From Process-Agnostic to Process-Aware Automation, Mining, and Prediction

Stefanie Rinderle-Ma(✉), Janik-Vasily Benzin, and Juergen Mangler

Technical University of Munich,
TUM School of Computation, Information and Technology,
Boltzmannstrasse 3, 85748 Garching, Germany
{stefanie.rinderle-ma,janik.benzin,juergen.mangler}@tum.de

Abstract. The entire research area of (business) process management has experienced a tremendous push with the advent of process mining, robotic process automation, and predictive process monitoring. While this development is highly appreciated, the current process-agnostic pipelines for process mining, robotic process automation, and predictive process monitoring have several limitations. Taking a system perspective, this keynote elaborates the limitations of process-agnostic automation. Then, it shows how a shift towards process-aware automation and predictive compliance monitoring can be achieved and how process-aware pipelines contribute to overcome the limitations of process-agnostic automation. Finally, research implications with a focus on Petri nets are derived.

Keywords: Process Automation · Process Mining · Predictive Process Monitoring · Predictive Compliance Monitoring

1 Introduction

Process mining and robotic process automation are two mega trends. *"The global process mining software market is projected to grow from $933.1 million in 2022 to $15,546.4 million by 2029, at a CAGR of 49.5% in the forecast period."*[1]. The combination of both technologies is expected to even increase their market penetration [6].

Process mining comprises a set of techniques for the discovery and analysis of process models and their executions based on process event logs [1] and the expectations in practice are high [35]. Robotic process automation refers to the automation of single process tasks by replacing human-task interaction with a software bot [2]. The currently applied *mine and automate* pipeline (e.g., [14]) is depicted in Fig. 1a). Process mining is applied to discover process models, and within these models tasks with the potential for automation are detected. As an intermediate step between process mining and the automation of tasks, [14]

[1] https://www.fortunebusinessinsights.com/process-mining-software-market-104792.

L. Gomes and R. Lorenz (Eds.): PETRI NETS 2023, LNCS 13929, pp. 3–15, 2023.
https://doi.org/10.1007/978-3-031-33620-1_1

advocate to standardize the process models by removing variations in the process that might result due to, e.g., product variants. Once tasks are automated, process mining can be used to continuously monitor their performance.

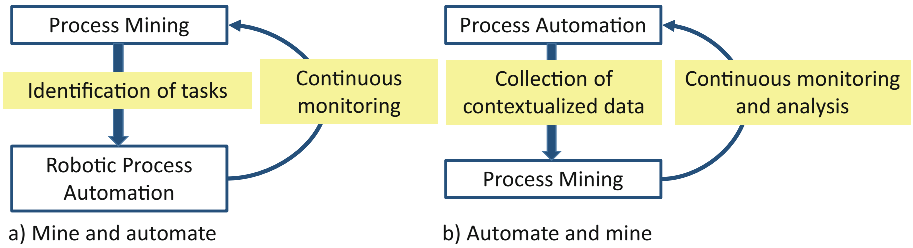

Fig. 1. Pipelines: a) Mine and automate; b) Automate and mine

However, the *mine and automate* pipeline as depicted in Fig. 1a) has several limitations:

1. *Task-oriented automation:* Robot process automation aims at the automation of single, often simple and repetitive interactions of humans with software. However, a process is a task-overarching, orchestrating concept. Real performance gains and analysis insights can only be achieved by taking an orchestration point of view for process automation.
2. *Data acquisition and preparation:* Process mining relies on process event logs emitted or extracted from information systems, e.g., ERP systems. If the underlying system is not process-aware or a black box (e.g., legacy systems), mechanisms for extraction and preparation of data are to be defined and employed. Moreover, if the data is spread over multiple and possibly heterogeneous information systems [18, 27], mechanisms for integrating the data are to be defined and employed. Existing commercial systems support a range of adaptors to different systems and data sources, e.g., data connections as supported in Celonis[2]. Using an object-centric approach offers the opportunity to capture objects and their life cycles in the process event logs [8] and can be used even if no case id is available or can be extracted form the underlying data. However, data connections are not robust towards changes in the data structures, i.e., data structure changes possibly require the adaptation of one, several, or all of the established data connections.
3. *Ex-post point of view:* Most of the process mining analysis tasks are conducted in an ex-post manner, i.e., based on process event logs that reflect already finished process executions. This holds true for all three pillars of process mining, i.e., process discovery, conformance checking, and process enhancement. However, the monitoring and analysis of process executions during runtime (online) based on process event streams provides current insights into the

