Petri Net Classes for Collaboration Mining: Assessment and Design Guidelines

Janik-Vasily Benzin[1] and Stefanie Rinderle-Ma[1]

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

Abstract. Collaboration mining develops discovery, conformance checking, and enhancement techniques for collaboration processes. The collaboration process model is key to represent the discovery result. As for process mining in general, Petri Net classes are candidates for collaboration process models due to their analytical power. However, a standard model class to represent collaboration processes is lacking due to the heterogeneity of collaboration and, thus, of collaboration mining techniques. Collaboration heterogeneity requires to cover, for example, intra-organizational collaborations as well as choreographies that span a process across multiple organizations. A standard collaboration model class would advance collaboration mining by focusing discovery through a common target model, supporting comparison, and enabling flexible mining pipelines. To find a standard model class, we aim at capturing collaboration heterogeneity in a meta model, assess Petri net classes as candidates for collaboration process models through the meta model, and derive design guidelines for the collaboration discovery.

Keywords: Collaboration Mining · Collaboration Process Models · Petri Net Classes · Design Guidelines

1 Introduction

Process mining research develops process discovery, conformance checking and enhancement techniques for *process orchestrations* that define what work is done in what order for similar *cases* [2,27,4]. In contrast, collaboration mining research develops the same for *collaboration processes* that define what work is done in what order for *collaborating cases*, i.e., collaboration processes correspond to multiple process orchestrations that collaborate via departments [33], services [24,46], agents [48,49,38], and organizations [55,54,29,15].

For mining process orchestrations the de-facto standard model class is *workflow nets* [3] to represent what work has to be done in what order as, for example, the workflow net concept underlies many models targeted by process discovery techniques [10]. Although declarative process models were also proposed, they are by far in the minority [10] and no discovery technique for collaboration processes targeting a declarative model is known to us. Hence, we focus on procedural models in the following. Nevertheless, a de-facto standard model class

for collaboration processes is missing, as the model classes targeted by discovery techniques are diverse, e.g., *communication nets* [46] vs. composed *RM_WF_nets* [33]. The heterogeneity of model classes is a consequence of the heterogeneity of collaborations, e.g., *message exchanges* [15] and *handover-of-work* [49], from various perspectives, e.g., *operational* [46] vs. *organizational* [49], and on various granularity levels, e.g., intra-organizational [33] vs. inter-organizational [15].

The goal of this work is to identify potential standard model classes for collaboration processes as *"the choice of target model is very important for the discovery process itself"* [1]. For this, we analyse the collaboration mining literature and related research areas with respect to the collaboration studied and integrate it in a *collaboration process meta model* in Sect. 2. Next, we derive assessment criteria from the collaboration process meta model to assess standard Petri net classes with respect to required properties of a standard model class for collaboration processes and findings from the assessment in Sect. 3. We focus on Petri net classes due to their expressiveness, graphical and integrating[1] nature, formal semantics, analysis techniques, and tool support [39,3,41]. Similar to process mining, collaboration mining starts with discovering a collaboration process model [4] such that the targeted model class determines applicable conformance checking and enhancement techniques. Thus, we translate the assessment findings into design guidelines for process discovery in Sect. 4 and conclude in Sect. 5.

2   A Collaboration Process Meta Model

The assessment of Petri net classes as standard candidates for collaboration process models in Sect. 3 necessitates an analysis of the heterogeneous collaboration studied in collaboration mining (cf. Sect. 1), as the collaboration determines what must be modelled in a collaboration process model. Due to the heterogeneity in what is considered as a collaboration process (CP), we take an integrating approach to understand what has to be modelled in a collaboration process (CP) model by presenting a collaboration process (CP) meta model.

The meta model integrates specifications of what has to be modelled from research areas that study CP models through a *top-down* approach that starts from scratch and models the CP as it should be executed. In that sense, collaboration mining represents the opposite *bottom-up* approach by assuming the CP has already been running for some time such that event logs can be extracted from information systems supporting the CP. Hence, the CP meta model brings process models from both top-down and bottom-up approaches together and is presented in Sect. 2.1 and instantiated for a real-world CP in Sect. 2.2.

