes and effects (cf. CMF9.1 and CMF9.3); provide visualizations for single and multiple process instances, possibly in an aggregated manner;*

CMF 9.5 explain and visualize effects of mitigation actions.

Example: “In case of suspected resurgence of such incidents, a problem management process should be undertaken with the aim of ascertaining their root causes and adopting the corresponding corrective and preventive procedures.” [71]

Prediction Requirements: *continuously visualize predicted compliance violations together with their mitigation actions and the effects of applying the mitigation actions; provide visualizations for single and multiple process instances, possibly in an aggregated manner;*

CMF 10.1 compliance degree of single traces.

Example “A passenger ship leaving Amsterdam has to moor in Newcastle within 16 h. It is desirable to judge with different degrees of violation a ship arriving in Newcastle after 16 h and 10 min and a ship arriving in Newcastle after 18 h” [64]

Prediction Requirements: continuously exploit predicted probabilities/likelihoods of compliance violations for continuously determining and updating the compliance degree of single process instances;

CMF 10.2 compliance degree of entire process/system.

Example “Several compliance constraints could be violated at the same time. The more are violated, the more serious.” [64]

Prediction Requirements: continuously exploit predicted probabilities/likelihoods of compliance violations for continuously determining and updating the compliance degree across all process instances and processes;

Wish list for user requirement prediction:

Compliance prediction with respect to user CMFs requires approaches to continuously predict and update conflicting compliance constraints and compliance violations as soon as possible with precise probability/likelihood and to exploit this information for precise user feedback. This user feedback comprises explanations at the algorithmic level as well as the visualization of compliance violation predictions and their probabilities. Moreover, the root cause for predicted compliance violations has to be investigated and presented/visualized to users. Finally, mitigation actions based on the compliance violation predictions and the root cause analysis are to be determined, continuously updated, and their effects and updates are to be visualized for users, as well.

4.4 Data requirements

CMF 11 efficiency/performance of PCM.

Example Timely detection of violations is crucial, e.g., if the systems needs more time for predicting a compliance violation, but the violation has already occurred.

Prediction Requirements: provision of performance optimization strategies for compliance prediction and its continuous update based on, e.g., delta approaches; benchmarks with respect to compliance prediction performance in offline and online settings;

CMF 12 integration of data from multiple sources.

Example: “if the loan request is greater or equal to one million, the solvency level of the customer needs to be at least A, a manager needs to process the request, and the solvency information must not be older than two days. [...] the information necessary to check this rule is distributed across multiple systems [57]

Prediction Requirements: transition from basing predictions on label equivalence to equivalence notions based on activity semantics, e.g., attribute equivalence [85] and integration of other process perspectives and case ids;

CMF 13 distributed processes.

Example: “Each Transport intermediate requires Permission of authority. Further on, the transporter must pass a Safety Check.” [28]

Prediction Requirements: transition from basing predictions on label equivalence to equivalence notions based on activity semantics, e.g., attribute equivalence [85] and integration of other process perspectives, case ids, and message ids; providing compliance predictions on event streams/compliance constraints with confidentiality requirements constituted by, e.g., hidden private process information;

CMF 14 context data (internal, external).

Example: “IF temperature > 25 FOR number measurements > 3 THEN discard goods.” [91]

Prediction Requirements: continuously exploit context data for compliance predictions, particularly for prediction at the presence of unseen process and data behavior and for predicting unseen context data behavior; continuously exploit context information for explaining prediction results;

CMF 15 data properties and quality.

Example: “For example, although the average number of activities in the EnvLog dataset is only 44

(compared to 20 for BPI12), the dataset only provides 787 instances for 331 possible activities resulting in a high sparsity of 0.42, whereas BPI12 has a comparably low sparsity of 0.0028.” [40]

Prediction Requirements: consider and exploit properties and quality of the input event streams; interpret data (quality) properties with respect to prediction results; elaborate strategies for dealing with data quality properties and problems under prediction result quality guarantees;

Wish list for data requirement prediction:

Compliance prediction with respect to data CMFs requires approaches for addressing and optimizing the performance of compliance predictions, especially for online predictions and at the presence of a multitude of compliance constraints and event data (\mapsto volume and velocity of the input data). Moreover, the heterogeneity of the input data, i.e., compliance constraints and event data from multiple sources, has to be addressed by novel data integration methods (\mapsto variety). This challenge is aggravated for distributed processes as the input data might also be subject to confidentiality requirements, resulting in partly hidden, invisible data. Finally, compliance prediction results are to be examined with respect to the properties and quality of the input data (\mapsto veracity).

5 Assessment of Existing PPM Approaches

Based on the extended CMF framework presented in Sect. 3 and the prediction requirements presented in Sect. 4, in this section, we assess the coverage of both, CMFs and prediction requirements, by existing approaches. These approaches mainly entail PPM approaches as they provide the predictive capabilities to be put to the test for PCM. The results of the coverage assessment are summarized in Table 5. The explanations and justification of the results are provided in subsequent Sect. 5.1 – 5.6. In detail, Table 5 distinguishes between assessment results for the case that only process behavior is used for PCM prediction that has been already observed and assessment results that are obtained at the presence of

unseen behavior, e.g., an activity/event that has not been observed so far.

Overall, Table 5 shows that in case of observed behavior only, some of the CMFs are supported by existing PPM approaches, either fully (+), partly (\sim), also in combination (c), e.g., when predicting resources, this is connected with predicting next activities, or not supported (–). In the following, the results of the assessment are discussed along the categorization of PPM approaches along their prediction goals such as next activity/event or outcome, partly adopting existing categorizations provided by [70, 34]. Note that, even if a CMF is assessed with $c/+$ this does not imply that a prediction is possible out-of-the-box with existing approaches since a combination has not been realized so far.

5.1 Next Activity/Event Prediction

From the PPM approaches harvested in the literature compilation (cf. Sect. 2.3), we identified 43 papers as next activity / event prediction approaches (note that some of the papers can be classified into multiple categories, for example, approaches that predict the next activity / event and the remaining time of an instance).

For PCM, next activity / event prediction means to make statements about upcoming activities / events that are referred to in one or several compliance constraints. Take the example for CMF0.2 (cf. Sect. 4): “If activity check-out is ever executed, then activity charge must never be executed.” [79]. This compliance constraint refers to activities *check-out* and *charge*.

A first observation is that existing approaches predict next activities if the activities in the compliance constraint have already been observed so far. Absence of an activity can then also be implicitly predicted, based on probabilities. Hence, regarding Table 5, we assess CMF0.1, CMF0.2, and CMF0.3 as covered under the condition that the activities in the compliance constraints have been observed yet (+), and as covered in a preliminary way for unseen behavior (–/ \sim).

When reviewing existing approaches, the output of the approaches did often not become entirely clear.

source	CMF	coverage	
		activities/events observed	unseen behavior
CMF framework [64]	CMF1.2 quantitative		- / ~
	CMF3.1 unary resource cond.	c / +	-
	CMF3.2 extended resource cond.		
	CMF1.1 qualitative		- / ~
	CMF8.1 early detection		
	CMF8.2 predict	c / ~	-
	CMF9.1 root cause		
	CMF10.2 comp. degree entire		
	CMF2.1 activity data		
	CMF2.2 case data		
	CMF4 non-atomic activities		
	CMF5 life cycle	-	-
	CMF8.4 mitigate		
	CMF9.2 effective communication		
CMF10.1 comp. degree single			
Sect. 3.1	CMF11 efficiency/performance	~	-
	CMF12 data integration		
	CMF13 distributed processes	-	
Sect. 3.2	CMF0.1 occurrence		
	CMF0.2 absence	+	- / ~
	CMF0.3 ordering		
	CMF14.1 context data (internal)	c / +	
	CMF14.2 context data (external)	~	
	CMF15 data prop. and quality		-
Sect. 3.4	CMF8.3 update	c / ~	
	CMF9.3 explain & viz. results		
	CMF6.1 simultaneous execution		
	CMF6.2 constrained execution	c / ~	
	CMF6.5 constrained start		
	CMF6.3 order		-
	CMF6.4 non-concurrent execution	-	
	CMF9.4 explain & viz. violations		
CMF9.5 explain & viz. actions			

Table 5: Coverage assessment of extended CMFs by existing literature; +:covered, ~:partly covered, -:not covered, c:combination necessary

For PCM, the prediction requirements state to predict the set of next activities / events, ranked by probability of occurrence and a distinction between immediately/eventually occurs. If we look at more complex compliance constraints referring to several activities and their occurrence/absence and order, a fine-granular prediction feedback with probabilities

would be desired, which is basically possible, but not explicitly provided by any of the approaches.

The next observation is that next activity / event prediction can serve as “anchor” for predicting the modeling requirements CMF1–3 referring to time, data, and resources. Some approaches combine next event/activity prediction with remaining time

and resource prediction. None of the analyzed approaches predicts data values; data, time, and resources are rather used as features to predict the next event/activity.

