Information Systems 141 (2026) 102728

Contents lists available at ScienceDirect

Information Systems

journal homepage: www.elsevier.com/locate/is

Object-centric process management: A research manifesto

Anjo Seidel [1,*], Mathias Weske [1,*], Marco Montali [2,*], Andrey Rivkin [3,*], Manfred Reichert [4,*], Jan Martijn E.M. van der Werf [5,*], Wil M.P. van der Aalst [6], Marius Breitmayer [4], Lukas Liss [6], Jan Niklas van Detten [6,36], Amin Jalali [7], Shahrzad Khayatbashi [8], Maximilian König [1], Tom Lichtenstein [1], Stefanie Rinderle-Ma [9], Barbara Weber [10], Pnina Soffer [11], Lorenzo Rossi [12], Daniel Calegari Garcia [13], Andrea Delgado [14], Remco Dijkman [15], Sarah Winkler [2], Matthias Weidlich [16], Sander J.J. Leemans [6], Dirk Fahland [15], Ava Swevels [15], Monique Snoeck [17], Giancarlo Guizzardi [18], Alessandro Gianola [19], Avigdor Gal [20], Ekkart Kindler [3], Irina Lomazova [21], Barbara Re [12], Giovanni Meroni [3], Andrea Morichetta [12], Alessandro Marcelletti [12], Sara Pettinari [22], Boudewijn van Dongen [15], Johannes De Smedt [17], Majid Rafiei [23], Julius Köpke [24], Thomas Hildebrandt [25], Francesca Zerbato [15], Luise Pufahl [9], Hajo Reijers [5], Artem Polyvyanyy [26], Chiara Di Francescomarino [27], Fabrizio Maria Maggi [2], Oscar Pastor [28], Stephan Haarmann [29], Henderik Proper [30], Xixi Lu [5], Hugo López [3], Tijs Slaats [25], Jochen De Weerdt [17], Massimiliano de Leoni [31], Niels Martin [32], Karolin Winter [15], Nick van Beest [33], Orlenys López-Pintado [34], Sebastiaan van Zelst [35], Chiara Ghidini [2], Arik Senderovich [36]

[1] *Hasso Plattner Institute, University of Postdam, Prof.-Dr.-Helmert-Straße 2-3, Potsdam, 14482, Germany*
[2] *Free University of Bozen-Bolzano, Italy*
[3] *Technical University of Denmark, Lyngby, Denmark*
[4] *Ulm University, Germany*
[5] *Utrecht University, The Netherlands*
[6] *RWTH Aachen University, Germany*
[7] *Stockholm University, Sweden*
[8] *Linköping University, Sweden*
[9] *Technical University of Munich, Germany*
[10] *University of St. Gallen, Switzerland*
[11] *University of Haifa, Israel*
[12] *Universita di Camerino, Italy*
[13] *Universidad ORT, Uruguay*
[14] *Universidad de la Republica, Uruguay*
[15] *Eindhoven University of Technology, The Netherlands*
[16] *Humboldt-Universität zu Berlin, Germany*
[17] *KU Leuven, Belgium*
[18] *University of Twente, The Netherlands*
[19] *INESC-ID/Instituto Superior Técnico, Universidade de Lisboa, Portugal*
[20] *Technion, Israel Institute of Technology, Israel*
[21] *HSE University, Moscow, Russia*
[22] *Gran Sasso Science Institute, L'Aquila, Italy*
[23] *SAP SE, Germany*
[24] *University Klagenfurt, Austria*
[25] *University of Copenhagen, Denmark*
[26] *University of Melbourne, Australia*
[27] *University of Trento, Italy*
[28] *Universidad Politécnica de Valencia, Spain*
[29] *Camunda Services GmbH, Germany*

* Corresponding authors.
E-mail addresses: anjo.seidel@hpi.de (A. Seidel), mathias.weske@hpi.de (M. Weske), montali@inf.unibz.it (M. Montali), ariv@dtu.dk (A. Rivkin), manfred.reichert@uni-ulm.de (M. Reichert), j.m.e.m.vanderwerf@uu.nl (J.M.E.M.v.d. Werf).

https://doi.org/10.1016/j.is.2026.102728
Received 30 March 2026; Accepted 2 April 2026
Available online 30 April 2026
0306-4379/

[30] Technische Universität Wien, Austria
[31] University of Padova, Italy
[32] Hasselt University, The Netherlands
[33] Commonwealth Scientific and Industrial Research Organization, Australia
[34] University of Tartu, Estonia
[35] Celonis SE, Germany
[36] York University, Canada

ARTICLE INFO

Recommended by Dennis Shasha

Keywords:
Information systems
Business process management
Object-centric process mining
Object-centric process modeling

ABSTRACT

Business process management employs process models and event logs to represent the behavior of the information systems under study. Traditional case-centric notions consider the order of activities and events in isolated process instances. The emerging field of object-centric processes challenges this assumption by putting objects in the center. Object-centric process mining and modeling approaches identify the structure of co-evolving data objects that influence the behavior of an information system to provide a comprehensive view of the system behavior. Object-centricity has been investigated independently in process modeling and in process mining, which resulted in the coexistence of seemingly contradictory assumptions and definitions. As a community effort, this research manifesto relates and aligns existing terminologies, definitions, and perspectives to provide a common ground for current and future research in object-centric business process management. Based on the current state of research, we propose a conceptualization that sets process models and event logs in relation to the information system's behavior and the execution data it generates. The conceptualization aims at aligning different terminologies and, thus, providing a basis to model and analyze behavioral characteristics. Building on this common ground, we identify open research challenges along the most relevant research areas in object-centric process management. For each research area, its current status is investigated and an outline of the most relevant research challenges is presented.

1. Introduction

Over the last decades, business process management has developed into a mature discipline with a solid conceptual foundation and significant impact on industrial practice [1,2]. Modern organizations model and improve their business processes to automate, execute, and evaluate them. For many years, different cases have been considered in isolation, such as individual purchase orders being processed. Although this approach proved useful in process modeling and automation, it led to silos, does not adequately represent the interdependencies between different cases, and to ignores the role of data objects. This shortcoming has led to a novel perspective on business processes called object-centricity [3].

In object-centric business processes, objects and their relationships to other objects and activities are in the center of attention [3]. This allows us to break the case silos and provide a comprehensive view on how processes are executed. Starting with process mining, object-centricity has become a major research field in business process management, shaking up all phases of the business process management lifecycle, including design, configuration, enactment, and evaluation [1].

This paper reviews current approaches in object-centric process management, provides a joint framework to represent object-centric business processes, and introduces a broad range of research challenges in this exciting research field.

To set the scene for the investigations in this paper, Ackhoff's hierarchy of data, information, and knowledge [4] is used to organize key artifacts in business process management. Information systems support the execution of business processes in organizations where humans interact with artifacts to create, store, and exchange information [5–7]. Business process management studies the behavior that emerges in information systems from different perspectives [8]. In this discipline, process models play a key role in describing the behavior of information systems as coordinated sets of activities that are carried out with the goal of creating value in organizations [1,2]. Meanwhile, event logs record changes that occurred in the information system that are relevant to the investigated process [9]. Consider the example of a hospital information system [10], where humans with different roles interact with medical hardware and software during patient treatment. A process model may capture clinical guidelines for treating patients or administrative processes, while an event log captures the events related to patient treatment that are recorded in software systems.

To illustrate the role of process models and event logs in relation to their subject under study, Fig. 1 applies them to the well-known hierarchy of data, information, and knowledge by Ackoff [4]. Data comprises the real-world information system, which is the subject of study, and the raw execution data. The information system comprises a setting in which humans perform work in coordination with machines. These interactions leave traces, which are stored as execution data. To analyze the relevant aspects of this data layer, it must be aggregated into structured information that holds its semantic content [11]. In business process management, process models and event logs play the role of information in Ackoff's hierarchy to serve as an intermediary between the knowledge about processes and the information system with its execution data. Here, information loss and an abstraction between the layers are inherent to manage the complexity of information systems.

The relationship between the original information system and its execution data on the one hand, and process models and event logs on the other hand, can be understood by referring to Stachowiak's general model theory [12]. A model has a purpose, according to which it maps certain features of the original and abstracts from others.

Traditionally, process models and event logs are case-centric. In process modeling and mining, the concept of a *case notion* defines the scope of the process. From a modeling perspective, process models based on domain knowledge are created to define the order of activities between a clear start and end to achieve a specific business goal [1]. A case is a process instance, that is, a sequence of activity executions from start to end. Such process models are to be implemented in the information system. Fig. 1 shows the levels of this implementation from top to bottom. Meanwhile, in process mining, the aim is to move from bottom to top in Fig. 1. Here, the first task is to identify cases, which represent process instances, and events, which capture activity executions, in the execution data. Event logs capture traces of temporally ordered events for all identified cases and serve as the basis for discovering process models and deriving process knowledge.

A key limitation of these traditional approaches is the assumption of isolated cases, which prevents process models and event logs from capturing any behavior that affects more than one case. However, real-world information systems typically contain entities that influence the

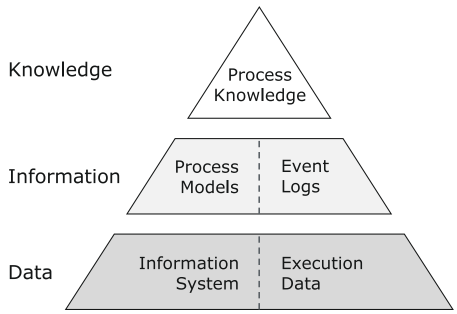

Fig. 1. The hierarchy of data, information, and knowledge in the context of object-centric processes, relating information systems and execution data, process models and event logs, and process knowledge.

behavior of not just one but many cases. Looking into the execution data, some identified events may correspond to multiple entities. A case notion is typically defined on the basis of attribute values that reference such entities. Hence, events may belong to multiple cases and even multiple different case notions, while some other events may not be captured by a certain case notion [13]. Especially in process mining, this raises the question of which case notion should be used to capture processes realistically and the relevant events as intended. The traditional single-case perspective does not account for different case notions simultaneously and can significantly affect the resulting business insights [3].

To overcome the single-case perspective and capture processes more realistically, the concept of *Object-Centric Process Mining* was introduced [3]. Here, multiple case notions can co-exist at the same time and are represented as *object types*. Events are correlated with all their related *objects* instead of only isolated cases. As a result, discovered object-centric process models explicitly represent how multiple case notions with potentially complex relations interact with each other [14,15]. This enables process models to capture the interrelation of structural and dynamic aspects of information systems.

In process modeling, the assumption of isolated cases has been tackled early on by identifying and modeling, for example, resources and data values that affect many cases. This is captured in established frameworks such as WfMC [16], ARIS [17], AMFIBIA [18], or Leymann & Roller [19]. Many other long-existing process modeling approaches refer to similar concepts of “objects” and “data” for modeling behavior in information systems with different terms: artifact-centric [20], data-aware [21], or object-aware [22]. All these approaches share the common goal of characterizing structures in an information system as certain kinds of “objects” to define and analyze their behavior. Despite the common goal and similar underlying concepts, the individual terminologies in these approaches differ and introduce different assumptions about information systems and the processes they support.

In summary, object-centric process mining and different process modeling approaches share the same goal of using process models that capture behavior in combination with structural information as objects. However, the concrete notion of an “object” and a useful set of object types can differ between approaches and is highly dependent on the intended purpose of the process model. On the one hand, different approaches may be of value for different tasks and different goals. For example, some modeling approaches are better suited for formal verification or comprehensive visualization than others. On the other hand, different stakeholders may be interested in individual perspectives on the same system, which requires a selection of relevant object types for their points of view.

Taken together, the above observations indicate that the object-centric process paradigm requires a deeper study and, in particular, the development of a holistic conceptual foundation that reconciles existing notions of object-centricity from process mining with historical perspectives from process modeling and workflow management.

This research manifesto is devoted to the goal of bringing together the different perspectives on object-centricity from process modeling and process mining. The purpose of this paper is to establish a common foundation for future research on object-centric process modeling and mining by relating and aligning concepts and terminology into the research area of *object-centric business process management* and identifying future research challenges. This paper elaborates on general model theory and provides an overview of existing object-centric process representations[1] in Section 2. Section 3 provides a conceptual model of the most relevant concepts in the object-centric paradigm. This conceptualization is then used to define a unified terminology for object-centric processes. Based on this conceptualization, Section 4 investigates the behavioral characteristics of object-centric processes.

To guide future research, Section 5 presents object-centric process management by unifying the most widely adopted business process management lifecycles from process mining and modeling and relating them to our conceptualization. A set of research areas is identified, and for each research area, an overview of existing work and the most significant research challenges is highlighted. Finally, Section 6 concludes the paper.

2. Object-centric process representations

As a philosophical underpinning for representing processes, we reflect on Stachowiak’s general model theory before providing an overview of the existing object-centric process modeling approaches and event log formats.

2.1. Modeling purposes

In his work, *general model theory*, Stachowiak [12] investigates the general concept of models from a philosophical perspective. This theory offers a characterization of models and their role in epistemology [23]. Stachowiak defines a *model* as a representation of an *original*. Each model serves a specific modeling purpose. This purpose determines which features of the original are to be represented by the model and which are abstracted from. Therefore, he defines three modeling features as illustrated in Fig. 2: (i) The *mapping* of features of the original that are represented in the model, (ii) the *abstraction* of the features of the original that are not represented in the model, and (iii) the *pragmatism* that requires the model to be useful for a certain purpose, that is, the modeling goal. Typically, several models describe a given original from different perspectives.

In the context of this paper, we can think of a business process as an original. The process is running in an organization, with people, information systems, and data objects involved. There are several models that define important aspects of the process, including process models, organizational models, and, for instance, data models. All these models map important aspects of the original business process while abstracting from aspects that are not relevant for their modeling goal.

Stachowiak distinguishes two types of models. A *descriptive* model represents an original, which already exists; it describes an original. A *prescriptive* model represents an original that does not yet exist in reality, i.e., it is envisioned to exist in the future based on the model.

Interestingly, business process models can serve as prescriptive or as descriptive models. A business process model that defines how a process should be implemented in an organization is certainly a prescriptive model. Process diagrams expressed in the Business Process Model and Notation (BPMN) [24] are often used to define prescriptive business processes or to document intended business processes. On the other hand, a process model that is discovered from an event log describes the

[1] This paper considers publications up to November 2025.

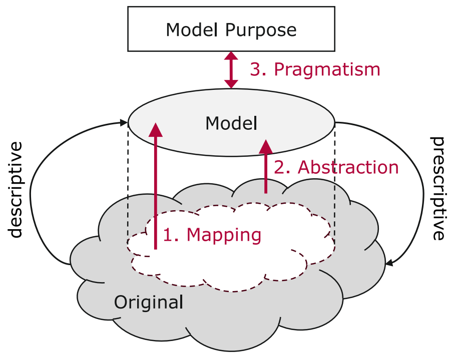

Fig. 2. The relation of original and model as described by Stachowiak [12]: 1. Mapping of features of the original to the model, 2. Abstraction of features that are not required for the modeling purpose, 3. Pragmatism of the model towards a purpose.

behavior of the process. Therefore, it can be considered a descriptive model. Directly Follows Graphs (DFG) [1] are often used as descriptive process models. In fact, an event log can be considered a model of an executed business process, as the business process execution (the "original", according to Stachowiak) is captured in the process execution data, and the event log maps only those attributes that are relevant and abstracts from the other attributes.

Object-centricity aims at modeling not only ordered activities but also relevant entities manipulated by those activities, like data objects. Related entities manifest structures that jointly influence behavior in an information system.

In summary, general model theory allows us to reason about the relation between a model and its original by investigating how it maps and abstracts features to serve a modeling purpose. Before elaborating on typical object-centric process management purposes in Section 5, we provide an outline of how existing object-centric process modeling approaches and event log formats map and abstract features of an original business process and its execution data.

2.2. Object-centric process modeling approaches

While object-centricity came up in process mining only recently [3], similar concepts have been discussed from a process modeling perspective for some time now. Many approaches have been introduced under different labels such as artifact-centric [25], data-centric [26], data-aware [27], object-aware [22], and object-centric [28] process modeling. Steinau et al. [26] and Berti et al. [29] provide comprehensive overviews of data-centric and object-centric process modeling approaches.

In the following, we outline how these approaches aim to achieve the same goal: modeling the interrelation of entity relationships and process behavior. These approaches are structured into (i) multi-model approaches, (ii) formal Petri net-based approaches, (iii) object-centric extensions of control flow-centric models, and (iv) declarative approaches.

2.2.1. Multi-model approaches

Combining several models [30] allows us to capture different aspects of a business process with dedicated notations.

FlexConnect [28] uses a hierarchical combination of activity diagrams, which can be interlinked through different types of objects. These objects can represent entities in the process and their relationships, establishing interrelations between the process fragments. The BALSA UML (BAUML) Framework [31,32] combines models from the UML standard [33]: a data model, state transition diagrams for each object, and activity diagrams are used to capture related objects. The Information System Model (ISM) [34] is inspired by case handling systems [35]. It combines Petri nets with identifiers and Object Role Models to design and analyze information systems. PHILharmonicFlows [22] combines a data model, object lifecycle models for every object type, and a declarative coordination model on the allowed state transitions for related objects. For each state, attribute-level operations are defined.

Similarly, MERODE [21,27] combines a data model, object lifecycles, and a correlation mechanism to bridge the gap between high-level BPMN process models and the data level of an information system. Data objects and their states are enriched with data operations. An explicit data model defines the object types, their attributes, and associations. An object-event table correlates related objects through events that process them jointly.

Different approaches have been introduced to bring object-centricity directly to the industry standard Business Process Model and Notation (BPMN) [24]. Hewelt et al. propose fragment-based Case Management (fCM) [36]. The fCM approach flexibly combines BPMN process fragments based on data objects and their states. These object-centric BPMN process models [37] are enriched with an explicit data model and object lifecycles for all object types. The process fragments define activities and their interactions with objects through a formal Petri net-based semantics [38]. For formal analysis, the deltaBPMN framework [39] combines BPMN process models with SQL-based manipulation of volatile and persistent data for verification via SMT-based model checking. The Process and Organizational Data Integrated Metamodel (BPODIM) [40] approach combines process execution and organizational data through an integrated metamodel. Raw data from process engines and databases are linked via matching mechanisms, enabling joint process and data mining for intra- and inter-organizational scenarios.

2.2.2. Petri net-based approaches

Petri nets are a well-established technique for formalizing concurrent behavior. Firing transitions represent activity executions by deleting tokens and producing tokens on places of the Petri net, realizing state transitions. Existing approaches differ in the representation of objects and their relationships.

Colored Petri nets extend this idea further by allowing tokens to carry data values, called color, which can be of an arbitrarily complex type, called color set. However, analysis of these nets is typically infeasible as reachability is undecidable in general [41]. An extension to Petri nets to model individual objects are Relation Nets [42], where tokens carry identifiers from a finite set of individual identities. In ν-Petri nets, transitions can create new identifiers [43]. While each token in ν-Petri nets carries a single identifier, Petri nets with identifiers allow tokens to carry a tuple of identifiers, thus introducing relations between identifiers [44].

A different approach is taken in the Nets in Nets paradigm. For example in object-oriented Petri nets [45], tokens refer to objects, and objects are again modeled as Petri nets, thus creating a hierarchy of Petri nets, where Petri nets on different layers in the hierarchy can synchronize. In this paradigm, objects can have their own state, behavior and identity, similar to the ideas of objects in object-oriented programming [46].