2.1   Elements of the Collaboration Process Meta Model

To guide the integration of existing CP models, the functional, behavioral, informational, operational and organizational process perspectives [27] are depicted

---

[1] Many alternative modelling languages can be transformed into Petri nets [41].

in Fig. 1. Aside from the analysis question defining the perspective on a process, the resulting main concept, the concept's abstraction relation instances/concept granularities, and elements of the perspective are presented. Existing work taking the top-down or bottom-up approach identifies a perspective's element as the answer to what is collaborating, i.e., the involved *collaboration concepts* (perspective's elements denoted in blue in Fig. 1).
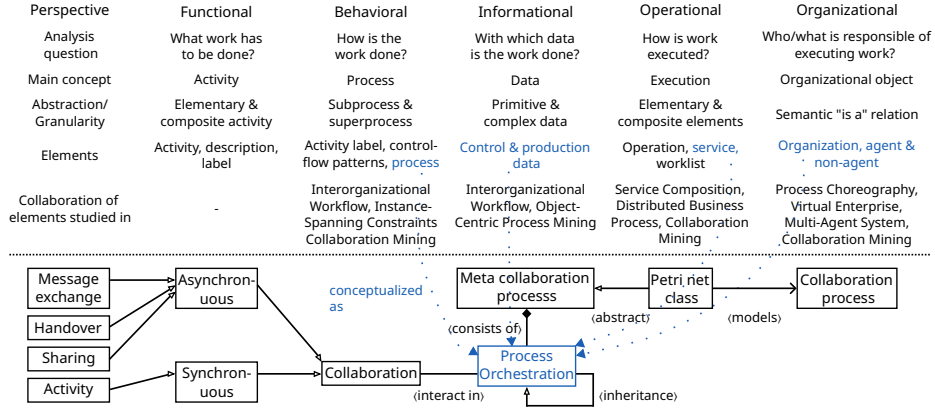
Perspective	Functional	Behavioral	Informational	Operational	Organizational
Analysis question	What work has to be done?	How is the work done?	With which data is the work done?	How is work executed?	Who/what is responsible of executing work?
Main concept	Activity	Process	Data	Execution	Organizational object
Abstraction/ Granularity	Elementary & composite activity	Subprocess & superprocess	Primitive & complex data	Elementary & composite elements	Semantic "is a" relation
Elements	Activity, description, label	Activity label, control-flow patterns, process	Control & production data	Operation, service, worklist	Organization, agent & non-agent
Collaboration of elements studied in	-	Interorganizational Workflow, Instance-Spanning Constraints Collaboration Mining	Interorganizational Workflow, Object-Centric Process Mining	Service Composition, Distributed Business Process, Collaboration Mining	Process Choreography, Virtual Enterprise, Multi-Agent System, Collaboration Mining

Fig. 1. Process perspectives [27] and research areas that study collaboration between perspective's elements and their relation to the collaboration process meta model.

Existing research areas taking the top-down approach are *interorganizational workflows* [2,7,9], *process choreographies* [16,17,20,37,22,21], *virtual enterprises* [14,26,25], *distributed business processes* [13,45], *service compositions* [43,6,4], *multi-agent systems* [47,48], and (process) *instance-spanning constraints* [19,53]. In contrast, the bottom-up approach is taken by *object-centric process mining* [5,11] and *collaboration mining* [24,55,46,54,15,29,38,33,49]. Considering both approaches, collaboration can be the result of multiple organizations (denoted in blue as an element of the organizational perspective in Fig. 1) or an organization's departments (non-agents in Fig. 1) collaborating towards achieving a common business goal. Also, collaboration can be the result of multiple services (element of the operational perspective) or agents (organizational perspective) collaborating to execute a CP. Similarly, collaboration can be the result of multiple objects (informational perspective) collaborating to be processed in a CP. Lastly, collaboration can be the result of multiple process instances (behavioral perspective) collaborating to meet the requirements stated in compliance constraints.

Hence, the main element of the CP meta model defining the various collaboration concepts can be an organization, (non-)agent, services, objects, and process instances. To cover the diverse nature of concepts, we conceptualize all these perspective's elements with their respective process orchestration, e.g., the process orchestration of a particular medical department [33] or of a particu-

lar web service [46] (cf. "conceptualized as" relation in dotted blue in Fig. 1). Our process orchestration conceptualization in the CP meta model is in line with top-down and bottom-up approaches, as work for both approaches typically models its collaboration concept by some process orchestration, e.g., the underlying workflow net for each medical department [33] or for each web service [46]. Hence, the *meta collaboration process* that abstracts the Petri net classes modelling the CP in Fig. 1 consists of process orchestrations that interact in various collaborations, i.e., the process orchestrations of a CP collaborate.