If the input event stream contains different event types (CMF4, CMF5) such as **start**, **complete**, or **running** (cf. life cycle model for XES [1]), the corresponding event labels are encoded as features, but a distinction of any kind of semantics of the event types is missing (–).

Regarding compliance constraints that span one or multiple process instances and/or processes, none of the existing approaches considers next activity/event prediction across several instances/processes for order (CMF6.3) and non-concurrent execution (CMF6.4). Simultaneous execution (CMF6.1), constrained execution (CMF6.2), and constrained start of following instances (CMF6.5) require the instance-spanning prediction of shared resources or data. Here, inter-case features have been considered w.r.t (remaining) time prediction, e.g., [93, 53] (c/\sim).

Regarding CMF8.1 and CMF8.2, existing approaches can basically provide early detection of compliance violations via next event/activity prediction (see CMF0.1–0.3), but only for control flow related violations (c/\sim). Updates of prediction results, especially compliance violations (CMF8.3), is addressed in a preliminary way by incremental learning approaches that focus on updating the prediction model when changes or drifts occur, e.g., [77, 69] (c/\sim). For CMF8.4, approaches mention that predictions can provide recommendations for users [67], but no mitigation actions/recommendations are actually provided, in particular not in connection with compliance constraints (–).

Root cause analysis (CMF9.1) can be implicitly based on probabilities and feature vectors, i.e., by answering the question whether certain data elements influence the prediction of the next activity/event (c/\sim). However, the effective communication of root causes to users (CMF9.2) is missing (–), although explainability (CMF9.3), e.g., based on features, is targeted by several approaches recently [98] (c/\sim). In particular, visualization approaches for explaining prediction results, predicted violations (CMF9.4), and the effects of mitigation actions (CMF9.5) are

missing (–). Explicit predictions of compliance degrees (CMF10.1 and CMF10.2) are also missing (–).

Regarding efficiency and performance (CMF11), first approaches contribute by applying, for example, scalable online learning algorithms [83] and hyperparameter optimization [32] (\sim).

Next activity/event prediction approaches providing solutions for integration input data streams from multiple, heterogeneous sources (CMF12) and for distributed process settings are missing (CMF13) (–).

The potential of contextual data (CMF14) is mentioned by several approaches. Internal contextual data is exploited by existing approaches such as [12, 45] by encoding them as features for next activity/event prediction ($c/+$). The influence of external contextual data such as sensor streams has not been investigated in great detail yet, only for concept drift prediction [97] (\sim).

The influence of data quality and properties (CMF15) is considered by first approaches that consider data properties [40] and deal with small data sets [47] (\sim).

5.2 Remaining Time, Delay, AnyIndicator

We categorized 60 PPM papers into the category dealing with any kind of temporal prediction. Whereas remaining time and delay are obviously closely related, ‘AnyIndicator’ approaches (cf. categorization as proposed in [70]) often also refer to temporal indicators. Hence, we merged ‘AnyIndicator’ into this category, although other performance indicators might be predicted, as well. We will comment on this within the respective category.

CMF1.1 on qualitative time is addressed implicitly by existing approaches using transition systems, e.g., [16, 2] in combination with next activity / event prediction approaches (c/\sim). Existing approaches for remaining time/delay in combination with next activity / event prediction cover CMF1.2 on quantitative time ($c/+$).

Activity and case data (CMF2.1 and CMF2.2), similarly to next activity / event prediction, is used for temporal prediction (as features), but not predicted by any of the existing approaches (–), al-

though this could be meaningful for temporal activity and case data.

Resource prediction (CMF3.1 and CMF3.2) in connection with temporal prediction is mostly seen from a scheduling perspective, i.e., how to determine and avoid potential temporal problems such as bottlenecks by assigning resources [89, 94]. Other approaches utilize resources as features for temporal prediction [31].

CMF4 and CMF5 are not covered (–).

For the simultaneous execution of process instances (CMF6.1), inter-case features for batching (i.e., executing process instances in one batch) are used in order to improve remaining time predictions [53, 81] (c/\sim). CMF6.2–6.4 are not covered by temporal prediction approaches (–). CMF6.5 is only touched upon, i.e., [31] predict how many instances will start in a particular time window (c/\sim).

CMF8.2 on predicting a set of possible future compliance violations is addressed by [8] in terms of SLA violations (c/\sim). Updates regarding the prediction of compliance violations (CMF8.3) are not explicitly mentioned, but could be tackled based on continuous task monitoring through sliding windows as proposed in, e.g., [22]. Alerts [13] and dashboards [80, 29] provide information on predictions to users and can help to avoid violations (CMF8.4), but do not yet suggest mitigation actions.

Root cause analysis (CMF9.1) can be prepared based on approaches such as [45] that explain the influence of attributes on the prediction results (c/\sim). CMF9.3 on explaining and visualizing results is implicitly supported via helping to choose parameters by [7] and by [53] in the context of inter-case features for batching. Explaining and visualization compliance violations and the effects of mitigation actions (CMF9.4 and CMF9.5) are missing (–).

For the assessment of compliance degrees, there is no approach for single instances (CMF10.1). By providing temporal predictions for inter-case features, [93] can contribute partly to CMF10.2, i.e., the prediction of the compliance degree (‘healthiness’) of entire process (c/\sim).

Performance/efficiency (CMF11) of temporal predictions is addressed by “*in a parallel and distributed manner, on top of a cloud-based service-oriented in-*

frastructure” [17] (\sim). Textual data (CMF12 on data from multiple sources) is utilized by [78] for temporal prediction (\sim). There are no approaches for temporal predictions in distributed processes (CMF13, –). Internal context data (CMF14.1) is utilized by [30] for temporal prediction (c/\sim), but none of the temporal prediction approaches exploits external context data (CMF14.2, –). Temporal prediction approaches concerned with data quality and properties (CMF15) are currently missing (–).

5.3 Outcome, AnyIndicator

26 approaches are categorized as outcome prediction approaches. Some ‘AnyIndicator’ approaches have been assessed in Sect. 5.2. Some ‘AnyIndicator’ approaches are also related to outcome and hence will be mentioned in this section, as well. Note that outcome prediction overlaps with remaining time prediction for several approaches, e.g., [92], i.e., prediction of the remaining time for a case is often the target for outcome prediction, too (CMF1.2, $c/+$). Data and resources are used as features for prediction, for example by [52], but neither serve as prediction goals themselves nor contribute to any compliance-related prediction.

Approaches that exploit life cycle information in the event logs/streams (CMF4 and CMF5) are missing (–).

CMF6.2 on constrained execution or processes is touched upon by ‘AnyIndicator’ approaches by enabling aggregated PPIs that might include data constraints, e.g., [22] (c/\sim). Approaches addressing CMF6.1 and CMF6.3–6.5 are missing (–).

Prescriptive monitoring approaches can foster the early detection of compliance violations (CMF8.1) and the preparation of mitigation actions (CMF8.4). [26], for example, enable the generation of alarms that trigger interventions to prevent an undesired outcome or mitigate its effect. Similarly, [109] “*supports the proactive handling of deviations, i.e. inserted and missing events in process instances, to reduce their potential harm*”. However, both approaches do not provide any recommendations or mitigation actions.

For root cause analysis of predicted compliance violations (CMF9.1), [38] provides input by visualization the impact of activities on the predicted outcome (c/\sim). There are no approaches that explain and visualize prediction results (CMF9.3), but rather provide metrics for the quality of the prediction results, for example, stability [51] and reliability [54]. Aside these quality metrics, CMF9.1–9.4 are not addressed by outcome prediction approaches. For CMF10.1 and CMF10.2, please see the discussion of [93] in Sect. 5.2 on temporal prediction.

For CMF12 on the integration of data from multiple sources, [78, 103] deal with structured and unstructured data, i.e., textual data, as input for PPM (\sim). CMF15 on data properties and quality is addressed by [54] with respect to reliability of the predictions (\sim). CMF11, CMF13, CMF14.1, and CMF14.2 are not addressed by existing outcome prediction approaches.

5.4 Resources

[14] predict the resource / resource pool an upcoming event will be assigned to. Hence, CMF4.1 and CMF4.2 can be assessed with $c/+$.

5.5 Predicates

We identified 2 distinct approaches that deal with predicting possible violations of predicates that range from simple SLA [59] to LTL based formulae [67]. The predicates refer to control flow and time. Hence, CMF0.1–0.3 are covered (+) if the activities have been already observed. [67] is also able to deal with quantitative time prediction (CMF1.2). Doing so, the approaches are able to predict certain compliance violations early (CMF8.1, c/\sim), but do not provide updates of the predictions, visualizations, root cause analysis, and mitigation actions. Moreover, existing approaches are not concerned with data integration or quality.