Synchronous proclets [47] model each object type as an individual Petri net, i.e., a proclet. The transitions of two proclets can be synchronized, thus correlating the two object types. Once a set of objects is synchronized, they need to be processed together throughout the execution. Different cardinality constraints and synchronization modes – namely exact and subset synchronization – are defined. Proclets without correlation constraints can be discovered from object-centric event data, as shown in [48,49].

Object-centric Petri nets (OCPN) [50] represent objects as tokens that flow through a combined net that contains dedicated places for each object type. Transitions may process a variable number of tokens for an adjunct place and hence allow flexible correlations of object types. However, the relationships between object types remain implicit.

Object-centric Petri nets with identifiers (OPID) [51] combine OCPNs with Petri nets with identifiers. In OPIDs, object identities and relationships are made explicit. They can be created, removed, and used to explicitly enforce the joint processing of related objects. Under certain assumptions, OCPNs can be directly mapped to the more precise OPIDs [52].

A parallel stream of work extends OCPNs by discovering synchronization- and specialization-sensitive object-centric Petri nets (SYSSOC-PN) [53]. Sets of objects that are always correlated are discovered and explicitly encoded into the Petri net through so-called *silent objects*. Furthermore, the discovery of object-centric process trees [54] serves as an intermediary step for OCPN discovery and guarantees the construction of sound OCPNs. The discovery of more complex relationships between objects allows inferring synchronization mechanisms analogous to OPIDs [55].

All these Petri net-based approaches share an abstraction of attributes and focus only on the processing of objects based on the control flow. Different types of Petri nets have been extended to support the analysis of data values during process execution. Catalog and object-aware nets (COA-nets) [56] and DB-nets [57] can inspect and manipulate the value of tokens compared to a read-only memory database. The PetriFlow language [58,59] allows us to model object lifecycles in the form of place/transition Petri nets with inhibitor arcs, reset arcs, read arcs, and data attributes, including references to other objects. However, these approaches abstract from event types that process multiple objects at once. Data-aware OPIDs (DOPID) [60] extend OPIDs by representing attribute values and distinguishing between exact and subset synchronization, mapping multi-object processing and data values.

2.2.3. Object-centric model extensions

In object-centric process mining, various process model discovery techniques have been proposed that extend well-established case-centric process modeling approaches to capture multiple object perspectives.

Object-centric directly follows graphs (OC-DFG) [3] are an extension of regular directly follows graphs (DFG). Similarly to DFGs, OC-DFGs represent event types as nodes. But additionally, OC-DFGs represent the *directly follows relations* of different object types as distinct colors. Relationships and attributes are abstracted for the purpose of focusing on event ordering. Object-centric variants [61] are clusters of similar process executions derived from the order of events of sets of related objects. Object-centric causal nets [62] extend causal nets [63] to represent causal relationships between event types for many object types at once. In addition, it has been demonstrated that balancing abstraction and mapping can lead to the discovery of simpler and more understandable object-centric causal nets [64,65].

2.2.4. Declarative approaches

The Guard-Stage-Milestone (GSM) approach constitutes one of the earliest artifact-centric declarative process modeling approaches, where tuples in the database are linked to guards and milestones in different stages of the process [20,66–68]. GSM defines a data model including attributes and represents processing steps and states as nested stages. Event-condition-action rules serve as guards for entering a stage and milestones for leaving a stage. Both connect stages to the data model and constrain how events operate on data and attributes. Although operating close to the data model, the definition of objects and their relations remains implicit. Furthermore, GSM models can be discovered from event logs [69,70].

Object-centric Behavioral Constraints (OCBC) [71,72] provides a declarative process modeling approach explicitly for object-centric processes. It combines DECLARE [73–75] with data models in the form of UML class diagrams [33]. A DECLARE model defines temporal ordering relations between activities, while the data model defines object types and their relationships. The combined model allows linking the two such that activities and temporal relations can be correlated to object types and their relationships. In a related approach, object-centric DECLARE [76] directly integrates objects and their relationships with a DECLARE model. Instead of representing them through an explicit data model, the temporal ordering relations are annotated with the object types involved.

Similarly, object-centric dynamic condition response graphs (OC-DCR) [77] extend on DCR graphs [78] to define a set of object types. Objects are defined as template class DCR graphs from which an object, i.e., a subgraph of activities and behavioral constraints, can be dynamically spawned and added to the top level DCR graph. Behavioral constraints can also be defined between object types, establishing object relationships. Recently, OC-DCR graphs have been studied in combination with Entity-Relationship-diagrams [79] and to specify choreographies between organizations [80].

Temporal object type models (TOTeM) [81] extend data models with temporal relationships between object types. While abstracting from concrete behavior, the temporal relationships provide the possibility to reason about the relationship of objects' processing states.

2.3. Object-centric event log formats

In recent years, several approaches have been introduced to define event log formats to store object-centric process execution data. We summarize the existing proposals for object-centric event logs, the advancements of the OCEL format, and the prospect of event knowledge graphs for storing object-centric event data.

The first attempts to extract object-centric event logs were made by storing collections of multiple event logs, where each log represents a distinct object type [49,70]. As the first approach to define unified object-centric event logs, Li et al. [82] extract extensible object-centric event logs from databases. With OCEL 1.0, the first widely accepted object-centric event log format was proposed [83]. OCEL 2.0 [84] extends the format by recording object relationships and attributes of object types to address limitations of the first version [85]. Still, OCEL 2.0 does not capture relationship changes. A set of benchmark event logs is provided in the OCEL format.[2] The transformation of traditional XES logs into OCEL is described by Rebmann et al. [86].

The IEEE Task Force on Process Mining works on a meta-model for the format of Object-Centric Event Data (OCED) [87–89]. In this context, Goossens et al. [90] provide an overview of existing approaches, and Verbruggen et al. [91] aim for a unified terminology.

The main agreement is that object-centric event data, in contrast to traditional event logs, correlate individual events not via case identifiers. Instead, the event log only captures the recorded events with timestamps and correlated objects. OCED suggests that objects can be linked to one another and have attribute values independent of events.

To support the manufacturing domain, an adaptation of the OCEL format was recently introduced for the Internet of Things (IoT) domain [92], enabling novel use cases.[3]

Object-Centric Process Querying (OCPQ) [93] aims at analyzing object-centric event logs efficiently. OCPQ provides a querying and constraint language, including a graphical representation where objects and events are considered first-class citizens. Languages like OCPQ are vital to navigate complex object-centric event logs.

In addition to the traditional XES [94] storage format for event logs, the use of knowledge graphs for object-centric event data promises improved analysis from multiple object-centric perspectives [95]. To facilitate the use of knowledge graphs, the translation of OCEL into event knowledge graphs [96], and back [97], is proposed. Different case studies show the value of the richness and advantages in querying event knowledge graphs [98,99]. A set of benchmark knowledge graphs is provided [95].

[2] https://www.ocel-standard.org/event-logs/overview/

[3] https://zenodo.org/records/8139445

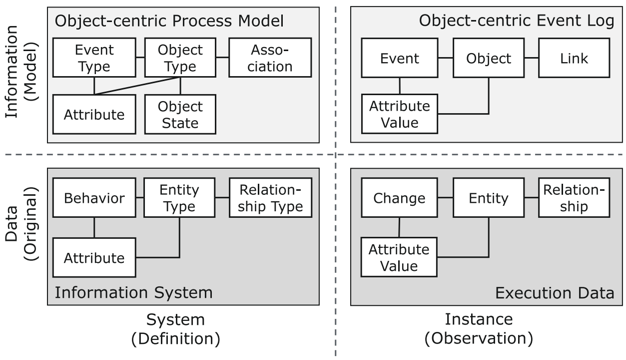

Fig. 3. A conceptual model relating the four quadrants of object-centric process models (top left) as representations of object-centric event logs (top right), which model the behavior in information systems (bottom left) and their execution data (bottom right).

3. Concepts and terminology

Based on the review of relevant approaches presented in Section 2, we now provide a conceptual model for object-centric process representations. It serves as a basis for defining a common terminology.

The conceptual model in Fig. 3 represents the main concepts of object-centric process models and the respective event log representations. It refines the data and information hierarchy in Fig. 1 by placing information systems and execution data in the data level at the bottom. As object-centric process models and event logs are models of an information system and of execution data, respectively, these appear at the top of the figure. To distinguish the top and bottom layers, we use the UML terminology [33] in the top and terminology from entity relationship modeling [100] in the bottom layer. Horizontally, a distinction is made between the system definition and the instance observations. The information system and its technical and organizational configurations define its behavior, while the produced data manifest observations of instances.

3.1. The four quadrants

The quadrants in Fig. 3 represent different aspects of object-centric processes. Starting at the bottom with the data level, the information systems define which execution data can be observed. At the information level, in the top of Fig. 3, process models define possible process instances that can be recorded in event logs.

As the basis for object-centric process management, an information system is a technical and organizational configuration in which behavior emerges as humans and machines exchange information [5–7]. It contains technical implementations, as well as explicit and tacit rules on the organization and behavior of different types of entities and their relationships. The information system is an important asset to execute business processes.

During business process execution, information systems generate and store data about entities, their relations, and the changes they undergo. Furthermore, system logs, data streams, and analog files can serve as sources for identifying discrete events that capture the changes of entities. For example, databases record both the current state of entities and any recent changes to them through redo logs [101]. Note that these records are subject to information loss since they cannot capture all continuous behavior in the real world.

Object-centric process models are representations of behavioral aspects of entity types in an information system. Guided by modeling decisions, they represent the relevant entity types as object types and their relationship types as associations. Furthermore, one object of an object type can even be used to model sets of entities or even virtual entities [54]. The behavior of individual objects is defined through discrete states that can be changed by events. As event types can be related to multiple object types, they define the combined behavior of multiple object types.

An object-centric event log captures instances of object-centric process models. On the one hand, it records events, objects, and their relationships as instances of the type-level definition in the process model. At the same time, these instances are abstractions from the original execution data on the data level. The object-centric event log represents concrete entities as objects with attribute values and object relationships as links. The recorded changes in the information system are stored as events with their corresponding objects.

3.2. Relationships between the quadrants

In the following, the concepts of the information level in Fig. 3 are described in relation to their original counterparts in the data level.

Object types are representations of entity types that exist in the real world. An object type may represent physical as well as virtual entity types. An object is an instance of an object type that represents entities of the corresponding entity type. Objects can be enriched with attribute values to capture more details of the real-world entities. In a hospital information system, for instance, entities such as patients and the results of laboratory tests could be represented as objects, while relevant aspects such as name or age can be modeled as attributes.

In modeling approaches, object types can be defined in explicit data models [22,27,36,40], as model templates [79], more implicitly through place types in Petri nets [47,50,51,53,60], or as colored annotations [3,76]. In addition, object-centric event logs define object types as a metamodel for recorded objects [84,87].

Associations between object types model the relationship types between real-world entities. Instances of associations are links between objects that represent concrete relationships. For the hospital example, the relationship between patients and their specific laboratory test results may need to be tracked through an association.

Associations can be explicitly represented in data models [22,27,36,71,79]. Some Petri net-based approaches implement synchronization mechanisms by representing associations as dedicated places and links as tokens [51,53]. Similarly, proclets define synchronization points to establish links between associated object types [47].

Object-centric process models capture the behavior of objects as discrete states. Such object states allow abstracting from continuous behavior, reducing complexity, and facilitating formal analysis and computability. The modeling approaches define different notions of object states for the object types [102]. A hospital process can differentiate between the state of a patient who remains ill and one who is now cured.

Most multi-model approaches described in Section 2.2 explicitly represent states and state transitions as object lifecycles [22,27,36], while Petri nets encode object states through the distribution of tokens in the net [34,50,51,54,60]. In declarative approaches, object states are typically modeled more implicitly, but can be inferred through the history of their executed events and links to other objects [71,76]. Meanwhile, in OC-DCR graphs [77,79], object states are explicitly modeled as data-aware markings of activities represented as object instance graphs.

An event type supports the representation of behavior in the information system. It models the processing of an entity as state transitions of its related object types. On the instance level, events can hence capture the changes related to entities in an information system as instances of event types and in correlation with the corresponding objects. Event logs natively capture event types as a metamodel for recorded events [84,87]. In the hospital example, a process model can capture examinations as an event type involving a patient and a physician.

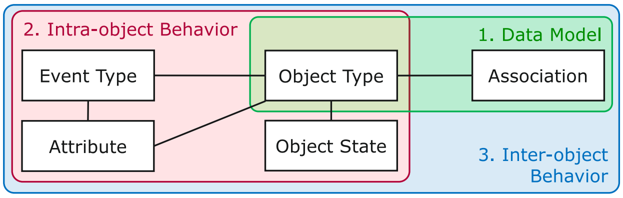

Fig. 4. The three object-centric process modeling aspects: data model, intra-object behavior, and inter-object behavior.

Traditionally, process modeling aims to capture behavior through high-level and non-atomic activities [1], which are also used in object-centric process modeling. However, approaches such as MERODE [27] and fCM [36] define an explicit mapping of high-level activities to many corresponding low-level event types. Other approaches remain implicit and assume a one-to-one correspondence between activity and event type.

4. Object-centric process characteristics

Based on the conceptualization of object-centric process management in Section 3, Fig. 4 illustrates the most prominent characteristics of object-centric processes: (i) a data model that defines object types and their associations, (ii) intra-object behavior defining object states and state transitions correlated with event types, and (iii) inter-object behavior correlating multiple objects and their intra-object behavior with each other. Together, these aspects advance the traditional case notion into a novel object-centric case notion.

4.1. Data model

A data model, like a UML class diagram, models object types and their associations, defining the domain in an object-centric process. For example, a hospital information system captures entities, such as patients and physicians, along with relevant attributes, and how they are associated through patient treatment processes.

As detailed in Section 2 some modeling approaches represent the data model explicitly, for instance as UML class diagrams [27,36] or ER diagrams [79]. In other approaches, the data model is embedded in the model as annotations [3,50,62,76].

4.1.1. Many-to-many associations

A common cardinality of a relation between entity types is many-to-many. For example, a patient is treated by many physicians, while one physician can treat many patients. Some formal modeling approaches are limited to many-to-one relationships [47,60]. They assume a behavioral normal form [47]. However, any many-to-many relationship can be reified into several many-to-one relationships with an additional association object type [51]. For example, modeling a treatment as a dedicated object can reify the patient-physician relationship.

4.1.2. Rigid & flexible links

The different approaches pose different assumptions about the stability of links between objects. So far, OCEL [84], and OCED [87] assume that links between objects are timeless, indicating that they are rigid and never change. This assumption applies to many discovered process model types, such as OCPN [52].

Other types of Petri nets explicitly allow for changing links. These flexible links can be created, used to correlate objects, and deleted during execution [47,60]. Modeling approaches [22,27,36,79] naturally capture flexible links to support realistic process executions.

In the hospital example, an association can define rigid links between a patient and all related physicians or one flexible and changing link with one physician at a time.

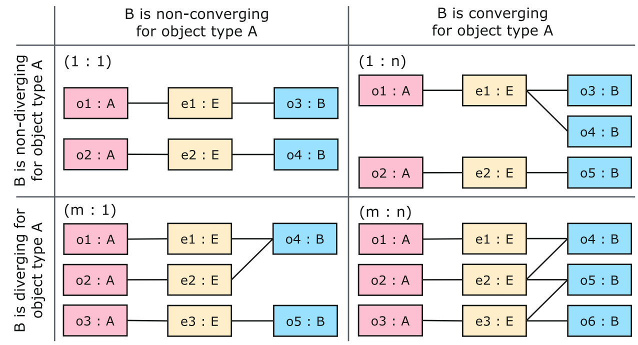

Fig. 5. The relationships between events of event type E and objects of the object types A and B exhibiting divergence and convergence between the two object types.

4.2. Intra-object behavior

The existing approaches use different ways to define the behavior of objects *in isolation* before specifying their joint behavior. This behavior of an object type is defined by a set of object states and transitions between them. The object state can rely on the attributes of the object or its prior behavior, and states can be changed by events [102]. Hence, the intra-object behavior defines how event types can change the state of an object.

While the described multi-model approaches model the intra-object behavior explicitly as object lifecycles, other approaches remain implicit. Typically, the concept of projection [3] or flattening [50] captures all event types and states of a single object type. For example, in OCPN [50], all places of one object type plus the related transitions manifest its intra-object behavior.

4.2.1. Multi-object processing & convergence

A very common feature in object-centric processes is multi-object processing [51], where one event processes multiple objects of one type. This represents observations in an information system where multiple entities of the same type participate in the same interaction. For instance, consider a surgery where many physicians are involved. In process modeling, this situation can be explicitly modeled by variable arcs in Petri nets [51], or as activities operating on data object collections in a BPMN process diagram [36,103]. In case-centric process models, where one activity instance may refer to a set of cases, this can be modeled as batch activities [104].

In process mining, multi-object processing may lead to so-called convergence between two potential case objects [3,15,54]. As illustrated in Fig. 5 in the top right (1:n), the event $e1$ is correlated to exactly one object $o1$ of type A, but processes multiple objects, $o3$ and $o4$, of type B. If A is chosen as the case notion, then all events are considered once for every case. However, if the object type B is considered as the case notion, $e1$ would be considered for two process instances, that is, for $o3$ and $o4$, even though it happened only once. As objects of type A are related to many objects of type B through events of type E, we can say that B is converging for A. This illustrates that only the consideration of intra-object behavior is not enough when considering multiple object types.

4.2.2. Recurring activities & divergence

In process modeling, a common behavioral pattern is recurring activities, typically modeled as loop structures, multi-instance subprocesses in BPMN [24], or declarative constraints [71]. Also, resources are often used to model entities that are involved in the same activity occurring in many process instances [105].

In process mining, recurring activities result in so-called divergence between two potential case notions [15]. In the bottom left of Fig. 5

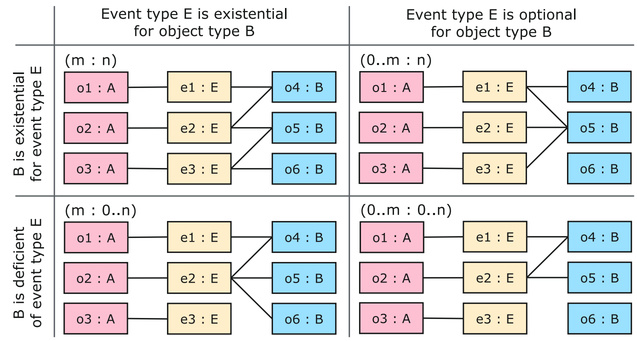

Fig. 6. The many-to-many relationships between object type A and B through the event type E, with and without optionality and deficiency between B and E.

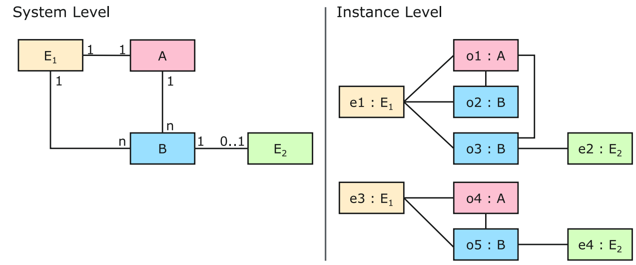

Fig. 7. An example of inter-object behavior of event types E_1 and E_2 and object types A and B illustrated on the type (left) and instance levels (right).

(m:1), $o4$ of type B appears twice, in the events $e1$ and $e2$. Meanwhile, every object of type A refers to exactly one event E. Therefore, the event type E appears as a recurring event type for B. We say that B is diverging for A through E. For example, a physician (B) may perform multiple surgeries (A) during one day. In traditional process discovery, recurring activities can lead to convoluted spaghetti models [3]. Divergent behavior can be an indicator for resource-like objects [106].