Despite the diverse range of represented concepts in terms of process perspective and granularity level, the relationships between process orchestrations in the CP meta model (cf. Fig. 1) are either a collaboration, e.g., two agent's exchange messages, or *inheritance* [12], e.g., a service and its subservice are abstracted into a single service. The inheritance relation between process orchestrations captures the various granularity levels of existing works' collaboration concepts (department vs. organization) in the CP meta model. The collaboration relationship can be generally defined as either asynchronous or synchronous [2] (cf. Fig. 1). *Message exchanges*, e.g., http-messages in web services [46], *handover-of-work* between process orchestrations, e.g., agent A executes activity "a" and hands the work over to agent B that executes activity "b" [49], and *sharing*, e.g., a doctor in a clinic can only do one task at a time [33], are studied for asynchronous collaboration between process orchestrations. On the contrary, synchronous collaboration is studied as an *activity* that is either executed by multiple process orchestrations together, e.g., an internist and a surgeon consult to determine necessary medication for a patient [33], or multiple process orchestrations are referenced in an activity, e.g., the order and package for packing the items of an order [5].

The applicability of the four collaboration types in the CP meta model to a real-world manufacturing CP are shown in the next section.

2.2   Collaboration Process Meta Model Instance in Manufacturing

To show the applicability of the CP meta model, we instantiate it (depicted in Fig. 2) for a real-world manufacturing CP [35]. The CP is a batch production of chess pieces orchestrated by the Cloud Process Execution Engine (CPEE) [36]. The CPEE executes the batch production process orchestration (denoted in blue in Fig. 2) that refers to the top-level orchestration process executed by the CPEE. The batch production produces all chess pieces that are purchased in an order (informational process orchestration related by synchronous collaboration `<sync>` in Fig. 2). The batch production process orchestration instantiates a production process orchestration for each produced chess piece by message exchange (`<msg>` relation between batch production and production) that is then executed in parallel by the CPEE. As soon as the production process orchestration is finished, it sends a confirmation message back to the batch production process orchestration. Cardinality constraints for the various collaboration relations are also depicted. For example, a single batch production produces all orders in the CP synchronously.
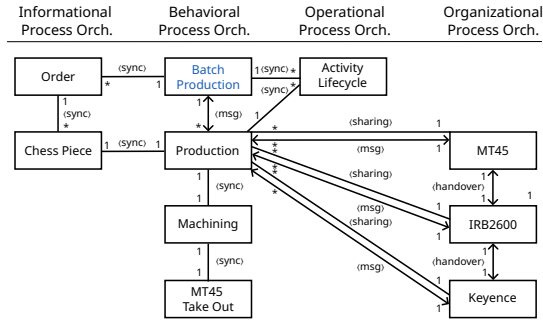
Fig. 2. Selected process orchestrations of a collaboration process meta model instance of the chess piece batch production recorded in [35] as multiple event logs.

Subprocesses instantiated by a behavioral process orchestration, i.e., process orchestrations instantiated with a blocking semantic [36], are a synchronous collaboration, e.g., production and machining in Fig. 2. The CPEE has a fine-grained activity lifecycle model, that is referred to by the activity lifecycle operational process orchestration in Fig. 2. Hence, the respective behavioral and the operational process orchestrations synchronously collaborate by executing the respective activity lifecycle transitions. A MT45 lathe machine, an IRB2600 industrial robot and a Keyence precision measurement machine are responsible for executing the CP. As there is only a single MT45, IRB2600 and Keyence machine, these limited resources are shared by the production process orchestration. The machines are orchestrated by message exchanges, i.e., their process orchestrations collaborate in a `<msg>` collaboration with the production.

Following the meta model in Fig. 1, the CP meta model instance in Fig. 2 abstracts a concrete Petri net CP model that can be mined for the batch production event logs. Depending on the analysis question motivating the mining of a CP model for the batch production, a subset of process orchestrations can be selected to scope the mined CP model. As it is not clear, what Petri net class should be targeted for the CP model by process discovery (cf. Sect. 1), the next section assesses candidate Petri net classes.

3   Assessment of Petri Net Classes in Modelling Collaboration Processes

As elaborated in Sect. 2.1, capturing CP in their entirety is a challenging task, in particular, w.r.t. the CP model class and the CP discovery technique which are tightly intertwined [1]. Hence, in this section, we assess existing Petri Net classes for their capability of modelling and discovering CPs.