5.6 Cost and Risk

3 papers deal with cost or risk prediction. There is a partial overlap with delay prediction as discussed

in Sect. 5.2, as delay can be seen as risk parameter. In [21], activity data (CMF2.1) is used for prediction, i.e., cost for executing tasks is seen as a risk parameter. Moreover, the approach aims to “*compute the optimal assignment of resources to tasks*” [21] (CMF3.1 and CMF3.2) and at risk aggregation over multiple instances (CMF6.2). However, for CMF3.1 and CMF3.2, no resource prediction is made. The risk prediction results rather in prediction violations of Performance Indicators or SLAs than in compliance violations (CMF8.2). These risk predictions are provided to users as recommendations (CMF8.4) which can also partly serve as mitigation actions for lowering risk for specific risk types. However, these recommendations do not target compliance violations. Regarding the analysis of root causes in the context of predictions (CMF9.1), [21] provide risk types. [7] support users in setting the parameters for the prediction which can be seen as a step towards explaining visualization results (9.3).

6 Open Challenges and Research Directions

The discussion of existing PPM approaches in Sect. 5.1 – 5.6 and the results shown in Table 5 summarize the potential of existing PPM approaches for a comprehensive PCM support. Within this section we want to summarize the open PCM challenges (cf. Sect. 6.1) and set out research directions for PCM and PPM (cf. Sect. 6.2).

6.1 Open PCM Challenges

1. **Treatment of unseen behavior and data:** Activity occurrence, absence, and ordering of activities in compliance constraints is covered by existing PPM approaches for those activities that have been already observed. Unseen behavior remains largely uncovered. Unseen behavior can occur in event streams if the underlying process model is not or only partly known or due to concept drift. In combination with compliance constraints, the requirement to predict unseen process behavior becomes even more likely

as the compliance constraints do not have to be part of either an underlying process model or the observed behavior in the event stream. Similar observations also hold for the treatment of unseen data and unseen data values (internal and external data).

2. **Prediction of data:** The prediction of time and resources is tied in with next activity prediction and covered by existing PPM approaches. What is still missing is the prediction of data values which are considered as features for prediction by existing approaches, but not predicted themselves. However, for the assessment of and taking decision on compliance the prediction of data values might be of great interest for constraints that refer to, for example, certain thresholds in medicine, logistics, and manufacturing. Here, prediction can either encounter already observed data values from historical data or even go beyond that by assuming that some data values have not been observed so far. How can we then update the models to consider newly observed behavior linked to data.
3. **Life cycle handling:** None of the existing PPM approaches exploits the life cycle of activities, i.e., exploits the semantics of distinct life cycle states/transitions of activities in the event stream. However, these life cycle states/transitions might contribute to predict, for example, the activity duration or might indicate exceptional behavior, resulting in unseen behavior or drift, and subsequently necessitating adequate mitigation actions.
4. **Instance and process spanning constraints:** Predicting the compliance of constraints that span multiple process instances and/or multiple processes has not been explicitly addressed by existing approaches, except for using inter-case features for predicting next activities or predicting performance indicators across multiple instances. However, many application domains crave for compliance support in instance and process spanning settings, e.g., logistics, medicine, and manufacturing.
5. **Visualization of predictions and violations:** Explainability of prediction results has gained attention. However, visualization approaches for prediction results, especially future compliance violations are missing. Moreover, root cause analysis has to be extended in order to deal with predicting violations of real-world compliance constraints.
6. **Provision of mitigation actions:** Though recommendations are used to support users in taking counteractions regarding delays or other risks, none of the approaches suggests mitigation actions to overcome compliance violations. In particular, approaches are missing that provide mitigation actions at different granularity levels, analyze and visualize the effects of applying mitigation actions, and provide users with estimations on their significance.
7. **Distributed process data and data quality:** First approaches start addressing data requirements in PPM and PCM. However, approaches for event and constraint data from distributed and heterogeneous sources and processes are missing. Moreover, the ongoing exploitation of contextual data as well as of data quality is promising, but under-researched yet.
8. **Compliance degree and update:** First approaches for predicting the compliance degree across multiple process instances have been presented. Yet, especially in combination with updating compliance violations, an open challenge remains how to define and update the compliance degree while new events arrive throughout the event stream and to predict compliance states of single instances.

6.2 Research Directions

Based on the list of open challenges discussed in Sect. 6.1, the following research directions can be derived:

Treatment of unseen behavior and data: Recently, strategies on how to update the prediction

model in case of unseen process behavior in the context of next activity prediction are proposed, including “do nothing”, “retrain without hyperparameter optimization”, “full retrain”, and “incremental update” [87, 77, 69]. In addition, we need to consider not just updating the model whenever unseen behavior has occurred, but also how to predict the unseen behavior as such, e.g., by considering available context data. Consider the following PCM challenges with unseen behavior:

- *Unseen tasks in compliance constraints:*

Example: Assume that $T_L = \{A, B, C, D, E\}$ is the set of observed tasks in the event log/stream L and $T_C = \{A, F\}$ is the set of observed tasks for given compliance constraint set C imposed on L (cmp. Def. 1). Assuming that $c \in C$ only refers to control flow and the given tasks, the following compliance constraints are conceivable

$$C = \{c_1 : A, c_2 : \neg A, c_3 : F, c_4 : \neg F, \\ c_5 : A \mapsto F, c_6 : A \mapsto \neg F, c_7 : \neg A \mapsto F, \\ c_8 : \neg A \mapsto \neg F\}^4.$$

Following [68], we distinguish compliance states possibly violated/satisfied and violated/satisfied. Possibly violated means that the violation can be still healed, i.e., by the occurrence of an activity that is mandatory according to the compliance constraint (cf. [65]). A (final) violation, by contrast, states that the constraint cannot be healed anymore, e.g., if the constraint is possibly violated and then the end event of a process instance or all end events occur. There is also a distinction between fully and partly violated/satisfied where full violation of a compliance constraint means that this constraint is violated for all process instances, and a partly violation means that the constraint is violated for at least one constraint [105].

For reasoning on the occurrence and treatment of unseen behavior, three cases illustrated by the example can be distinguished:

1. Case 1: task F has not been observed for L (denoted as $\neg F$).

⁴ c_1 and c_2 are not (directly) affected by the observation of F . Hence, we will not follow up on these constraints.

Compliance constraint c_3 is possibly violated and c_4 possibly satisfied for all process instances. To reason about (final) violation or satisfaction of c_3, c_4 , either F has to be observed (see next case) or we have to assume the existence of explicit end events in the event stream. For c_5 , as soon as A as been observed, the constraint will be flagged as as possibly violated as long as F is not observed (Case 2) or an end event (if known) occurs. The same considerations as for c_5 hold for c_7 as long as A is not observed. For c_6 and c_8 , the occurrence/non-occurrence of A triggers the non-occurrence of F which is possibly satisfied for this case (and satisfied if an end event occurs).

2. Case 2: task F is observed (denoted as F). The compliance state of constraint c_3 is updated from possibly violated to satisfied and the state of c_4 is updated to violated. For c_5 , if A has been observed and no end event has been observed yet, the compliance state is updated from possibly violated to satisfied. For c_6 , if A has been observed and no end event has been observed yet, the compliance state of c_6 is set from satisfied to violated. For c_7 , if A has not been observed yet, and F is observed, the compliance state has to be updated from possibly violated to satisfied. In case of c_8 , if A has not been observed yet, and F is observed, the compliance state has to be updated from possibly satisfied to violated.
3. Case 3: Knowledge that F is referred to by compliance constraints is exploited for prediction. Even if F has not been observed yet, the knowledge that one or several compliance constraints that are imposed on L , refer to F (e.g., c_3), could be incorporated into the prediction model, to reason about compliance at least with some adapted probabilities. This can support the prediction that some unseen behavior might occur or even more precisely that F will occur with some

probability. This can be refined by considering the occurrence of F depending on the occurrence of an already observed (or even not-observed) activity (e.g., c_5).

- **Unseen behavior with respect to data, time, and resources:** This is still an open challenge in PPM and PCM overall. Temporal and resource prediction approaches exist (cf. Sect. 6), but have not dealt with unseen temporal and resource behavior yet. Even for historical data, data prediction approaches are missing. Consequently there are no approaches for predicting unseen data and their values on event streams during runtime.

Overall, the treatment of unseen behavior, data, and resources necessitates update strategies of prediction models and the continuous update of the violations predictions (see also subsequent challenges).

Data prediction: As stated before approaches for predicting data values are missing. However, for compliance constraints, the prediction of data and data values is crucial. Assume, for example, a compliance constraint from the logistics domain on a transportation process that states if the transport takes up to 5 hours, the destination of the transport is 'London', if the transport takes longer than 5 hours, the destination is 'Berlin'. Predictions regarding this constraint necessitate the prediction of remaining time plus location. Additionally, data value prediction might not only refer to process data, but also to contextual data such as time series, for example, if decision rules are based on time series data such as temperature [91]. Here the combination of PCM with time series prediction approaches constitutes a promising research direction [43].

