Forgetting the past is not always the solution

How to keep previous versions of knowledge graphs available and queryable

Contact persons: Katja Hose (khose@cs.aau.dk)

Typically, semantic data sources, i.e., knowledge graphs[1], are considered static resources that never change. They are hosted by a triple store (potentially a SPARQL[2] endpoint with a Web-accessible interface) and can be queried using SPARQL. Sometimes also the combination of multiple SPARQL endpoints is needed to answer a particular query (federated query processing[3] and P2P[4]). However, we all know that knowledge is not static, some knowledge graphs contain errors introduced accidentally by automatic information extraction processes[5], and once in a while there is a major update of the entire knowledge graph, which so far simply replaces the previous version so that previous versions disappear and can no longer be accessed and queried. Hence, archiving solutions[6,7] have become available. Still, these solutions focus more on the support of storing and reconstructing previous versions. Efficient support of efficient processing of "time travel queries"[6,8], i.e., queries on (combinations of) previous versions of the knowledge graph, is still lacking.

Theme 1: Existing standards and solutions

While one way to approach the matter is to develop specialized archiving solutions, the other one is to try to keep multiple versions of a knowledge graph available with existing standards that can encode metadata about triples, i.e., in our case the metadata would be information about the version a triple belongs to. Standard methods include reification[9], singleton

---

[1] Aidan Hogan et al.: Knowledge Graphs. Morgan & Claypool Publishers, 2021, https://kgbook.org/
[2] https://www.w3.org/TR/sparql11-overview/
[3] Lars Heling, Maribel Acosta. Federated SPARQL Query Processing over Heterogeneous Linked Data Fragments. WWW 2022: 1047-1057, https://doi.org/10.1145/3485447.3511947
[4] Christian Aebeloe, Gabriela Montoya, Katja Hose. ColChain: Collaborative Linked Data Networks. WWW 2021: 1385-1396, https://doi.org/10.1145/3442381.3450037
[5] Let's exclude malicious intentions and assume that providers do not knowingly provide erroneous information with bad intentions.
[6] Olivier Pelgrin, Luis Galárraga, Katja Hose. Towards fully-fledged archiving for RDF datasets. Semantic Web 12(6): 903-925 (2021), https://doi.org/10.3233/SW-210434
[7] Ruben Taelman, Thibault Mahieu, Martin Vanbrabant, Ruben Verborgh. Optimizing storage of RDF archives using bidirectional delta chains. Semantic Web 13(4): 705-734 (2022), https://doi.org/10.3233/SW-210449
[8] Katja Hose. Knowledge Graph (R)Evolution and the Web of Data. MEPDaW@ISWC 2021: 1-7, http://ceur-ws.org/Vol-3225/paper1.pdf
[9] https://www.w3.org/wiki/RdfReification

properties[10], and named graphs[11], which all show different types of shortcomings[6,12,13]. However, recently RDF-star[14] was proposed, which enables nesting of triples. Hence, aspects that projects in this theme could target are, for example, evaluating the suitability of existing standards (in particular RDF-star, for which commercial engines have become available) for archiving and queries on versioned knowledge graphs, supporting combinations of multiple versioned knowledge graphs, developing a benchmark on a collected set of DBpedia Live changesets (available in our group), etc.

Theme 2: Time-travel queries

There are plenty of types of time-travel queries[6,15] ; some "simply" querying a previous version of a knowledge graph while others require comparisons of particular versions, finding versions where a query holds answers, etc. However, so far only Ostrich[7] is able to load a relatively large amount of data and query it. Hence, aspects that projects in this theme could target are, for example, developing alternative storage and querying techniques for efficient support of time travel queries, or in combination with aspects of Theme 1 looking into how existing standards for metadata encoding can support such queries, or investigating how time-travel queries (so far restricted to a single knowledge graphs) can support combinations of multiple versioned knowledge graphs, or developing efficient interactive solutions that help users formulate time travel queries and explore their results, etc.

[10] V. Nguyen, O. Bodenreider, A. Sheth,.Don't Like RDF Reification? Making Statements about Statements Using Singleton Property, in: TheWebConf, 2014, pp. 759–770

[11] J. J. Carroll, C. Bizer, P. Hayes, P. Stickler. Named graphs, Journal of Web Semantics (2005), pp. 247–267

[12] J. Frey, K. Müller, S. Hellmann, E. Rahm. M.-E. Vidal, Evaluation of Metadata Representations in RDF stores, Semantic Web (2019) 205–229.

[13] D. Hernández, L. Galárraga, K. Hose. Computing How-Provenance for SPARQL Queries via Query Rewriting, PVLDB (2021) 3389–3401

[14] https://w3c.github.io/rdf-star/cg-spec/editors_draft.html

[15] Javier D Fernández, Jürgen Umbrich, Axel Polleres, and Mag- nus Knuth. Evaluating Query and Storage Strategies for RDF Archives. ICSS 2016, pp. 41–48