3.1   Assessment Criteria

Given the existing research areas that study CPs in Sect. 2, through snowballing and expert knowledge, we identify 20 Petri net classes that are target candidates as CP models for collaboration mining as depicted in Tab. 1. The assessment aims to determine the properties of each Petri net class in modelling CP models. The Petri net classes are presented in reference to their main study and year of publication. To assess the Petri net classes, *general properties* characterize the class from a general conceptual and theoretical perspective, *mining properties* characterize the class with respect to collaboration mining, and *collaboration properties* characterize the class with respect to the four types of collaboration occurring in the CP meta model (cf. Fig. 1).

Table 1. Overview of the 20 Petri net model classes with CP modelling assessment.

Class	Main Study	Year	General Properties				Mining Properties				Collaboration Properties			
			Appr.	Turing Compl.	Data	Semantics	CP Repr.	PD	Rediscover.	Soundness	Messages	Handover	Sharing	Activity
Petri Net	[40]	1962	T	−[39]	I	I/T	I	I	−	NA	I	I	I	I
Colored Petri Net	[28]	1981	T	−[28]	E	I	I	I	−	NA	E	E	E	E
Object System	[50]	1996	T	+[31]	E	I	E	I	−	NA	E	E	E	E
Interorgan. Workflow	[2]	1998	T	−[39]	I	I	E	I	−	D	E	I	I	E
Interaction Petri Net	[16]	2007	T	−[39]	I	I	E	[24][15]	−	?	E²	I	I	−
Compositional Service Tree	[6]	2009	T	−[39]	I	I	E	?	−	D	E	I	I	−
v-PN	[44]	2011	T	−[44]	E	I	I	?	−	U	I	I	I	I
Integrated RM_WF_net	[55]	2013	B	−[39]	I	I	E	[55]	−	D	E	I	E	E
Healthcare Petri Net	[34]	2018	T	−?	I	I	E	?	−	?	I	E	E	−
Synchronuous Proclet System	[18]	2019	T	−?[18]	E	T	E	?	−	NA	I	I	I	E
t-PNID	[41]	2019	T	−[39]	E	I	E	I	−	U	I	I	I	E
Communication Net	[46]	2019	B	−?	I	I	E	[46]	−	?	E	I	I	−
Top-Level Process Model	[54]	2020	B	−?	I	I	E	[54]	−	D	E	I	I	I
System Net	[42]	2020	T	−?	E	T	E	?	−	NA	E	E	I	I
Object-centric Petri Net	[5]	2020	B	−?	E	I	E	[5]	−	?	I	I	I	E
Industry Net	[30]	2022	T	−?	I	I	E	[29]	−	?	E	I	I	−
Multi-Agent System Net	[49]	2023	B	−[39]	I	I	E	[49]	−	D	I	E	I	I
Generalized Workflow Net	[38]	2023	B	−?	I	I	E	[38]	−	D	E	I	I	E
Composed RM_WF_net	[33]	2023	B	−[39]	I	I	E	[33]	−	?	E	I	E	E
Typed Jackson Net	[11]	2023	B	−[11]	E	I	E	[11]	+	S	I	I	I	E

General properties of Petri net classes are the *approach* [37], *Turing completeness* [39], ability to represent *data* (cf. control vs. production data in Sect. 2) and the proposed type of *semantics* [51]. To build a CP model, two approaches exist: *(T)op-down* starts from scratch and models the CP manually by starting with the definition of process orchestrations and collaborations and further refining the workflow nets, while *(B)ottom-up* starts with an event log recorded from information systems supporting the CP execution and mines a CP model. Next, Petri net classes can either be Turing complete $(+)$ or not $(-)$. If the Turing completeness is only conjected, the conjecture is denoted with a ?, e.g., $-^?$.

As the ability to represent a concept in a Petri net class can either be $-$, i.e., the class definition prohibits the representation, *(I)mplicit*, i.e., the concept can theoretically be represented in the class, but has no dedicated element in the definition, or *(E)xplicit*, i.e., the concept has a dedicated element in the definition, the properties data, *CP representation* and all four collaboration properties are based on this distinction. Petri net semantics can either be *(I)nterleaving*, i.e., semantics is defined as traces of fired transitions such that transition labels always interleave/are totally ordered, and *(T)rue concurrency*, i.e., semantics is defined as a *causal net* [51] such that transition firings can be partially ordered or "truly concurrent".