Continuous update of prediction results and compliance violations: This aspect is closely related to the consideration on unseen behavior. Even for simple compliance constraints, unseen behavior might necessitate updates of the compliance states: an example is c_3 which is currently violated. As soon as F is observed, the compliance state has to be updated to fulfilled. The need for continuous update is

even aggravated for compliance constraints that refer to time, data, and resources, in particular, if contextual, time series data with possibly continuously changing values, is considered.

Online PCM, online (re-)training of prediction models: It is a difference to have historical data available to train the prediction model and then constantly update it whenever new information comes in and starting from scratch and having to learn and train without any previous knowledge. This has been addressed by online process mining approaches and would need to be investigated for the PCM problem. Recapitulate Example 2 from Sect. 1 containing an evolving event stream. If we, e.g., know the set of compliance constraints and their possible violations, we can try to estimate the probability of a violation already after the first incoming event $e_1 = (A, t_1, 30)$. Here, one could assume that since, based on the constraint we know that A, B or C are in principal conceivable, we could go for a violation probability of $2/3$. Yet, the certainty of the prediction would not be very high at this stage. After the second event $e_2 = (B, t_2)$ has arrived, we could update the prediction model since we have seen that no violation has occurred so far and could conclude that the probability of a violation is lower than initially expected.

Assess compliance violation risk: Prediction results include probabilities, e.g., how likely is the occurrence of a certain activity. For more complex compliance constraints referring to activities, data, time, and resources, the probabilities of satisfying/violating these constraints have to be calculated in an adequate manner, e.g., how likely is the occurrence of a certain activity producing a certain data value in a given time span? If these probabilities can be determined, in turn, the risk of violations can be assessed. As for the continuous update of compliance violations predictions, their probabilities and, in turn, risks have to be continuously updated.

Relate classifier with compliance constraints: This challenge is related to enabling a fine-grained root cause analysis. Some PPM approaches train

classifiers based on available data and use them to, e.g., predict the next event/activity. Consider, e.g., a compliance constraint stating that “for premium customers any loan request below € 100 will in any case be granted without performing an additional check”. If a decision tree algorithm is trained to predict the next activity based on event attributes *customer type* and *requested loan amount*, we would be able to determine the underlying decision rule, i.e., (*customer type = premium & requested loan amount < 100*) \rightarrow *grant loan immediately*, otherwise *perform additional check*. In order to enable a fine-grained root cause analysis, we need to link this decision rule to the given constraint.

Behavioral aspect of constraints spanning multiple processes and process instances:

State-of-the-art PPM approaches mostly focus on predicting next activity/event within the context of single instances. However, when considering compliance constraints that span across multiple processes and process instances, it becomes necessary to predict interactions between the affected process instances and their behavior, as well. In particular, such compliance constraints refer to data and/or resources shared by processes/instances. Take as an example the compliance constraint: “*Each clerk is allowed to issue approve loan as long as a threshold (around \$1M) is not reached. Otherwise he has to delay this event to the following day*” [110]. First of all, we can see that the constraint imposes a condition across several process instances that refers to data element ‘threshold’ and implicitly to time (‘within one day’). Hence, compliance predictions across multiple processes and process instances have to consider a combination of the control flow, data, time, and resource perspective. Moreover, constraints spanning multiple instances and processes typically state and trigger actions, e.g., delaying instance execution until the following day. Incorporating the effects of this behavior imposed by the constraints into the prediction is of utmost importance.

Life cycle handling: Incorporating and exploiting life cycle states and transitions into PPM and

PCM might tremendously increase prediction quality and applicability in real-world settings. Consider, for example, the transportation of goods to different locations. By distinguishing the **start** and **complete** events of activities, activity duration can be considered in the prediction. By exploiting more life cycle states such as **suspend** and **abort**, upcoming exceptions might be predicted and exception handling actions be defined and taken. Assume that, for example, for activity ‘transport’ a **suspend** event occurs. This might result in delay which should be incorporated in a temporal prediction, e.g., of the remaining time of the affected instance. If an **abort** event occurs for ‘transport’, we can conclude that the transport will not be completed (i.e., event **complete** will not occur for ‘transport’) and this might result in a compliance violation.

Mitigation actions: Based on compliance violation predictions combined with root cause analysis, the effects of mitigation actions can be assessed; either by simulating what will happen if a user applies a specific countermeasure or by determining and suggesting mitigation actions for avoiding the compliance violation. Consider again the transportation example provided in research direction *Life cycle handling*. Assume that based on data gathered before and during transportation, the transportation is predicted to be aborted for a certain process instance. Based on the prediction, we can immediately start to define countermeasures for avoiding the compliance violation of not arriving at the destination, together with predicting their effects. One possible mitigation action in this case is to start another transportation process arriving on time. The prediction can then estimate whether or not the application of this countermeasure compensates the failure.

Provision, explanation, and visualization of compliance violations (beyond SLA):

PCM requires an aggregated view on several perspectives, including the compliance constraints and the process respectively process instance perspective, i.e., a view on the current event stream combined with continuous updates. Though some approaches already pro-

vide visualizations for simple SLA violations including color coding, e.g., red means the SLA is violated, green the SLA is not violated, there is still room for improvement and extensions when considering complex compliance constraints. Therefore, visualization approaches are required to depict possible complex information at once, i.e., all compliance states for all processes and process instances captured by the event stream, as well as the definition and visualization of single views, e.g., visualization of compliance predictions for one constraint, one particular instance, or one perspective such as time. Moreover, information on root causes for compliance violations, the current prediction model in use, and mitigation actions together with their effects should be conveyed to users based on visualization approaches.

Input data (quality, volume, variety, velocity, confidentiality): One drawback of existing PPM approaches with respect to the input data is the assumption of label equivalence, i.e., the prediction are based on labels of events. Label equivalence is not sufficient, particularly when merging event streams from heterogeneous input sources (variety). Here, PPM and PCM can benefit from equivalence notions that aim at the semantics and functionality of activities, e.g., attribute equivalence [57]. Another challenge is the size of the input data which can be too small or too big (volume). First approaches for boosting small data sets [48, 42] have been proposed; approaches aiming at efficiency and performance of PPM and PCM with respect to both, volume and high velocity event streams are missing (note that for PCM also a large set of compliance constraints might exist). For distributed processes, event streams might contain information on message exchanges between the partners –how can they be exploited for prediction or being predicted themselves?– and might also contain hidden/invisible parts due to confidentiality requirements of the partners. An initial hurdle is the lack of data sets. Hence, the collection, provision, and preparation of (real-world) data for different process scenarios containing multiple perspectives remains an ongoing community challenge.

External context data: Including context data into PPM and PCM can significantly increase prediction capabilities and quality. In manufacturing processes, for example, several sensor data streams are measured continuously that report the environment state/context of the process, e.g., the room/machine temperature or the fluid level in the machine. Detecting deviations in the context data can increase prediction effectiveness, e.g., concept drifts might be predicted early [97]. This can be furthered by augmenting PCM by predictions of the context data. One challenge is encode context data, especially for a multitude of data streams that might also influence each other. Moreover, strategies for updating prediction models at the presence of continuous context data have to be elaborated, e.g., constant updates versus updates if significant changes in the context data occurs.

Systematic assessment of data mining/machine learning techniques: In the light of a multitude of challenges and research directions, the systematic assessment of (existing) prediction techniques to address these challenges is required. It is likely that not each problem can be tackled by existing data mining/machine learning techniques. Therefore it might be necessary to develop new techniques.

7 Limitations

The survey follows the research method depicted in Fig. 1. Each of the steps produces an output which serves as input for the subsequent step with the goal to generate and present contributions in a rigorous way. We followed selected principles of conducting a systematic literature review [10] and adapted them in terms of incorporating existing surveys as basis whenever possible. In this spirit, we took the established CMF framework [64] and extended it based on more recent findings and utilized existing PPM categorizations such as [70]. Despite this careful method design, the following limitations for this work can be identified.

- New approaches on PPM are published constantly. Therefore, one limitation of this work is that new approaches since the literature compilation in February might have been published which are consequently not covered within this paper yet. A search on Google Scholar with `allintitle:predictive process monitoring` and selecting papers after 2022 results in 12 hits⁵. From these 12 hits, this survey covers [86, 45, 52]. 9 papers are not covered, out of which 7 papers have been published as technical reports and 1 as PhD thesis. Looking at this most recent work, the majority of the approaches is concerned with explainability, some combined with data issues such as [25] (cf CMF9, CMF14, and CMF15).
- The main focus of this paper is on prediction tasks and compliance monitoring. Hence, further related areas such as online process mining, concept drift detection, and anomaly detection approaches have only been considered if papers from these areas were detected during the systematic literature review. Online process mining is often geared towards concept drift detection. This work covers several concept drift detection approaches [69, 97], also in connection with updating prediction models at the presence of concept drift [87, 77]. Anomaly detection can provide insights to PCM. However, most anomaly detection approaches work offline, some can be applied on event streams, e.g., [56], but process anomaly predictions beyond the approaches studied in this work such as [9] are missing.
- If surveys exist for the investigated research areas, i.e., for CM and PPM, we used these surveys as a basis for our further literature analysis. Doing so might result in missing papers that have been published prior to the existing surveys and not having been treated by them. However, we conducted a full search without restricting the publication dates first and then compared the identified set of papers to existing surveys. Doing so, we limit the risk of missing out relevant

prior work.