It is important to note that two object types can be converging and diverging at the same time, as illustrated in the lower right corner of Fig. 5 (m:n). For example, many physicians (B) may be involved with many patients (A) through overlaps in surgery events (E).

4.2.3. *Optionality & deficiency*

Orthogonal to convergence and divergence, an event type E and an object type B can show deficiency [54] and optionality [90], illustrated in Fig. 6.

An optional event type E for a given object type B allows an object of type B to exist without a correlation to an event of type E (cf. $o6$ in the upper right corner in Fig. 6). Process models can capture this as decisions or optional events. With A as the case notion, information that is captured in the objects of B will not be considered for some cases. In a setting where all physicians (A) perform surgeries (E), there might be some patients (B) who do not receive any surgery and are therefore not captured at all.

If an event type E is deficient of the object type B, an event of type E can exist without any object of type B (cf. $e3$ lower left corner in Fig. 6). For example, when admitting a patient, the patient (A) is existential, while a physician (B) may be deficient. When choosing B as the case notion, some events of type E might not be recorded.

Event types may be optional while object types may simultaneously be deficient. Optionality and deficiency also emphasize the need to represent the inter-object behavior for a comprehensive picture of the information system's behavior.

4.3. *Inter-object behavior*

The inter-object behavior defines how multiple associated objects co-evolve. Extending the intra-object behavior, one event can occur for a set of objects of different types. This behavioral perspective allows us to correlate many objects through events. Although multi-model approaches typically use one dedicated coordination model [22] or a high-level BPMN diagram [27,36], other approaches represent the intra- and inter-object behavior as a single graph [50,51].

For example, Fig. 7 illustrates a setting where two event types E_1 and E_2 and two object types A and B are related in a one-to-many relationship. The complete object-centric process only emerges when considering the combined inter-object behavior. Consider events of E_1 are surgeries being performed. E_2 represents documentations after surgeries. Objects of the type A are patients, and of the type B are physicians. Here, the data model combines the object types A and B such that one patient is associated with many physicians. Referring to the characteristics described in Section 4.2, the many physicians (B) are converging for the patient (A), as one surgery event (e.g., $e1$ of type E_1) is performed for exactly one patient, but many physicians are involved. In this example, we do not observe divergence. However, patients (A) are deficient in documentation events (e.g., $e2$ and $e4$ of type E_2), while they are optional for physicians (e.g., $o2$ of type B). Only if the two object types are considered jointly can we observe the true nature in which objects and events are related.

4.3.1. *Synchronization*

In order to define object-centric process executions where objects are only linked in accordance with the data model, the inter-object behavior must consider the multiplicity of their associations. Synchronization refers to mechanisms in the inter-object behavior that enforce the occurrence of specific events only for objects that are linked [47,51,53]. For example, a patient should only be associated with their own test results.

Two notions of synchronization exist [47]: Exact synchronization enforces that an object can only be processed together with all its linked objects; subset synchronization allows an object to be processed with only a non-empty subset of its related objects. As synchronization is inferred from the data model, different types of synchronization patterns can be discovered [52,53,55].

4.4. *Object-centric case notions*

In traditional business process management, process models aim to accurately define the start and end of a process, relevant activities, and their ordering. Hence, a process typically has a case notion, which defines the perspective of the process model and implies how events or activities are correlated. One of the first tasks when extracting a traditional event log is choosing a case identifier that allows for correlating events. This case identifier defines the case notion and can be interpreted as one object type. In process modeling, objects typically refer to business artifacts and are considered from a modeled case notion, which acts as the anchor of modeling process perspectives. For example, a hospital process can be considered from the perspective of a patient or a physician, resulting in very different models.

The inter-object behavior of an object-centric process model combines multiple object types and, hence, bears more than only one case object. Object-centric cases may, therefore, lack a precise start or end, challenging typical assumptions in case-centric models. Still, in object-centric process instances, cases can be identified as groups of events and objects that form one connected component [61]. Therefore, the notion of an object-centric case depends on the scope of object types and event types that are considered. For the example considering Fig. 7, there could be two cases. One comprises the events $e1$ and $e2$ and the objects $o1$, $o2$, and $o3$. The second case includes $e3$, $e4$, $o4$, and $o5$.

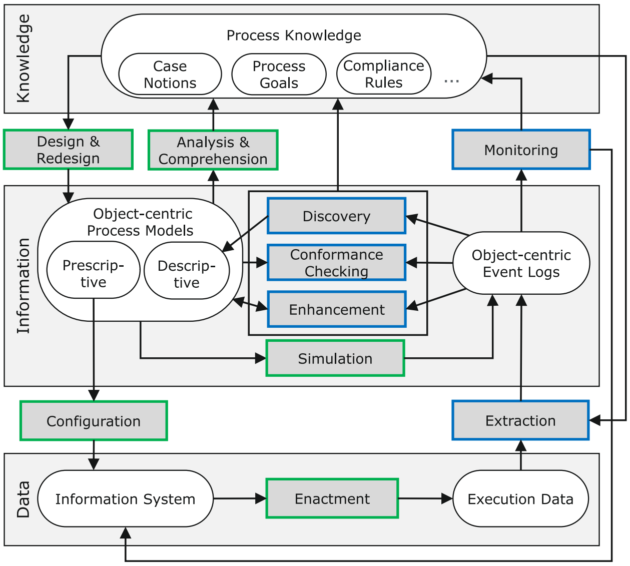

Fig. 8. In object-centric process management, tasks are correlated with input and output from the three levels. Typical model-driven engineering tasks are colored green, process mining tasks blue.

On the instance level, a case is defined as objects that are related plus events occurring in the context of these objects. However, the concrete object-centric cases cannot be modeled by the data model or intra-object behavior alone. The explicit definition of an object-centric case notion needs to consider the combined inter-object behavior. Given an object-centric process model and a data model that define an object-centric case notion, we can flexibly identify different traditional case notions. As traditional case notions consider only one object in isolation, it can be interpreted as a simple filter on a complex object-centric case notion. This allows us to relate any object-centric process model back to traditional case notions as well.

5. Research challenges and opportunities

The discussed approaches to object-centric process management differ in the way they map and abstract features from information systems. In BPM, it is important to choose an appropriate process representation for a particular task. To guide these decisions, we characterize object-centric process management by aligning the most relevant research areas to the conceptualization introduced above. For each area, the status quo in research is outlined, and the most relevant research challenges and opportunities are identified.

Different BPM lifecycles have been proposed for traditional business process management, relating typical BPM tasks. From a model-driven engineering perspective [1,2], processes are designed and analyzed, configured in the information system, and enacted. Running processes in the information system can be controlled, monitored, and evaluated, providing new input for process redesign. From a process mining perspective [9,107], process models represent real-world information systems that record event logs, which can be used to discover descriptive process models, check conformance with the process model, and enhance existing process models with additional information. Process models can be simulated to generate event logs, allowing for what-if analyses. Running process instances can be monitored by visualizing them or predicting their outcome.

As a unification of the existing BPM lifecycles for case-centric processes, we characterize object-centric process management as shown in Fig. 8, which relates typical process management tasks to their input and output artifacts, i.e., process knowledge, process models, event logs, information systems, and data.

The architecture encompasses and extends traditional case-centric processes. Model-driven engineering uses an object-centric case notion to design process models based on process knowledge. These models can be analyzed and redesigned until they meet the desired purpose. Such models prescribe the behavior in the information system. To do so, these models are enriched to the required technical level of detail, so that they serve as configurations of the information system. The enactment of process implementations produces process instances, which generate process execution data.

Process mining aims to gain knowledge from execution data. Event logs must be extracted at the beginning of every process mining effort. In this step, it is crucial to define a suitable object-centric case notion that identifies required object types and relevant event types. From event logs, descriptive process models can be discovered, which define the as-is processes in the original information system. Conformance checking compares a process model with an event log to identify deviating executions between the two. Existing process models can be enhanced with additional information observed in an event log to provide more descriptive models. Process models can be simulated under different assumptions to generate synthetic event logs. Analysts can then perform what-if analyses by comparing those logs to assess the impact of alternative process configurations. Finally, monitoring aims to extract knowledge about running process instances. This knowledge can be retrieved from structured event logs.

5.1. Challenges in conceptualization and methodology

In a process management setting where multiple tasks from Fig. 8 are employed, different models are used to obtain process knowledge. Some models focus on a single aspect of the system, such as behavior or data. Other models focus on multiple aspects, such as the integration of processes and data. As these models represent the same business process, ensuring consistency between them is essential. On the one hand, the combination of different models can help to overcome blind spots. On the other hand, contradictory properties of different models should be avoided. The use of different models for different aspects of the system brings both conceptual and methodological challenges.

Concept-1 (Modeling Purposes & Capabilities) Picking useful modeling approaches for managing object-centric processes is crucial. Therefore, it is important to identify specific object-centric modeling purposes in real-world settings. In a second step, we have to study which conceptual definitions of objects and modeling paradigms are needed to fit these purposes. The features of the modeling approaches need to be analyzed to identify their capabilities and limitations in applying them for different purposes. For example, the desired comprehensibility of process models requires a visual representation, while automated analysis requires a formal definition and computational decidability.

Concept-2 (Case Notions) Object-centric process models can facilitate case notions in an integrated fashion. As different approaches model case notions differently, it is important to investigate which object-centric case notions match the purposes for a given use case or data set.

Concept-3 (Common Terminology) Modeling paradigms use different conceptualizations and differing metamodels. Often, they incorporate implicit assumptions on real-world information systems. It is crucial to make these assumptions explicit and study their interrelation to harmonize them. This, for example, may require the development of a common terminology [91,108].

Following the questions tackling conceptualization, the following methodological challenges must also be addressed:

Method-1 (Representation) Methodologies are needed to guide the modeling of object-centric processes from different perspectives. In software architectures, views and perspectives can be modeled explicitly [109], which needs to be adapted for object-centric process management.

Method-2 (Purpose) The capabilities of the different object-centric process representations need to be reflected when applying them for the purpose of specific BPM tasks. For different tasks in the lifecycle (cf. Fig. 8), different process models and combinations of models can show more or less useful. Methodologies need to be developed and evaluated that guide the selection of appropriate models for identifying and satisfying the purposes for a given use case.

Method-3 (Adoption) To adopt the object-centric paradigm in practice, it is necessary to investigate the business value of object-centric process management to introduce it to stakeholders. The roles of stakeholders in business process management should be revised in the new object-centric paradigm.

Method-4 (Case Notion) Whereas traditional case-centric process models implicitly employed a single case representation, in the object-centric paradigm, this case notion needs to be modeled explicitly, potentially as a collection of objects. This allows introducing object-centric concepts into existing process management initiatives. Process managers can benefit from using new perspectives on their information system's behavior. However, this must be supported by new best practices and guidelines facilitating the adoption of the object-centric paradigm.

Method-5 (Interoperability) Since object-centric processes can cross the boundaries of organizational units, it has to be studied how to ensure that objects that are represented in different systems can be unequivocally identified and matched. This calls for robust methodologies, tools, and techniques to support such interoperability between different object-centric systems, as well as between object-centric and case-centric systems.

Method-6 (Ethical Considerations) As objects can explicitly represent human resources, object-centric modeling and analysis of respective systems can reveal privacy-sensitive information about these subjects. Consequently, object-centric process analyses raise all kinds of ethical questions that need to be addressed, both in methodology and analysis techniques.

5.2. Challenges in process design & redesign

When multiple models describe the same system, they must be designed consistently. This holds true for multi-model approaches and for the combination of different modeling approaches. Consistency is studied for system architectures [110,111] and process behavior [112]. Modeling tools, which support consistency by design, exist for the multi-model approaches fragment-based Case Management [113], Information System Modeling [114], MERODE [115], and PHILharmonicFlows [116].

Object-centric Petri nets can be modeled as colored Petri nets [117], for which there is tool support [118–120]. Recognizing that resources are special types of objects, Lo Bianco et al. [121] provide an extension to Petri nets that is a first step towards including decisions on the use of resources by other objects.

Design-1 (Granularity and Abstraction) Moving from high-level process models and architecture models to low-level implementations is the typical task of model-driven engineering, where a broad spectrum of concepts is captured in the UML modeling standard [33]. These concepts can serve as a rich toolbox to support object-centric process modeling as well.

Design-2 (Data Modeling) The expressiveness of data models and the integration of data models with process models need further investigation. For example, the data modeling concepts of inheritance, aggregation, and composition induce behavioral constraints on process models, which have only been considered in few approaches [21].

Design-3 (Resource Modeling) Active resources are special types of objects behaving differently from passive objects. This is a widely accepted observation in resource-centric [122–125] and simulation-focused [121,126] approaches. In particular, resources exhibit distinct behavioral patterns across cases and objects as they perform batch activities [127] and prioritize cases or objects [128], affecting process performance and outcomes [129]. Yet, in the object-centric paradigm, active resources are often simply modeled as object types [53] without accommodating their special role.

Design-4 (Inter-object Relationships) The inter-object behavior needs to be properly defined considering data modeling aspects, such as different types of associations and cardinalities between data classes. Existing synchronization mechanisms are so far limited to a uniform level of granularity and many-to-one relationships. More perspectives should be investigated, including dynamic relationships, hierarchies of object types, and, analogously to queuing theory, objects that may be prioritized over other related objects.

Design-5 (Multi-model Approaches) Modeling object-centric processes is challenging because objects might be accessed and manipulated by several traditional business processes [34,130]. In addition to traditional approaches like imperative and declarative process modeling, object-centric multi-model approaches need to be investigated. This includes the use of views and filters on specific object types and event types to focus on relevant aspects.

Design-6 (Instance-spanning Constraints) The interrelation of multiple traditional case-centric processes is studied as instance-spanning constraints [131–134]. Multiple process instances are constrained by implicit dependencies that are made explicit through formal constraints. In light of object-centric processes, these constraints can now be explicitly represented as many interrelated objects.

Design-7 (Distributed Processes) In distributed settings, parties may only have partial, delayed, or conditional views of objects and their states. These states can only be synchronized through message exchange or object transfer, which necessitates explicit modeling constructs for visibility and information exchange. Although initial attempts capture subsets of these aspects [79,135–138], a refined expressive language for distributed object-centric choreographies remains an open design challenge.

Design-8 (Design Methodologies) Design methodologies, such as object-centric workflow patterns, reference models, and standards, are needed to guide modelers through the complexity of object-centric processes. In addition, redesign methodologies are needed to meaningfully incorporate the insights from prior analysis into an existing model.

Design-9 (Generative AI) The potentials of generative AI need to be explored in all parts of process design, ranging from process model creation using natural language processing to label analysis and process improvement.

5.3. Challenges in process model analysis

By process model analysis, we refer to the formal and mathematical verification of such models. To formally analyze an object-centric business process, it must be defined in a mathematically precise way. Petri nets are among the most widely used formalisms for this purpose [139]. They support the analysis of numerous semantic and syntactic properties, the study of different behavioral equivalences, and the verification of temporal-logic properties [140].

Although Petri net-based formalisms have been used in business process management to cover the control-flow dimension of the process [1], dealing with object-centricity requires incorporating the data dimension in a suitable way. This is inherently challenging from a formal analysis point of view, as evidenced by a long-standing series of results within the foundations of data management [67,141] and artificial intelligence [142,143].

In the following, we list the challenges which are relevant for analyzing object-centric processes.

Analyze-1 (Synchronization) In object-centric processes, objects are synchronized based on their correlations with other objects. Although various notions of synchronization behavior in object-centric processes have recently been explored (e.g., [44,47,51,53,56]), the problem of correlating objects throughout a process requires further investigation. This is particularly relevant in scenarios where objects are connected through complex relationships (e.g., inheritance, aggregation, or composition known from data modeling) and where these links may change during process execution.

Analyze-2 (Correctness Notions) Object-centric processes require new, appropriate correctness notions as well as formal languages for specifying them. One of the most well-known correctness notions in the context of business processes (or, more specifically, workflow nets) is soundness [144]. A natural question is whether standard notions of soundness, or data-aware variants of them [145,146], can be successfully applied to object-centric processes. Even for object-centric Petri nets proposed in [50], which exhibit behavioral similarities to standard Petri nets [147], it was discussed in [148] that traditional soundness notions have limited applicability in the object-centric setting. Following these observations, preliminary works (e.g., [44,148]) have introduced more suitable soundness notions for different classes of object-centric process formalisms. Nevertheless, this research area requires significantly more attention, both in terms of developing new correctness notions and in devising procedures for their automated verification.

Analyze-3 (Temporal Properties) In complex object-centric settings, studying temporal properties is essential. Due to the presence of objects and relationships in object-centric processes, such properties have to simultaneously account for the dimensions of time and data values. This calls for first-order variants of temporal properties and the corresponding verification techniques. So far, relevant works have focused on fundamental properties such as reachability and safety [143,149], or first-order extensions of temporal logics [150–152]. These approaches adopt an interleaving semantics for concurrency and do not directly lend themselves to dealing with object-centric processes with true concurrency. First-order extensions of CTL or CTL* [153] can help to better represent suitable temporal properties in this context, but further research is needed towards suitable temporal logics for object-centric processes and their efficient verification.

Analyze-4 (Compliance Checking) Compliance checking analyzes process models for their conformity to compliance rules that are extracted from policies and legal regulations, which are not process-centric per se. While case-centric process models aim to map rules to activities, the object-centric paradigm makes it possible to relate compliance rules explicitly with objects, allowing for a more accurate representation. Although OC-DECLARE [76], OC-DCR [77,79] or OCBC [71] provide declarative formalisms, the increasing computational complexity of object-centric process models must be addressed.

Analyze-5 (Probabilistic Processes) The analysis of probabilistic process executions for traditional case-centric process models is investigated in probabilistic modeling and queueing theory. In probabilistic object-centric processes, multiple object perspectives need to be considered.

5.4. Challenges in model comprehension

By process model comprehension, we refer to the task of assessing a process model by human experts in order to validate it with respect to reality. Most prominently, graphical models are used, but also formal or textual representations, to understand a process. For object-centric processes, the aggregation of several perspectives into one model or the combination of multiple models for different perspectives results in more complex process models, which are more difficult to comprehend for human users [130]. Therefore, model comprehension comprises a set of challenges. For some declarative case-centric notations, such as GSM [154], a decreased perceived ease of use can be observed. Yet, such notations can be more useful at conveying circumstantial information [155], which may render them more suitable for the object-centric paradigm. The comprehensibility of different graphical object-centric process notations is investigated by Holz et al. [156]. Schreiber et al. investigate the cognitive effects of fragmented process models [157].

Berti et al. [158] provide tools for analyzing process models discovered from OCEL logs. Layouting algorithms are proposed to support the visualization of object-centric Petri nets [159]. Initial research shows that balancing abstraction and mapping can result in the discovery of simpler and more understandable object-centric models [64,65]. In [160], process analysts are supported by defining process scopes to visualize high-level inter-process interactions.

Comprehension-1 (Model Comprehension) To improve the comprehensibility of object-centric process models for human users, more insight is needed into the cognitive effects of object-centric process modeling paradigms. Modeling languages need to be adapted or developed for better comprehensibility.

Comprehension-2 (Model Visualization) Users may pose individual requirements towards the visualization of an object-centric process model and may be interested in selecting specific object-centric case notions [106,160]. Flexible derivations of personalized views on object-centric process models are needed to guide users.

Comprehension-3 (Model Transformations) To bridge the comprehensibility and formal analysis of models, model transformations can serve as intermediaries between low-level formalisms and comprehensible graphical notations. As they may map features differently and abstract from only some characteristics, equivalent mappings are difficult to achieve.