Mining properties are the CP representation, *discovery technique (PD)*, *rediscoverability* [8], and *soundness* [2]. As CPs are a composition of process orchestrations, e.g. [2,33], CP representation refers to the representation of workflow net compositions. To mine a CP model that is an instance of the Petri net class, a discovery technique targeting that Petri net class has to either exist, is unknown to exist *(?)*, or is *(I)nherited* from a Petri net subclass. An important and desired property of a Petri net for collaboration mining is its soundness (and generalized definitions such as *identifier soundness* [11]) that can either be *(N)ot (A)pplicable* as the class is too general, unknown *(?)* as a restated definition is missing, *(U)ndecidable* or *(D)ecidable* as a decision problem, and *(S)ound by construction*.

3.2   Findings

From the assessment result in Tab. 1, we deduce nine findings. The first finding is that research on candidate Petri net classes started with the top-down approach and has gradually put more focus on the bottom-up approach. The second finding is that for 19 Petri net classes, the approach assessment is straightforward, while the typed Jackson net proposal [11] fuses the top-down with the bottom-up approach by taking both approaches in turn and bringing them together through a framework for rediscoverability. However, their motivation puts more focus on the bottom-up approach. The third finding is the observation that Typed Jackson nets and object-centric Petri nets [5] are the only bottom-up approaches

---

[2] Interaction Petri nets are the only model class in our overview, that conceptualizes a message exchange as an atomic, synchronous firing of a transition in a Petri net [16].

with an explicit data representation, a consequence of recent efforts to mine CP models for the informational process perspective (cf. Fig. 1) in which process orchestrations are objects and collaboration is synchronous. Considering the general properties, a fourth finding is that there is no existing discovery technique that defines a true concurrency semantics for its Petri net class. The fourth finding contrasts a recent effort on true concurrency through *systems mining* by [23] and related efforts on partial order-based process mining [32].

Considering the mining properties assessment in Tab. 1, a fifth finding is that early Petri net classes inherit discovery techniques from later proposals for Petri net subclasses, e.g. the *Colliery technique* [15] is a discovery technique for the Interaction Petri net class, which is a subclass of the interorganizational workflow proposed nine years earlier. The sixth finding is that discovery techniques were only proposed three times for an existing Petri net class proposed in a top-down approach, i.e., the two approaches are brought together through distinct publications in [24,15] for the interaction Petri net and in [29] for the industry net. We conject that both the interaction Petri net and industry net are equivalent both in terms of the assessement and theoretically, which emphasizes the need for a central Petri net class for collaboration mining to alleviate potential redundancies in research. The seventh finding is that rediscoverability and sound by construction are rare and coincide in typed Jackson nets, while the importance of soundness in mining is confirmed through classes having a discovery technique, soundness is either *?*, decidable or sound by construction.

Considering the collaboration properties assessment in Tab. 1, the eighth finding is that colored Petri nets and object systems are the only classes that explicitly represent all four collaboration types, but lack a native discovery technique targeting that class. Although the discovery technique [5] for object-centric Petri nets results in an inherited discovery technique for colored Petri nets, the former class lacks explicit representation of three of the four collaboration types. In contrast, the classes integrated RM_WF_net and composed RM_WF_net targeted by the discovery techniques in [55] and [33] explicitly represent three of the four collaboration types. The multi-agent system net targeted by the agent miner [49] is the only class particularly designed for explicitly representing the missing handover collaboration type, pointing to a potential fused class with explicit representation of all four classes and a discovery technique that comes with it. Overall, the eighth finding highlights that the early classes colored Petri net and object systems are too general and powerful to be targeted by a discovery technique (yet), but proposed subclasses for which a discovery technique exists do not yet come with the explicit modelling of all four collaboration types. The ninth and final finding is that there is yet no class for collaboration mining that the workflow net constitutes for process orchestration mining. The next section translates the nine findings into design guidelines for collaboration miners.