- As already mentioned within the research direction *systematic assessment of data mining techniques* the aim of this paper is not to assess PPM approaches in terms of machine learning or data mining techniques in detail, i.e., the goal is not to identify the PPM approach currently performing best. Instead, the goal is to provide a comprehensive outline and analysis of the PCM problem and how it is addressed by current literature. Hence, at this point, we do not investigate or propose particular techniques from a technical point of view.
- Though there are case studies for compliance monitoring available (\mapsto Sect. 3.4), we still need to have a detailed look and investigate whether these are suitable for evaluating approaches tackling the mentioned research directions.

8 Conclusion

This work provides a comprehensive overview and analysis of the Predictive Compliance Monitoring (PCM) problem and has tackled research questions RQ1 – RQ4 (cf. Sect. 1) as summarized in the following. In addition to findings on PCM requirements and approaches, the study particularly provides findings on predictive process monitoring and its capabilities.

RQ1: Which PCM approaches exist? RQ1 is addressed by an extensive compilation of literature on PCM and its related fields compliance monitoring, predictive process monitoring, and service level agreement prediction. Based on analyzing the literature, we conclude that the PCM problem in general has not been addressed by now, i.e., no specific PCM approaches exist.

RQ2: Which functionalities must PCM approaches address? The selected literature from compliance monitoring, predictive process monitoring and service level agreement prediction emphasizes that the compliance monitoring functionalities as originally

⁵accessed 2022-05-07

proposed by [64] in 2015 are still valid and can serve as requirements for the PCM problem. The CMF framework is extended based on analyzing the literature compilation regarding compliance monitoring directions after 2015, most prominently, towards the explainability of the prediction results and input data requirements.

RQ3: Which PCM functionalities are covered by existing approaches? How are they covered? Papers explicitly addressing the PCM problem are missing. In general, predictive process monitoring holds the capabilities to tackle PCM challenges. These capabilities are formulated as “PPM-PCM wish list” based on the extended CMF framework. In order to put existing approaches from the literature compilation to the test, we categorize and assess them along the wish list. The assessment finds some capabilities to be (partly) supported, but there is no comprehensive solution for all PCM challenges, i.e., the assessment eventually results in a list of open PCM challenges.

RQ4: Which open challenges and research directions remain for full PCM support? Based on the identified open PCM challenges, together with the PPM-PCM wish list, research directions for predictive process and compliance monitoring are elaborated. These research directions comprise the prediction of data and data values, the treatment of unseen process behavior and data, the explainability of compliance violation predictions, and the explicit treatment of data quality and properties, also for distributed and heterogeneous processes and data sources. All of these open challenges constitute key success factors for predictive process and compliance monitoring.

The research directions point to several future research opportunities. Working on the research directions will necessitate a comprehensive assessment of existing machine learning and data mining techniques and might result in the development of extended or even new techniques. Moreover, this work assumes that compliance constraints are formalized using some notion. In future work, we will incorporate the sources, e.g., regulatory documents, into predictive compliance monitoring. In order to eval-

uate and compare new techniques and approaches, appropriate data sets are crucial, i.e., event streams, contextual data, data from different sources and processes, and unseen data.

References

- [1] IEEE standard for extensible event stream (xes) for achieving interoperability in event logs and event streams. *IEEE Std 1849-2016*, pages 1–50, 2016. doi:10.1109/IEEESTD.2016.7740858.
- [2] Ahmad Aburomman, Manuel Lama, and Alberto Bugarín. A vector-based classification approach for remaining time prediction in business processes. *IEEE Access*, 7:128198–128212, 2019. doi:10.1109/ACCESS.2019.2939631.
- [3] Grigoris Antoniou, George Baryannis, Sotiris Batsakis, Guido Governatori, Mohammad Badiul Islam, Qing Liu, Livio Robaldo, Giovanni Siragusa, and Ilias Tachmazidis. Large-scale legal reasoning with rules and databases. *FLAP*, 8(4):911–940, 2021.
- [4] Ahmed Awad, Sherif Sakr, and Amal Elgammal. Compliance Monitoring as a Service: Requirements, Architecture and Implementation. In *Cloud Computing*, pages 1–7, 2015. doi:10.1109/CLOUDCOMP.2015.7149636.
- [5] Ahmed Barnawi, Ahmed Awad, Amal Elgammal, Radwa Elshawi, Abdullallah Almalaise, and Sherif Sakr. An Anti-Pattern-based Runtime Business Process Compliance Monitoring Framework. *Advanced Computer Science and Applications*, 7, 2016.
- [6] Ahmed Barnawi, Ahmed Awad, Amal Elgammal, Radwa El Shawi, Abdullallah Almalaise, and Sherif Sakr. Runtime self-monitoring approach of business process compliance in cloud environments. *Cluster Computing*, 18(4):1503–1526, December 2015. doi:10.1007/s10586-015-0494-0.

- [7] Nico Bartmann, Stefan Hill, Carl Corea, Christoph Drodt, and Patrick Delfmann. Applied predictive process monitoring and hyper parameter optimization in camunda. In *CAiSE Forum 2021*, pages 129–136, 2021. doi:10.1007/978-3-030-79108-7_15.
- [8] Antonio Bevacqua, Marco Carnuccio, Francesco Folino, Massimo Guarascio, and Luigi Pontieri. A data-driven prediction framework for analyzing and monitoring business process performances. In *Enterprise Information Systems*, pages 100–117, 2013. doi:10.1007/978-3-319-09492-2_7.
- [9] Kristof Böhmer and Stefanie Rinderle-Ma. Mining association rules for anomaly detection in dynamic process runtime behavior and explaining the root cause to users. *Inf. Syst.*, 90:101438, 2020. doi:10.1016/j.is.2019.101438.
- [10] Pearl Brereton, Barbara A. Kitchenham, David Budgen, Mark Turner, and Mohamed Khalil. Lessons from applying the systematic literature review process within the software engineering domain. *J. Syst. Softw.*, 80(4):571–583, 2007. doi:10.1016/j.jss.2006.07.009.
- [11] Jens Brunk. Structuring business process context information for process monitoring and prediction. In *Business Informatics*, pages 39–48, 2020. doi:10.1109/CBI49978.2020.00012.
- [12] Jens Brunk, Matthias Stierle, Leon Papke, Kate Revoreda, Martin Matzner, and Jörg Becker. Cause vs. effect in context-sensitive prediction of business process instances. *Inf. Syst.*, 95:101635, 2021. doi:10.1016/j.is.2020.101635.
- [13] Cristina Cabanillas, Claudio Di Ciccio, Jan Mendling, and Anne Baumgrass. Predictive task monitoring for business processes. In *Business Process Management*, pages 424–432, 2014. doi:10.1007/978-3-319-10172-9_31.
- [14] Manuel Camargo, Marlon Dumas, and Oscar González Rojas. Learning accurate LSTM models of business processes. In *Business Process Management*, pages 286–302, 2019. doi:10.1007/978-3-030-26619-6_19.
- [15] Malu Castellanos, Fabio Casati, Umeshwar Dayal, and Ming-Chien Shan. A Comprehensive and Automated Approach to Intelligent Business Processes Execution Analysis. *Distributed and Parallel Databases*, 16(3):239–273, November 2004. doi:10.1023/B:DAPD.0000031635.88567.65.
- [16] Michelangelo Ceci, Pasqua Fabiana Lanotte, Fabio Fumarola, Dario Pietro Cavallo, and Donato Malerba. Completion time and next activity prediction of processes using sequential pattern mining. In *Discovery Science*, volume 8777, pages 49–61, 2014. doi:10.1007/978-3-319-11812-3_5.
- [17] Eugenio Cesario, Francesco Folino, Massimo Guarascio, and Luigi Pontieri. A cloud-based prediction framework for analyzing business process performances. In *Workshop on Privacy Aware Machine Learning for Health Data Science*, pages 63–80, 2016. doi:10.1007/978-3-319-45507-5_5.
- [18] Giuseppe Cicotti, Luigi Coppolino, Salvatore D’Antonio, and Luigi Romano. Runtime model checking for SLA compliance monitoring and qos prediction. *J. Wirel. Mob. Networks Ubiquitous Comput. Dependable Appl.*, 6(2):4–20, 2015. doi:10.22667/JOWUA.2015.06.31.004.
- [19] Marco Comuzzi. Alignment of process compliance and monitoring requirements in dynamic business collaborations. *Enterprise Information Systems*, 11(6):884–908, 2017. doi:10.1080/17517575.2015.1135482.
- [20] Marco Comuzzi, Alfonso E. Márquez Chamorro, and Manuel Resinas. A hybrid reliability metric for SLA predictive monitoring. In *Symposium on Applied Computing*, pages 32–39, 2019. doi:10.1145/3297280.3297285.

