

Extraction of Object-Centric Event Logs through Virtual Knowledge Graphs

Extended Abstract

Jing Xiong¹, Guohui Xiao^{2,3}, Tahir Emre Kalayci⁴, Marco Montali^{1,3}, Zhenzhen Gu¹ and Diego Calvanese^{1,3,5}

¹KRDB Research Centre for Knowledge and Data, Free University of Bozen-Bolzano, 39100 Bolzano, Italy

²Department of Information Science and Media Studies, University of Bergen, 5007 Bergen, Norway

³Ontopic S.R.L., 39100 Bolzano, Italy

⁴Virtual Vehicle Research GmbH, 8010 Graz, Austria

⁵Department of Computing Science, Umeå University, 901 87 Umeå, Sweden

Abstract


Process mining is a family of techniques that supports the analysis of operational processes based on event logs. Among the existing event log formats, the IEEE standard eXtensible Event Stream (XES) is the most widely adopted. In XES, each event must be related to a single case object, which may lead to convergence and divergence problems. To solve such issues, object-centric approaches become promising, where objects are the central notion, and one event may refer to multiple objects. In particular, the *Object-Centric Event Logs* (OCEL) standard has been proposed recently. However, the crucial problem of extracting OCEL logs from external sources is still largely unexplored. In this paper, we try to fill this gap by leveraging the Virtual Knowledge Graph (VKG) approach to access data in relational databases. We have implemented this approach in the OnProm system, extending it from XES to OCEL support. The full version of this article has been submitted to an international conference.

Keywords


Process mining, object-centric event logs, virtual knowledge graph, ontology-based data access

1. Introduction

Process mining [1, 2] is a family of techniques relating the fields of data science and process management to support the analysis of operational processes based on event logs. To perform process mining, normally the algorithms and tools expect that the event logs are following certain standards. However, in reality, most IT systems in companies and organizations do not directly produce such logs, and the relevant information is spread in legacy systems, in particular, relational databases. Hence, event log extraction from legacy systems is a key enabler for process mining [3, 4, 5, 6].

 DL 2022: 35th International Workshop on Description Logics, August 7–10, 2022, Haifa, Israel

 jing.xiong@unibz.it (J. Xiong); guohui.xiao@uib.no (G. Xiao); emre.kalayci@v2c2.at (T. E. Kalayci); montali@inf.unibz.it (M. Montali); zhenzhen.gu@unibz.it (Z. Gu); calvanese@inf.unibz.it (D. Calvanese)

 0000-0002-3604-9645 (J. Xiong); 0000-0002-5115-4769 (G. Xiao); 0000-0001-6228-1221 (T. E. Kalayci); 0000-0002-8021-3430 (M. Montali); 0000-0002-7346-6093 (Z. Gu); 0000-0001-5174-9693 (D. Calvanese)



© 2022 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

 CEUR Workshop Proceedings (CEUR-WS.org)

There have been several proposals for the representation of event logs, e.g., eXtensible Event Stream (XES) [7], JSON Support for XES (JXES) [8], Open SQL Log Exchange (OpenSLEX) [9], and eXtensible Object-Centric (XOC) [10], where XES is the most adopted one, being the IEEE standard for interoperability in event logs [11]. In XES (and other similar proposals), each event is related to a single case object, which leads to problems with convergence (when an event is related to multiple cases and occurs repetitively) and divergence (when multiple events are in a single case and are hard to separate) [12]. To solve these issues, object-centric approaches become promising, where objects are the central notion, and one event may refer to multiple objects. In particular, along this direction, the *Object-Centric Event Logs* (OCEL) standard [13] has been proposed recently.

To the best of our knowledge, the crucial problem of extracting OCEL logs from external sources is still largely unexplored. The only exception is [14], where OCEL logs are extracted by identifying the so-called master and relevant tables in the underlying database and building a *Graph of Relations* (GoR). Though promising, this approach might be difficult to adopt when the underlying tables are complex and the GoR is hard to model, because it does not separate the storage level (i.e., the database) from the concept level (i.e., domain knowledge about events).