4   Design Guidelines for Process Discovery

The assessment of existing Petri net classes as standard candidates for CPs Sect. 3 resulted in nine findings that are the foundation of the design guidelines for future process discovery in collaboration mining. The search for the standard Petri net class for collaboration mining continues and underlies the first three guidelines:

G1 Take the balance between (explicit) modelling and decision power of your targeted Petri net class actively into account. Existing discovery techniques target either the one end, e.g. [11] with good decision power on soundness and rediscoverability, but lack of three collaboration types, or the other end of the bargain, [33] with good modelling power, but lack of known decision power on soundness and rediscoverability. A future discovery technique targeting a (potentially new) class that has modelling power to represent all four collaboration types and good decision power not only brings together the top-down and bottom-up approach, but also is applicable to the challenges studied in a diverse range of research areas and their respective application domains (cf. Fig. 1).
G2 Match your discovery technique's design rationale with the properties of existing classes to access the existing theoretical and practical knowledge and improve rather algorithmically instead of by definition of a new class. If a new class is still necessary, take the first guideline into account.
G3 State your targeted class explicitly and formally, as it allows straightforward assessment similar to Sect. 3.1 and is part of the first maturity stage [52] of your discovery technique. Have all four maturity stages in mind.
G4 Think three times about not taking a divide-and-conquer approach to discovery, as the CP is a composition. If you have another approach, motivate it well.
G5 Balance your design guided by the previous four guidelines with your discovery technique's complexity to achieve sufficient efficiency for practical purposes.

By taking these guidelines into account, collaboration mining techniques can be flexibly combined, easily compared and assessed, and come with a well-researched theoretical foundation.
Limitations: Despite the extensive set of covered techniques and research areas, the heterogeneity of collaboration results may result in having missed existing work on collaboration. Also, the set of assessed Petri net classes is likely not completely comprehensive. Hence, a Petri net class might exists already that is the best candidate for a standard CP model. Additionally, the criteria can be further refined to improve the characterization of the standard CP model.

5   Conclusion

The integration of heterogeneous collaboration in our CP meta model enables the assessment of existing Petri net classes with respect to properties of a standard

model class for CP and can guide new collaboration mining techniques beginning with process discovery. As a clear standard Petri net class for CP is still missing, future techniques should take this lack of standardization into account to advance collaboration mining in a coherent way. Additionally, a fusion of the top-down and bottom-up approach opens up yet to be utilized ways of combining results from both sides.

References

1. van der Aalst, W.M.P.: On the Representational Bias in Process Mining. In: 2011 IEEE 20th International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises. pp. 2–7 (Jun 2011)
2. van der Aalst, W.M.P.: Modeling and analyzing interorganizational workflows. In: Proceedings 1998 International Conference on Application of Concurrency to System Design. pp. 262–272 (1998)
3. van der Aalst, W.M.P.: Interorganizational workflows: An approach based on message sequence charts and petri nets. Systems Analysis Modelling Simulation **34**, 335–367 (1999)
4. van der Aalst, W.M.P.: Service Mining: Using Process Mining to Discover, Check, and Improve Service Behavior. IEEE Trans. Serv. Comput. **6**(4), 525–535 (Oct 2013)
5. van der Aalst, W.M.P., Berti, A.: Discovering object-centric Petri nets. Fundamenta informaticae **175**(1-4), 1–40 (2020)
6. van der Aalst, W.M.P., van Hee, K.M., Massuthe, P., et al., S.: Compositional Service Trees. In: Applications and Theory of Petri Nets. pp. 283–302. LNCS, Springer, Berlin, Heidelberg (2009)
7. van der Aalst, W.M.P., Moldt, D., Valk, R., Wienberg, F.: Enacting interorganizational workflows using nets in nets. In: Proceedings of the 1999 Workflow Management Conference. vol. 70, pp. 117–136. Citeseer (1999)
8. van der Aalst, W.M.P., Weijters, T., Maruster, L.: Workflow mining: discovering process models from event logs. IEEE Trans Knowl Data Eng **16**(9), 1128–1142 (Sep 2004)
9. van der Aalst, W.M.P., Weske, M.: The P2P Approach to Interorganizational Workflows. In: Advanced Information Systems Engineering. pp. 140–156. LNCS, Springer, Berlin, Heidelberg (2001)
10. Augusto, A., Conforti, R., Dumas, M., Rosa et al., M.L.: Automated Discovery of Process Models from Event Logs: Review and Benchmark. IEEE Trans Knowl Data Eng **31**(4), 686–705 (Apr 2019)
11. Barenholz, D., Montali, M., Polyvyanyy, A., Reijers et al., H.A.: There and Back Again. In: Application and Theory of Petri Nets and Concurrency. pp. 37–58. LNCS, Springer, Cham (2023)
12. Basten, T., van Der Aalst, W.: Inheritance of behavior. The Journal of Logic and Algebraic Programming **47**(2), 47–145 (2001)
13. Borkowski, M., Fdhila, W., Nardelli, M., Rinderle-Ma, S., Schulte, S.: Event-based failure prediction in distributed business processes. Inf. Syst. **81**, 220–235 (2019)
14. Chu, X.N., Tso, S.K., Zhang, W.J., Li, Q.: Partnership Synthesis for Virtual Enterprises. Int. J. Adv. Manuf. Technol. **19**(5), 384–391 (Mar 2002)
15. Corradini, F., Re, B., Rossi, L., Tiezzi, F.: A Technique for Collaboration Discovery. In: Enterprise, Business-Process and Information Systems Modeling. pp. 63–78. LNBIP, Springer, Cham (2022)