- [21] Raffaele Conforti, Massimiliano de Leoni, Marcello La Rosa, Wil M. P. van der Aalst, and Arthur H. M. ter Hofstede. A recommendation system for predicting risks across multiple business process instances. *Decis. Support Syst.*, 69:1–19, 2015. doi:10.1016/j.dss.2014.10.006.
- [22] Alfredo Cuzzocrea, Francesco Folino, Massimo Guarascio, and Luigi Pontieri. A predictive learning framework for monitoring aggregated performance indicators over business process events. In *International Database Engineering & Applications Symposium*, pages 165–174, 2018. doi:10.1145/3216122.3216143.
- [23] Massimiliano de Leoni, Wil M. P. van der Aalst, and Marcus Dees. A general process mining framework for correlating, predicting and clustering dynamic behavior based on event logs. *Inf. Syst.*, 56:235–257, 2016. doi:10.1016/j.is.2015.07.003.
- [24] Matthias Ehrendorfer, Juergen Mangler, and Stefanie Rinderle-Ma. Assessing the impact of context data on process outcomes during runtime. In *Service-Oriented Computing*, volume 13121, pages 3–18, 2021. doi:10.1007/978-3-030-91431-8_1.
- [25] Ghada El-Khawaga, Mervat Abu-Elkheir, and Manfred Reichert. Explainability of predictive process monitoring results: Can you see my data issues? *CoRR*, abs/2202.08041, 2022. URL: <https://arxiv.org/abs/2202.08041>.
- [26] Stephan A. Fahrenkrog-Petersen, Niek Tax, Irene Teinemaa, Marlon Dumas, Massimiliano de Leoni, Fabrizio Maria Maggi, and Matthias Weidlich. Fire now, fire later: alarm-based systems for prescriptive process monitoring. *Knowl. Inf. Syst.*, 64(2):559–587, 2022. doi:10.1007/s10115-021-01633-w.
- [27] Walid Fdhila, Manuel Gall, Stefanie Rinderle-Ma, Juergen Mangler, and Conrad Indiono. Classification and formalization of instance-spanning constraints in process-driven applications. In *Business Process Management*, pages 348–364, 2016. doi:10.1007/978-3-319-45348-4_20.
- [28] Walid Fdhila, David Knuplesch, Stefanie Rinderle-Ma, and Manfred Reichert. Verifying compliance in process choreographies: Foundations, algorithms, and implementation. *Information Systems*, page 101983, 2022. doi:<https://doi.org/10.1016/j.is.2022.101983>.
- [29] Filipe Ferreira, Ahm Shamsuzzoha, Américo Lopes Azevedo, and Petri Helo. Virtual enterprise process monitoring: An approach towards predictive industrial maintenance. In *Progress in Systems Engineering*, pages 285–291, 2014. doi:10.1007/978-3-319-08422-0_43.
- [30] Francesco Folino, Massimo Guarascio, and Luigi Pontieri. Discovering context-aware models for predicting business process performances. In *On the Move to Meaningful Internet Systems*, pages 287–304, 2012. doi:10.1007/978-3-642-33606-5_18.
- [31] Francesco Folino, Massimo Guarascio, and Luigi Pontieri. A prediction framework for proactively monitoring aggregate process-performance indicators. In *Enterprise Distributed Object Computing*, pages 128–133, 2015. doi:10.1109/EDOC.2015.27.
- [32] Chiara Di Francescomarino, Marlon Dumas, Marco Federici, Chiara Ghidini, Fabrizio Maria Maggi, Williams Rizzi, and Luca Simonetto. Genetic algorithms for hyperparameter optimization in predictive business process monitoring. *Inf. Syst.*, 74:67–83, 2018. doi:10.1016/j.is.2018.01.003.
- [33] Chiara Di Francescomarino, Marlon Dumas, Fabrizio Maria Maggi, and Irene Teinemaa. Clustering-Based Predictive Process Monitoring. *IEEE Transactions on Services Computing*, 12(6):896–909, November 2019. Conference Name: IEEE Transactions on Services Computing. doi:10.1109/TSC.2016.2645153.

- [34] Chiara Di Francescomarino, Chiara Ghidini, Fabrizio Maria Maggi, and Fredrik Milani. Predictive process monitoring methods: Which one suits me best? In *Business Process Management*, pages 462–479, 2018. doi:10.1007/978-3-319-98648-7_27.
- [35] Manuel Gall and Stefanie Rinderle-Ma. Evaluating compliance state visualizations for multiple process models and instances. In *Business Process Management Forum*, volume 427, pages 126–142, 2021. doi:10.1007/978-3-030-85440-9_8.
- [36] Ping Gong, David Knuplesch, Zaiwen Feng, and Jianmin Jiang. bpCMon: A Rule-Based Monitoring Framework for Business Processes Compliance. *Web Services Research*, 14(2):81–103, 2017. doi:10.4018/IJWSR.2017040105.
- [37] Nitin Harane and Sheetal Rathi. Comprehensive Survey on Deep Learning Approaches in Predictive Business Process Monitoring. In *Modern Approaches in Machine Learning and Cognitive Science: A Walkthrough: Latest Trends in AI*. 2020. doi:10.1007/978-3-030-38445-6_9.
- [38] Maximilian Harl, Sven Weinzierl, Mathias Stierle, and Martin Matzner. Explainable predictive business process monitoring using gated graph neural networks. *Journal of Decision Systems*, 29:312–327, 2020. doi:10.1080/12460125.2020.1780780.
- [39] Mouna Hedjeres, Abdellah Boukerram, Samir Sebahi, and Mohand-Said Hacid. Temporal Event based Compliance Monitoring. In *Advances in Computing and Communication Engineering*, pages 17–22, 2018. doi:10.1109/ICACCE.2018.8441700.
- [40] Kai Heinrich, Patrick Zschech, Christian Janiesch, and Markus Bonin. Process data properties matter: Introducing gated convolutional neural networks (GCNN) and key-value-predict attention networks (KVP) for next event prediction with deep learning. *Decis. Support Syst.*, 143:113494, 2021. doi:10.1016/j.dss.2021.113494.
- [41] Robert Heinrich, Philipp Merkle, Jörg Henss, and Barbara Paech. Integrating business process simulation and information system simulation for performance prediction. *Softw. Syst. Model.*, 16(1):257–277, 2017. doi:10.1007/s10270-015-0457-1.
- [42] Tobias Herbert, Juergen Mangler, and Stefanie Rinderle-Ma. Generating reliable process event streams and time series data based on neural networks. In *Enterprise, Business-Process and Information Systems Modeling*, volume 421, pages 81–95, 2021. doi:10.1007/978-3-030-79186-5_6.
- [43] Yuxiu Hua, Zhifeng Zhao, Rongpeng Li, Xianfu Chen, Zhiming Liu, and Honggang Zhang. Deep learning with long short-term memory for time series prediction. *IEEE Commun. Mag.*, 57(6):114–119, 2019. doi:10.1109/MCOM.2019.1800155.
- [44] Dragan Ivanovic, Manuel Carro, and Manuel V. Hermenegildo. Constraint-based runtime prediction of SLA violations in service orchestrations. In *Service-Oriented Computing*, pages 62–76, 2011. doi:10.1007/978-3-642-25535-9_5.
- [45] Abdulrahman Jalayer, Mohsen Kahani, Asef Pourmasoumi, and Amin Beheshti. Ham-net: Predictive business process monitoring with a hierarchical attention mechanism. *Knowl. Based Syst.*, 236:107722, 2022. doi:10.1016/j.knosys.2021.107722.
- [46] Martin Käppel, Stefan Jablonski, and Stefan Schöning. Cost-sensitive predictive business process monitoring. In *New Trends in Database and Information Systems*, volume 1450, pages 14–26, 2021. doi:10.1007/978-3-030-85082-1_2.
- [47] Martin Käppel, Stefan Jablonski, and Stefan Schöning. Evaluating predictive business process monitoring approaches on small event logs.