[2] https://docs.celonis.com/en/data-connections.html.

process, e.g., detecting exceptions when they are actually happening, and hence enabling a quicker reaction to potential problems such as compliance violations [9,19,36]. Moreover, in practice, many analysis questions refer to the monitoring of the process perspectives time, resources, and data, e.g., a temperature sensor exceeding a certain threshold or temporal deviations that *"are mostly caused by humans, e.g., someone stepping into the safety area of a machine causing a delay, and hint to problems with work organization"* [34]. Even predictive process monitoring, though suggesting to be applied in an online manner due to the term "monitoring", is mostly applied in an ex-post way. More precisely, a process event log is split into training and test data. One or several prediction models are learned based on the training data. These prediction models are then applied to the test data, i.e., prefixes from the test data are used to reflect a process event stream. Prediction goals comprise, for example, the remaining time of cases, the next activity, and the outcome of a process [13].

4. *Dealing with uncertainty and concept drift:* Ex-post mining allows to obtain a picture of the past. However, an ever changing process environment and uncertainties force processes to adapt constantly [5,9]. In the manufacturing domain, for example, if new processes are set up, several adaptation cycles are necessary until a process runs in a robust way. In health care, due to unforeseen situations, ad-hoc changes of process instances can be frequently required, e.g., the blood pressure exceeds a threshold such that the surgery has to be delayed. *"This uncertainty often manifests itself in significant changes in the executed processes"* [5]. Process changes, in turn, manifest as *concept drifts* in process event logs [5] and as *unseen behavior* in process event streams [23]. A selection of use cases for process changes from different domains can be found in [17].

Limitations 1. and 2. refer to the system and data perspective and Limitations 3. and 4. to the mining and analysis perspective of a process. In order to address Limitations 1. and 2., we advocate an inversion of the *mine and automate* pipeline into an *automate and mine* pipeline as depicted in Fig. 1b). The *automate and mine* pipeline starts with automated and orchestrated processes, driven and managed by process engines or process-aware information systems. These systems can be exploited to collect data in an integrated, orchestrated, and contextualized manner at arbitrary granularity which, in turn, offers novel process mining insights [29], for example, the combined analysis process event logs/streams and sensors streams [11,37].

Limitations 3. and 4. emphasize the need to move towards approaches applied during runtime when mining and monitoring processes. Most promising here are approaches for online process mining such as [9] and predictive process monitoring (cf., e.g., survey in [13]). One of the most crucial (business) goals of predictive process monitoring is the prediction of possible compliance violations [26]. For this, in existing approaches, the compliance constraint of interest, for example, service level agreement "90% of the orders must be processed within 2 h", is encoded as prediction goal in a prediction model (referred to as *predicate prediction* [21]). Predicate prediction is illustrated through the *comply and predict* pipeline depicted in Fig. 2a): compliance constraints are encoded as prediction goals (comply) into a prediction model each, based on which violations of the constraint are predicted (predict). The *comply and predict* pipeline for predicate prediction comes with the following limitations (ctd.):