Comprehension-4 (Layouting) The comprehensibility of object-centric process models can be supported by suitable layouts and automatic computation via layouting algorithms, which need to be studied in more detail.

5.5. Challenges in process configuration

The configuration phase refers to the implementation of the behavior defined by an object-centric process model. This is mainly achieved in two ways. A process-oriented information system can take a detailed and configured process model as input [1], and a process engine is then responsible for executing the process as defined in the process model. Alternatively, the process model can serve as a blueprint to manually implement the process in an existing system.

In model-driven engineering, multiple object-centric approaches, such as the Guard-Stage-Milestone approach [20], OC-DCR [161,162], MERODE [27], and PHILharmonicFlows [22,163], enrich process models to a level of detail where they can be enacted by a process engine immediately.

Config-1 (Standards) Novel object-centric process modeling standards are required which combine concepts from high-level process modeling notations and detailed data modeling capabilities. For instance, established standards such as BPMN 2.0 [24] to model business processes and UML [33] to model data structures and object behavior may be combined, similar to DCR graphs combined with ER diagrams [79].

Config-2 (System Architectures) System architectures for process-oriented information systems need to be designed that support the coexistence of multiple intertwined process models jointly realizing process executions. In contrast, current architectures designed for a case-centric perspective primarily execute process instances independently of each other.

Config-3 (Resource Configuration) The modeling and configuration of resources in an object-centric process raises the question of the extent to which resources can be modeled as objects. Resources may also access multiple other objects, which requires a configuration of access control and permission handling.

Config-4 (Data Granularity) The configuration of high-level process models results in very detailed process implementations. To cope with the additional complexity in the object-centric setting, online analytical processing (OLAP) operations have proven useful [164], such as roll-up and drill-down. More techniques and methodologies for handling these different levels of granularity are needed.

Config-5 (Programmed Implementations) To support a direct implementation of object-centric processes into executable programs, object-oriented programming [165] provides features that can be harnessed, while proposals such as behavioral programming [166] may specifically help in implementing the interactions of objects.

Config-6 (Version Management) Process configurations should be integrated into the running information systems. Integration, maintenance, and version updating of object-centric process configurations require updated methodologies.

5.6. Challenges in process enactment

Given a process implementation, which prescribes how the process should be executed, process enactment aims to provide run-time support to make sure that the process is actually executed as defined.

Enact-1 (Process Engines) Traditionally, process execution engines are software components that use process models to steer the execution of process instances. Since object-centric processes combine multiple object perspectives, they offer to solve conflicting data access between different traditional process instances. First object-centric process engines based on GSM [167,168] and on PetriFlow [58,59] have been proposed. Comparable process engines for other recently established object-centric modeling approaches are still missing.

Enact-2 (Interfaces) During process enactment, typically several databases or, more generally, data stores are accessed. Appropriate interfaces have to be designed to provide an object view on the data stores. These might use non-object representations, such as tuples in a relational database. Relational views might be employed to provide an object perspective on data that is stored in a relational database system.

Enact-3 (Transactions & Compensation) To achieve robustness in process execution, process models may define transactional contexts and compensation mechanisms. In traditional models, the scope of transactions and compensation handlers can be defined on the level of individual process instances. In object-centric models, however, more flexible definitions are required, as transactions may span across object types and a single object may be included in multiple different processes. Approaches to capture these scenarios and provide unambiguous semantics need to be developed for object-centric process engines, along with implementation strategies that link transactional concepts at the process level to those available in database management systems.

Enact-4 (Distributed Execution) Novel software architectures for object-centric process enactment must be designed and evaluated. Analogously to microservices [169], objects may relate to distributed or heterogeneous entities in the information system. Furthermore, agent-based processes [122] define the behavior of individual distributed agents, which can be transferred to define inter- and intra-object behavior. Traditional architectures in which a central process engine invokes services to provide functionality need to be extended to capture distributed objects.

Enact-5 (Data Formats) The enactment of object-centric business processes creates large datasets, which are valuable assets for process mining. Standardized data formats for execution data of object-centric business processes will be important to facilitate process evaluation in general and process mining in particular. Existing initiatives like OCEL and OCED need to be consolidated and extended to capture the required features.

Enact-6 (Logging) To ensure that the execution data holds the information that is required for later event log extraction, methods and guidelines should be developed to automatically log relevant events, entities, and attributes during process execution. In addition, information systems could directly generate object-centric event logs during enactment.

5.7. Challenges in event log extraction

The extraction or generation of event logs from process execution data is a key task in any process mining endeavor. It is obvious that the quality of the generated event log influences the quality of the process mining task. The extraction of object-centric event logs is studied for different sources: databases [49,82,170], virtual knowledge graphs [171], and blockchain applications [172,173].

To query process-centric data more effectively from a database, process querying languages are developed. Related work includes academic efforts [174], as well as industry works [175,176].

Traditional process querying and constraint checking assume a case-centric view of event data. Therefore, a querying and constraint framework and language that works with Object-Centric Event Data is required. OCPQ (Object-Centric Process Querying) [93] provides a querying and constraint language in which objects and events are considered first-class citizens. To support the navigation of related objects and events, languages like OCPQ are vital for extracting object-centric event data. Furthermore, graph query languages such as Cypher are effective in querying object-centric event data [95] across a large variety of industrial use cases [177].

Methodologies from traditional process mining have been extended to the extraction of object-centric event logs, emphasizing the role of data models to specify case notions [108]. To extract more suitable logs, the definition of advanced case notions [106] and scopes for aggregating and abstracting objects and events [160] are required.

Extract-1 (Data Sources) A wide variety of data sources for object-centric processes should be investigated, such as data streams, database tables, and redo logs. For instance, existing case-centric event log extraction methods based on sensor data and video material [178,179] should be extended to the object-centric setting.

Extract-2 (Data Quality) The different types of data sources need to be analyzed with respect to data quality issues. While existing methods to assess the quality of the data sources provide a good starting point [180], they have to be critically evaluated.

Extract-3 (Temporal Databases) With the object-centric paradigm, event logs can be regarded as declarative views over temporal databases that store the objects in question. This assumption allows explicating the implicit mapping between the evolution of data and the corresponding log entries but raises the question of how to define or derive such views.

Extract-4 (Heterogeneous & Federated Sources) Given the fact that object-centric business processes use a wide variety of objects stored in different information systems, mapping entities from heterogeneous or distributed data sources to objects is challenging. These distributed sources may contain divergent representations of entities while referring to the same object. While central data storage of event logs is studied [181], federated process mining [182] needs to be extended with modeling concepts and algorithms that harmonize such distributed data into a coherent object-centric event log. Frameworks are needed to determine indicators of which entities and which attributes are relevant for certain questions. A balance is required between the complexity of the model and its comprehensiveness.

Extract-5 (Abstraction & Privacy) Guidelines are needed to abstract objects depending on the required level of granularity and the requirements of the resulting event log. On the one hand, the resulting log needs to be rich enough to answer relevant questions. On the other hand, the representation of execution data in objects should comply with ethical and legal considerations. For example, objects representing human resources should be anonymized in accordance with data protection regulations.

Extract-6 (ETL Pipelines) Event log extraction induces various design choices, e.g., related to the identification of objects and events in a data collection, the abstraction of their states, and their correlation. These design choices that are commonly managed as part of an extract-transform-load (ETL) process impact any downstream analysis. As such, they should be made explicit and their implications on analysis results, e.g., in terms of accuracy, timeliness, and trustworthiness, shall be made transparent to achieve traceability. Alternatively, extract-load-transform (ELT) approaches as proposed by Esser et al. [95] and Swevels et al. [88,183] can be pursued, which do provide flexibility in realizing data transformations in line with analysis goals, while the use of graph data models allows trace transformations and their impact on analysis results. Yet these approaches currently lack scalability to industrial workloads.

Extract-7 (Object Relationships) Currently, the role of links between objects is not yet represented in object-centric event logs in detail. As relationships such as foreign keys in a database can change over time, volatile links between objects should also be investigated. To this end, techniques from data profiling may be adapted to support the identification of object links and to monitor their evolution. However, traditional data profiling techniques are largely agnostic to the phenomena that lead to changes in the data, whereas in object-centric processes, these changes are structured and systematic, which raises the challenge of integrating process knowledge into the respective mining techniques.

Extract-8 (Event Log Formats) Current standardization efforts for object-centric event log formats do not cover all possible data modeling concepts [87]. While various use cases require the extension of an event log format based on OCEL and OCED, a standardization effort has to balance the desired expressiveness (covering various modeling concepts and use cases) with the simplicity required for industrial-scale implementation and adoption by a large ecosystem of source systems and process mining solutions.

Extract-9 (Storage Formats) Given their complexity, object-centric event logs have become larger and computationally more expensive to process. Therefore, different storage formats for event logs, such as event knowledge graphs, can be considered. For practical applications, trade-offs between the required grade of detail in the log and the technical feasibility need to be studied.

5.8. Challenges in process discovery

As process discovery aims at generating process models that describe how the processes in an organization are actually executed, it is one of the most important tasks in process mining. So far, several process discovery approaches have been proposed: object-centric Petri nets [50,53], process trees [54], typed Jackson Nets [184], process variants [61], TOTeM data models [81], directly follows graphs [185], causal nets [62,64], object-centric DECLARE [76], and object-centric DCR [77]. Other approaches have been considered primarily from a modeling perspective and consider process discovery as well: GSM [186], PhilharmonicFlows [187], OCBC [188], and DMN [189].

Discover-1 (Purpose) Methodologies are needed to determine appropriate approaches that satisfy required model purposes for different real-world use cases. Depending on the task, different discovered model types might be more or less useful. For example, different process representations are needed for comprehension and for process verification.

Discover-2 (Object Relationships) The discovery of complex object relationships needs to be studied in more detail. While initial approaches discover object relationships [55,81,190], the discovery of more advanced data modeling concepts such as inheritance, aggregation, composition, and the difference between static and changing links during execution need to be studied.

Discover-3 (Intra-object Behavior) Currently, discovered intra-object behavior is restricted to the observation that events are happening in correlation with objects without qualifying the type of correlation. Discovering CRUD operations on objects and object relationships is crucial for discovering process models with advanced execution semantics. This requires an appropriate level of granularity in the event logs.

Discover-4 (Inter-object Behavior) Discovering appropriate inter-object behavior of multiple object types is challenging. Although first approaches aim to discover synchronizing behavior [52,55,76], more work is needed to discover complex synchronization patterns. Also, temporal dependencies [81] that can be discovered between object types can be used to discover more appropriate process models. Object-centric discovery must consider inter-case dynamics, the second-order effects that arise when many objects compete for shared resources. Queue mining research shows that system performance is shaped not only by logical dependencies but also by congestion phenomena such as workload coupling, queue buildup, blocking, and resource contention [191]. Incorporating such inter-case interactions is essential to uncovering the operational drivers of delay and instability across object lifecycles.

Discover-5 (Instance-spanning Constraints) As highlighted in Design-6, instance-spanning constraints capture dependencies between multiple instances of case-centric processes. In event logs, instance-spanning constraints manifest as observations tracing back to batch activities. Discovery of instance-spanning constraints and batch activities from case-centric event logs [192–194] can be explicitly tied to related objects in object-centric process discovery.

Discover-6 (Comprehensibility) The increased scope, detail, and structure of the data and behavior will make it more likely that the discovered process models are too complex to understand. At the same time, users may be interested in specific aspects of the behavior. This creates the challenge of discovering partial models in line with user interests or a collection of object-centric local process models.

Discover-7 (Computational Complexity) Although more expressive models shall be discovered, their rising complexity unavoidably will come with higher computational complexity. Efficient discovery algorithms with useful trade-offs for a given purpose are required.

Discover-8 (Controllability) While different discovery algorithms have been proposed, they offer limited control over parameters. Hence, these algorithms impede the discovery of models that fit individual purposes. The focus should be on algorithms that offer greater control to process scientists.

Discover-9 (Concept Drift) Concept drift characterizes a change in process instances over time [195]. The object-centric paradigm offers an analysis of the root cause in more detail, which might stem from changes in entities and their relationships. As object-centric logs can incorporate multiple case notions, they facilitate the analysis of process changes, enabling the analysis and comparison of process behavior for changed processes [130].

5.9. Challenges in conformance checking

Conformance checking studies the relationships between process execution data, represented by event logs, and process models. Several approaches have been proposed, including object-centric behavioral constraints [196,197] and object-centric directly follows graphs [185]. Specific metrics like fitness and precision aim to quantify the level of conformance between model and log. For object-centric Petri nets, these metrics have been defined by Adams et al. [198]. In conformance checking, a typical task is to identify the best alignment of behavioral patterns between the model and the log [199]. The alignment of object-centric Petri nets and object-centric event logs is defined by Liss et al. [200], while Gianola et al. extend it for object-centric Petri nets with identifiers to respect the links of objects to capture synchronization [51] and also to respect objects' attributes [60]. To study the relationships between these formalisms, Seidel et al. [52] investigate equivalences between them. Abstractions can be used to compare the conformance results [201]. Sommers et al. [202] show that alignments over shared objects require re-thinking how deviations between model and log are diagnosed when finding the most likely explanation for deviations.

Conformance-1 (Conformance Metrics) In general, conformance checking compares the similarity between process model and log. Either the model or the log needs to be trusted in representing the original information system accurately [203,204]. The object-centric paradigm can help to represent the information system more comprehensively, promising more accurate conformance checking once appropriate conformance metrics are defined.

Conformance-2 (Missing Objects) As events are correlated through objects, missing objects in the log or missing activities in the model can drastically influence the observed events and their alignment. While current work assumes consistent objects in the model and log, future studies need to investigate the role of missing objects to better align models and logs.

Conformance-3 (Object Relationships) Current object-centric conformance checking considers mostly event ordering per object. However, object-centric processes offer more complex dimensions such as attribute values, object states, and object synchronization. Capturing [131] and quantifying such additional dimensions can help in capturing conformance more comprehensively.

Conformance-4 (Granularity) Different levels of granularity between detailed event logs and more abstract process models need to be bridged. An execution state of a process model may be defined at a more abstract level than the recorded events and their attributes in an event log. A mapping between states in the model and the temporal position in the log, and between event occurrences in the log and state transitions in the model, is needed.

Conformance-5 (Computational Costs) The computation of conformance metrics is computationally demanding for object-centric cases [51], which can limit their application in real-world use cases [205]. Methodologies such as relaxations [206], for finding suitable trade-offs between desired complexity and computational feasibility, as well as performant algorithms are needed. In addition, more robust and efficient algorithms are needed for conformance checking in real-world logs, especially when dealing with high-volume data [205].

5.10. Challenges in process enhancement

Process enhancement aims to enrich existing process models with insights gathered from event logs [107,207]. Most importantly, the process models are enriched with the perspectives of performance, data, resources, and time [13]. Enhancement of process performance is studied for GSM [208], object-centric Petri nets [209], and object-centric DFGs [185]. Knowledge about the data model can be used to enhance process models. Temporal relations between object types can indicate distinct process areas [210], while rigid associations can imply synchronization [52]. Distance metrics are proposed to group similar objects for different analysis purposes [211], which can be used for enhancement.

Enhance-1 (Visualization) Analogous to model comprehension, the visualization of object-centric enhancements must express the required complexity comprehensibly. Novel designs and visualizations must be developed and empirically evaluated.

Enhance-2 (Backward Compatibility) Traditional process models can be enhanced with object-centric information on additional object perspectives. Correctly representing the object-centric enhancements requires an investigation of the relation between objects and the traditional case notion.

Enhance-3 (Temporal Information) To enhance process models with the dimension of time, process models need to capture temporal aspects in the form of activity duration or waiting times, while event logs typically capture atomic events. Event logs need to be rich enough to distinguish passive waiting from actively running activities, motivating comprehensive logging during enactment and extraction of these features.

5.11. Challenges in process simulation

In traditional process simulation, process models are enriched with execution contexts such as branching probabilities, resource constraints, and execution times for activities. Existing data-aware process simulation approaches [212] extend case-centric simulation where data attributes, which determine the process flow, can be modified throughout the simulation. Batching policies used in existing simulation [213,214] provide a basis to model both convergence and divergence between simulated cases. Recent works on process simulation with differentiated resources discover and integrate resource calendars, heterogeneous performance, and intermittent availability into simulations [126,215], where resources are entities with distinct behaviors and constraints. Multi-agent approaches such as AgentSimulator [122] allow modeling resources as agents accordingly.

As a first dedicated approach for object-centric process simulation, Knopp et al. [216] discover object-centric Petri nets with simulation configurations from object-centric event logs. Object-centric process simulation promises new and valuable insights [217] but still needs to be combined with the existing work on data-driven and agent-based simulation.

Simulate-1 (Generating OCELs) Simulation can produce meaningful benchmark OCELs for evaluating process mining approaches.

Real-world process execution may be confidential and proprietary. Simulation can help to create realistic but publicaly-available data sets. The challenge is to find methodologies that perform realistic simulations for a given purpose and with desired features.

Simulate-2 (Resources) While existing data-driven simulators support concurrent execution of many cases, with data-aware update rules and constraints to guide the execution [212] and allocation of resources with differentiated availability and performance [122,126,215], these approaches remain case-centric. Accessible and configurable methods are needed to extend these foundations to object-centric models, where activity execution may depend on the joint state and availability of multiple interrelated objects.

Simulate-3 (Simulation Configurations) The configurations of object-centric simulation models are inherently more complex than in traditional process simulation. Because the execution of activities depends not only on processing times, branching probabilities, and resource capacities, but also on the availability of objects, the configuration needs to define parameters that define the effect of objects on the simulation outcome. Such parameters include the waiting times and processing times of objects for different activities and resources. For relationships between objects, for each object, the number of related objects needs to be specified.

Simulate-4 (Scheduling) In object-centric simulation, orchestration must consider events whose enablement depends on multiple interrelated objects, which are also not bound to a single case lifecycle. This creates convergence and divergence that can span multiple concurrent cases, which reduces the simulation controllability and increases side effects that impact other pending events and objects.

Simulate-5 (Object Creation) In object-centric simulation, new objects are not created in a single stream as in case-centric models. Instead, different types of objects may be created at different times and need to be dynamically linked during the simulation. Mechanisms are needed to represent how objects are created asynchronously and how their relations control the execution.

Simulate-6 (What-if Analysis) The application of object-centric simulation for what-if analysis and process optimization has so far remained largely unexplored. In the object-centric domain, where interactions between multiple object types can lead to several undesired side effects, there is a huge potential to optimize the underlying processes.

Simulate-7 (Simulation Model Quality) Evaluating object-centric simulation models is a challenging and unexplored area. Existing metrics applied to case-centric settings mainly rely on distance measures and compare original and simulated event logs globally, focusing on control flow, time, congestion, and resource dimensions [218]. In the object-centric setting, measures need to be defined that quantify the data perspective for object-centric scenarios. New evaluation methods are required to capture the individual impact of objects on the different dimensions of the simulation outcomes and their combined effect.

5.12. Challenges in process monitoring

Process monitoring encompasses two tasks. First, operational monitoring analyzes running process instances and evaluates their performance. Second, predictive process monitoring goes beyond analyzing historical instances to forecast the future progression of ongoing instances.

Park and van der Aalst [219] provide an approach for operational monitoring of object-centric event logs. Monitoring offers the ability to analyze business processes according to their performance [185,209] or specific constraints [220]. Model-driven engineering approaches, such as PHILharmonicFlows [221], provide monitoring facilities for their configured and implemented process models at runtime. BPMN in combination with GSM can be used for the decentralized monitoring of objects, as shown in [222].

