Mining For Meaning: Formalization and Implementation of Ontology-Driven Process Mining

Riley Moher[1]

[1]Department of Mechanical & Industrial Engineering, University of Toronto, Canada

1. Introduction: Process Mining and Ontologies

Process mining has fundamentally transformed how organizations discover, monitor, and optimize operational workflows [1]. Despite its data-centric foundation, real-world process mining is not a purely data-driven practice. For example, repeated patient intake events for the same patient may be interpreted either as evidence of finer-grained clinical activities or as erroneous duplicates that should be removed. Such decisions—alongside filtering event logs, correcting timestamps, re-labelling activities or objects, resolving case ambiguities, and abstracting events [2, 3]—show that transforming raw event data into meaningful insights inherently depends on interpretive commitments. Rather than treating these interventions as auxiliary "pre-mining" steps [3], they reflect the continual application of domain knowledge throughout the analytical lifecycle, where analysts iteratively refine both data and assumptions through modelling, transformation, and validation.

This doctoral work argues that these actions are central to the analysis itself, constituting the active construction of a *data theory* [4]. A data theory is a formal specification of the interpretive commitments that govern how raw event data is transformed into a computable representation. Rather than remaining an implicit by-product of ad hoc pre-mining decisions, this work provides a formalization of such theories together with a computational mechanism for applying them in practice, enabling analysts to inspect, apply, and revise interpretive commitments without requiring expertise in ontology engineering.

The challenge of combining process knowledge with event data is well-recognized within the process mining community. Recent efforts have targeted classification of data quality issues [2], challenging event log data formats [5], and introducing new paradigms like object-centric event logs [6, 7]. While these initiatives represent important progress, they do not fully address the broader problem of formally capturing the knowledge and assumptions that shape the analysis itself [8]. Consequently, many critical analytical decisions remain external to, and unrecorded by, the process models and event data on which conclusions are ultimately based.

In parallel, the applied ontology community has developed rich conceptual models for representing complex process entities and relationships [9, 10, 11, 12]. While these are widely used as reference models across domains such as biomedicine [13], criminal law [14], manufacturing [15], and satellite telemetry [16], process mining requires more than referential models. For example, when identifying hand-offs in an event log, an analyst may wish to exclude events which amount to a support ticket sitting in a queue, as no work is being done by any human resource [1]. Rather than obfuscating domain knowledge or goals behind filtering steps, we wish for this interpretation of the data to be explicit. Encoding such choices as part of a formal representation enables them to be applied consistently, reused across analyses, and modified independently of other modelling decisions, while directly shaping the resulting analysis.

Realizing this integration, however, presents a significant interdisciplinary challenge. Foundational ontologies prioritize faithful, detailed representations of reality, resulting in highly granular concep-

Proceedings of the BPM 2026 Best Dissertation Award, Doctoral Consortium, and Demonstration & Resources Forum (BPM 2026), Toronto, Canada, September 27th to October 2nd, 2026.
riley.moher@mail.utoronto.ca (R. Moher)

[1]This specific example is a real issue that arises in a widely-utilized benchmark dataset, BPIC 2013 [17]

CEUR Workshop Proceedings ceur-ws.org ISSN 1613-0073
published 2026-09-28

tualizations with vast and diverse entities and axioms. While semantically robust, their *operational realization* [4] over high-volume enterprise event logs is often intractable.

On the more lightweight side, Semantic Web approaches such as OWL ontologies, RDF, and virtual knowledge graphs [18, 19, 20, 21] achieve scalability by restricting expressive power and delegating more sophisticated reasoning to external software. Consequently, ontology-based process mining approaches [22, 23, 24, 25] have largely focused on semantic annotation, data integration, and vocabulary alignment rather than representing the interpretative decisions that shape process mining analyses. More broadly, prior evaluations show that the expressive restrictions of Description Logic formalisms limit the ability of Semantic Web ontologies to capture fundamental process dynamics and relationships [26, 27].

As a result, much of the knowledge guiding process mining remains opaquely embedded within scattered scripts and software. This lack of operational transparency is especially critical for enterprises under regulatory scrutiny (e.g., healthcare, finance, and insurance), where a formal, verifiable record of how data were transformed is just as critical as the conclusions themselves.