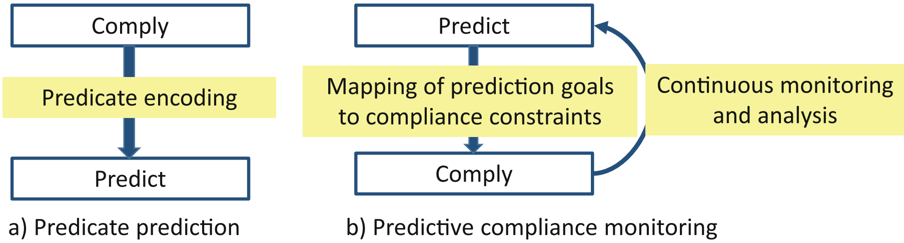

Fig. 2. Pipelines: a) Predicate prediction; b) Predictive compliance monitoring

5. *Performance:* In literature, predicate prediction, is mostly applied in the context of simple scenarios. Simple here refers to i) compliance constraints of limited complexity such as service level agreements and ii) a limited number of predicates. The reason is that the encoding of i) is more manageable for simple compliance constraints and ii) keeps the number of prediction models limited that are necessary for predicate prediction (recall that for n compliance constraints, n prediction models are to be created). However, real-world scenarios can look very different [32]: contrary to i), compliance constraints that stem from regulatory documents such as the GDPR are complex and refer to multiple process perspectives. Contrary to ii) there might be several hundred compliance constraints that are imposed on one process [28]. Supporting predicate prediction for full-blown real-world scenarios would possibly lead to a large number of complex prediction models, resulting in performance issues.
6. *Transparency:* Predicate prediction yields a binary answer, i.e., either "the predicate is violated" (possibly with a counterexample) or "the predicate is not violated". Though this constitutes an essential information, in particular in the case of violations, often some sort of reaction is required. At least, it should become transparent why a violation occurred and for which instance(s) (root cause). Without this information, it is difficult for users to decide on remedy actions.
7. *Maintainability:* In predicate and compliance prediction in general, two sources of change might occur. First of all, changes of the process and its instances might become necessary, reflected by concept drift in the process event log. Secondly, changes in the set of compliance constraints might be performed by adding, deleting, and updating compliance constraints.

Compliance constraint changes can occur frequently, e.g.: *"Bank regulations change about every 12 min"*[3]. For predicate prediction, a compliance constraint change requires the adaptation of the associated prediction model, i.e., m compliance constraint changes result in the adaptation of m prediction models.

In order to tackle Limitations 5.– 7., again, we advocate an inversion of the *comply and predict* pipeline shown in Fig. 2a). Instead of encoding compliance constraints and predicting their violations afterwards, we suggest the *predict and comply* pipeline denoted as *predictive compliance monitoring* [32], depicted in Fig. 2b): at first, predicting takes places through process monitoring approaches with different prediction goals such as next activity, remaining time, outcome, and other key performance indicators are applied (predict), followed by a mapping to the set of compliance constraints (comply).

In the following, we will contrast the different pipelines and approaches. For this, we take the perspective of a holistic system and generalize the pipelines into process-agnostic and process-aware automation (cf. Sect. 2). Finally, research implications with a focus on Petri nets will be provided in Sect. 3.

2 Process-Agnostic and Process-Aware Automation

In the introduction, pipelines for process automation and mining as well as prediction and compliance are shown, i.e., the current *mine and automate* and the inverted *automate and mine* pipeline as well as the current *comply and predict* and the inverted *predict and comply* pipeline. From a system perspective, the two pipelines are not separated from each other, i.e., a holistic system can support both. Figure 3 shows the system perspective realizing the *mine and automate* and *comply and predict* pipelines on the left side and the system perspective realizing the *automate and mine* and *predict and comply* pipelines on the right side. Due to the fact, that the system perspective on the right side takes an explicit process-aware point of view by employing a process engine or process-aware information system, we refer to it as *PAWA: process-aware automation*. Symmetrically, we refer to the system on the left side, where automation is restricted to single tasks, as *PAGA: process-agnostic automation*.

In current PAGA systems, the event and data stream is extracted by ETL pipelines from logs of the machine, ERP systems, and further systems as depicted in Fig. 3. A multi-perspective process model is mined through process discovery, conformance checking, and enriching the process model with additional perspectives using further mining methods, e.g., decision and organizational mining [12]. The machine, ERP systems and further systems are then *enhanced* through process analysts, domain experts, and/or developers as a result of insights gained from analysing the multi-perspective process model. Enhancing refines robotic process automation as shown for the *mine and automate* pipeline depicted in

[3] https://thefinanser.com/2017/01/bank-regulations-change-every-12-minutes (last accessed 2023-04-03).

Fig. 1 by additionally optimizing existing automatic activities in the process model, ad-hoc activities to mitigate possible problems that conformance checking has unveiled, or circumventing bottlenecks by assigning further resources to an activity.

The current state is dominated by the relational perspective of ERP systems that comes with major drawbacks. First, directly connecting to ERP and further relational information systems necessitates sophisticated ETL pipelines that emphasize ex-post over ex-ante views. Second, the lack of the process perspective in relational systems nudges the analysis to choose the traditional mine and automate line of action (cf. Sect. 1 and Fig. 1) such that the corresponding disadvantages apply.