Existing approaches aim to provide efficient analysis of event logs in order to support process monitoring. Kumar et al. [223] compare different database schemas for querying monitoring results from object-centric event logs. Jalali [224] groups similar behavioral patterns to improve event log access.

Monitor-1 (Visualization) To show process managers meaningful monitoring results, the visualization of object-centric process insights needs to find a balance between simple visualizations that are easy to comprehend and visualizations that expose the complex interrelations of different objects [160]. Novel approaches for visualizing monitoring results and dashboards are needed, and empirical research should investigate their practical applicability.

Monitor-2 (Querying) To retrieve monitoring results from running process executions as manifested in databases, the relation between process models and efficient querying with querying languages [174] such as PQL or SIGNAL should be studied. Benchmarks are needed for comparable evaluations of queries.

Monitor-3 (Run-time Analysis) Operational monitoring involves the application of conformance and compliance checking techniques at runtime. This requires the analysis of object-centric constraints over incomplete process executions. Existing techniques, respectively, aim at monitoring cross-instance constraints [225] and run-time monitoring [226–228]. The combination of both perspectives still needs to be studied.

The predictive process monitoring subfield uses the information from event logs as input to machine learning approaches to predict the outcome, remaining time, or next events of running business process instances [229]. In contrast to case-centric event logs, object-centric event logs also store information on objects and their links. This can be used by novel feature extraction techniques [230]. The first investigations indicate that object-centric predictive process monitoring can provide better predictions considering object relationships [231–234].

Monitor-4 (Reproducibility) Research on object-centric prediction models needs frameworks, public data sets, and benchmarks to support the reproducibility and comparability of approaches.

Monitor-5 (Training Objectives) Machine learning models need to be trained on suitable data for relevant features. Although object-centric event logs may contain more detailed information than traditional logs, it is not obvious which features contribute to specific prediction tasks. Hence, it should be investigated to what extent and at what abstraction level attribute values, links, and inter-object behavior are relevant for training.

Monitor-6 (Object-centric Outcomes) Without clear boundaries of cases, outcome prediction first needs to determine what a suitable outcome comprises, based on involved objects. Analogously, time predictions need to consider that the process executions are not sequences, but complex graphs, involving different objects and their links. Methods are needed to assist in selecting suitable prediction outcomes. For example, existing works [232,234] propose the concept of viewpoints. Viewpoints are special object types that are especially relevant for the process performance. Yet, the choice of viewpoints depends on the domain and the specific outcome and, hence, not always straightforward.

Monitor-7 (Prescriptive Monitoring) Prescriptive process monitoring employs predictions that can be enacted in the running information system right away [229]. In the object-centric setting, this may lead to side effects on other objects. Model-driven predictions that explicitly track such side effects can help uncover unintended interactions [235]. Another opportunity is shifting from local prediction models – which consider each execution in isolation – to a global prediction approach that simultaneously accounts for all ongoing executions, thereby revealing hidden interferences and improving predictive accuracy [236].

Monitor-8 (Explainability) Predictions made using machine learning models need to be explainable to become a trustworthy basis for decision-making. Such prediction models are often probabilistic and intransparent. Although predictive models contain rich levels of information, conveying useful explanations to the users of predictive technology is already challenging for case-centric processes. Here, counterfactual explanations can be used, where counterfactual cases exemplify minimal changes to the case, which modify the predicted outcome [237,238]. The absence of case notions raises the question of what constitutes an appropriate counterfactual for a given prediction.

6. Conclusion

Object-centric process management uses object-centric process models and event logs as intermediaries between information systems and knowledge to describe, analyze, and monitor the system's behavior. This research manifesto provides an overview of the status quo in the research of object-centric processes. Based on an overview of the literature, a conceptualization is provided that encompasses the work in process modeling and process mining by defining a common terminology for object-centric process models and event logs. We explain the most relevant behavioral characteristics that emerge in object-centric processes. Finally, a unified domain model of object-centric process management relates the most prominent research areas in BPM to the conceptualization. We observe that a variety of different process representations are suitable for different purposes in process management. There is not only one process modeling approach to satisfy all purposes, but object-centric process management needs to identify appropriate models for different tasks. Therefore, we elaborate on the most pressing research challenges in academia and practice to guide future research on object-centric process management.

As the purpose of this paper is to arrive at a common understanding of object-centric processes and their context, the proposed conceptualization only provides a general overview of the relevant concepts, while a detailed discussion on standardization and research on object-centric methodologies remains open. However, future research along the lines of identified research challenges will contribute to clarifying these aspects. This manifesto may serve as a common ground for the novel and developing field of research on object-centric process management. Future research will help to adopt the object-centric paradigm in practice for a more precise and comprehensive management of processes in information systems.

CRediT authorship contribution statement

Anjo Seidel: Conceptualization, Writing – original draft. **Mathias Weske:** Writing – original draft, Conceptualization. **Marco Montali:** Conceptualization, Writing – original draft. **Andrey Rivkin:** Conceptualization, Writing – original draft. **Manfred Reichert:** Conceptualization, Writing – review & editing. **Jan Martijn E.M. van der Werf:** Conceptualization, Writing – original draft. **Wil M.P. van der Aalst:** Conceptualization, Writing – review & editing. **Marius Breitmayer:** Conceptualization, Writing – review & editing. **Lukas Liss:** Writing – review & editing. **Jan Niklas van Detten:** Writing – review & editing. **Amin Jalali:** Writing – review & editing. **Shahrzad Khayatbashi:** Writing – review & editing. **Maximilian König:** Writing – review & editing. **Tom Lichtenstein:** Writing – review & editing. **Stefanie Rinderle-Ma:** Writing – review & editing. **Barbara Weber:** Writing – review & editing. **Pnina Soffer:** Writing – review & editing. **Lorenzo Rossi:** Writing – review & editing. **Daniel Calegari Garcia:** Writing – review & editing. **Andrea Delgado:** Writing – review & editing. **Remco Dijkman:** Writing – review & editing. **Sarah Winkler:** Writing – review & editing. **Matthias Weidlich:** Writing – review & editing. **Sander J.J. Leemans:** Writing – review & editing. **Dirk Fahland:** Writing – review & editing. **Ava Swevels:** Writing – review & editing. **Monique Snoeck:** Writing – review & editing. **Giancarlo Guizzardi:** Writing – review & editing. **Alessandro Gianola:** Writing – review & editing. **Avigdor Gal:** Writing – review & editing. **Ekkart Kindler:** Writing – review & editing. **Irina Lomazova:** Writing – review & editing. **Barbara Re:** Writing – review & editing. **Giovanni Meroni:** Writing – review & editing. **Andrea Morichetta:** Writing – review & editing. **Alessandro Marcelletti:** Writing – review & editing. **Sara Pettinari:** Writing – review & editing. **Boudewijn van Dongen:** Writing – review & editing. **Johannes De Smedt:** Writing – review & editing. **Majid Rafiei:** Writing – review & editing. **Julius Köpke:** Writing – review & editing. **Thomas Hildebrandt:** Writing – review & editing. **Francesca Zerbato:** Writing – review & editing. **Luise Pufahl:** Writing – review & editing. **Hajo Reijers:** Writing – review & editing. **Artem Polyvyanyy:** Writing – review & editing. **Chiara Di Francescomarino:** Writing – review & editing. **Fabrizio Maria Maggi:** Writing – review & editing. **Oscar Pastor:** Writing – review & editing. **Stephan Haarmann:** Writing – review & editing. **Henderik Proper:** Writing – review & editing. **Xixi Lu:** Writing – review & editing. **Hugo López:** Writing – review & editing. **Tijs Slaats:** Writing – review & editing. **Jochen De Weerdt:** Writing – review & editing. **Massimiliano de Leoni:** Writing – review & editing. **Niels Martin:** Writing – review & editing. **Karolin Winter:** Writing – review & editing. **Nick van Beest:** Writing – review & editing. **Orlenys López-Pintado:** Writing – review & editing. **Sebastiaan van Zelst:** Writing – review & editing. **Chiara Ghidini:** Writing – review & editing. **Arik Senderovich:** Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

References

[1] M. Weske, Business Process Management: Concepts, Languages, Architectures, Springer, Berlin, Heidelberg, 2024, http://dx.doi.org/10.1007/978-3-662-69518-0.

[2] M. Dumas, M. La Rosa, J. Mendling, H.A. Reijers, Fundamentals of Business Process Management, Springer, Berlin, Heidelberg, 2018, http://dx.doi.org/10.1007/978-3-662-56509-4.

[3] W.M.P. van der Aalst, Object-centric process mining: dealing with divergence and convergence in event data, in: Software Engineering and Formal Methods, Springer International Publishing, Cham, 2019, pp. 3–25, http://dx.doi.org/10.1007/978-3-030-30446-1_1.

[4] R.L. Ackoff, From data to wisdom, J. Appl. Syst. Anal. 16.1 (1989) 3–9.

[5] S. Alter, Information Systems, Addison-Wesley Longman Publishing Co. Inc., 1998.

[6] G.M. Marakas, J.A. O'Brien, Introduction to Information Systems, McGraw-Hill/Irwin New York, 2013.

[7] R.K. Rainer, B. Prince, C. Sanchez-Rodriguez, I. Splettstoesser-Hogeterp, S. Ebrahimi, Introduction to Information Systems, John Wiley & Sons, 2020.

[8] J. vom Brocke, W.M. van der Aalst, N. Berente, B.F. van Dongen, T. Grisold, W. Kremser, J. Mendling, B.T. Pentland, M. Roeglinger, M. Rosemann, et al., Process science: the interdisciplinary study of socio-technical change, Process. Sci. 1 (1) (2024) 1.

[9] W.M.P. van der Aalst, et al., Process mining manifesto, in: Business Process Management Workshops, Springer, Berlin, Heidelberg, 2012, pp. 169–194, http://dx.doi.org/10.1007/978-3-642-28108-2_19.

[10] G.J. Kuperman, R.M. Gardner, T.A. Pryor, HELP: A Dynamic Hospital Information System, Springer Science & Business Media, 2013.

[11] L. Floridi, Information: A Very Short Introduction, vol. 225, Oxford University Press, 2010.

[12] H. Stachowiak, Allgemeine Modelltheorie, Springer-Verlag/Wien, 1973.
[13] W.M.P. van der Aalst, Process Mining - Data Science in Action, second ed., Springer, 2016, http://dx.doi.org/10.1007/978-3-662-49851-4.
[14] W.M.P. van der Aalst, Object-centric process mining: An introduction, in: A. Cerone (Ed.), Formal Methods for an Informal World - ICTAC 2021 Summer School, Tutorial Lectures, in: Lecture Notes in Computer Science, vol. 13490, Springer, 2021, pp. 73–105, http://dx.doi.org/10.1007/978-3-031-43678-9_3.
[15] W.M.P. van der Aalst, Object-centric process mining: Unraveling the fabric of real processes, Mathematics 11 (12) (2023) 2691.
[16] D. Hollingsworth, The Workflow Reference Model, Technical Report TC00-1003, The Workflow Management Coalition (WfMC), 1995.
[17] A.-W. Scheer, ARIS — Business Process Modeling, Springer Science & Business Media, 2000.
[18] B. Axenath, E. Kindler, V. Rubin, AMFIBIA: A meta-model for the integration of business process modelling aspects, Int. J. Bus. Process. Integr. Manag. 2 (2) (2007) 120–131.
[19] F. Leymann, D. Roller, Production Workflow: Concepts and Techniques, Upper Saddle River, New Jersey, USA, Prentice-Hall PTR, 1999.
[20] R. Hull, E. Damaggio, F. Fournier, M. Gupta, F.T. Heath, S. Hobson, M. Linehan, S. Maradugu, A. Nigam, P. Sukaviriya, R. Vaculin, Introducing the guard-stage-milestone approach for specifying business entity lifecycles, in: Web Services and Formal Methods, Springer, Berlin, Heidelberg, 2011, pp. 1–24, http://dx.doi.org/10.1007/978-3-642-19589-1_1.
[21] M. Snoeck, Enterprise information systems engineering: the MERODE approach, in: The Enterprise Engineering Series, Springer International Publishing, Cham, 2014, http://dx.doi.org/10.1007/978-3-319-10145-3.
[22] V. Künzle, M. Reichert, PHILharmonicFlows: Towards a framework for object-aware process management, J. Softw. Maint. Evol.: Res. Pr. 23 (4) (2011) 205–244, http://dx.doi.org/10.1002/smr.524.
[23] H. Stachowiak, Der modellbegriff in der erkenntnistheorie, Z. Allg. Wiss. 11 (1) (1980) 53–68, http://dx.doi.org/10.1007/BF01801279.
[24] OMG, Business process model and notation (BPMN), version 2.0.2, formal/13-12-09, ISO/IEC 19510:2013, 2014.
[25] A. Nigam, N.S. Caswell, Business artifacts: An approach to operational specification, IBM Syst. J. 42 (3) (2003) 428–445, http://dx.doi.org/10.1147/SJ.423.0428.
[26] S. Steinau, A. Marrella, K. Andrews, F. Leotta, M. Mecella, M. Reichert, DALEC: A framework for the systematic evaluation of data-centric approaches to process management software, Softw. Syst. Model. 18 (4) (2019) 2679–2716, http://dx.doi.org/10.1007/s10270-018-0695-0.
[27] M. Snoeck, C. Verbruggen, J. De Smedt, J. De Weerdt, Supporting data-aware processes with MERODE, Softw. Syst. Model. 22 (6) (2023) 1779–1802, http://dx.doi.org/10.1007/s10270-023-01095-4.
[28] G. Redding, M. Dumas, A.H.M. ter Hofstede, A. Iordachescu, A flexible, object-centric approach for business process modelling, Serv. Oriented Comput. Appl. 4 (3) (2010) 191–201, http://dx.doi.org/10.1007/S11761-010-0065-4.
[29] A. Berti, M. Montali, W.M.P. van der Aalst, Advancements and challenges in object-centric process mining: a systematic literature review, 2023, arXiv:2311.08795.
[30] A. Alman, F.M. Maggi, S. Rinderle-Ma, A. Rivkin, K. Winter, Towards a multi-model paradigm for business process management, in: Advanced Information Systems Engineering - 36th International Conference, CAiSE 2024, in: Lecture Notes in Computer Science, vol. 14663, Springer, 2024, pp. 178–194, http://dx.doi.org/10.1007/978-3-031-61057-8_11.
[31] D. Calvanese, M. Montali, M. Estañol, E. Teniente, Verifiable UML artifact-centric business process models, in: Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management, in: CIKM '14, Association for Computing Machinery, New York, NY, USA, 2014, pp. 1289–1298, http://dx.doi.org/10.1145/2661829.2662050.
[32] M. Estañol, A. Queralt, M. Sancho, E. Teniente, Artifact-centric business process models in UML, in: Business Process Management Workshops - BPM 2012 International Workshops, in: Lecture Notes in Business Information Processing, vol. 132, Springer, 2012, pp. 292–303, http://dx.doi.org/10.1007/978-3-642-36285-9_34.
[33] OMG, Unified modeling language (UML) 2.5.1, formal/17-12-05, 2017.
[34] A. Polyvyanyy, J.M.E.M. van der Werf, S. Overbeek, R. Brouwers, Information systems modeling: Language, verification, and tool support, in: P. Giorgini, B. Weber (Eds.), Advanced Information Systems Engineering - 31st International Conference, CAiSE 2019, Rome, Italy, June 3-7, 2019, Proceedings, in: Lecture Notes in Computer Science, vol. 11483, Springer, 2019, pp. 194–212, http://dx.doi.org/10.1007/978-3-030-21290-2_13.
[35] K.M. van Hee, J. Keiren, R. Post, N. Sidorova, J.M.E.M. van der Werf, Designing case handling systems, Trans. Petri Nets Other Model. Concurr. 1 (2008) 119–133, http://dx.doi.org/10.1007/978-3-540-89287-8_8.
[36] M. Hewelt, M. Weske, A hybrid approach for flexible case modeling and execution, in: Business Process Management Forum, Springer International Publishing, Cham, 2016, pp. 38–54, http://dx.doi.org/10.1007/978-3-319-45468-9_3.
[37] A. Seidel, M. König, M. Weske, Towards object-centric BPMN process models, in: Proceedings of the Best BPM Dissertation Award, Doctoral Consortium, and Demonstrations & Resources Forum Co-Located with 22nd International Conference on Business Process Management, BPM 2024, 2024.
[38] S. Haarmann, M. Montali, M. Weske, Refining case models using cardinality constraints, in: Advanced Information Systems Engineering - 33rd International Conference, CAiSE 2021, in: Lecture Notes in Computer Science, vol. 12751, Springer, 2021, pp. 296–310, http://dx.doi.org/10.1007/978-3-030-79382-1_18.
[39] S. Ghilardi, A. Gianola, M. Montali, A. Rivkin, Delta-BPMN: A concrete language and verifier for data-aware BPMN, in: Business Process Management - 19th International Conference, BPM 2021, in: Lecture Notes in Computer Science, vol. 12875, Springer, 2021, pp. 179–196, http://dx.doi.org/10.1007/978-3-030-85469-0_13.
[40] A. Delgado, D. Calegari, Towards a unified vision of business process and organizational data, in: XLVI Latin American Computing Conference, CLEI 2020, IEEE, 2020, pp. 108–117, http://dx.doi.org/10.1109/CLEI52000.2020.00020.
[41] K. Jensen, Coloured Petri Nets: Basic Concepts, Analysis Methods and Practical Use, vol. 1, Springer, 1996, URL https://link.springer.com/book/10.1007/978-3-662-03241-1.
[42] W. Reisig, Petri Nets: An Introduction, Springer, 1985.
[43] F. Rosa-Velardo, D. de Frutos-Escrig, Decidability and complexity of Petri nets with unordered data, Theoret. Comput. Sci. 412 (34) (2011) 4439–4451, http://dx.doi.org/10.1016/J.TCS.2011.05.007.
[44] J.M.E.M. van der Werf, A. Rivkin, M. Montali, A. Polyvyanyy, Correctness notions for Petri nets with identifiers, Fund. Inform. 190 (2–4) (2024) 159–207, http://dx.doi.org/10.3233/FI-242169, arXiv:2212.07363.
[45] D. Moldt, R. Valk, Object oriented Petri nets in business process modeling, in: Business Process Management, Models, Techniques, and Empirical Studies, in: Lecture Notes in Computer Science, vol. 1806, Springer, 2000, pp. 254–273, http://dx.doi.org/10.1007/3-540-45594-9_16.
[46] G. Booch, Object-oriented analysis and design with applications, second ed., in: Benjamin/cummings Series in Object-Oriented Software Engineering, Addison-Wesley, 1995.
[47] D. Fahland, Describing behavior of processes with many-to-many interactions, in: Application and Theory of Petri Nets and Concurrency, Springer International Publishing, Cham, 2019, pp. 3–24, http://dx.doi.org/10.1007/978-3-030-21571-2_1.
[48] D. Fahland, Process mining over multiple behavioral dimensions with event knowledge graphs, in: W.M.P. van der Aalst, J. Carmona (Eds.), Process Mining Handbook, in: Lecture Notes in Business Information Processing, vol. 448, Springer, 2022, pp. 274–319, http://dx.doi.org/10.1007/978-3-031-08848-3_9.
[49] X. Lu, M. Nagelkerke, D. van de Wiel, D. Fahland, Discovering interacting artifacts from ERP systems, IEEE Trans. Serv. Comput. 8 (6) (2015) 861–873, http://dx.doi.org/10.1109/TSC.2015.2474358.
[50] W.M.P. van der Aalst, A. Berti, Discovering object-centric Petri nets, Fund. Inform. 175 (1–4) (2020) 1–40, http://dx.doi.org/10.3233/FI-2020-1946.
[51] A. Gianola, M. Montali, S. Winkler, Object-centric conformance alignments with synchronization, in: Advanced Information Systems Engineering - 36th International Conference, CAiSE 2024, in: Lecture Notes in Computer Science, vol. 14663, Springer, 2024, pp. 3–19, http://dx.doi.org/10.1007/978-3-031-61057-8_1.
[52] A. Seidel, S. Winkler, A. Gianola, M. Montali, M. Weske, To bind or not to bind? Discovering stable relationships in object-centric processes, in: Conceptual Modeling - 44th International Conference, ER 2025, Proceedings, in: Lecture Notes in Computer Science, vol. 16189, Springer, 2025, pp. 223–241, http://dx.doi.org/10.1007/978-3-032-08623-5_12.
[53] J.N. van Detten, P. Schumacher, S.J.J. Leemans, Object synchronizations and specializations with silent objects in object-centric Petri nets, in: Business Process Management, Springer Nature Switzerland, Cham, 2024, pp. 57–74, http://dx.doi.org/10.1007/978-3-031-70396-6_4.
[54] J.N. van Detten, P. Schumacher, S.J.J. Leemans, Discovering compact, live and identifier-sound object-centric process models, in: 2024 6th International Conference on Process Mining, ICPM, 2024, pp. 113–120, http://dx.doi.org/10.1109/ICPM63005.2024.10680659.
[55] J.N. van Detten, P. Schumacher, S.J. Leemans, Modeling and discovering dynamic identity relations in object-centric process mining, in: 2025 7th International Conference on Process Mining, ICPM, 2025.
[56] S. Ghilardi, A. Gianola, M. Montali, A. Rivkin, Petri net-based object-centric processes with read-only data, Inf. Syst. 107 (2022) 102011, http://dx.doi.org/10.1016/j.is.2022.102011.
[57] M. Montali, A. Rivkin, DB-nets: On the marriage of colored Petri nets and relational databases, Trans. Petri Nets Other Model. Concurr. 12 (2017) 91–118, http://dx.doi.org/10.1007/978-3-662-55862-1_5.
[58] G. Juhás, T. Kovácik, J. Kovár, M. Kranec, L. Petrovic, Petriflow language and netgrif application builder, in: Proceedings of the Best Dissertation Award, Doctoral Consortium, and Demonstration & Resources Track at BPM 2021 Co-Located with 19th International Conference on Business Process Management, BPM 2021, in: CEUR Workshop Proceedings, vol. 2973, CEUR-WS.org, 2021, pp. 171–175.