16. Decker, G., Weske, M.: Local Enforceability in Interaction Petri Nets. In: BPM 2007, vol. 4714, pp. 305–319. Springer, Berlin, Heidelberg (2007)
17. Decker, G., Weske, M.: Interaction-centric modeling of process choreographies. Information Systems **36**(2), 292–312 (Apr 2011)
18. Fahland, D.: Describing Behavior of Processes with Many-to-Many Interactions. In: Application and Theory of Petri Nets and Concurrency. pp. 3–24. LNCS, Springer, Cham (2019)
19. Fdhila, W., Gall, M., Rinderle-Ma, S., Mangler et al., J.: Classification and Formalization of Instance-Spanning Constraints in Process-Driven Applications. In: BPM 2016. LNCS, vol. 9850, pp. 348–364. Springer (2016)
20. Fdhila, W., Indiono, C., Rinderle-Ma, S., Reichert, M.: Dealing with change in process choreographies: Design and implementation of propagation algorithms. Information Systems **49**, 1–24 (Apr 2015)
21. Fdhila, W., Knuplesch, D., Rinderle-Ma, S., Reichert, M.: Verifying compliance in process choreographies: Foundations, algorithms, and implementation. Information Systems p. 101983 (Jan 2022)
22. Fdhila, W., Rinderle-Ma, S., Knuplesch, D., Reichert, M.: Change and Compliance in Collaborative Processes. In: 2015 IEEE International Conference on Services Computing. pp. 162–169. IEEE, New York City, NY, USA (Jun 2015)
23. Fettke, P., Reisig, W.: Systems Mining with Heraklit: The Next Step (Jun 2022), arXiv:2202.01289 [cs]
24. Gaaloul, W., Baïna, K., Godart, C.: Log-based mining techniques applied to Web service composition reengineering. Serv. Oriented Comput. Appl. **2**(2), 93–110 (Jul 2008)
25. Garcia, E., Giret, A., Botti, V.: Designing normative open virtual enterprises. Enterp. Inf. Syst. **10**(3), 303–324 (Mar 2016)
26. Grefen, P., Mehandjiev, N., Kouvas, G., Weichhart, G., Eshuis, R.: Dynamic business network process management in instant virtual enterprises. Computers in Industry **60**(2), 86–103 (Feb 2009)
27. Jablonski, S., Bussler, C.: Workflow management: modeling concepts, architecture and implementation. ITP New Media (1996)
28. Jensen, K.: Coloured petri nets and the invariant-method. Theor Comput Sci . **14**(3), 317–336 (1981)
29. Kwantes, P., Kleijn, J.: Distributed Synthesis of Asynchronously Communicating Distributed Process Models. In: Transactions on Petri Nets and Other Models of Concurrency XVI, pp. 49–72. LNCS, Springer, Berlin, Heidelberg (2022)
30. Kwantes, P.M., Kleijn, J.: On the synthesis of industry level process models from enterprise level process models. In: ATAED@ Petri Nets/ACSD. pp. 6–22 (2018)
31. Köhler, M., Rölke, H.: Properties of Object Petri Nets. In: Applications and Theory of Petri Nets 2004. pp. 278–297. LNCS, Springer (2004)
32. Leemans, S.J.J., van Zelst, S.J., Lu, X.: Partial-order-based process mining: a survey and outlook. Knowledge and Information Systems (Nov 2022)
33. Liu, C., Li, H., Zhang, S., Cheng et al., L.: Cross-Department Collaborative Healthcare Process Model Discovery From Event Logs. IEEE Trans. Autom. Sci. Eng. **20**(3), 2115–2125 (Jul 2023)
34. Mahulea, C., Mahulea, L., García Soriano, J.M., Colom, J.M.: Modular Petri net modeling of healthcare systems. Flex. Serv. Manuf. J. **30**(1), 329–357 (Jun 2018)
35. Mangler, J., Ehrendorfer, M.: XES Chess Pieces Production (May 2023). https://doi.org/https://zenodo.org/record/7477845
36. Mangler, J., Rinderle-Ma, S.: Cloud Process Execution Engine: Architecture and Interfaces (Sep 2022), arXiv:2208.12214 [cs]