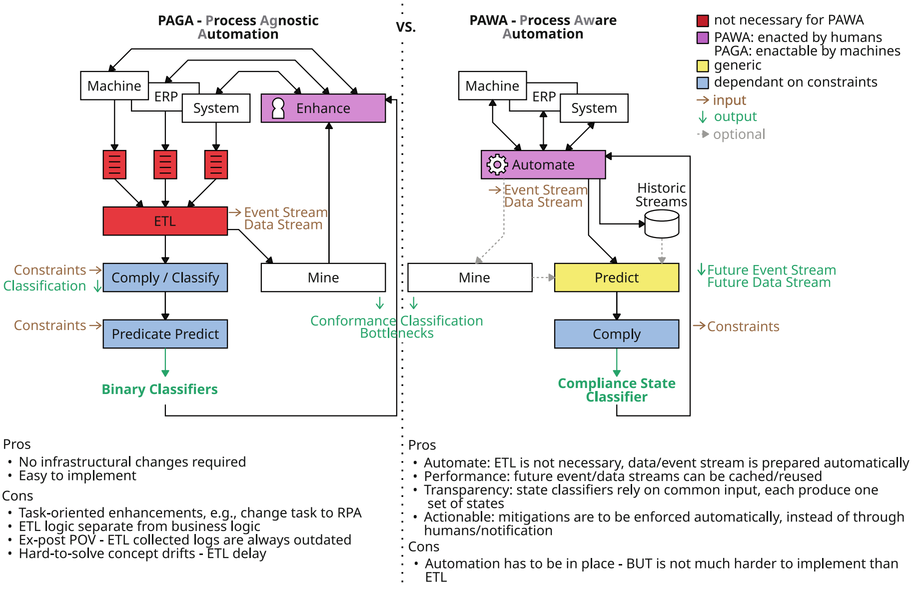

Fig. 3. System View Comparison

PAWA systems serve as an orchestration and automation environment that integrates the machine, ERP, and the system views (cf. Fig. 3). This enables the implementation and execution of arbitrary processes (→ Limitation 1). To illustrate this, in [29], we provide a classification of process automation scenarios in manufacturing along the two dimensions of "human involvement" and "green field – brown field". This results in four automation classes that we have found and realized across 16 real-world process scenarios. More precisely, the process scenarios were modeled, implemented, and hence automated using the cloud process execution engine cpee.org [25]. The scenarios comprise i) a robotic process automation scenario (low human involvement, brown field), ii) fully automated process orchestration (low human involvement, green field), iii) process-oriented user support (high human involvement, brown field), and iv) interactive process automation (high human involvement, green field). i) was chosen to automate a task due to a black box application system to be invoked. ii) orchestrates the tasks of a robot, a machine, and measurement equipment. A video of the execution of the process orchestration can be found here[4]. iii) includes the automatically generated instructions to be shown at work stations for staff in a process with more than 20.000 variants. iv) features the on the fly creation and routing in process models based on interactions between human users and physical devices such as machines [22] or other utilities, e.g., in the care domain [33]. Such process scenarios are not only prevalent for the manufacturing domain, but also for other domains such as health care and logistics which integrate "physical" aspects (machines, vehicles) and human work. The variety of scenarios underpins that robotic process automation can be supported by a PAWA system, but is only one piece. PAWA systems are able to support any process orchestration and integrate different systems, human work, and physical devices along the process logic.

Moreover, PAWA systems can be employed to collect data in a systematic, integrated, and contextualized manner (→ Limitation 2), i.e., they log every event emitted during process execution and on top of that, PAWA systems can collect and log process context data, e.g., IoT data in domains such as production, health care, and logistics. The combined collection of process and IoT data has gained interest lately, resulting in an extension of the process event log standard eXtensible event stream (XES)[5], i.e., the XES Sensor Stream extension [24]. This way, process engines and process-aware information systems serve as systems for the process-oriented and contextualized collection of process data at an arbitrary granularity (as defined in the process models) and a trusted, high quality level (****(*) star level according to the L^* data quality model for process mining [3]) [30]. Using, for example, cpee.org as process collection system, we collected and published three real-world process event logs with additional context data[6]. Two data sets comprise data from public transport, augmented with context data on weather, traffic, etc. and one data set stems from the production domain on producing a chess piece.