[59] G. Juhás, M. Mladoniczky, L. Petrovic, Practical experience with Petriflow: Enriched process models serving as implementation, in: Modellierung 2024 - Workshop Proceedings, Potsdam, Germany, Gesellschaft für Informatik e.V., 2024, p. 24, http://dx.doi.org/10.18420/MODELLIERUNG2024-WS-024.

[60] A. Gianola, M. Montali, S. Winkler, Object-centric processes with structured data and exact synchronization - formal modelling and conformance checking, in: Advanced Information Systems Engineering - 37th International Conference, CAiSE 2025, in: Lecture Notes in Computer Science, vol. 15702, Springer, 2025, pp. 185–202, http://dx.doi.org/10.1007/978-3-031-94571-7_11.

[61] J.N. Adams, D. Schuster, S. Schmitz, G. Schuh, W.M.P. van der Aalst, Defining cases and variants for object-centric event data, in: 2022 4th International Conference on Process Mining, ICPM, 2022, pp. 128–135, http://dx.doi.org/10.1109/ICPM57379.2022.9980730.

[62] L. Liss, C. Mensing, W.M.P. van der Aalst, Object-centric causal nets, in: Advanced Information Systems Engineering - 37th International Conference, CAiSE 2025, in: Lecture Notes in Computer Science, vol. 15702, Springer, 2025, pp. 94–110, http://dx.doi.org/10.1007/978-3-031-94571-7_6.

[63] W.M.P. van der Aalst, A. Adriansyah, B.F. van Dongen, Causal nets: A modeling language tailored towards process discovery, in: CONCUR 2011, in: Lecture Notes in Computer Science, vol. 6901, Springer, 2011, pp. 28–42, http://dx.doi.org/10.1007/978-3-642-23217-6_3.

[64] E. de Moura Figueiredo, A. Jalali, Discovering object-centric causal nets with edge-coarse-graining in process mining, in: Perspectives in Business Informatics Research: 24th International Conference, BIR 2025, Springer Nature, 2025, p. 201.

[65] E.d. Figueiredo, A. Jalali, Object-centric process mining for public sector transformation, in: BIR 2025 Workshops, 24th International Conference on Perspectives in Business Informatics Research, in: CEUR-WS, vol. 4034, CEUR Workshop Proceedings, Riga, Latvia, 2025, URL https://ceur-ws.org/Vol-4034/paper87.pdf.

[66] R. Hull, E. Damaggio, R.D. Masellis, F. Fournier, M. Gupta, F.F.T.H. III, S. Hobson, M.H. Linehan, S. Maradugu, A. Nigam, P.N. Sukaviriya, R. Vaculín, Business artifacts with guard-stage-milestone lifecycles: managing artifact interactions with conditions and events, in: Proceedings of the Fifth ACM International Conference on Distributed Event-Based Systems, DEBS 2011, ACM, 2011, pp. 51–62, http://dx.doi.org/10.1145/2002259.2002270.

[67] A. Deutsch, R. Hull, Y. Li, V. Vianu, Automatic verification of database-centric systems, ACM SIGLOG News 5 (2) (2018) 37–56, http://dx.doi.org/10.1145/3212019.3212025.

[68] A. Deutsch, Y. Li, V. Vianu, Verification of hierarchical artifact systems, ACM Trans. Database Syst. 44 (3) (2019) 12:1–12:68, http://dx.doi.org/10.1145/3321487.

[69] V. Popova, M. Dumas, Discovering unbounded synchronization conditions in artifact-centric process models, in: Business Process Management Workshops - BPM 2013 International Workshops, in: Lecture Notes in Business Information Processing, vol. 171, Springer, 2013, pp. 28–40, http://dx.doi.org/10.1007/978-3-319-06257-0_3.

[70] V. Popova, D. Fahland, M. Dumas, Artifact lifecycle discovery, Int. J. Coop. Inf. Syst. 24 (1) (2015) 1550001:1–1550001:44, http://dx.doi.org/10.1142/S021884301550001X.

[71] W.M.P. van der Aalst, G. Li, M. Montali, Object-centric behavioral constraints, 2017, http://dx.doi.org/10.48550/arXiv.1703.05740, arXiv:1703.05740.

[72] A. Artale, A. Kovtunova, M. Montali, W.M.P. van der Aalst, Modeling and reasoning over declarative data-aware processes with object-centric behavioral constraints, in: Business Process Management - 17th International Conference, BPM 2019, in: Lecture Notes in Computer Science, vol. 11675, Springer, 2019, pp. 139–156, http://dx.doi.org/10.1007/978-3-030-26619-6_11.

[73] M. Pesic, H. Schonenberg, W.M.P. van der Aalst, DECLARE: full support for loosely-structured processes, in: 11th IEEE International Enterprise Distributed Object Computing Conference, EDOC 2007, IEEE Computer Society, 2007, pp. 287–300, http://dx.doi.org/10.1109/EDOC.2007.14.

[74] W.M.P. van der Aalst, M. Pesic, H. Schonenberg, Declarative workflows: Balancing between flexibility and support, Comput. Sci. Res. Dev. 23 (2) (2009) 99–113, http://dx.doi.org/10.1007/S00450-009-0057-9.

[75] C. Di Ciccio, M. Montali, Declarative process specifications: Reasoning, discovery, monitoring, in: Process Mining Handbook, in: Lecture Notes in Business Information Processing, vol. 448, Springer, 2022, pp. 108–152, http://dx.doi.org/10.1007/978-3-031-08848-3_4.

[76] A. Küsters, W.M.P. van der Aalst, OC-DECLARE: discovering object-centric declarative patterns with synchronization, in: Business Process Management - 23rd International Conference, BPM 2025, in: Lecture Notes in Computer Science, vol. 16044, Springer, 2025, pp. 162–179, http://dx.doi.org/10.1007/978-3-032-02867-9_11.

[77] A.K. Christfort, A. Rivkin, D. Fahland, T.T. Hildebrandt, T. Slaats, Discovery of object-centric declarative models, in: 2024 6th International Conference on Process Mining, ICPM, 2024, pp. 121–128, http://dx.doi.org/10.1109/ICPM63005.2024.10680680.

[78] T.T. Hildebrandt, R.R. Mukkamala, Declarative event-based workflow as distributed dynamic condition response graphs, in: Proceedings Third Workshop on Programming Language Approaches To Concurrency and Communication-CEntric Software, PLACES 2010, in: EPTCS, vol. 69, 2010, pp. 59–73, http://dx.doi.org/10.4204/EPTCS.69.5.

[79] T. Zuckmantel, H.A. López, Y. Zhou, B. Düdder, T. Hildebrandt, Data integrity-by-design: Combining declarative object-centric choreographies and entity relationship models, in: Proceedings of the 31st International Conference on Cooperative Information Systems, CoopIS 2025, Springer, 2025.

[80] T. Zuckmantel, Y. Zhou, B. Düdder, T.T. Hildebrandt, DACEO: declarative asynchronous choreographies with general data-dependent event-ordering and objects, in: Coordination Models and Languages - 27th IFIP WG 6.1 International Conference, COORDINATION 2025, Held As Part of the 20th International Federated Conference on Distributed Computing Techniques, DisCoTec 2025, in: Lecture Notes in Computer Science, vol. 15731, Springer, 2025, pp. 197–216, http://dx.doi.org/10.1007/978-3-031-95589-1_10.

[81] L. Liss, J.N. Adams, W.M.P. van der Aalst, Totem: temporal object type model for object-centric process mining, in: Business Process Management Forum, Springer Nature Switzerland, Cham, 2024, pp. 107–123, http://dx.doi.org/10.1007/978-3-031-70418-5_7.

[82] G. Li, E.G.L. de Murillas, R.M. de Carvalho, W.M.P. van der Aalst, Extracting object-centric event logs to support process mining on databases, in: Information Systems in the Big Data Era, Springer International Publishing, Cham, 2018, pp. 182–199, http://dx.doi.org/10.1007/978-3-319-92901-9_16.

[83] A.F. Ghahfarokhi, G. Park, A. Berti, W.M.P. van der Aalst, OCEL: A standard for object-centric event logs, in: New Trends in Database and Information Systems, Springer International Publishing, Cham, 2021, pp. 169–175, http://dx.doi.org/10.1007/978-3-030-85082-1_16.

[84] A. Berti, I. Koren, J.N. Adams, G. Park, B. Knopp, N. Graves, M. Rafiei, L. Liß, L.T.G. Unterberg, Y. Zhang, C. Schwanen, M. Pegoraro, W.M.P. van der Aalst, OCEL (object-centric event log) 2.0 specification, 2024, http://dx.doi.org/10.48550/arXiv.2403.01975, arXiv:2403.01975.

[85] A. Goossens, J. De Smedt, J. Vanthienen, W.M.P. van der Aalst, Enhancing data-awareness of object-centric event logs, in: Process Mining Workshops, Springer Nature Switzerland, Cham, 2023, pp. 18–30, http://dx.doi.org/10.1007/978-3-031-27815-0_2.

[86] A. Rebmann, J.-R. Rehse, H. van der Aa, Uncovering object-centric data in classical event logs for the automated transformation from XES to OCEL, in: Business Process Management, Springer International Publishing, Cham, 2022, pp. 379–396, http://dx.doi.org/10.1007/978-3-031-16103-2_25.

[87] D. Fahland, M. Montali, J. Lebherz, W.M.P. van der Aalst, M. van Asseldonk, P. Blank, L. Bosmans, M. Brenscheidt, C. Di Ciccio, A. Delgado, D. Calegari, J. Peeperkorn, E. Verbeek, L. Vugs, M.T. Wynn, Towards a simple and extensible standard for object-centric event data (OCED) - core model, design space, and lessons learned, 2024, http://dx.doi.org/10.48550/ARXIV.2410.14495, CoRR abs/2410.14495, arXiv:2410.14495.

[88] A. Swevels, D. Fahland, M. Montali, Implementing object-centric event data models in event knowledge graphs, in: Process Mining Workshops - ICPM 2023 International Workshops, in: Lecture Notes in Business Information Processing, vol. 503, Springer, 2023, pp. 431–443, http://dx.doi.org/10.1007/978-3-031-56107-8_33.

[89] D. Calegari, A. Delgado, A model-driven engineering perspective for the object-centric event data (OCED) metamodel, in: Business Process Management Workshops - BPM 2023 International Workshops, in: Lecture Notes in Business Information Processing, vol. 492, Springer, 2023, pp. 508–520, http://dx.doi.org/10.1007/978-3-031-50974-2_38.

[90] A. Goossens, J. De Smedt, J. Vanthienen, Object-centric event logs: characteristics, comparative analysis and road map, in: Business Process Management Forum, Springer Nature Switzerland, Cham, 2024, pp. 37–54, http://dx.doi.org/10.1007/978-3-031-70418-5_3.

[91] C. Verbruggen, A. Goossens, J. De Smedt, J. Vanthienen, M. Snoeck, iDOCEM: Defining a common terminology for object-centric event logging and data-centric process modelling, Softw. Syst. Model. (2024) http://dx.doi.org/10.1007/s10270-024-01191-z.

[92] Y. Bertrand, C. Imenkamp, L. Malburg, M. Ehrendorfer, M. Franceschetti, J. Grüger, F. Leotta, J. Mangler, R. Seiger, A. Koschmider, S. Rinderle-Ma, B. Weber, E. Serral, An object-centric core metamodel for IoT-enhanced event logs, 2025, http://dx.doi.org/10.48550/ARXIV.2506.21300, CoRR abs/2506.21300, arXiv:2506.21300.

[93] A. Küsters, W.M.P. van der Aalst, OCPQ: object-centric process querying and constraints, in: Research Challenges in Information Science - 19th International Conference, RCIS 2025, in: Lecture Notes in Business Information Processing, vol. 547, Springer, 2025, pp. 383–400.

[94] C.W. Gunther, H. Verbeek, XES-standard definition, 2014.

[95] S. Esser, D. Fahland, Multi-dimensional event data in graph databases, J. Data Semant. 10 (1) (2021) 109–141, http://dx.doi.org/10.1007/s13740-021-00122-1.

[96] S. Khayatbashi, O. Hartig, A. Jalali, Transforming object-centric event logs to temporal event knowledge graphs, in: Business Process Management Workshops - BPM 2024 International Workshops, in: Lecture Notes in Business Information

Processing, vol. 534, Springer, 2024, pp. 300–313, http://dx.doi.org/10.1007/978-3-031-78666-2_23.

[97] S. Khayatbashi, O. Hartig, A. Jalali, Transforming event knowledge graph to object-centric event logs: A comparative study for multi-dimensional process analysis, in: Conceptual Modeling, Springer Nature Switzerland, Cham, 2023, pp. 220–238, http://dx.doi.org/10.1007/978-3-031-47262-6_12.

[98] E.L. Klijn, F. Mannhardt, D. Fahland, Aggregating event knowledge graphs for task analysis, in: Process Mining Workshops - ICPM 2022 International Workshops, in: Lecture Notes in Business Information Processing, vol. 468, Springer, 2022, pp. 493–505, http://dx.doi.org/10.1007/978-3-031-27815-0_36.

[99] E.L. Klijn, D. Preuss, L. Imeri, F. Baumann, F. Mannhardt, D. Fahland, Event knowledge graphs for auditing: A case study, in: Process Mining Workshops - ICPM 2023 International Workshops, in: Lecture Notes in Business Information Processing, vol. 503, Springer, 2023, pp. 84–97, http://dx.doi.org/10.1007/978-3-031-56107-8_7.

[100] P.P.-S. Chen, The entity-relationship model—toward a unified view of data, ACM Trans. Database Syst. 1 (1) (1976) 9–36, http://dx.doi.org/10.1145/320434.320440.

[101] D. Bano, T. Lichtenstein, F. Klessascheck, M. Weske, Database-less extraction of event logs from redo logs, in: 24th International Conference on Business Information Systems, BIS 2021, 2021, pp. 73–82, http://dx.doi.org/10.52825/BIS.V1I.66.

[102] M. König, R. Gießler, W. Brandt, A. Seidel, M. Weske, A unified view on data object states, in: Advanced Information Systems Engineering - 37th International Conference, CAiSE 2025, in: Lecture Notes in Computer Science, vol. 15702, Springer, 2025, pp. 259–276, http://dx.doi.org/10.1007/978-3-031-94571-7_15.

[103] M. König, T. Lichtenstein, A. Seidel, M. Weske, Introducing variables to data objects in BPMN, in: Enterprise Design, Operations, and Computing. EDOC 2024 Workshops - IRESEARCH, MIDas4CS, Doctoral Consortium, Joint CBI-EDOC Forum and Other Joint CBI-EDOC Events, in: Lecture Notes in Business Information Processing, vol. 537, Springer, 2024, pp. 137–154, http://dx.doi.org/10.1007/978-3-031-79059-1_9.

[104] L. Pufahl, M. Weske, Batch activities in process modeling and execution, in: Service-Oriented Computing - 11th International Conference, ICSOC 2013, in: Lecture Notes in Computer Science, vol. 8274, Springer, 2013, pp. 283–297, http://dx.doi.org/10.1007/978-3-642-45005-1_20.

[105] S. Ihde, L. Pufahl, M. Lin, A. Goel, M. Weske, Optimized resource allocations in business process models, in: Business Process Management Forum - BPM Forum 2019, in: Lecture Notes in Business Information Processing, vol. 360, Springer, 2019, pp. 55–71, http://dx.doi.org/10.1007/978-3-030-26643-1_4.

[106] J.N. van Detten, P. Schumacher, S.J.J. Leemans, A framework for advanced case notions in object-centric process mining, in: Process Mining Workshops - ICPM 2024 International Workshops, in: Lecture Notes in Business Information Processing, vol. 533, Springer, 2024, pp. 402–414, http://dx.doi.org/10.1007/978-3-031-82225-4_30.

[107] W.M.P. van der Aalst, J. Carmona (Eds.), Process Mining Handbook, Springer Nature, 2022, http://dx.doi.org/10.1007/978-3-031-08848-3.

[108] N. Miri, S. Khayatbashi, J. Zdravkovic, A. Jalali, OCPM^2: Extending the process mining methodology for object-centric event data extraction, in: Enterprise, Business-Process and Information Systems Modeling - 26th International Conference, BPMDS 2025, and 30th International Conference, EMMSAD 2025, in: Lecture Notes in Business Information Processing, vol. 558, Springer, 2025, pp. 123–140, http://dx.doi.org/10.1007/978-3-031-95397-2_8.

[109] N. Rozanski, E. Woods, Software Systems Architecture: Working With Stakeholders Using Viewpoints and Perspectives, Addison-Wesley, 2012.

[110] R.M. Dijkman, Consistency in Multi-Viewpoint Architectural Design (Ph.D. thesis), University of Twente, Enschede, Netherlands, 2006, URL http://eprints.eemcs.utwente.nl/2729/.

[111] A.A. da Costa Junior, S. Misra, M.S. Soares, A systematic mapping study on software architectures description based on ISO/IEC/IEEE 42010: 2011, in: Computational Science and Its Applications - ICCSA 2019 - 19th International Conference, in: Lecture Notes in Computer Science, vol. 11623, Springer, 2019, pp. 17–30, http://dx.doi.org/10.1007/978-3-030-24308-1_2.

[112] M. Weidlich, R.M. Dijkman, M. Weske, Behaviour equivalence and compatibility of business process models with complex correspondences, Comput. J. 55 (11) (2012) 1398–1418, http://dx.doi.org/10.1093/COMJNL/BXS014.

[113] K. Andree, L. Bein, M. König, C. Mandel, M. Rosenau, C. Terboven, D. Bano, S. Haarmann, M. Weske, Design-time support for fragment-based case management, in: Business Process Management Workshops, Springer International Publishing, Cham, 2023, pp. 231–242, http://dx.doi.org/10.1007/978-3-031-25383-6_17.

[114] J.M.E.M. van der Werf, A. Polyvyanyy, The information systems modeling suite - modeling the interplay between information and processes, in: Application and Theory of Petri Nets and Concurrency - 41st International Conference, PETRI NETS 2020, Paris, France, June 24-25, 2020, Proceedings, in: Lecture Notes in Computer Science, vol. 12152, Springer, 2020, pp. 414–425, http://dx.doi.org/10.1007/978-3-030-51831-8_22.

[115] M. Snoeck, MERLIN: An intelligent tool for creating domain models, in: Research Challenges in Information Science - 14th International Conference, RCIS 2020, in: Lecture Notes in Business Information Processing, vol. 385, Springer, 2020, pp. 549–555, http://dx.doi.org/10.1007/978-3-030-50316-1_37.

[116] L. Arnold, M. Breitmayer, M. Reichert, A web-based modelling tool for object-centric business processes, in: Enterprise Design, Operations, and Computing. EDOC 2024 Workshops - IRESEARCH, MIDas4CS, Doctoral Consortium, Joint CBI-EDOC Forum and Other Joint CBI-EDOC Events, in: Lecture Notes in Business Information Processing, vol. 537, Springer, 2024, pp. 267–275, http://dx.doi.org/10.1007/978-3-031-79059-1_16.

[117] W.M.P. van der Aalst, Learning colored Petri nets using object-centric event data (OCED2CPN), in: 7th IEEE Congress on Information Science and Technology, CiSt 2023, IEEE, 2023, pp. 1–6, http://dx.doi.org/10.1109/CIST56084.2023.10410001.

[118] K. Jensen, L.M. Kristensen, L. Wells, Coloured Petri nets and CPN tools for modelling and validation of concurrent systems, Int. J. Softw. Tools Technol. Transf. 9 (3–4) (2007) 213–254, http://dx.doi.org/10.1007/S10009-007-0038-X.

[119] A. Berti, W.M.P. van der Aalst, CPN-py: A python-based tool for modeling and analyzing colored Petri nets, 2025, http://dx.doi.org/10.48550/ARXIV.2506.12238, CoRR abs/2506.12238, arXiv:2506.12238.

[120] R.M. Dijkman, Simpn: A python library for modeling and simulating timed, colored Petri nets, in: Proceedings of the Best Dissertation Award, Doctoral Consortium, and Demonstration & Resources Forum at BPM 2024 Co-Located with 22nd International Conference on Business Process Management, BPM 2024, in: CEUR Workshop Proceedings, vol. 3758, CEUR-WS.org, 2024, pp. 71–75, URL https://ceur-ws.org/Vol-3758/paper-12.pdf.

[121] R.L. Bianco, R.M. Dijkman, W. Nuijten, W. van Jaarsveld, Action-evolution Petri nets: A framework for modeling and solving dynamic task assignment problems, in: Business Process Management - 21st International Conference, BPM 2023, in: Lecture Notes in Computer Science, vol. 14159, Springer, 2023, pp. 216–231, http://dx.doi.org/10.1007/978-3-031-41620-0_13.

[122] L. Kirchdorfer, R. Blümel, T. Kampik, H. van der Aa, H. Stuckenschmidt, Agentsimulator: An agent-based approach for data-driven business process simulation, in: 6th International Conference on Process Mining, ICPM 2024, IEEE, 2024, pp. 97–104, http://dx.doi.org/10.1109/ICPM63005.2024.10680660.

[123] A. Tour, A. Polyvyanyy, A.A. Kalenkova, A. Senderovich, Agent miner: An algorithm for discovering agent systems from event data, in: Business Process Management - 21st International Conference, BPM 2023, in: Lecture Notes in Computer Science, vol. 14159, Springer, 2023, pp. 284–302, http://dx.doi.org/10.1007/978-3-031-41620-0_17.

[124] D. Sommers, N. Sidorova, B.F. van Dongen, Aligning event logs to resource-constrained ν-Petri nets, in: Application and Theory of Petri Nets and Concurrency - 43rd International Conference, PETRI NETS 2022, Bergen, Norway, June 19-24, 2022, Proceedings, in: Lecture Notes in Computer Science, vol. 13288, Springer, 2022, pp. 325–345, http://dx.doi.org/10.1007/978-3-031-06653-5_17.

[125] A. Tour, A. Polyvyanyy, A.A. Kalenkova, Agent system mining: Vision, benefits, and challenges, IEEE Access 9 (2021) 99480–99494, http://dx.doi.org/10.1109/ACCESS.2021.3095464.

[126] O. López-Pintado, M. Dumas, Business process simulation: Probabilistic modeling of intermittent resource availability and multitasking behavior, Inf. Syst. 127 (2025) 102471, http://dx.doi.org/10.1016/J.IS.2024.102471.

[127] E.L. Klijn, F. Mannhardt, D. Fahland, Classifying and detecting task executions and routines in processes using event graphs, in: Business Process Management Forum - BPM Forum 2021, in: Lecture Notes in Business Information Processing, vol. 427, Springer, 2021, pp. 212–229, http://dx.doi.org/10.1007/978-3-030-85440-9_13.

[128] E.L. Klijn, I. Tentina, D. Fahland, F. Mannhardt, Decomposing process performance based on actor behavior, in: 6th International Conference on Process Mining, ICPM 2024, IEEE, 2024, pp. 129–136, http://dx.doi.org/10.1109/ICPM63005.2024.10680657.

[129] E.L. Klijn, F. Mannhardt, D. Fahland, Multi-perspective concept drift detection: Including the actor perspective, in: Advanced Information Systems Engineering - 36th International Conference, CAiSE 2024, in: Lecture Notes in Computer Science, vol. 14663, Springer, 2024, pp. 141–157, http://dx.doi.org/10.1007/978-3-031-61057-8_9.

[130] S. Khayatbashi, V. Sjölind, A. Granåker, A. Jalali, AI-enhanced business process automation: A case study in the insurance domain using object-centric process mining, in: Enterprise, Business-Process and Information Systems Modeling - 26th International Conference, BPMDS 2025, and 30th International Conference, EMMSAD 2025, in: Lecture Notes in Business Information Processing, vol. 558, Springer, 2025, pp. 3–18, http://dx.doi.org/10.1007/978-3-031-95397-2_1.

[131] J.M.E.M. van der Werf, H.M.W. Verbeek, W.M.P. van der Aalst, Context-aware compliance checking, in: Business Process Management - 10th International Conference, BPM 2012, Tallinn, Estonia, September 3-6, 2012. Proceedings, in: Lecture Notes in Computer Science, vol. 7481, Springer, 2012, pp. 98–113, http://dx.doi.org/10.1007/978-3-642-32885-5_7.

[132] W. Fdhila, M. Gall, S. Rinderle-Ma, J. Mangler, C. Indiono, Classification and formalization of instance-spanning constraints in process-driven applications, in: Business Process Management - 14th International Conference, BPM 2016, in: Lecture Notes in Computer Science, vol. 9850, Springer, 2016, pp. 348–364, http://dx.doi.org/10.1007/978-3-319-45348-4_20.

[133] M. Leitner, J. Mangler, S. Rinderle-Ma, Definition and enactment of instance-spanning process constraints, in: Web Information Systems Engineering - WISE 2012 - 13th International Conference, in: Lecture Notes in Computer Science, vol. 7651, Springer, 2012, pp. 652–658, http://dx.doi.org/10.1007/978-3-642-35063-4_49.

[134] K. Winter, S. Rinderle-Ma, Defining instance spanning constraint patterns for business processes based on proclets, in: Conceptual Modeling - 39th International Conference, ER 2020, in: Lecture Notes in Computer Science, vol. 12400, Springer, 2020, pp. 149–163, http://dx.doi.org/10.1007/978-3-030-62522-1_11.

[135] T. Lichtenstein, M. Weske, Execution semantics for process choreographies with data, in: Business Process Management Forum - BPM 2023 Forum, in: Lecture Notes in Business Information Processing, vol. 490, Springer, 2023, pp. 90–106, http://dx.doi.org/10.1007/978-3-031-41623-1_6.

[136] T. Lichtenstein, A. Seidel, M. Weske, Detecting conflicting object views in object-centric process choreographies, in: Conceptual Modeling - 44th International Conference, ER 2025, in: Lecture Notes in Computer Science, vol. 16189, Springer, 2025, pp. 204–222, http://dx.doi.org/10.1007/978-3-032-08623-5_11.

[137] N. Lohmann, K. Wolf, Artifact-centric choreographies, in: Service-Oriented Computing - 8th International Conference, ICSOC 2010, in: Lecture Notes in Computer Science, vol. 6470, 2010, pp. 32–46, http://dx.doi.org/10.1007/978-3-642-17358-5_3.

[138] A. Meyer, L. Pufahl, K. Batoulis, D. Fahland, M. Weske, Automating data exchange in process choreographies, Inf. Syst. 53 (2015) 296–329, http://dx.doi.org/10.1016/J.IS.2015.03.008.

[139] W. Reisig, Understanding Petri Nets - Modeling Techniques, Analysis Methods, Case Studies, Springer, 2013, http://dx.doi.org/10.1007/978-3-642-33278-4.

[140] J. Esparza, M. Nielsen, Decidability issues for Petri nets - a survey, 2024, http://dx.doi.org/10.48550/ARXIV.2411.01592, CoRR abs/2411.01592, arXiv:2411.01592.

[141] D. Calvanese, G. De Giacomo, M. Montali, Foundations of data-aware process analysis: A database theory perspective, in: Proceedings of PODS 2013, 2013, pp. 1–12.

[142] C. Baral, G.D. Giacomo, Knowledge representation and reasoning: What's hot, in: B. Bonet, S. Koenig (Eds.), Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, AAAI 2015, AAAI Press, 2015, pp. 4316–4317.

[143] A. Gianola, Verification of data-aware processes via satisfiability modulo theories, in: Lecture Notes in Business Information Processing, vol. 470, Springer, 2023, http://dx.doi.org/10.1007/978-3-031-42746-6.

[144] W.M.P. van der Aalst, K.M. van Hee, A.H.M. ter Hofstede, N. Sidorova, H.M.W. Verbeek, M. Voorhoeve, M.T. Wynn, Soundness of workflow nets: classification, decidability, and analysis, Form. Asp. Comput. 23 (3) (2011) 333–363, http://dx.doi.org/10.1007/S00165-010-0161-4.

[145] M. Montali, D. Calvanese, Soundness of data-aware, case-centric processes, Int. J. Softw. Tools Technol. Transf. 18 (5) (2016) 535–558, http://dx.doi.org/10.1007/S10009-016-0417-2.

[146] M. de Leoni, P. Felli, M. Montali, A holistic approach for soundness verification of decision-aware process models, in: ER, in: LNCS, vol. 11157, Springer, 2018, pp. 219–235.

[147] I.A. Lomazova, A.A. Mitsyuk, A. Rivkin, Typed Petri nets with variable arc weights, in: J.D. Weerdt, L. Pufahl (Eds.), Business Process Management Workshops - BPM 2023 International Workshops, Utrecht, the Netherlands, September 11-15, 2023, Revised Selected Papers, in: Lecture Notes in Business Information Processing, vol. 492, Springer, 2023, pp. 483–495, http://dx.doi.org/10.1007/978-3-031-50974-2_36.

[148] I.A. Lomazova, A.A. Mitsyuk, A. Rivkin, Soundness in object-centric workflow Petri nets, 2021, CoRR abs/2112.14994, URL https://arxiv.org/abs/2112.14994, arXiv:2112.14994.

[149] D. Calvanese, S. Ghilardi, A. Gianola, M. Montali, A. Rivkin, SMT-based verification of data-aware processes: a model-theoretic approach, Math. Structures Comput. Sci. 30 (3) (2020) 271–313, http://dx.doi.org/10.1017/S0960129520000067.

[150] B. Bagheri Hariri, D. Calvanese, G. De Giacomo, A. Deutsch, M. Montali, Verification of relational data-centric dynamic systems with external services, in: Proceedings of the 32nd ACM SIGMOD-SIGACT-SIGAI Symposium on Principles of Database Systems, in: PODS '13, Association for Computing Machinery, New York, NY, USA, 2013, pp. 163–174, http://dx.doi.org/10.1145/2463664.2465221.

[151] L. Geatti, A. Gianola, N. Gigante, Linear temporal logic modulo theories over finite traces., in: Proceedings of IJCAI, vol. 22, 2022, pp. 2641–2647.

[152] A. Gianola, M. Montali, S. Winkler, Linear-time verification of data-aware processes modulo theories via covers and automata, in: Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI Press, 2024, pp. 10525–10534, http://dx.doi.org/10.1609/AAAI.V38I9.28922.

[153] P. Felli, M. Montali, S. Winkler, CTL* model checking for data-aware dynamic systems with arithmetic, in: Automated Reasoning - 11th International Joint Conference, IJCAR 2022, in: Lecture Notes in Computer Science, vol. 13385, Springer, 2022, pp. 36–56, http://dx.doi.org/10.1007/978-3-031-10769-6_4.

[154] H.A. Reijers, I. Vanderfeesten, M.G.A. Plomp, P. Van Gorp, D. Fahland, W.L.M. van der Crommert, H.D.D. Garcia, Evaluating data-centric process approaches: does the human factor factor in? Softw. Syst. Model. 16 (3) (2017) 649–662, http://dx.doi.org/10.1007/s10270-015-0491-z.

[155] D. Fahland, D. Lübke, J. Mendling, H.A. Reijers, B. Weber, M. Weidlich, S. Zugal, Declarative versus imperative process modeling languages: The issue of understandability, in: Enterprise, Business-Process and Information Systems Modeling, 10th International Workshop, BPMDS 2009, and 14th International Conference, EMMSAD 2009, Held at CAiSE 2009, in: Lecture Notes in Business Information Processing, vol. 29, Springer, 2009, pp. 353–366, http://dx.doi.org/10.1007/978-3-642-01862-6_29.

[156] J. Holz, L. Pufahl, I. Weber, A systematic comparison of case management languages, in: Business Process Management Workshops, Springer International Publishing, Cham, 2023, pp. 257–273, http://dx.doi.org/10.1007/978-3-031-25383-6_19.

[157] C. Schreiber, A. Abbad-Andaloussi, B. Weber, On the cognitive and behavioral effects of abstraction and fragmentation in modularized process models, Inf. Syst. 125 (2024) 102424, http://dx.doi.org/10.1016/j.is.2024.102424.

[158] A. Berti, W.M.P. van der Aalst, OC-PM: Analyzing object-centric event logs and process models, Int. J. Softw. Tools Technol. Transf. 25 (1) (2023) 1–17, http://dx.doi.org/10.1007/s10009-022-00668-w.

[159] T. Brachmann, I. Koren, L. Liss, W. van der Aalst, Visualizing object-centric Petri nets, in: International Conference on Business Process Management, Springer, 2025.

[160] S. Khayatbashi, M. Rafiei, J. Chen, T. Kampik, G. Berg, A. Jalali, Enriching object-centric event data with process scopes: A framework for aggregation and analysis, 2025, http://dx.doi.org/10.48550/ARXIV.2508.18830, CoRR abs/2508.18830, arXiv:2508.18830.

[161] T.T. Hildebrandt, A.A. Andaloussi, L.R. Christensen, S. Debois, N.P. Healy, H.A. López, M. Marquard, N.L.H. Møller, A.C.M. Petersen, T. Slaats, B. Weber, Ecoknow: Engineering effective, co-created and compliant adaptive case management systems for knowledge workers, in: ICSSP '20: International Conference on Software and System Processes 2020, ACM, 2020, pp. 155–164, http://dx.doi.org/10.1145/3379177.3388908.

[162] S. Debois, T.T. Hildebrandt, T. Slaats, M. Marquard, A case for declarative process modelling: Agile development of a grant application system, in: 18th IEEE International Enterprise Distributed Object Computing Conference Workshops and Demonstrations, EDOC Workshops 2014, IEEE Computer Society, 2014, pp. 126–133, http://dx.doi.org/10.1109/EDOCW.2014.27.

[163] M. Breitmayer, L. Arnold, M. Reichert, Data-driven evolution of activity forms in object- and process-aware information systems, in: Cooperative Information Systems - 28th International Conference, CoopIS 2022, in: Lecture Notes in Computer Science, vol. 13591, Springer, 2022, pp. 186–204, http://dx.doi.org/10.1007/978-3-031-17834-4_11.

[164] S. Khayatbashi, N. Miri, A. Jalali, OLAP operations for object-centric process mining, in: Intelligent Information Systems - CAiSE 2025 Forum and Doctoral Consortium, in: Lecture Notes in Business Information Processing, vol. 557, Springer, 2025, pp. 111–118, http://dx.doi.org/10.1007/978-3-031-94590-8_14.

[165] P. Wegner, Concepts and paradigms of object-oriented programming, OOPS Messenger 1 (1) (1990) 7–87, http://dx.doi.org/10.1145/382192.383004.

[166] D. Harel, A. Kantor, G. Katz, A. Marron, G. Weiss, G. Wiener, Towards behavioral programming in distributed architectures, Sci. Comput. Program. 98 (2015) 233–267, http://dx.doi.org/10.1016/J.SCICO.2014.03.003.

[167] D. Boaz, T. Heath, M. Gupta, L. Limonad, Y. Sun, R. Hull, R. Vaculín, The ACSI hub: A data-centric environment for service interoperation, in: Proceedings of the BPM Demo Sessions 2014 Co-Located with the 12th International Conference on Business Process Management, BPM 2014, in: CEUR Workshop Proceedings, vol. 1295, CEUR-WS.org, 2014, p. 11.

[168] F.F.T. Heath, D. Boaz, M. Gupta, R. Vaculín, Y. Sun, R. Hull, L. Limonad, Barcelona: A design and runtime environment for declarative artifact-centric BPM, in: Service-Oriented Computing - 11th International Conference, ICSOC 2013, in: Lecture Notes in Computer Science, vol. 8274, Springer, 2013, pp. 705–709, http://dx.doi.org/10.1007/978-3-642-45005-1_65.

[169] N. Dragoni, S. Giallorenzo, A. Lluch-Lafuente, M. Mazzara, F. Montesi, R. Mustafin, L. Safina, Microservices: Yesterday, today, and tomorrow, in: Present and Ulterior Software Engineering, Springer, 2017, pp. 195–216, http://dx.doi.org/10.1007/978-3-319-67425-4_12.

[170] A. Delgado, D. Calegari, Process and organizational data integration from BPMS and relational/nosql sources for process mining, in: Proceedings of the 17th International Conference on Software Technologies, ICSOFT 2022, SCITEPRESS, 2022, pp. 557–566, http://dx.doi.org/10.5220/0011322500003266.