In this work, we try to fill this gap by leveraging the OnProm (<http://onprom.inf.unibz.it/>) framework [4, 5] for extracting event logs from legacy information systems. OnProm v1 was already relying on the technology of Virtual Knowledge Graphs (VKG) [15] to expose databases as Knowledge Graphs that conform to a conceptual model, and to query this conceptual model and eventually generate logs by using ontology and mapping-based query processing. It came with a toolchain to process the conceptual model, and to automatically extract XES event logs, by relying on the VKG system Ontop [16]. We present here OnProm v2, which we have modularized so that it becomes easier to extend, and in which we have implemented OCEL-specific features to extract OCEL logs.

2. The OnProm Framework for Event Log Extraction

We describe now the OnProm approach for event log extraction, as shown in Figure 1. To extract from a legacy *information system* $\mathcal{I} = \langle \mathcal{R}, \mathcal{D} \rangle$, with relational schema \mathcal{R} and database \mathcal{D} , event logs that conform to an event log standard X , OnProm works as follows:

- (1) A domain ontology is a high-level abstraction of business logic concerned in a domain of interest. The user can design a domain ontology \mathcal{T} using the standard ontology language OWL 2 QL using any ontology editing tool, e.g., the *Ontology Editor* of the OnProm tool chain. Then the user creates a VKG mapping \mathcal{M} (using, e.g., the Ontop plugin for Protégé [17]) to declare how the instances of classes and properties in \mathcal{T} are populated from \mathcal{I} . This step is only concerned with modeling the domain of interest and is agnostic to the event log standard.
- (2) OnProm assumes that for the event log standard X , a specific (domain-independent) event ontology \mathcal{E}_X is available. The *Annotation Editor* of OnProm imports \mathcal{E}_X , and allows the user to create annotations \mathcal{L}_X , which are based on the classes in \mathcal{E}_X , over the classes in \mathcal{T} .
- (3) OnProm assumes that for the standard X also a set of SPARQL queries for extracting the log information is defined. By relying on a conceptual schema transformation approach [6]

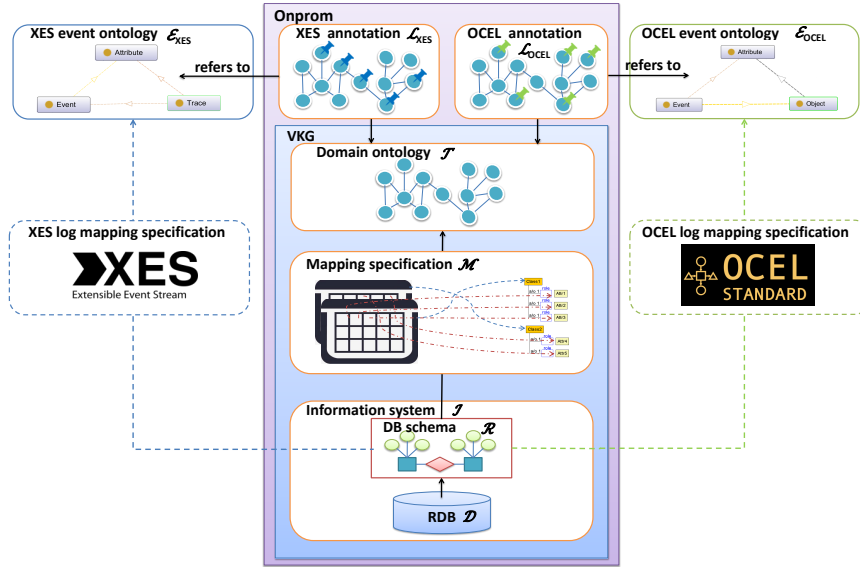


Figure 1: OnProm event log extraction framework

and query reformulation of Ontop, using \mathcal{L}_X , \mathcal{T} , \mathcal{M} , and \mathcal{R} , these SPARQL queries are internally translated to SQL queries over \mathcal{I} . OnProm evaluates the generated SQL queries to construct corresponding Java objects and serialize them into log files compliant with X . As mentioned, OnProm v1 only supported the XES standard. In this work, we have first modularized the system, by separating the above steps in different software components, so as to make it more extensible. Then we have introduced OCEL-specific features in Steps (2) and (3). Hence, OnProm v2 is now able to extract OCEL logs from relational databases.

Next we illustrate the functionality of OnProm for extracting OCEL logs through an example. The OCEL event ontology $\mathcal{E}_{\text{OCEL}}$ is a very simple ontology with only three classes: Object, Event, and Attribute. We consider Dolibarr [18] v14, a popular open source Enterprise Resource Planning (ERP) system. We have designed a *Sale Orders* domain ontology and the mapping in the Ontop system (Figure 2). We have then used the Annotation Editor of OnProm to annotate this ontology with $\mathcal{E}_{\text{OCEL}}$ classes (Figure 3). Based on the provided information, OnProm is able to extract OCEL logs automatically. Figure 4 shows a fragment of the extracted log in XML, and its graphical visualization.

Acknowledgments

This research has been supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation, by the Italian PRIN project HOPE, and by the EU H2020 project INODE, grant n. 863410.

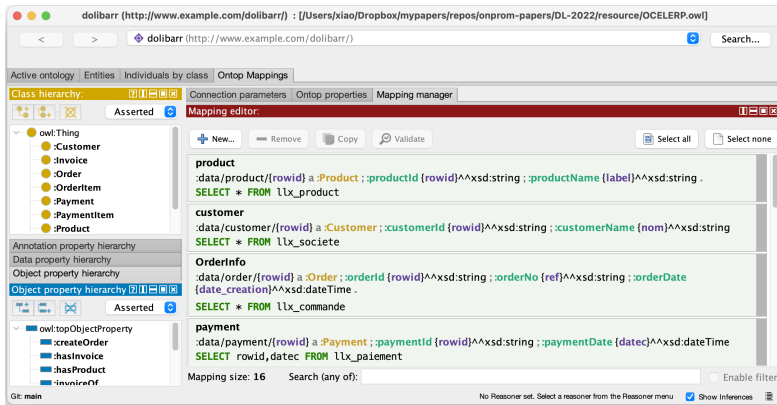


Figure 2: Ontology and mappings shown in Ontop

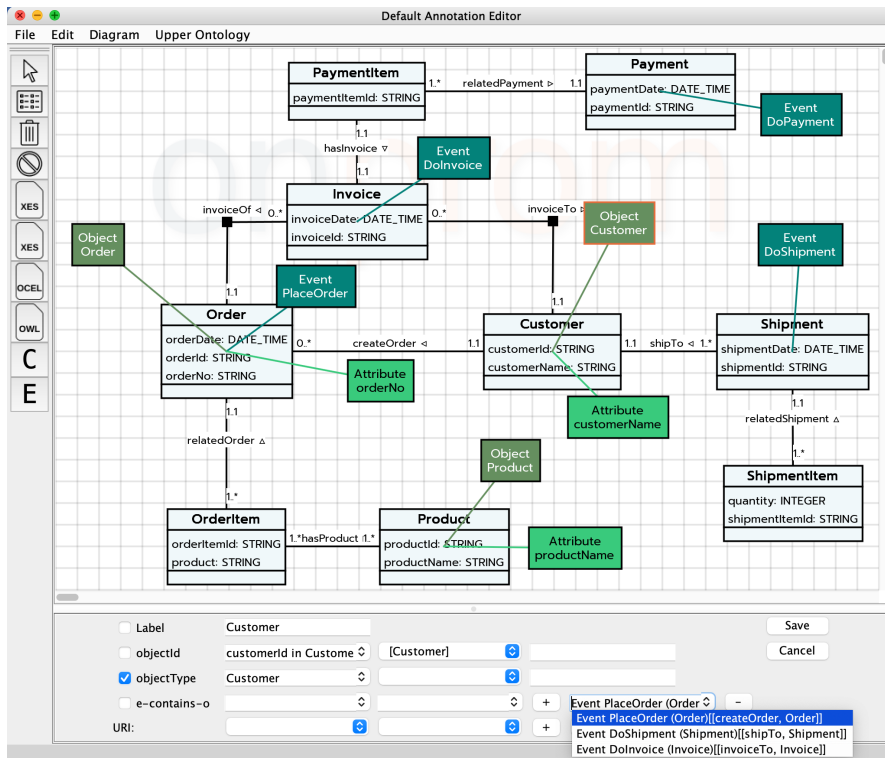


Figure 3: OnProm Annotation Editor showing the annotated ontology of the Dolibarr ERP system

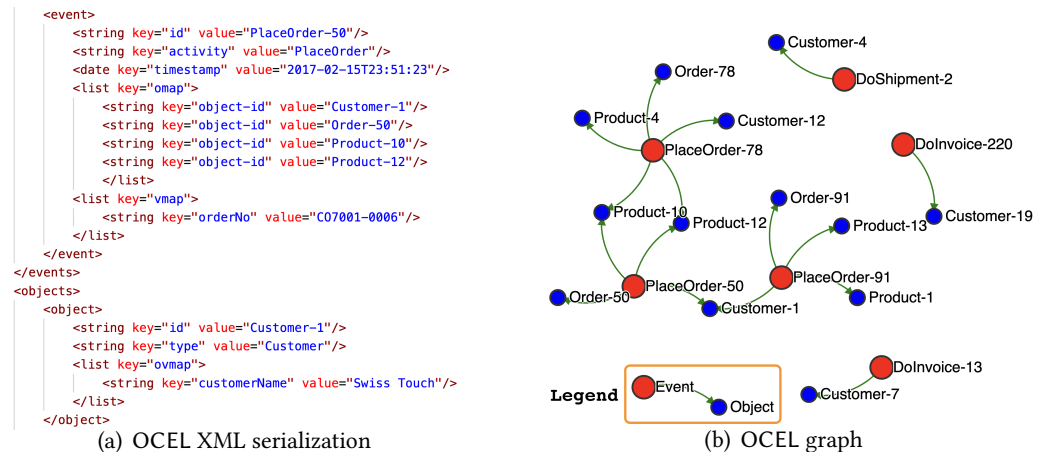


Figure 4: A fragment of the extracted OCEL log from the Dolibarr ERP system

References

- [1] W. M. P. van der Aalst, et al., Process Mining Manifesto, in: Proc. of the Business Process Management Workshops (BPM-WS), volume 99 of *Lecture Notes in Business Information Processing*, Springer, 2011, pp. 169–194. doi:10.1007/978-3-642-28108-2_19.
- [2] W. van der Aalst, *Process Mining – Data Science in Action*, 2nd ed., Springer, 2016.
- [3] D. Calvanese, M. Montali, A. Syamsiyah, W. M. P. van der Aalst, Ontology-driven extraction of event logs from relational databases, in: Proc. of the 11th Int. Workshop on Business Process Intelligence (BPI), volume 256 of *Lecture Notes in Business Information Processing*, Springer, 2016, pp. 140–153. doi:10.1007/978-3-319-42887-1_12.
- [4] D. Calvanese, T. E. Kalayci, M. Montali, A. Santoso, The onprom toolchain for extracting business process logs using ontology-based data access, in: Proc. of the BPM Demo Track and BPM Dissertation Award (BPM-D&DA), volume 1920 of *CEUR Workshop Proceedings*, CEUR-WS.org, 2017. URL: http://ceur-ws.org/Vol-1920/BPM_2017_paper_207.pdf.
- [5] 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: Proc. of the 20th Int. Conf. on Business Information Systems (BIS), volume 288 of *Lecture Notes in Business Information Processing*, Springer, 2017, pp. 220–236. doi:10.1007/978-3-319-59336-4_16.
- [6] D. Calvanese, T. E. Kalayci, M. Montali, A. Santoso, W. van der Aalst, Conceptual schema transformation in ontology-based data access, in: Proc. of the 21st Int. Conf. on Knowledge Engineering and Knowledge Management (EKAW), volume 11313 of *Lecture Notes in Computer Science*, Springer, 2018, pp. 50–67. doi:10.1007/978-3-030-03667-6_4.
- [7] H. M. W. Verbeek, J. C. A. M. Buijs, B. F. van Dongen, W. M. P. van der Aalst, XES, XESame, and ProM 6, in: *Information Systems Evolution: Selected Extended Papers of CAiSE Forum 2010*, volume 72 of *Lecture Notes in Business Information Processing*, Springer, 2010, pp. 60–75. doi:10.1007/978-3-642-17722-4_5.
- [8] M. B. Shankara Narayana, H. Khalifa, W. M. P. van der Aalst, JXES: JSON Support for the XES Event Log Standard, CoRR Technical Report arXiv:2009.06363, arXiv.org e-Print archive, 2020. URL: <https://arxiv.org/abs/2009.06363>.
- [9] E. G. L. de Murillas, H. A. Reijers, W. M. P. van der Aalst, Connecting databases with process mining: A meta model and toolset, *Software and System Modeling* 18 (2019) 1209–1247. doi:10.1007/s10270-018-0664-7.
- [10] G. Li, E. G. López de Murillas, R. M. de Carvalho, W. M. P. van der Aalst, Extracting object-centric event logs to support process mining on databases, in: J. Mendling, H. Mouratidis (Eds.), Proc. of CAiSE Forum, volume 317 of *Lecture Notes in Business Information Processing*, Springer, 2018, pp. 182–199. doi:10.1007/978-3-319-92901-9_16.
- [11] XES, 1849-2016 - IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams, IEEE Computer Society, 2016. doi:10.1109/IEEESTD.2016.7740858.
- [12] W. M. P. van der Aalst, Object-centric process mining: Dealing with divergence and convergence in event data, in: Proc. of the 17th Int. Conf. on Software Engineering and Formal Methods (SEFM), volume 11724 of *Lecture Notes in Computer Science*, Springer, 2019, pp. 3–25. doi:10.1007/978-3-030-30446-1_1.
- [13] A. F. Ghahfarokhi, G. Park, A. Berti, W. M. P. van der Aalst, OCEL: A standard for object-

- centric event logs, in: ADBIS 2021 Short Papers, Doctoral Consortium and Workshops: DOING, SIMPDA, MADEISD, MegaData, CAoNS, volume 1450 of *Communications in Computer and Information Science*, Springer, 2021, pp. 169–175.
- [14] A. Berti, G. Park, M. Rafiei, W. van der Aalst, An Event Data Extraction Approach from SAP ERP for Process Mining, CoRR Technical Report arXiv:2110.03467, arXiv.org e-Print archive, 2021. URL: <https://arxiv.org/abs/2110.03467>.
- [15] G. Xiao, L. Ding, B. Cogrel, D. Calvanese, Virtual Knowledge Graphs: An overview of systems and use cases, *Data Intelligence* 1 (2019) 201–223. doi:10.1162/dint_a_00011.
- [16] G. Xiao, D. Lanti, R. Kontchakov, S. Komla-Ebri, E. Güzel-Kalayci, L. Ding, J. Corman, B. Cogrel, D. Calvanese, E. Botoeva, The virtual knowledge graph system Ontop, in: Proc. of the 19th Int. Semantic Web Conf. (ISWC), volume 12507 of *Lecture Notes in Computer Science*, Springer, 2020, pp. 259–277. doi:10.1007/978-3-030-62466-8_17.
- [17] A. Poggi, M. Rodriguez-Muro, M. Ruzzi, Ontology-based database access with DIG-Mastro and the OBDA Plugin for Protégé, in: K. Clark, P. F. Patel-Schneider (Eds.), Proc. of the 4th Int. Workshop on OWL: Experiences and Directions (OWLED), 2008.
- [18] Dolibarr Open Source ERP CRM, Web suite for business, <https://www.dolibarr.org/>, 2021. (Last accessed on 5 May 2022).