In addition, PAWA systems collect and log data at an time, i.e., in an ex-post manner as process event logs and during runtime as process event streams (→ Limitation 3). This also includes the runtime collection of context data such as sensor streams. In particular, the online collection of event streams facilitates the early detection of concept drifts [35] (→ Limitation 4).

Up to this point, we discussed how PAGA and PAWA systems realize the *mine and automate* and *automate and mine* pipelines shown in Fig. 1. PAGA and PAWA systems can also realize the *comply and predict* and *predict and*

[4] https://lehre.bpm.in.tum.de/~mangler/.Slides/media/media1.mp4, last accessed 2023-04-04.
[5] www.xes-standard.org.
[6] https://zenodo.org/communities/processmining.

comply pipelines shown in Fig. 2, i.e., on the PAGA side by predicate prediction and on the PAWA side as predictive compliance monitoring components.

We conducted a comprehensive literature review covering the research areas of predictive process monitoring and compliance monitoring (see, e.g., [20]) with respect to functionalities required for building a predictive compliance monitoring system [31]. A system that supports predictive compliance monitoring employs the *predict and comply* pipeline (cf. Fig. 2b) to predict the future progress of the monitored system and to monitor compliance on top of the predictions and interprets them from a systems perspective. An abstract view on how to integrate predictive compliance monitoring into a PAWA system is depicted in Fig. 3, contrasted by the current state of how predicate prediction is conceptualized and implemented in a PAGA system.

In PAGA systems, due to the current lack of the process perspective in the monitored system and in the prediction models, the results of predicting compliance violations have to be manually transformed into actions that can be executed on an ERP system by notifying the respective employee (*enhance*).

In PAWA systems, the goal of predictive compliance monitoring centers around the process perspective (cf. Fig. 3). By automating existing ERP systems or substituting existing systems through a PAWA system, ETL pipelines are replaced by a simple connection to the logging service of PAWA system. The optional *mine* and the compulsory *predict* separately consume the event and data stream from the logging services. While *mine* is concerned with discovering process models, analysing structural and behavioral properties of process models and checking conformance, *predict* focuses on a single prediction model trained to predict the future event and data stream of the overall process, i.e., the prediction goal is a stream prediction (→ Limitations 3. and 4.). The prediction model can additionally take the mined process model as input such that the prediction of the event and data stream is based on the respective execution states of running instances in the process model. Overall, the prediction goal consists of future events and, in particular, data attributes. If required for very important compliance constraints, the inverted, specialized *comply and predict* pipeline (predicate prediction) can be added to the predictive compliance monitoring system such that an independent prediction model for the very important compliance constraint is trained. Stream predictions of the process are the input to *comply*, while predicted violations of independent prediction models can directly trigger mitigation actions in the monitored system. Given a stream prediction, *comply* checks compliance of the compliance constraints resulting in various compliance states. Due to the process perspective inherent in PAWA systems, predicted compliance states can automatically trigger mitigation actions, e.g., by adding ad-hoc activities to an ongoing process instance.

Note that predictive compliance monitoring could also be integrated into the PAGA system, inheriting its limitations due to enhance and the data collection. More importantly, note that the distinction into predicate prediction and predictive compliance monitoring does not only apply to the domain of process mining and automation, but also to the more general area of *event prediction* [7].

At this point, we have to say that there is no solution for predictive compliance monitoring yet and the “sweet spot” between predicate prediction and predictive compliance monitoring w.r.t. prediction quality and limitations has to be investigated [32].

Due to its process centricity, the PAWA system comes with the following advantages regarding predictive compliance monitoring:

– *Performance and maintainability of the prediction model (→ Limitations 5. and 7.):* If the set of compliance constraints is updated, no retraining of the prediction model is necessary due to the clear separation of the prediction model and the compliance checking. Furthermore, no new prediction models have to be trained for new or updated constraints.
– *Transparency and explainability of the predictve process monitoring system (→ Limitation 6.):* As the prediction model predicts the future event and data streams, violations of compliance states can be pinpointed to their respective events or data attributes in the stream. Hence, the predicted violation is transparent and explainable.
– *Actionable mitigations:* Due to the process centricity of the PAGA system, compliance states can directly trigger actions in the process engine, e.g., through adding ad-hoc activities, or spawning instances of specialized mitigation processes.