- In *Quality of Information and Communications Technology*, pages 167–182, 2021. doi:10.1007/978-3-030-85347-1_13.
- [48] Martin Käppel, Stefan Schönig, and Stefan Jablonski. Leveraging small sample learning for business process management. *Inf. Softw. Technol.*, 132:106472, 2021. doi:10.1016/j.infsof.2020.106472.
- [49] Naveed Khan, Zulfiqar Ali, Aftab Ali, Sally I. McClean, Darryl Charles, Paul N. Taylor, and Detlef D. Nauck. A generic model for end state prediction of business processes towards target compliance. In *Artificial Intelligence*, volume 11927, pages 325–335, 2019. doi:10.1007/978-3-030-34885-4_25.
- [50] Jongchan Kim and Marco Comuzzi. A diagnostic framework for imbalanced classification in business process predictive monitoring. *Expert Syst. Appl.*, 184:115536, 2021. doi:10.1016/j.eswa.2021.115536.
- [51] Jongchan Kim and Marco Comuzzi. Stability metrics for enhancing the evaluation of outcome-based business process predictive monitoring. *IEEE Access*, 9:133461–133471, 2021. doi:10.1109/ACCESS.2021.3115759.
- [52] Jongchan Kim, Marco Comuzzi, Marlon Dumas, Fabrizio Maria Maggi, and Irene Teinmaa. Encoding resource experience for predictive process monitoring. *Decis. Support Syst.*, 153:113669, 2022. doi:10.1016/j.dss.2021.113669.
- [53] Eva L. Klijn and Dirk Fahland. Identifying and reducing errors in remaining time prediction due to inter-case dynamics. In *Process Mining*, pages 25–32, 2020. doi:10.1109/ICPM49681.2020.00015.
- [54] Christopher Klinkmüller, N. R. T. P. van Beest, and Ingo Weber. Towards reliable predictive process monitoring. In *CAiSE Forum*, volume 317, pages 163–181. Springer, 2018. doi:10.1007/978-3-319-92901-9_15.
- [55] David Knuplesch, Manfred Reichert, and Akhil Kumar. A framework for visually monitoring business process compliance. *Inf. Systems*, 64:381–409, 2017. doi:10.1016/j.is.2016.10.006.
- [56] Jonghyeon Ko and Marco Comuzzi. Keeping our rivers clean: Information-theoretic online anomaly detection for streaming business process events. *Inf. Syst.*, 104:101894, 2022. doi:10.1016/j.is.2021.101894.
- [57] Patrik Koenig, Juergen Mangler, and Stefanie Rinderle-Ma. Compliance Monitoring on Process Event Streams from Multiple Sources. In *Process Mining*, 2019. doi:10.1109/ICPM.2019.00026.
- [58] Kala Kotamarthi, Xianzhi Wang, Georg Grossmann, Quan Z. Sheng, and Sarath Indrakanti. A Framework Towards Model Driven Business Process Compliance and Monitoring. In *Enterprise Distributed Object Computing Workshop*, pages 24–32, 2015. doi:10.1109/EDOCW.2015.27.
- [59] Philipp Leitner, Johannes Ferner, Waldemar Hummer, and Schahram Dustdar. Data-driven and automated prediction of service level agreement violations in service compositions. *Distributed Parallel Databases*, 31(3):447–470, 2013. doi:10.1007/s10619-013-7125-7.
- [60] Philipp Leitner, Johannes Ferner, Waldemar Hummer, and Schahram Dustdar. Data-driven and automated prediction of service level agreement violations in service compositions. *Distributed Parallel Databases*, 31(3):447–470, 2013. doi:10.1007/s10619-013-7125-7.
- [61] Philipp Leitner, Anton Michlmayr, Florian Rosenberg, and Schahram Dustdar. Monitoring, prediction and prevention of SLA violations in composite services. In *Web Services*, pages 369–376, 2010. doi:10.1109/ICWS.2010.21.
- [62] Daniela Loreti, Federico Chesani, Anna Ciampolini, and Paola Mello. A distributed

- approach to compliance monitoring of business process event streams. *Future Generation Computer Systems*, 82:104–118, 2018. doi:10.1016/j.future.2017.12.043.
- [63] Linh Thao Ly, Fabrizio Maria Maggi, Marco Montali, Stefanie Rinderle-Ma, and Wil M. P. van der Aalst. A framework for the systematic comparison and evaluation of compliance monitoring approaches. In *Enterprise Distributed Object Computing*, pages 7–16, 2013. doi:10.1109/EDOC.2013.11.
- [64] Linh Thao Ly, Fabrizio Maria Maggi, Marco Montali, Stefanie Rinderle-Ma, and Wil M. P. van der Aalst. Compliance monitoring in business processes: Functionalities, application, and tool-support. *Inf. Syst.*, 54:209–234, 2015. doi:10.1016/j.is.2015.02.007.
- [65] Linh Thao Ly, Stefanie Rinderle-Ma, Kevin Göser, and Peter Dadam. On enabling integrated process compliance with semantic constraints in process management systems - requirements, challenges, solutions. *Inf. Syst. Frontiers*, pages 195–219, 2012. doi:10.1007/s10796-009-9185-9.
- [66] Linh Thao Ly, Stefanie Rinderle-Ma, David Knuplesch, and Peter Dadam. Monitoring business process compliance using compliance rule graphs. In *On the Move to Meaningful Internet Systems*, pages 82–99, 2011. doi:10.1007/978-3-642-25109-2_7.
- [67] Fabrizio Maria Maggi, Chiara Di Francescomarino, Marlon Dumas, and Chiara Ghidini. Predictive monitoring of business processes. In *Advanced Information Systems Engineering*, pages 457–472, 2014. doi:10.1007/978-3-319-07881-6_31.
- [68] Fabrizio Maria Maggi, Marco Montali, and Ubaier Bhat. Compliance Monitoring of Multi-Perspective Declarative Process Models. In *Enterprise Distributed Object Computing Conference*, pages 151–160, 2019. doi:10.1109/EDOC.2019.00027.
- [69] Marco Maisenbacher and Matthias Weidlich. Handling concept drift in predictive process monitoring. In *Services Computing*, pages 1–8, 2017. doi:10.1109/SCC.2017.10.
- [70] Alfonso Eduardo Márquez-Chamorro, Manuel Resinas, and Antonio Ruiz-Cortés. Predictive monitoring of business processes: A survey. *IEEE Trans. Serv. Comput.*, 11(6):962–977, 2018. doi:10.1109/TSC.2017.2772256.
- [71] Nijat Mehdiyev and Peter Fettke. *Explainable Artificial Intelligence for Process Mining: A General Overview and Application of a Novel Local Explanation Approach for Predictive Process Monitoring*, pages 1–28. 2021. doi:10.1007/978-3-030-64949-4_1.
- [72] Nijat Mehdiyev and Peter Fettke. Local post-hoc explanations for predictive process monitoring in manufacturing. In *European Conference on Information Systems*, 2021.
- [73] Giovanni Meroni, Luciano Baresi, Marco Montali, and Pierluigi Plebani. Multi-party business process compliance monitoring through IoT-enabled artifacts. *Information Systems*, 73:61–78, 2018. doi:10.1016/j.is.2017.12.009.
- [74] Dominic A. Neu, Johannes Lahann, and Peter Fettke. A systematic literature review on state-of-the-art deep learning methods for process prediction. *Artif. Intell. Rev.*, 55(2):801–827, 2022. doi:10.1007/s10462-021-09960-8.
- [75] O. Ogunbiyi, A. Basukoski, and T. J. Chausalet. Comparative analysis of clustering-based remaining-time predictive process monitoring approaches. *International Journal of Business Process Integration and Management*, 2020.
- [76] Kasey Panetta. Gartner top 10 data and analytics trends for 2021, 2021. (accessed 2022-01-13). URL: <https://www.gartner.com/smarterwithgartner/gartner-top-10-data-and-analytics-trends-for-2021>.