[171] J. Xiong, G. Xiao, T.E. Kalayci, M. Montali, Z. Gu, D. Calvanese, A virtual knowledge graph based approach for object-centric event logs extraction, in: Process Mining Workshops, Springer Nature Switzerland, Cham, 2023, pp. 466–478, http://dx.doi.org/10.1007/978-3-031-27815-0_34.

[172] L.M. M'Baba, N. Assy, M. Sellami, W. Gaaloul, M.F. Nanne, Extracting artifact-centric event logs from blockchain applications, in: 2022 IEEE International Conference on Services Computing, SCC, 2022, pp. 274–283, http://dx.doi.org/10.1109/SCC55611.2022.00048.

[173] F. Corradini, A. Marcelletti, A. Morichetta, B. Re, L. Verducci, A mapping methodology enabling object-centric process mining on blockchain applications, Blockchain: Res. Appl. (2025) 100392.

[174] A. Polyvyanyy, Process Querying Methods, Springer, 2022.

[175] T. Vogelgesang, J. Ambrosy, D. Becher, R. Seilbeck, J. Geyer-Klingeberg, M. Klenk, Celonis PQL: A query language for process mining, in: A. Polyvyanyy (Ed.), Process Querying Methods, Springer, 2022, pp. 377–408, http://dx.doi.org/10.1007/978-3-030-92875-9_13.

[176] J. Brand, T. Kampik, C. Okulmus, M. Weidlich, One language to rule them all: Behavioural querying of process data using SQL, in: Process Mining Workshops - ICPM 2024 International Workshops, in: Lecture Notes in Business Information Processing, vol. 533, Springer, 2024, pp. 18–30, http://dx.doi.org/10.1007/978-3-031-82225-4_2.

[177] A. Swevels, R.M. Dijkman, D. Fahland, Inferring missing entity identifiers from context using event knowledge graphs, in: Business Process Management - 21st International Conference, BPM 2023, in: Lecture Notes in Computer Science, vol. 14159, Springer, 2023, pp. 180–197, http://dx.doi.org/10.1007/978-3-031-41620-0_11.

[178] A. Gavric, D. Bork, H. Proper, Multimodal process mining, in: 2024 26th International Conference on Business Informatics, CBI, IEEE, 2024, pp. 99–108.

[179] D. Janssen, F. Mannhardt, A. Koschmider, S.J. van Zelst, Process model discovery from sensor event data, in: International Conference on Process Mining, Springer, 2020, pp. 69–81.

[180] S. Mohammed, L. Ehrlinger, H. Harmouch, F. Naumann, D. Srivastava, The five facets of data quality assessment, SIGMOD Rec. 54 (2) (2025) 18–27, http://dx.doi.org/10.1145/3749116.3749120.

[181] L. Bosmans, J. Peeperkorn, A. Goossens, G. Lugaresi, J.D. Smedt, J.D. Weerdt, Dynamic and scalable data preparation for object-centric process mining, 2024, http://dx.doi.org/10.48550/ARXIV.2410.00596, CoRR abs/2410.00596, arXiv:2410.00596.

[182] W.M.P. van der Aalst, Federated process mining: Exploiting event data across organizational boundaries, in: IEEE International Conference on Smart Data Services, SMDS 2021, IEEE, 2021, pp. 1–7, http://dx.doi.org/10.1109/SMDS53860.2021.00011.

[183] A. Swevels, E.L. Klijn, D. Fahland, Object-centric process mining (and more) using a graph-based approach with promg, in: Doctoral Consortium and Demo Track 2023 At the International Conference on Process Mining 2023 Co-Located with the 5th International Conference on Process Mining, ICPM 2023, in: CEUR Workshop Proceedings, vol. 3648, CEUR-WS.org, 2023, URL https://ceur-ws.org/Vol-3648/paper_9922.pdf.

[184] D. Barenholz, M. Montali, A. Polyvyanyy, H.A. Reijers, A. Rivkin, J.M.E.M. van der Werf, There and back again - on the reconstructability and rediscoverability of typed jackson nets, in: Application and Theory of Petri Nets and Concurrency - 44th International Conference, PETRI NETS 2023, Lisbon, Portugal, June 25-30, 2023, Proceedings, in: Lecture Notes in Computer Science, vol. 13929, Springer, 2023, pp. 37–58, http://dx.doi.org/10.1007/978-3-031-33620-1_3.

[185] G. Park, J.N. Adams, W.M.P. van der Aalst, Conformance checking and performance analysis using object-centric directly-follows graphs, in: Business Process Management Forum, Springer Nature Switzerland, Cham, 2024, pp. 179–196, http://dx.doi.org/10.1007/978-3-031-70418-5_11.

[186] L.M. M'Baba, M. Sellami, N. Assy, W. Gaaloul, M.F. Nanne, Discovering guard stage milestone models through hierarchical clustering, in: Cooperative Information Systems, Springer Nature Switzerland, Cham, 2024, pp. 239–256, http://dx.doi.org/10.1007/978-3-031-46846-9_13.

[187] M. Breitmayer, L. Arnold, D. Goth, M. Reichert, An approach for discovering data-driven object lifecycle processes, in: Research Challenges in Information Science, Springer Nature Switzerland, Cham, 2024, pp. 237–254, http://dx.doi.org/10.1007/978-3-031-59465-6_15.

[188] G. Li, R.M. de Carvalho, W.M.P. van der Aalst, Automatic discovery of object-centric behavioral constraint models, in: Business Information Systems, Springer International Publishing, Cham, 2017, pp. 43–58, http://dx.doi.org/10.1007/978-3-319-59336-4_4.

[189] A. Goossens, J.D. Smedt, J. Vanthienen, Extracting process-aware decision models from object-centric process data, Inf. Sci. 682 (2024) 121263, http://dx.doi.org/10.1016/J.INS.2024.121263.

[190] A. Giacché, S. Pettinari, L. Rossi, Revealing one-to-many event relationships in event knowledge graphs, in: Process Mining Workshops - ICPM 2024 International Workshops, in: Lecture Notes in Business Information Processing, vol. 533, Springer, 2024, pp. 184–196, http://dx.doi.org/10.1007/978-3-031-82225-4_14.

[191] A. Senderovich, M. Weidlich, A. Gal, A. Mandelbaum, Queue mining for delay prediction in multi-class service processes, Inf. Syst. 53 (2015) 278–295.

[192] K. Winter, F. Stertz, S. Rinderle-Ma, Discovering instance and process spanning constraints from process execution logs, Inf. Syst. 89 (2020) 101484, http://dx.doi.org/10.1016/J.IS.2019.101484.

[193] K. Winter, S. Rinderle-Ma, Discovering instance-spanning constraints from process execution logs based on classification techniques, in: 21st IEEE International Enterprise Distributed Object Computing Conference, EDOC 2017, IEEE Computer Society, 2017, pp. 79–88, http://dx.doi.org/10.1109/EDOC.2017.20.

[194] N. Martin, L. Pufahl, F. Mannhardt, Detection of batch activities from event logs, Inf. Syst. 95 (2021) 101642, http://dx.doi.org/10.1016/J.IS.2020.101642.

[195] R.P.J.C. Bose, W.M.P. van der Aalst, I. Zliobaite, M. Pechenizkiy, Handling concept drift in process mining, in: Advanced Information Systems Engineering - 23rd International Conference, CAiSE 2011, in: Lecture Notes in Computer Science, vol. 6741, Springer, 2011, pp. 391–405, http://dx.doi.org/10.1007/978-3-642-21640-4_30.

[196] B. Xiu, G. Li, Diagnosing conformance between object-centric event logs and models, IEEE Access 11 (2023) 110837–110849, http://dx.doi.org/10.1109/ACCESS.2023.3322366.

[197] B. Xiu, G. Li, Quantifying conformance between object-centric event logs and models, https://ieeexplore.ieee.org/abstract/document/10580902.

[198] J.N. Adams, W.M. van der Aalst, Precision and fitness in object-centric process mining, in: 2021 3rd International Conference on Process Mining, ICPM, 2021, pp. 128–135, http://dx.doi.org/10.1109/ICPM53251.2021.9576886.

[199] J. Carmona, B.F. van Dongen, M. Weidlich, Conformance checking: Foundations, milestones and challenges, in: W.M.P. van der Aalst, J. Carmona (Eds.), Process Mining Handbook, in: Lecture Notes in Business Information Processing, vol. 448, Springer, 2022, pp. 155–190, http://dx.doi.org/10.1007/978-3-031-08848-3_5.

[200] L. Liss, J.N. Adams, W.M.P. van der Aalst, Object-centric alignments, in: Conceptual Modeling, Springer Nature Switzerland, Cham, 2023, pp. 201–219, http://dx.doi.org/10.1007/978-3-031-47262-6_11.

[201] E. Wrede, J.N. van Detten, L. Liss, S.J. Leemans, Out of babylon: Object-centric conformance checking on graph-based abstractions, in: International Conference on Cooperative Information Systems, 2025.

[202] D. Sommers, N. Sidorova, B.F. van Dongen, Conformance checking with model projections - rethinking log-model alignments for processes with interacting objects, in: Application and Theory of Petri Nets and Concurrency - 45th International Conference, PETRI NETS 2024, in: Lecture Notes in Computer Science, vol. 14628, Springer, 2024, pp. 61–82, http://dx.doi.org/10.1007/978-3-031-61433-0_4.

[203] A. Rogge-Solti, A. Senderovich, M. Weidlich, J. Mendling, A. Gal, In log and model we trust? A generalized conformance checking framework, in: Business Process Management - 14th International Conference, BPM 2016, in: Lecture Notes in Computer Science, vol. 9850, Springer, 2016, pp. 179–196, http://dx.doi.org/10.1007/978-3-319-45348-4_11.

[204] M. Kabierski, J.-R. Rehse, J.M.E. van der Werf, Comparing apples with oranges: An assessment framework for model-system similarity, in: Business Process Management Forum: BPM 2025 Forum, Springer Nature, 2025, p. 75.

[205] S. Khayatbashi, N. Miri, A. Jalali, Advancing object-centric process mining with multi-dimensional data operations, 2024, http://dx.doi.org/10.48550/ARXIV.2412.00393, CoRR abs/2412.00393, arXiv:2412.00393.

[206] D. Sommers, N. Sidorova, B.F. van Dongen, In system alignments we trust! explainable alignments via projections, Inf. Syst. 136 (2026) 102631, http://dx.doi.org/10.1016/J.IS.2025.102631.

[207] A. Polyvyanyy, W.M.P. van der Aalst, A.H.M. ter Hofstede, M.T. Wynn, Impact-driven process model repair, ACM Trans. Softw. Eng. Methodol. 25 (4) (2017) 28:1–28:60, http://dx.doi.org/10.1145/2980764.

[208] M.L. van Eck, N. Sidorova, W.M.P. van der Aalst, Guided interaction exploration and performance analysis in artifact-centric process models, Bus. Inf. Syst. Eng. 61 (6) (2019) 649–663, http://dx.doi.org/10.1007/S12599-018-0546-0.

[209] G. Park, J.N. Adams, W.M.P. van der Aalst, OPerA: object-centric performance analysis, in: Conceptual Modeling, Springer International Publishing, Cham, 2022, pp. 281–292, http://dx.doi.org/10.1007/978-3-031-17995-2_20.

[210] L. Liss, W.M.P. van der Aalst, Process area extraction by multilevel resource detection for object-centric process mining, in: Business Process Management - 23rd International Conference, BPM 2025, in: Lecture Notes in Computer Science, vol. 16044, Springer, 2025, pp. 197–214, http://dx.doi.org/10.1007/978-3-032-02867-9_13.

[211] J.S. Hildebrand, J.N. van Detten, S.J. Leemans, A distance metric for object-centric process mining, in: 2025 7th International Conference on Process Mining, ICPM, 2025.

[212] O. López-Pintado, S. Murashko, M. Dumas, Discovery and simulation of data-aware business processes, in: 6th International Conference on Process Mining, ICPM 2024, IEEE, 2024, pp. 105–112, http://dx.doi.org/10.1109/ICPM63005.2024.10680675.

[213] O. López-Pintado, J. Rosenbaum, M. Dumas, Optimization of activity batching policies in business processes, in: Business Process Management - 23rd International Conference, BPM 2025, in: Lecture Notes in Computer Science, vol. 16044, Springer, 2025, pp. 397–414, http://dx.doi.org/10.1007/978-3-032-02867-9_24.

[214] N. Martin, A. Solti, J. Mendling, B. Depaire, A. Caris, Mining batch activation rules from event logs, IEEE Trans. Serv. Comput. 14 (6) (2021) 1837–1848, http://dx.doi.org/10.1109/TSC.2019.2912163.

[215] O. López-Pintado, M. Dumas, J. Berx, Discovery, simulation, and optimization of business processes with differentiated resources, Inf. Syst. 120 (2024) 102289, http://dx.doi.org/10.1016/J.IS.2023.102289.
[216] B. Knopp, M. Pourbafrani, W.M.P. van der Aalst, Discovering object-centric process simulation models, in: 2023 5th International Conference on Process Mining, ICPM, 2023, pp. 81–88, http://dx.doi.org/10.1109/ICPM60904.2023.10271944.
[217] W.M.P. van der Aalst, Toward more realistic simulation models using object-centric process mining, in: ECMS 2023 Proceedings Edited By Enrico Vicario, Romeo Bandinelli, Virginia Fani, Michele Mastroianni, ECMS, 2023, pp. 5–13, http://dx.doi.org/10.7148/2023-0005.
[218] D. Chapela-Campa, I. Benchekroun, O. Baron, M. Dumas, D. Krass, A. Senderovich, A framework for measuring the quality of business process simulation models, Inf. Syst. 127 (2025) 102447, http://dx.doi.org/10.1016/J.IS.2024.102447.
[219] G. Park, W.M.P. van der Aalst, Operational process monitoring: An object-centric approach, Comput. Ind. 164 (2025) 104170, http://dx.doi.org/10.1016/j.compind.2024.104170.
[220] G. Park, W.M.P. van der Aalst, Monitoring constraints in business processes using object-centric constraint graphs, in: Process Mining Workshops, Springer Nature Switzerland, Cham, 2023, pp. 479–492, http://dx.doi.org/10.1007/978-3-031-27815-0_35.
[221] L. Arnold, M. Breitmayer, M. Reichert, Monitoring object-centric business processes: an empirical study, in: Research Challenges in Information Science: Information Science and the Connected World, Springer Nature Switzerland, Cham, 2023, pp. 327–342, http://dx.doi.org/10.1007/978-3-031-33080-3_20.
[222] G. Meroni, L. Baresi, M. Montali, P. Plebani, Multi-party business process compliance monitoring through IoT-enabled artifacts, Inf. Syst. 73 (2018) 61–78, http://dx.doi.org/10.1016/J.IS.2017.12.009.
[223] A. Kumar, P. Soffer, A. Tsoury, Normalizing object-centric process logs by applying database principles, Inf. Syst. 115 (2023) 102196, http://dx.doi.org/10.1016/j.is.2023.102196.
[224] A. Jalali, Object type clustering using Markov directly-follow multigraph in object-centric process mining, IEEE Access 10 (2022) 126569–126579, http://dx.doi.org/10.1109/ACCESS.2022.3226573.
[225] N. van Beest, H. Groefsema, A. Cryer, G. Governatori, S.C. Tosatto, H.M.S. Burke, Cross-instance regulatory compliance checking of business process event logs, IEEE Trans. Softw. Eng. 49 (11) (2023) 4917–4931, http://dx.doi.org/10.1109/TSE.2023.3319086.
[226] Q. Luo, Y. Zhang, C. Lee, D. Jin, P.O. Meredith, T. Serbanuta, G. Rosu, RV-monitor: Efficient parametric runtime verification with simultaneous properties, in: Runtime Verification - 5th International Conference, RV 2014, in: Lecture Notes in Computer Science, vol. 8734, Springer, 2014, pp. 285–300, http://dx.doi.org/10.1007/978-3-319-11164-3_24.
[227] K. Havelund, Rule-based runtime verification revisited, Int. J. Softw. Tools Technol. Transf. 17 (2) (2015) 143–170, http://dx.doi.org/10.1007/S10009-014-0309-2.
[228] D.A. Basin, F. Klaedtke, E. Zalinescu, The MonPoly monitoring tool, in: RV-CuBES 2017. an International Workshop on Competitions, Usability, Benchmarks, Evaluation, and Standardisation for Runtime Verification Tools, in: Kalpa Publications in Computing, vol. 3, EasyChair, 2017, pp. 19–28, http://dx.doi.org/10.29007/89HS.
[229] P. Ceravolo, M. Comuzzi, J. De Weerdt, C. Di Francescomarino, F.M. Maggi, Predictive process monitoring: concepts, challenges, and future research directions, Process. Sci. 1 (1) (2024) 2.
[230] A. Berti, J. Herforth, M.S. Qafari, W.M.P. van der Aalst, Graph-based feature extraction on object-centric event logs, Int. J. Data Sci. Anal. 18 (2) (2024) 139–155, http://dx.doi.org/10.1007/s41060-023-00428-2.
[231] W. Gherissi, J. El Haddad, D. Grigori, Object-centric predictive process monitoring, in: Service-Oriented Computing – ICSOC 2022 Workshops, Springer Nature Switzerland, Cham, 2023, pp. 27–39, http://dx.doi.org/10.1007/978-3-031-26507-5_3.
[232] R. Galanti, M. de Leoni, N. Navarin, A. Marazzi, Object-centric process predictive analytics, Expert Syst. Appl. 213 (2023) 119173, http://dx.doi.org/10.1016/j.eswa.2022.119173.
[233] T.K. Smit, H.A. Reijers, X. Lu, HOEG: A new approach for object-centric predictive process monitoring, in: Advanced Information Systems Engineering, Springer Nature Switzerland, Cham, 2024, pp. 231–247, http://dx.doi.org/10.1007/978-3-031-61057-8_14.
[234] J.N. Adams, H. Drescher, A. Swoboda, N. Günnemann, G. Park, W.M.P. van der Aalst, Improving predictive process monitoring using object-centric process mining, in: 32nd European Conference on Information Systems - People First: Constructing Digital Futures Together, ECIS 2024, 2024, URL https://aisel.aisnet.org/ecis2024/track08_bpm_di/track08_bpm_di/7.
[235] A. Seidel, S. Haarmann, M. Weske, Model-based recommendations for next-best actions in knowledge-intensive processes, in: Advanced Information Systems Engineering - 36th International Conference, CAiSE 2024, in: Lecture Notes in Computer Science, vol. 14663, Springer, 2024, pp. 195–211, http://dx.doi.org/10.1007/978-3-031-61057-8_12.
[236] M. de Leoni, D.P. Volpato, Global predictive monitoring of object-centric processes, in: Business Process Management - 23rd International Conference, BPM 2025, in: Lecture Notes in Computer Science, vol. 16044, Springer, 2025, pp. 255–272, http://dx.doi.org/10.1007/978-3-032-02867-9_16.
[237] A. Buliga, C. Di Francescomarino, C. Ghidini, F.M. Maggi, Counterfactuals and ways to build them: Evaluating approaches in predictive process monitoring, in: Advanced Information Systems Engineering - 35th International Conference, CAiSE 2023, in: Lecture Notes in Computer Science, vol. 13901, Springer, 2023, pp. 558–574, http://dx.doi.org/10.1007/978-3-031-34560-9_33.
[238] O. Hundogan, X. Lu, Y. Du, H.A. Reijers, CREATED: generating viable counterfactual sequences for predictive process analytics, in: Advanced Information Systems Engineering - 35th International Conference, CAiSE 2023, in: Lecture Notes in Computer Science, vol. 13901, Springer, 2023, pp. 541–557, http://dx.doi.org/10.1007/978-3-031-34560-9_32.