3 Implications on Research

In the introduction, we raise seven limitations with current *mine and automate* as well as *comply and predict* pipelines which are integrated and analyzed through the systems perspective (PAGA vs. PAWA in Fig. 3). In the following, we will derive research implications with a focus on Petri net based research.

Soundness Verification for Automatic Changes to Automation. The system view comparison in Fig. 3 shows the two extreme sides of a continuous automation scale supported by process mining. A company on the move to process-aware automation can exhibit both automation systems, i.e., PAGA and PAWA, at the same time, as not all parts of the company are yet shifted to PAWA. During the transition, companies can benefit from support on how to shift from the manual *enhance* to the machine-enactable *automate* (cf. Fig. 3).

Petri Nets for Process-Aware Automation. Although Petri nets have been proposed and applied for process execution in the past (cmp. FUNSOFT Nets [10] in 1998), it remains not fully clear which Petri net class is sufficient to be used as execution model in PAWA. Recent candidates include object-centric Petri nets [4], Petri nets with identifiers [38], and colored Petri nets [16]. The main question is to keep the balance between expressive power to model all process perspectives and preventing problems such as checking soundness from becoming undecidable. Hence, research on Petri nets classes such as object-centric Petri nets or Petri nets with identifiers is ongoing.

Conformance Checking on Petri Net Process Models of Collaborative Systems. Conformance checking techniques for object-centric Petri nets and Petri nets with identifiers comparable to alignment-based conformance checking for sound workflow nets are missing. Also, replay-based techniques are yet missing, as the only existing object-centric Petri net implementation in PM4PY[7] does not feature replay.

Rescheduling Processes Execution - Checking and Balancing Resource Utilization. Whenever automatic changes are made, resource utilization may be affected. As multiple processes may share the same resources, optimization regarding resource utilization leads to better throughput. Scheduling of resource allocation with timed Petri nets (cmp. [15]) based on process models, can allow for simple, automatic and explainable solutions.

Instance and Process Spanning Constraints. Research on predicting and checking compliance has focused on intra-instance constraints so far. Predicting compliance states for instance and process spanning constraints remains an open research problem [31].

Provision of Mitigation Actions. Automatically providing mitigation actions for compliance violations, in particular at different granularity levels, and analyzing and visualizing their effects is relevant for both, predicate prediction and predictive process monitoring, but yet to be solved [31].

Visualization and explanation of predictions and violations. Visualization approaches for prediction results and future compliance violations are mostly missing. Moreover, root cause analysis has to be extended in order to deal with predicting violations of real-world compliance constraints [31].

Online Predictive Process Monitoring and Updating Compliance States. Since predictive process monitoring predicts future event and data streams given current event and data streams, prediction methods such as deep learning cannot be applied for cases with frequent process adaptations. It is not clear for which process environments existing prediction methods are capable of updating the prediction model after each incoming event with data or which batching methods are required such that existing prediction methods exhibit a sufficient performance. Continuous update of prediction models and predictions also results in continuous update of compliance states. It is open which granularity levels for compliance states, i.e., event-level, instance-level, process-level, multi-process-level, and multi-organisation-level, are supporting the users in understanding the current system state. Moreover, it is unclear how compliance states can be transformed between different granularity levels [31].

[7] https://pm4py.fit.fraunhofer.de/.

Data Properties and Quality. Exploiting data properties and quality is an emerging research topic. Considering data quality, data values of low quality may point to a compliance violation, e.g., redundant sensors fail quickly after each other. The relation of data quality with compliance violation that goes beyond merely removing low quality data points or imputating data values may reveal further insights.