This doctoral work approaches these issues through a middle ground: a hybrid reasoning paradigm that combines scalable ontology-based data integration techniques with expressive logical formalisms. By formalizing the process *data theories* [4] that analysts iteratively construct, this research provides a framework for integrating foundational process ontologies with real-world event data while balancing semantic expressiveness, transparency, and computational tractability.

To guide this work, the following research questions are posed:

- **RQ1:** What are the most significant and common kinds of interpretative choices made in analyzing event log data, and how might they be characterized through ontological commitments?
- **RQ2:** How can applied ontology be utilized to formalize the iterative, interpretative decisions that shape process mining analyses?
- **RQ3:** What logical and architectural choices are needed to enable an implementation of expressive ontology-driven process mining while maintaining computational tractability?

Substantial progress has already been made toward these objectives. Prior work during the PhD established theoretical foundations of process data theories and their operational realization with expressive ontologies [28, 4]. These contributions are supported by a working software implementation [2] that couples scalable data integration technologies [21, 29] with datalog and first-order logic process ontologies [9, 30], demonstrating the practical feasibility of the proposed approach on real-world process mining datasets.

The remainder of this paper outlines the research trajectory. Section 2 situates the project within the broader literature. Section 3 presents the completed contributions and implementation status. Finally, Section 4 discusses ongoing work regarding event abstraction and object-centric process ontologies, outlining future directions toward a comprehensive ontology-driven process mining framework.

2. Identifying the Gap

Recent work has increasingly introduced formal and semantic techniques into process mining, ranging from models based on temporal logics [31, 32] to ontology-driven querying of event data [33]. However, these approaches remain structured around an *onto/logical divide*: formal mechanisms are used for representation or reasoning, but the ontological commitments they encode about processes, events, and objects are rarely made explicit or operationalised. As a result, what we refer to as a *process data theory*—the structured interpretation of event data through domain assumptions—remains implicit, with design decisions driven primarily by implementation concerns such as scalability or decidability rather than explicit semantic commitments. This limits support for principled reasoning about process knowledge across analytical workflows.

[2]GitHub repository: https://github.com/Ontology-Benchmarks/opal

Event Knowledge Graph (EKG) approaches extend process mining representations by integrating heterogeneous behavioural data into graph-based structures [34]. While this improves representational flexibility, transformations are encoded directly in query logic rather than through explicit interpretive mappings. For example, assumptions such as "a hand-off excludes transfers via system queues" or "multiple intake events correspond to a single real-world patient admission" would be embedded implicitly in queries rather than represented in a more formal and reusable manner. Object-centric extensions (e.g., OCEL/OCED [35]) further improve structural expressiveness but remain schema-driven, without an explicit mechanism to represent and reuse such interpretive assumptions.

Declarative and data-aware process models, such as DECLARE [31], provide a formal foundation based on Linear Temporal Logic (LTL) [36], enabling specification of processes in terms of constraints over event sequences. However, they rely on a rigid assumption of events being homogeneous and totally ordered, which clashes with the messiness of real-world event logs. Extensions incorporating data and objects [32, 37] partially address this limitation by enriching events with attributes or object relations, but still rely on flattening heterogeneous domain interpretations into simple attribute-value encodings, leaving the underlying interpretive commitments implicit.

Ontology-Based Data Access (OBDA) frameworks, exemplified by ONPROM in process mining [33], use lightweight ontologies to mediate queries over underlying event data. While effective for structured data access, OBDA is primarily a retrieval mechanism: it supports querying a fixed dataset via an ontology, but does not support data transformation or the representation of alternative interpretations of the data. As a result, interpretive operations such as filtering, correction, or re-labelling remain external to the framework and must be handled outside the ontology layer.

Semantic Web vocabularies and event ontologies, including BPMO [38] and SUPER [22, 23], primarily function as lightweight annotation or classification schemes. They are limited by the usage of less expressive semantic web formalisms, prioritizing taxonomic relations over complex relations like temporal structures or causal dependencies. Empirical evaluations against foundational ontologies such as PSL [9] and UFO [10] highlight these gaps in the ability of these lightweight approaches to represent competency questions about rich process concepts [26, 27].