- [77] Stephen Pauwels and Toon Calders. Incremental predictive process monitoring: The next activity case. In *Business Process Management*, volume 12875, pages 123–140, 2021. doi: 10.1007/978-3-030-85469-0_10.
- [78] Marco Pegoraro, Merih Seran Uysal, David Benedikt Georgi, and Wil M. P. van der Aalst. Text-aware predictive monitoring of business processes. In *Business Information Systems*, pages 221–232, 2021. doi:10.52825/bis.v1i.62.
- [79] Maja Pesic, Helen Schonenberg, and Wil M. P. van der Aalst. DECLARE: full support for loosely-structured processes. In *Enterprise Distributed Object Computing Conference*, pages 287–300. IEEE Computer Society, 2007. doi: 10.1109/EDOC.2007.14.
- [80] Mirko Polato, Alessandro Sperduti, Andrea Burattin, and Massimiliano de Leoni. Time and activity sequence prediction of business process instances. *Computing*, 100(9):1005–1031, 2018. doi:10.1007/s00607-018-0593-x.
- [81] Mahsa Pourbafrani, Shreya Kar, Sebastian Kaiser, and Wil M. P. van der Aalst. Remaining time prediction for processes with inter-case dynamics. In *Process Mining Workshops*, pages 140–153, 2021. doi:10.1007/978-3-030-98581-3_11.
- [82] Efrén Rama-Maneiro, Juan C. Vidal, and Manuel Lama. Deep Learning for Predictive Business Process Monitoring: Review and Benchmark. *arXiv:2009.13251 [cs]*, 2021.
- [83] Pedro Rico, Félix Cuadrado, Juan C. Dueñas, Javier Andión, and Hugo A. Parada G. Business process event prediction through scalable online learning. *IEEE Access*, 9:136313–136333, 2021. doi:10.1109/ACCESS.2021.3117147.
- [84] Stefanie Rinderle-Ma, Manuel Gall, Walid Fdihila, Jürgen Mangler, and Conrad Indiono. Collecting examples for instance-spanning constraints. *CoRR*, abs/1603.01523, 2016. URL: <http://arxiv.org/abs/1603.01523>.
- [85] Stefanie Rinderle-Ma, Manfred Reichert, and Martin Jurisch. On utilizing web service equivalence for supporting the composition life cycle. *Int. J. Web Serv. Res.*, 8(1):41–67, 2011. doi:10.4018/jwsr.2011010103.
- [86] Williams Rizzi, Chiara Di Francescomarino, and Fabrizio Maria Maggi. Explainability in Predictive Process Monitoring: When Understanding Helps Improving. In *Business Process Management Forum*, pages 141–158, 2020. doi:10.1007/978-3-030-58638-6_9.
- [87] Williams Rizzi, Chiara Di Francescomarino, Chiara Ghidini, and Fabrizio Maria Maggi. How do I update my model? on the resilience of predictive process monitoring models to change. *Knowl Inf Syst*, 2022. doi:https://doi.org/10.1007/s10115-022-01666-9.
- [88] Carlos Rodríguez, Patrícia Silveira, Florian Daniel, and Fabio Casati. Analyzing compliance of service-based business processes for root-cause analysis and prediction. In *Current Trends in Web Engineering*, volume 6385, pages 277–288, 2010. doi:10.1007/978-3-642-16985-4_25.
- [89] Andreas Rogge-Solti and Mathias Weske. Prediction of business process durations using non-markovian stochastic petri nets. *Inf. Syst.*, 54:1–14, 2015. doi:10.1016/j.is.2015.04.004.
- [90] Sherif Sakr, Ahmed Awad, and Amal Elgammal. Compliance Monitoring as a Service: Requirements, Architecture and Implementation. In *Cloud Computing*, 2015. doi:10.1109/CLOUDCOMP.2015.7149636.
- [91] Beate Scheibel and Stefanie Rinderle-Ma. Decision mining with time series data based on automatic feature generation. In *Advanced Information Systems Engineering*, 2022. (accepted for publication).

- [92] Luis Galdo Seara and Renata Medeiros de Carvalho. An approach for workflow improvement based on outcome and time remaining prediction. In *Model-Driven Engineering and Software Development*, pages 473–480, 2019. doi:10.5220/0007577504750482.
- [93] Arik Senderovich, Chiara Di Francescomarino, and Fabrizio Maria Maggi. From knowledge-driven to data-driven inter-case feature encoding in predictive process monitoring. *Inf. Syst.*, 84:255–264, 2019. doi:10.1016/j.is.2019.01.007.
- [94] Arik Senderovich, Matthias Weidlich, Avigdor Gal, and Avishai Mandelbaum. Queue mining for delay prediction in multi-class service processes. *Inf. Syst.*, 53:278–295, 2015. doi:10.1016/j.is.2015.03.010.
- [95] Renuka Sindhgatta, Chun Ouyang, and Catarina Moreira. Exploring interpretability for predictive process analytics. In *Service-Oriented Computing*, volume 12571, pages 439–447, 2020. doi:10.1007/978-3-030-65310-1_31.
- [96] Florian Spree. Predictive Process Monitoring-A Use-Case-Driven Literature Review. In *EMISA Forum: Vol. 40, No. 1*. De Gruyter, 2020.
- [97] Florian Stertz, Stefanie Rinderle-Ma, and Juer-gen Mangler. Analyzing process concept drifts based on sensor event streams during runtime. In *Business Process Management*, pages 202–219, 2020. doi:10.1007/978-3-030-58666-9_12.
- [98] Matthias Stierle, Jens Brunk, Sven Weinzierl, Sandra Zilker, Martin Matzner, and Jörg Becker. Bringing light into the darkness - A systematic literature review on explainable predictive business process monitoring techniques. In *European Conference on Information Systems*, 2021.
- [99] Oleg Svatoš. Requirements for Business Process Legal Compliance Monitoring. *Journal of Systems Integration*, 8(2), 2017. doi:10.20470/jsi.v8i2.301.
- [100] Bayu Adhi Tama and Marco Comuzzi. An empirical comparison of classification techniques for next event prediction using business process event logs. *Expert Syst. Appl.*, 129:233–245, 2019. doi:10.1016/j.eswa.2019.04.016.
- [101] Bayu Adhi Tama and Marco Comuzzi. An empirical comparison of classification techniques for next event prediction using business process event logs. *Expert Systems with Applications*, 129:233–245, 2019. doi:10.1016/j.eswa.2019.04.016.
- [102] Johan J C Tambotuh, Harjanto Prabowo, Sani M Isa, and Wahyu Pudjianto. Process mining in governance, risk management, compliance (GRC), and auditing: A systematic literature review. *JATIT*, (18), 2021.
- [103] Irene Teinemaa, Marlon Dumas, Fabrizio Maria Maggi, and Chiara Di Francescomarino. Predictive business process monitoring with structured and unstructured data. In *Business Process Management*, pages 401–417, 2016. doi:10.1007/978-3-319-45348-4_23.
- [104] Irene Teinemaa, Marlon Dumas, Marcello La Rosa, and Fabrizio Maria Maggi. Outcome-Oriented Predictive Process Monitoring: Review and Benchmark. *ACM Trans. Knowl. Discov. Data*, 13(2):17:1–17:57, 2019. doi:10.1145/3301300.
- [105] Silvano Colombo Tosatto, Guido Governatori, and Nick van Beest. Checking regulatory compliance: Will we live to see it? In *Business Process Management*, pages 119–138, 2019. doi:10.1007/978-3-030-26619-6_10.
- [106] J. M. E. M. van der Werf and H. M. W. Verbeek. Online Compliance Monitoring of Service Landscapes. In *Business Process Management Workshops*, pages 89–95, 2015. doi:10.1007/978-3-319-15895-2_8.
- [107] Ilya Verenich, Marlon Dumas, Marcello La Rosa, Fabrizio Maria Maggi, and Irene Teinemaa. Survey and cross-benchmark comparison

- of remaining time prediction methods in business process monitoring. *ACM Trans. Intell. Syst. Technol.*, 10(4):34:1–34:34, 2019. doi: 10.1145/3331449.
- [108] Thomas Voglhofer and Stefanie Rinderle-Ma. Collection and elicitation of business process compliance patterns with focus on data aspects. *Bus. Inf. Syst. Eng.*, 62(4):361–377, 2020. doi: 10.1007/s12599-019-00594-3.
- [109] Sven Weinzierl, Sebastian Dunzer, Johannes Christian Tenschert, Sandra Zilker, and Martin Matzner. Predictive business process deviation monitoring. In *European Conference on Information Systems*, 2021.
- [110] Karolin Winter and Stefanie Rinderle-Ma. Discovering instance-spanning constraints from process execution logs based on classification techniques. In *Enterprise Distributed Object Computing Conference*, pages 79–88, 2017. doi:10.1109/EDOC.2017.20.
- [111] Karolin Winter and Stefanie Rinderle-Ma. Untangling the GDPR using conrelminer. *CoRR*, abs/1811.03399, 2018.
- [112] Karolin Winter and Stefanie Rinderle-Ma. Deriving and combining mixed graphs from regulatory documents based on constraint relations. In *Advanced Information Systems Engineering*, volume 11483, pages 430–445, 2019. doi:10.1007/978-3-030-21290-2_27.
- [113] Karolin Winter and Stefanie Rinderle-Ma. Defining instance spanning constraint patterns for business processes based on proclets. In *Conceptual Modeling*, pages 149–163, 2020. doi:10.1007/978-3-030-62522-1_11.
- [114] Karolin Winter, Florian Stertz, and Stefanie Rinderle-Ma. Discovering instance and process spanning constraints from process execution logs. *Inf. Syst.*, 89:101484, 2020. doi: 10.1016/j.is.2019.101484.
- [115] Frederik Wolf, Jens Brunk, and Jörg Becker. A Framework of Business Process Monitoring and Prediction Techniques. In *Innovation Through Information Systems*, 2021. doi: 10.1007/978-3-030-86797-3_47.
- [116] Marwa Hussein Zaki, Ahmed Awad, and Osman Hegazy. Enabling Compliance Monitoring for Process Execution Engines. *BPMDS 2017 RADAR*, page 8, 2017.