References

1. van der Aalst, W.M.P.: Process Mining - Data Science in Action. 2nd edn. Springer, Heidelberg (2016). https://doi.org/10.1007/978-3-662-49851-4
2. van der Aalst, W.M.P., Bichler, M., Heinzl, A.: Robotic process automation. Bus. Inf. Syst. Eng. **60**(4), 269–272 (2018). https://doi.org/10.1007/s12599-018-0542-4
3. van der Aalst, W.M.P., et al.: Process mining manifesto. In: Business Process Management Workshops, pp. 169–194 (2011). https://doi.org/10.1007/978-3-642-28108-2_19
4. van der Aalst, W.M., Berti, A.: Discovering object-centric Petri nets. Fundamenta informaticae **175**(1–4), 1–40 (2020)
5. Adams, J.N., van Zelst, S.J., Rose, T., van der Aalst, W.M.P.: Explainable concept drift in process mining. Inf. Syst. **114**, 102177 (2023). https://doi.org/10.1016/j.is.2023.102177
6. Badakhshan, P., Wurm, B., Grisold, T., Geyer-Klingeberg, J., Mendling, J., vom Brocke, J.: Creating business value with process mining. J. Strateg. Inf. Syst. **31**(4), 101745 (2022). https://doi.org/10.1016/j.jsis.2022.101745
7. Benzin, J.V., Rinderle-Ma, S.: A survey on event prediction methods from a systems perspective: bringing together disparate research areas, February 2023. http://arxiv.org/abs/2302.04018
8. Berti, A., van der Aalst, W.M.P.: OC-PM: analyzing object-centric event logs and process models. Int. J. Softw. Tools Technol. Transf. **25**(1), 1–17 (2023). https://doi.org/10.1007/s10009-022-00668-w
9. Ceravolo, P., Tavares, G.M., Junior, S.B., Damiani, E.: Evaluation goals for online process mining: a concept drift perspective. IEEE Trans. Serv. Comput. **15**(4), 2473–2489 (2022). https://doi.org/10.1109/TSC.2020.3004532
10. Deiters, W., Gruhn, V.: Process management in practice applying the FUNSOFT net approach to large-scale processes. Autom. Softw. Eng. **5**(1), 7–25 (1998). https://doi.org/10.1023/A:1008654224389
11. Ehrendorfer, M., Mangler, J., Rinderle-Ma, S.: Assessing the impact of context data on process outcomes during runtime. In: Service-Oriented Computing, pp. 3–18 (2021). https://doi.org/10.1007/978-3-030-91431-8_1
12. Fahland, D.: Multi-dimensional process analysis. In: Business Process Management, pp. 27–33 (2022). https://doi.org/10.1007/978-3-031-16103-2_3
13. Francescomarino, C.D., Ghidini, C., Maggi, F.M., Milani, F.: Predictive process monitoring methods: which one suits me best? In: Business Process Management, pp. 462–479 (2018). https://doi.org/10.1007/978-3-319-98648-7_27
14. Geyer-Klingeberg, J., Nakladal, J., Baldauf, F., Veit, F.: Process mining and robotic process automation: a perfect match. In: Dissertation Award, Demonstration, and Industrial Track at BPM, pp. 124–131. CEUR-WS.org (2018)
15. Huang, B., Zhou, M., Lu, X.S., Abusorrah, A.: Scheduling of resource allocation systems with timed petri nets: a survey. ACM Comput. Surv. **55**(11), 230:1–230:27 (2023). https://doi.org/10.1145/3570326

16. Jensen, K.: Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use, vol. 1. Springer, Heidelberg (1996). https://doi.org/10.1007/978-3-662-03241-1
17. Kaes, G., Rinderle-Ma, S., Vigne, R., Mangler, J.: Flexibility requirements in real-world process scenarios and prototypical realization in the care domain. In: Meersman, R., et al. (eds.) OTM 2014. LNCS, vol. 8842, pp. 55–64. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-662-45550-0_8
18. Koenig, P., Mangler, J., Rinderle-Ma, S.: Compliance monitoring on process event streams from multiple sources. In: Process Mining, pp. 113–120 (2019). https://doi.org/10.1109/ICPM.2019.00026
19. Lee, W.L.J., Burattin, A., Munoz-Gama, J., Sepúlveda, M.: Orientation and conformance: a hmm-based approach to online conformance checking. Inf. Syst. **102**, 101674 (2021). https://doi.org/10.1016/j.is.2020.101674
20. Ly, L.T., Maggi, F.M., Montali, M., Rinderle-Ma, S., van der Aalst, W.M.P.: Compliance monitoring in business processes: functionalities, application, and tool-support. Inf. Syst. **54**, 209–234 (2015). https://doi.org/10.1016/j.is.2015.02.007
21. Maggi, F.M., Francescomarino, C.D., Dumas, M., Ghidini, C.: Predictive monitoring of business processes. In: Advanced Information Systems Engineering, vol. 8484, pp. 457–472 (2014). https://doi.org/10.1007/978-3-319-07881-6_31
22. Mangat, A.S., Mangler, J., Rinderle-Ma, S.: Interactive process automation based on lightweight object detection in manufacturing processes. Comput. Ind. **130**, 103482 (2021). https://doi.org/10.1016/j.compind.2021.103482
23. Mangat, A.S., Rinderle-Ma, S.: Next-activity prediction for non-stationary processes with unseen data variability. In: Enterprise Design, Operations, and Computing, pp. 145–161 (2022). https://doi.org/10.1007/978-3-031-17604-3_9
24. Mangler, J., et al.: Datastream XES extension: embedding IoT sensor data into extensible event stream logs. Future Internet **15**(3) (2023). https://doi.org/10.3390/fi15030109
25. Mangler, J., Rinderle-Ma, S.: CPEE - cloud process execution engine. In: BPM Demo Sessions, p. 51 (2014). http://ceur-ws.org/Vol-1295/paper22.pdf
26. Márquez-Chamorro, A.E., Resinas, M., Ruiz-Cortés, A.: Predictive monitoring of business processes: a survey. IEEE Trans. Serv. Comput. **11**(6), 962–977 (2018). https://doi.org/10.1109/TSC.2017.2772256
27. Oberdorf, F., Schaschek, M., Weinzierl, S., Stein, N., Matzner, M., Flath, C.M.: Predictive end-to-end enterprise process network monitoring. Bus. Inf. Syst. Eng. **65**(1), 49–64 (2023). https://doi.org/10.1007/s12599-022-00778-4
28. Rinderle-Ma, S., Kabicher-Fuchs, S.: An indexing technique for compliance checking and maintenance in large process and rule repositories. Enterp. Model. Inf. Syst. Archit. Int. J. Concept. Model. **11**, 2:1–2:24 (2016). https://doi.org/10.18417/emisa.11.2
29. Rinderle-Ma, S., Mangler, J.: Process automation and process mining in manufacturing. In: Business Process Management, pp. 3–14 (2021). https://doi.org/10.1007/978-3-030-85469-0_1
30. Rinderle-Ma, S., Stertz, F., Mangler, J., Pauker, F.: Process Mining-Discovery, Conformance, and Enhancement of Manufacturing Processes, pp. 363–383. Springer, Heidelberg (2023). https://doi.org/10.1007/978-3-662-65004-2_15
31. Rinderle-Ma, S., Winter, K., Benzin, J.V.: Predictive compliance monitoring in process-aware information systems: state of the art, functionalities, research directions, March 2023. http://arxiv.org/abs/2205.05446, accepted in Information Systems

32. Rinderle-Ma, S., Karolin Winter, J.V.B.: Predictive compliance monitoring in process-aware information systems: state of the art, functionalities, research directions. Inf. Syst. (2023). https://doi.org/10.1016/j.is.2023.102210
33. Stertz, F., Mangler, J., Rinderle-Ma, S.: Balancing patient care and paperwork automatic task enactment and comprehensive documentation in treatment processes. Enterp. Model. Inf. Syst. Archit. Int. J. Concept. Model. **15**, 11:1–11:28 (2020). https://doi.org/10.18417/emisa.15.11
34. Stertz, F., Mangler, J., Rinderle-Ma, S.: The role of time and data: online conformance checking in the manufacturing domain. arXiv:2105.01454 (2021)
35. Stertz, F., Mangler, J., Scheibel, B., Rinderle-Ma, S.: Expectations vs. experiences - process mining in small and medium sized manufacturing companies. In: Business Process Management Forum, pp. 195–211 (2021). https://doi.org/10.1007/978-3-030-85440-9_12
36. Stertz, F., Rinderle-Ma, S.: Process histories - detecting and representing concept drifts based on event streams. In: On the Move to Meaningful Internet Systems, pp. 318–335 (2018). https://doi.org/10.1007/978-3-030-02610-3_18
37. Stertz, F., Rinderle-Ma, S., Mangler, J.: Analyzing process concept drifts based on sensor event streams during runtime. In: Business Process Management, pp. 202–219 (2020). https://doi.org/10.1007/978-3-030-58666-9_12
38. van der Werf, J.M.E.M., Rivkin, A., Montali, M., Polyvyanyy, A.: Correctness notions for Petri nets with identifiers, December 2022. http://arxiv.org/abs/2212.07363, arXiv:2212.07363 [cs]