Collectively, these paradigms illustrate a persistent limitation: existing semantic and formal approaches either scale by restricting expressive interpretive power or increase expressiveness at the cost of computational feasibility, without providing a mechanism to explicitly represent and execute process data theories as part of the analytical workflow.

3. Research Progress and Implemented Contributions

This PhD work has advanced the research questions with several key areas of focus: (i) the construction of interpretive process knowledge, (ii) its formal representation, and (iii) its execution over event data. Figure 1 illustrates the supporting architecture for this structure and the relationships between its components.

Interpretive foundation: process data theories (RQ1). At the foundation of the approach is the characterization of interpretive decisions involved in transforming raw event data into meaningful process representations. These include decisions such as event filtering, activity and object identification, case construction, and abstraction. This work formalizes these as elements of a *process data theory*, a structured specification of how a dataset is interpreted through domain assumptions grounded in a process ontology. The specifiation of data theories is partly explored in our previous work [4], but an ongoing submission to the Applied Ontology journal aims to flesh this out in much greater detail. While data theories themselves are not strictly a system component, they provide the conceptual basis for all subsequent formalization.

Formalization through ontology mapping and patterns (RQ2). The second element operationalises process data theories through formal representations based on ontology mappings and reusable templates. Our earlier work introduced a pattern-based methodology for constructing such theories using ontology-driven knowledge patterns [28]. These patterns encode various interpretive

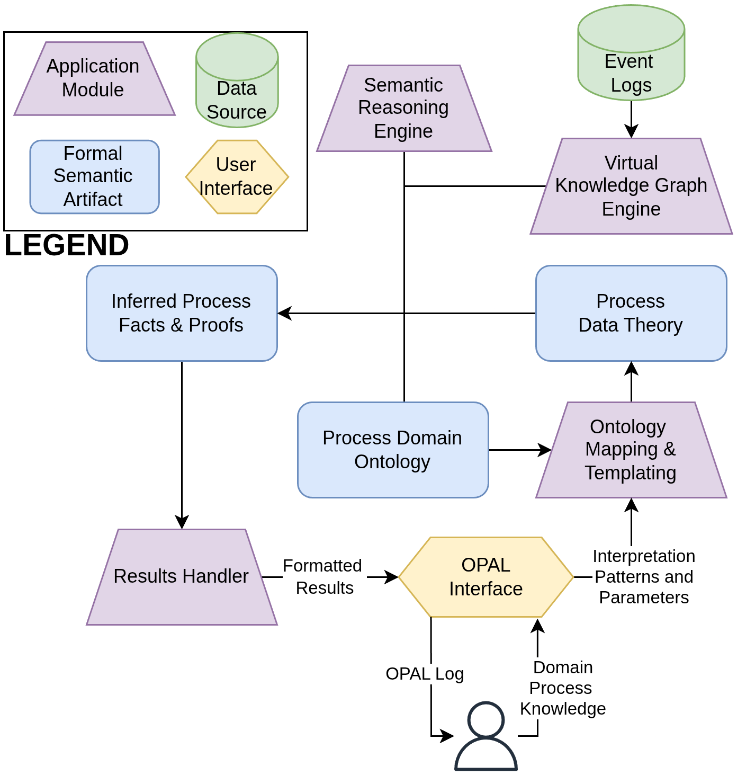

Figure 1: High-level architecture of OPAL

decisions—like event abstraction, or elements of a process model—as parameterized and reusable artifacts. These patterns are intended to make interpretive decisions explicit and reusable across analyses. Take for example modelling causal dependencies: a practitioner simply instantiates a pattern like `Precedes(QualityCheck, Ship)`. While similar in usage to DECLARE constraints, these templates are significantly more expressive; because they are grounded in a foundational ontology, they can capture complex semantics like participation, occurrences over time intervals, and nested subactivities. This mapping and templating is intended to enable the creation of data theories without requiring practitioners to be experts in knowledge representation.

Execution layer: OPAL system (RQ3).

To address the challenge of balancing computational tractability with expressive process reasoning, the Ontology-driven Process Mining Analysis Library (OPAL) employs a *hybrid reasoning architecture*. Standard Semantic Web ontologies alone lack the expressivity required for deep process logic [26, 27]. We therefore leverages mature, highly scalable Semantic Web mapping tools (Ontop [21] and DuckDB [29]) for efficient virtualisation, and utilize more an expressive Datalog engine (Soufflé [30]) for heavier reasoning.

The system materializes these results as inferred process facts—such as state changes, and participation constraints. Crucially, OPAL also outputs a metadata artifact that records the process data theory applied, capturing all mappings, templates, and interpretive assumptions to enable reproducibility and verifiability.

Integration and workflow. As illustrated in Figure 1, interpretive decisions (RQ1) are formally encoded as data theories, instantiated via expressive ontology templates (RQ2), and executed through OPAL's hybrid architecture (RQ3). This strict modular separation ensures that explicit, formally verified domain knowledge—rather than opaque analytical assumptions—drives process mining analysis.

Ongoing work. Our current efforts to extend this foundational architecture are advancing along three targeted tracks:

- **Process Data Theories:** The core formalization of pattern-based data theories and interpretive mappings is being expanded for submission to the *Applied Ontology* journal.

- **Mereological Event Abstraction:** To yield deeper insights into how raw data abstracts to high-level behavior, we are formalizing event abstraction not as a pre-processing step, but as ontological commitments grounded in process mereology (targeting ICPM 2027).
- **Ontology Discovery:** We are developing automated, implementation-focused methods to discover implicit ontologies from object-centric (OCEL) datasets by analyzing object co-participation and synchronization patterns (targeting the *Business & Information Systems Engineering* (BISE) journal).

Together, these contributions demonstrate a clear progression from formalizing interpretive structures to executing them within a scalable, ontology-driven process mining framework.

4. Discussion and Conclusion

This doctoral project investigates how applied ontology can be used to make the interpretative knowledge underlying process mining analyses explicit, reusable, and computationally tractable. While process mining is often presented as a data-driven discipline, practical analyses depend upon numerous assumptions and interpretations rooted in external domain knowledge. These assumptions frequently remain embedded within software pipelines or analyst expertise, limiting transparency, reproducibility, and verification.

The contributions completed to date establish the foundations for addressing this challenge. The introduction of process data theories provides a formal mechanism for representing interpretative process mining knowledge, while the OPAL architecture demonstrates how expressive process ontologies can be connected to real-world process data. The accompanying software implementation further shows that these ideas can be operationalized using existing data integration and reasoning technologies without significant sacrifices to reasoning capabilities of practical process mining applications.

A central contribution of this work is the identification of a middle ground between two extremes that have traditionally characterized ontology-based approaches to process analysis. At one extreme, lightweight semantic technologies provide scalable data integration but offer limited support for representing complex process knowledge. At the other, expressive foundational ontologies provide rich conceptual models but often lack practical mechanisms for operational realization over enterprise-scale datasets. The proposed framework combines the strengths of both approaches by maintaining explicit ontological commitments while leveraging scalable data processing infrastructure.

Many process mining tasks, particularly those involving statistical analysis, machine learning, and large-scale discovery algorithms, are better addressed through specialized computational methods. We do not aim to replace these methods but to complement these approaches by providing formal representations of the knowledge and assumptions that shape process mining analyses. In doing so, ontology-driven process mining can improve transparency, reproducibility, interoperability, and verification across the broader process mining lifecycle.

More broadly, this research contributes to both the process mining and applied ontology communities. For process mining, it provides mechanisms to formally represent and characterize the knowledge used to interpret process data. For applied ontology, it offers a means of grounded, data-driven application of process ontologies through operational realization with real-world datasets. By bridging these communities, this doctoral project aims to establish both *ontology-driven process mining* and *data-driven process ontology* as practical and semantically rigorous paradigms for making process mining more meaningful.

Declaration on Generative AI

During the preparation of this work, the author(s) used Gemini Pro and ChatGPT in order to: make minor edits for word choice and phrasing. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the publication's content.

References

[1] W. Van Der Aalst, A. Adriansyah, A. K. A. De Medeiros, F. Arcieri, T. Baier, T. Blickle, J. C. Bose, P. Van Den Brand, R. Brandtjen, J. Buijs, et al., Process mining manifesto, in: Business Process Management Workshops: BPM 2011 International Workshops, Clermont-Ferrand, France, August 29, 2011, Revised Selected Papers, Part I 9, Springer, 2012, pp. 169–194.

[2] S. Suriadi, R. Andrews, A. ter Hofstede, M. T. Wynn, Event log imperfection patterns for process mining: Towards a systematic approach to cleaning event logs, Information systems 64 (2017) 132–150.

[3] S. K. Pradhan, M. Jans, N. Martin, Getting the data in shape for your process mining analysis: An in-depth analysis of the pre-analysis stage, ACM Computing Surveys 57 (2025) 1–37.

[4] R. Moher, M. Gruninger, Process representation meets operational realization: An architecture for data-driven process ontology application through process mining, in: Formal Ontology in Information Systems, IOS Press, 2025, pp. 196–209.

[5] R. Accorsi, C. Di Ciccio, L. Jayarathna, H. Verbeek, Rethinking the input for process mining: Insights from the xes survey and workshop, in: Process Mining Workshops, 2022, p. 3.

[6] D. Fahland, M. Montali, J. Lebherz, W. M. van der Aalst, M. van Asseldonk, P. Blank, L. Bosmans, M. Brenscheidt, C. Di Ciccio, A. Delgado, et al., Towards a simple and extensible standard for object-centric event data (oced)–core model, design space, and lessons learned, arXiv preprint arXiv:2410.14495 (2024).

[7] A. Berti, I. Koren, J. N. Adams, G. Park, B. Knopp, N. Graves, M. Rafiei, L. Liß, L. T. G. Unterberg, Y. Zhang, et al., Ocel (object-centric event log) 2.0 specification, arXiv preprint arXiv:2403.01975 (2024).

[8] H. Hooshyar, M. Fumagalli, M. Montali, G. Guizzardi, Time and relations into focus: Ontological foundations of object-centric event data, arXiv preprint arXiv:2512.14425 (2025).

[9] M. Grüninger, Ontology of the process specification language, in: Handbook on ontologies, Springer, 2004, pp. 575–592.

[10] G. Guizzardi, A. Botti Benevides, C. M. Fonseca, D. Porello, J. P. A. Almeida, T. Prince Sales, Ufo: Unified foundational ontology, Applied ontology 17 (2022) 167–210.

[11] S. Borgo, R. Ferrario, A. Gangemi, N. Guarino, C. Masolo, D. Porello, E. M. Sanfilippo, L. Vieu, Dolce: A descriptive ontology for linguistic and cognitive engineering, Applied ontology 17 (2022) 45–69.

[12] J. N. Otte, J. Beverley, A. Ruttenberg, Bfo: Basic formal ontology, Applied ontology 17 (2022) 17–43.

[13] C. H. Bernabé, N. Queralt-Rosinach, V. E. Silva Souza, L. O. Bonino da Silva Santos, B. Mons, A. Jacobsen, M. Roos, The use of foundational ontologies in biomedical research, Journal of Biomedical Semantics 14 (2023) 21.

[14] M. El Ghosh, H. Naja, H. Abdulrab, M. Khalil, Crimonto: A generalized domain-specific ontology for modeling procedural norms of the lebanese criminal law, Data & Knowledge Engineering 158 (2025) 102419.

[15] M. Grüninger, S. Shapiro, M. S. Fox, H. Weppner, Combining rfid with ontologies to create smart objects, International journal of production research 48 (2010) 2633–2654.

[16] A. A. Brandão, G. Loureiro, An overview of the bfo-basic formal ontology-and its applicability for satellite systems (2020).

[17] C. J. Kang, Y. S. Kang, Y. S. Lee, S. Noh, H. C. Kim, W. C. Lim, J. Kim, R. Hong, Process mining-based understanding and analysis of volvo it's incident and problem management processes., BPIC@ BPM 85 (2013).

[18] W. W. W. Consortium, et al., Owl 2 web ontology language document overview (2012).

[19] J. Z. Pan, Resource description framework, in: Handbook on ontologies, Springer, 2009, pp. 71–90.

[20] A. Dimou, M. Vander Sande, P. Colpaert, R. Verborgh, E. Mannens, R. Van de Walle, Rml: A generic language for integrated rdf mappings of heterogeneous data., Ldow 1184 (2014).

[21] T. Bagosi, D. Calvanese, J. Hardi, S. Komla-Ebri, D. Lanti, M. Rezk, M. Rodriguez-Muro, M. Slusnys,

G. Xiao, The ontop framework for ontology based data access, in: The Semantic Web and Web Science: 8th Chinese Conference, CSWS 2014, Wuhan, China, August 8-12, 2014, Revised Selected Papers 8, Springer, 2014, pp. 67–77.
[22] C. Pedrinaci, J. Domingue, Towards an ontology for process monitoring and mining, in: CEUR Workshop Proceedings, volume 251, 2007, pp. 76–87.
[23] Deliverable 1.1: Business process ontology framework, Project IST 026850 SUPER, 2007.
[24] M. Hepp, F. Leymann, J. Domingue, A. Wahler, D. Fensel, Semantic business process management: A vision towards using semantic web services for business process management, in: IEEE International Conference on e-Business Engineering (ICEBE'05), IEEE, 2005, pp. 535–540.
[25] G. Leonardi, M. Striani, S. Quaglini, A. Cavallini, S. Montani, Towards semantic process mining through knowledge-based trace abstraction, in: International Symposium on Data-Driven Process Discovery and Analysis, Springer, 2017, pp. 45–64.
[26] M. Katsumi, M. Grüninger, Using psl to extend and evaluate event ontologies, in: International Symposium on Rules and Rule Markup Languages for the Semantic Web, Springer, 2015, pp. 225–240.
[27] A. Botti Benevides, J.-R. Bourguet, G. Guizzardi, R. Peñaloza, J. P. A. Almeida, Representing a reference foundational ontology of events in sroiq, Applied Ontology 14 (2019) 293–334.
[28] R. Moher, M. Gruninger, Mining for meaning: Ontology-aware process mining methods through knowledge patterns, in: International Conference on Research Challenges in Information Science, Springer, 2025.
[29] M. Raasveldt, H. Mühleisen, Duckdb: an embeddable analytical database, in: Proceedings of the 2019 international conference on management of data, 2019, pp. 1981–1984.
[30] H. Jordan, B. Scholz, P. Subotić, Soufflé: On synthesis of program analyzers, in: International Conference on Computer Aided Verification, Springer, 2016, pp. 422–430.
[31] C. Di Ciccio, M. Montali, et al., Declarative process specifications: Reasoning, discovery, monitoring., Process mining handbook 448 (2022) 108–152.
[32] G. Bergami, F. M. Maggi, A. Marrella, M. Montali, Aligning data-aware declarative process models and event logs, in: Business Process Management: 19th International Conference, BPM 2021, Rome, Italy, September 06–10, 2021, Proceedings 19, Springer, 2021, pp. 235–251.
[33] D. Calvanese, T. E. Kalayci, M. Montali, S. Tinella, Ontology-based data access for extracting event logs from legacy data: the onprom tool and methodology, in: International Conference on Business Information Systems, Springer, 2017, pp. 220–236.
[34] D. Fahland, Process mining over multiple behavioral dimensions with event knowledge graphs, in: Process mining handbook, Springer, 2022, pp. 274–319.
[35] S. Khayatbashi, O. Hartig, A. Jalali, Transforming event knowledge graph to object-centric event logs: A comparative study for multi-dimensional process analysis, in: International Conference on Conceptual Modeling, Springer, 2023, pp. 220–238.
[36] G. De Giacomo, M. Y. Vardi, Linear temporal logic and linear dynamic logic on finite traces (2013).
[37] A. Küsters, W. M. van der Aalst, Oc-declare: Discovering object-centric declarative patterns with synchronization, in: International Conference on Business Process Management, Springer, 2025, pp. 162–179.
[38] Z. Yan, E. Cimpian, M. Zaremba, M. Mazzara, Bpmo: semantic business process modeling and wsmo extension, in: IEEE International Conference on Web Services (ICWS 2007), IEEE, 2007, pp. 1185–1186.