Versionnement de graphe pour des données urbaines évolutives

Jey PUGET GIL, Emmanuel Coquery, John Samuel, Gilles Gesquière

Graph versioning for evolving urban data
Versionnement de graphe pour les données urbaines évolutives

Jey Puget Gil
jey.puget-gil@liris.cnrs.fr
Université de Lyon, Université Claude Bernard, LIRIS, UMR-CNRS 5205
Villeurbanne, FRANCE

Emmanuel Coquery
emmanuel.coquery@univ-lyon1.fr
Université Claude Bernard, LIRIS, UMR-CNRS 5205
Villeurbanne, FRANCE

John Samuel
john.samuel@cpe.fr
Université de Lyon, CPE Lyon, LIRIS, UMR-CNRS 5205
Villeurbanne, FRANCE

Gilles Gesquière
gilles.gesquiere@univ-lyon2.fr
Université de Lyon, Université Lumière Lyon 2, LIRIS, UMR-CNRS 5205
Villeurbanne, FRANCE

ABSTRACT

The continuous evolution of cities poses significant challenges in terms of managing and understanding their complex dynamics. With the increasing demand for transparency and the growing availability of open urban data, it has become important to ensure the reproducibility of scientific research and computations in urban planning. To understand past decisions and other possible scenarios, we require solutions that go beyond the management of urban knowledge graphs. In this work, we explore existing solutions and their limits and explain the need and possible approaches for querying across multiple graph versions.

RÉSUMÉ

L'évolution continue des villes pose des défis importants en termes de gestion et de compréhension de leurs dynamiques complexes. Avec la demande croissante de transparence et la disponibilité grandissante de données urbaines ouvertes, il est devenu important d'assurer la reproductibilité de la recherche scientifique et des calculs dans le domaine de l'urbanisme. Pour comprendre les décisions passées et d'autres scénarios possibles, nous avons besoin de solutions qui vont au-delà de la gestion des graphes de connaissances urbaines. Dans ce travail, nous explorons les solutions existantes et leurs limites, et expliquons le besoin et les approches possibles pour l'interrogation à travers de multiples versions de graphes.

CCS CONCEPTS

• **Information systems** → *Geographic information systems*; **Resource Description Framework (RDF)**; **Web Ontology Language (OWL)**.

KEYWORDS

RDF, versioning, graph, urban data, deduction

MOTS CLÉS

RDF, versionnement, graphe, données urbaines, déduction

1 INTRODUCTION AND MOTIVATION

Urban planners, historians, archaeologists, and researchers are continuously analyzing the constant development of cities. They are interested in an understanding of the possible versions and scenarios of the city [2], both in the past and in the future, if certain decisions were to be made. The choices made and the lessons learned in urban planning in the past serve as a guide for future decisions.

As the availability of open data increases across all sectors, so too does the demand for transparency in decision-making. Urban data come in different forms, they can be structured (sensors, building data, ...), semi-structured (urban system logs, ...) or unstructured (images, text, ...). Decisions are made on the basis of the data available at a given point in time. In other words, both the most recent version of the city and the previous versions are taken into account. In certain cases, complex calculations on this existing data guide the decision-makers. Reproducibility of these calculations is therefore also a requirement for transparency.

We provide the following sample queries from urban planning project proposals to better illustrate our research work:

- **Q1**: *Which city versions have a metro station accessible to people with disabilities?*
- **Q2**: *Across multiple concurrent city versions, what is the maximum known height of a particular building?* (aggregation)

Our previous research work proposed the use of graph formats [4] for the transformation and management of heterogeneous and concurrent urban data. In this work, we want to go beyond this and explore and develop a system that can query multiple versions of the graph simultaneously to answer complex queries like the ones above. This requires not only versioning of code (complex calculations) and data. It also requires efficient querying techniques across versions.

This article briefly reviews different ways to address the need for effective tools and methodologies to analyze urban development, emphasizing the importance of versioned data management.

2 STATE OF THE ART

Data and code evolution have been at the heart of many recent research and industrial advances. Taken together, they make up an important part of urban knowledge evolution. Given their growing use, our research is particularly focused on version control systems.

2.1 Code and Data versioning

Versioned repositories are systems that track and manage changes to data and code over time, allowing researchers to maintain a historical record of their work and facilitate collaboration. When these two concepts are integrated, they provide several benefits in scientific research, such as reproducibility, transparency, and collaboration. Version control allows different deductive paths to be explored and merged, facilitating collaboration among researchers with different expertise. Code versioning systems like GIT and SVN play a critical role in software development, enabling collaborative work, code reuse, and traceability. There also exist some dedicated solutions for versioning data such as DVC, DagsHub, Delta Lake, Dolt, Qri, Weights and Biases, Git LFS, Comet and LakeFS.

These different approaches are not suitable for our case study, since we want to work with concurrent versions and scenarios [2]. To answer even simple queries like **Q1**, current solutions require querying multiple database instances or checking out multiple version commits, which limits query response times.

2.2 Database versioning

A recent interesting solution called DoltHub, an online platform and hosting service provides version control for databases. This technology allows users to create, manage, and collaborate on databases using Git-style workflows and supports branching and merging, enabling teams to work on different features or versions of a database. However, cross-version querying of RDF data with such a solution is a challenge. Native SPARQL (SPARQL Protocol and RDF Query Language) queries are not inherently version-aware. Another solution called QuitStore [1] is an RDF data versioning system that addresses the need for efficient data retrieval across different versions. By implementing an RDF-based approach, QuitStore allows users to track changes and revisions to their semantic data over time. However, these versioning systems provide little support for cross-version queries. Indeed, they can either query metadata on multiple versions or query data on a single version.

Temporal databases, also known as historicized databases, are specialized databases that are designed to capture and store historical snapshots of data over a while. