Compatibility Challenges of the Current State-of-the-Art Provenance Tools*

Rudolf Wittner[1,2] [0000−0002−0003−2024] and Matúš Formánek[1] [0009−0009−3219−0631]

[1] Masaryk University, Žerotínovo nám. 617/9, 601 77 Brno, CZ
{wittner,formanekmato}@mail.muni.cz
[2] BBMRI-ERIC, Neue Stiftingtalstrasse 2/B/6, 8010 Graz, AU
rudolf.wittner@bbmri-eric.eu

Abstract. Provenance is information about the history of a described object. The current standard for provenance representation, W3C PROV, results from several years of efforts in the semantic web, linked data, computational workflows, databases, and other computer science-related communities. The standard is currently used as a groundwork for developing the ISO 23494 provenance standard series. During the development of the ISO standard, the PROV model's two major implementations – Prov Python and ProvToolbox – were used and found not fully compatible. This paper introduces the current standardization effort and related projects, describes issues encountered during the usage of the implementations, and discusses the potential causes and conclusions.

Keywords: provenance · standardization · Common Provenance Model · W3C PROV · ISO 23494.

1 Introduction

Provenance is a record that describes people, institutions, entities, and activities involved in producing, influencing, or delivering a piece of data or a thing [1]. Depending on the actual content of provenance, it can be used for various purposes, such as to assess the trustworthiness or quality of a described object [10].

Provenance information has been investigated by various computer science communities since the eighties [7]. Database provenance [2] aimed to explain the results of a database query. Workflows provenance [4] aimed to support reproducibility of computational workflows. Semantic web, linked data, and librarian communities developed multiple ontologies for provenance for various purposes (e.g., [3, 9]). The plethora of available provenance representations and ontologies motivated the researchers to understand the different representations used for provenance, the common aspects, and the reasons for differences. Consequently,

* This work has been supported by EU's Horizon Europe research and innovation programme under grant agreement No 101046203 (BY-COVID project), and under grant agreement No 101131701 (EvolveBBMRI project).

a consensus on the need for a common provenance standard emerged [7]. As a result, the W3C PROV standard [5] was developed.

W3C PROV is the current major standard that supports interoperable interchange of provenance information in heterogeneous environments such as the Web [5]. The standard's core is a conceptual data model, PROV-DM [1], representing provenance as a graph, where graph nodes represent entities, activities, or agents, and edges represent their relations. The data model is expressed in PROV Ontology (PROV-O [6]) using the OWL2 Web Ontology Language (OWL2).

One of the current research focuses in the provenance domain is to enable a unified traversal, processing, and analysis of distributed provenance chains [12] – sets of mutually interconnected provenance graphs, where each of the graphs is possibly stored and managed by a different organization. Such a provenance chain documents an object that traverses multiple organizations during its life cycle, so each can provide only a part of the documentation of the object's history.

The Common Provenance Model[3] (CPM) is a current extension of the PROV-DM that supports the creation of such distributed multi-organizational provenance chains. The CPM was developed as part of the EOSC-Life project[4] and is currently being adopted and refined in other European projects, namely BY-COVID[5], BIOINDUSTRY 4.0[6], or EvolveBBMRI[7]. The CPM serves as an open conceptual foundation for the *ISO 23494 Provenance information model for biological material and data* [11] provenance standard series, which is currently under development. However, despite more than ten years of presence of the accepted W3C PROV standard, the two major implementations of the PROV-DM – Prov library[8] for Python and ProvToolbox library[9] for Java – were found not fully compatible[10] during their adoption in the aforementioned projects.

2 Results

The ProvToolbox and Prov Python libraries enable Java/Python representation of PROV-DM and support conversions between various formats, such as PROV-O (RDF) or PROV-JSON. As the libraries implement the same standard for provenance representation, they are naturally expected to be compatible, meaning that we can serialize provenance using one library and deserialize it with the other. Despite the presence of proper compatibility tests of the libraries[11],

[3] https://commonprovenancemodel.org/
[4] https://www.eosc-life.eu/
[5] https://by-covid.org/
[6] https://bioindustry4.hub.inrae.fr/
[7] https://www.bbmri-eric.eu/scientific-collaboration/evolvebbmri/
[8] https://pypi.org/project/prov/
[9] https://lucmoreau.github.io/ProvToolbox/
[10] The term compatibility is used in the sense that one library can generate a provenance representation that can not be parsed or is misinterpreted by the other.
[11] https://github.com/openprov/interop-test-harness

several compatibility issues were found during the current standardization and adoption efforts, for instance:

1. Identifiers in PROV are qualified names (IRIs) that consist of a namespace and a local part. Both libraries enable serialization of provenance containing an identifier with a space in the local part of the identifier, but ProvToolbox can not deserialize such a document when it is serialized in PROV-N notation [8].
2. ProvToolbox expects that the "prov" and "xsd" namespaces are explicitly defined in the PROV-JSON serialization. However, according to PROV-JSON specification, the namespaces are implicit, which causes deserialization issues when a PROV-JSON file is serialized using the Prov Python library, which does not explicitly define the namespaces.
3. There is a difference in how the ProvToolbox and the Prov Python represent microseconds in timestamps. During deserialization between the implementations, both libraries can experience some loss of information.
4. If a PROV-JSON document contains "prefix.default" node, the ProvToolbox does not consider it as a default namespace but adds it to *regular* namespaces and adds an implicit default namespace, which negatively affects the interpretation of identifiers with the original default namespace.

A demonstration and descriptions of the complete list of the issues experienced are available at Github repository[12]. The issues were found between ProvToolbox version 2.0.2 and Prov Python version 2.0.0. The issues were reported to the authors of the libraries and have been fixed already.

3 Discussion and Conclusions

The presence of SW bugs is common, and fixing bugs is a standard part of a SW development process. Additionally, the PROV standard is relatively extensive, and it may be very difficult to capture all bugs and potential compatibility issues during the development of libraries, so it can be expected that some bugs emerge during the proper adoption of the tools. This is to say that we do not consider the issues encountered to be the fault of the libraries' authors. Based on these presumptions, we deduce that the implemented tools were probably used **in an isolated way**, so adopters of the libraries had no chance to encounter the aforementioned issues during their adoption. However, isolated usage of tools in heterogeneous environments, such as the Web, for which the underlying PROV standard is intended, can hardly be feasible.

In addition, as the provenance-related research in the last decade was mostly focused on prototyping new technologies, adoption of the PROV-DM in various domains, or demonstrating new research outcomes, minor bugs, which are otherwise critically important from the compatibility perspective, could have been ignored (e.g., the issue 4. in the list above). As a result, we encourage everyone

[12] https://github.com/mf-16/bakalarka

to report any bugs they encounter despite not preventing the usage of a particular tool for intended usage. Continuous responsible reporting of bugs could have accelerated the adoption of the tools and the PROV standard in the ongoing research and standardization efforts.

Acknowledgements. We want to acknowledge the authors of the provenance libraries, namely Luc Moreau and Trung Dong Huynh, who rapidly fixed reported bugs and provided insights into how the libraries are meant to be used.

References

1. Belhajjame, K., B'Far, R., Cheney, J., Coppens, S., Cresswell, S., Gil, Y., Groth, P., Klyne, G., Lebo, T., McCusker, J., Miles, S., Meyers, J., Sahoo, S., Tilmes, C.: PROV-DM: The PROV data model. W3C Recommendation (2013)
2. Buneman, P., Khanna, S., Wang-Chiew, T.: Why and where: A characterization of data provenance. In: Van den Bussche, J., Vianu, V. (eds.) Database Theory — ICDT 2001. pp. 316–330. Springer Berlin Heidelberg, Berlin, Heidelberg (2001)
3. da Silva, P.P., McGuinness, D.L., Fikes, R.: A proof markup language for semantic web services. Information Systems **31**(4), 381–395 (2006), https://doi.org/10.1016/j.is.2005.02.003, the Semantic Web and Web Services
4. Davidson, S.B., Freire, J.: Provenance and scientific workflows: challenges and opportunities. In: Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data. p. 1345–1350. SIGMOD '08, Association for Computing Machinery, New York, NY, USA (2008), https://doi.org/10.1145/1376616.1376772
5. Groth, P., Moreau, L.: PROV-overview. W3C Working Group Note (2013)
6. Lebo, T., Sahoo, S., McGuinness, D., Belhajjame, K., Cheney, J., Corsar, D., Garijo, D., Soiland-Reyes, S., Zednik, S., Zhao, J.: Prov-o: The prov ontology. W3C Recommendation (2013)
7. Moreau, L., Groth, P., Cheney, J., Lebo, T., Miles, S.: The rationale of prov. Journal of Web Semantics **35**, 235–257 (2015), https://doi.org/10.1016/j.websem.2015.04.001
8. Moreau, L., Missier, P., Cheney, J., Soiland-Reyes, S.: Prov-n: The provenance notation. W3C Recommendation (2013)
9. Sahoo, S.S., Sheth, A.P.: Provenir ontology: Towards a framework for escience provenance management (2009)
10. Simmhan, Y.L., Plale, B., Gannon, D.: A survey of data provenance in e-science. SIGMOD Rec. **34**(3), 31–36 (Sep 2005), https://doi.org/10.1145/1084805.1084812
11. Wittner, R., Holub, P., Mascia, C., Frexia, F., Müller, H., Plass, M., Allocca, C., Betsou, F., Burdett, T., Cancio, I., Chapman, A., Chapman, M., Courtot, M., Curcin, V., Eder, J., Elliot, M., Exter, K., Goble, C., Golebiewski, M., Kisler, B., Kremer, A., Leo, S., Lin-Gibson, S., Marsano, A., Mattavelli, M., Moore, J., Nakae, H., Perseil, I., Salman, A., Sluka, J., Soiland-Reyes, S., Strambio-De-Castillia, C., Sussman, M., Swedlow, J.R., Zatloukal, K., Geiger, J.: Toward a common standard for data and specimen provenance in life sciences. Learning Health Systems **8**(1), e10365 (2024), https://doi.org/10.1002/lrh2.10365
12. Wittner, R., Mascia, C., Gallo, M., Frexia, F., Müller, H., Plass, M., Geiger, J., Holub, P.: Lightweight distributed provenance model for complex real–world environments. Scientific Data **9**(1), 503 (Aug 2022), https://doi.org/10.1038/s41597-022-01537-6