37. Meyer, A., Pufahl, L., Batoulis, K., Fahland, D., Weske, M.: Automating data exchange in process choreographies. Information Systems **53**, 296–329 (Oct 2015)
38. Nesterov, R., Bernardinello, L., Lomazova, I., Pomello, L.: Discovering architecture-aware and sound process models of multi-agent systems: a compositional approach. Softw Syst Model **22**(1), 351–375 (Feb 2023)
39. Peterson, J.L.: Petri Nets. ACM Computing Surveys **9**(3), 223–252 (Sep 1977)
40. Petri, C.A.: Kommunikation mit automaten (1962)
41. Polyvyanyy, A., van der Werf, J.M.E.M., Overbeek, S., Brouwers, R.: Information Systems Modeling: Language, Verification, and Tool Support. In: Advanced Information Systems Engineering. pp. 194–212. LNCS, Springer, Cham (2019)
42. Reisig, W.: Composition of Component Models - A Key to Construct Big Systems. In: Leveraging Applications of Formal Methods, Verification and Validation: Engineering Principles. pp. 171–188. LNCS, Springer, Cham (2020)
43. Rinderle-Ma, S., Reichert, M., Jurisch, M.: Equivalence of Web Services in Process-Aware Service Compositions. In: 2009 IEEE International Conference on Web Services. pp. 501–508 (Jul 2009)
44. Rosa-Velardo, F., de Frutos-Escrig, D.: Decidability and complexity of Petri nets with unordered data. Theor Comput Sci . **412**(34), 4439–4451 (Aug 2011)
45. Schulte, S., Hoenisch, P., Hochreiner, C., Dustdar et al., S.: Towards Process Support for Cloud Manufacturing. In: 2014 IEEE 18th International Enterprise Distributed Object Computing Conference. pp. 142–149. Ulm, Germany (Sep 2014)
46. Stroiński, A., Dwornikowski, D., Brzeziński, J.: A Distributed Discovery of Communicating Resource Systems Models. IEEE Trans. Serv. Comput. **12**(2), 172–185 (Mar 2019)
47. Tan, W., Xu, W., Yang, F., Xu et al., L.: A framework for service enterprise workflow simulation with multi-agents cooperation. Enterp. Inf. Syst. **7**(4), 523–542 (2013)
48. Tour, A., Polyvyanyy, A., Kalenkova, A.: Agent System Mining: Vision, Benefits, and Challenges. IEEE Access **9**, 99480–99494 (2021)
49. Tour, A., Polyvyanyy, A., Kalenkova, A., Senderovich, A.: Agent Miner: An Algorithm for Discovering Agent Systems from Event Data (Jul 2023), arXiv:2212.01454
50. Valk, R.: On Processes of Object Petri Nets (1996)
51. Van Glabbeek, R., Vaandrager, F.: Petri net models for algebraic theories of concurrency. In: PARLE Parallel Architectures and Languages Europe: Volume II: Parallel Languages. pp. 224–242. Springer (1987)
52. van der Werf, J.M.E.M., Polyvyanyy, A., van Wensveen, B.R., Brinkhuis, M., Reijers, H.A.: All that glitters is not gold: Four maturity stages of process discovery algorithms. Information Systems **114**, 102155 (Mar 2023)
53. Winter, K., Rinderle-Ma, S.: Defining Instance Spanning Constraint Patterns for Business Processes Based on Proclets. In: Conceptual Modeling. pp. 149–163. LNCS, Springer, Cham (2020)
54. Zeng, Q., Duan, H., Liu, C.: Top-Down Process Mining From Multi-Source Running Logs Based on Refinement of Petri Nets. IEEE Access **8**, 61355–61369 (2020)
55. Zeng, Q., Sun, S., Duan, H., Liu et al., C.: Cross-organizational collaborative workflow mining from a multi-source log. Decis Support Syst **54**, 1280–1301 (Feb 2013)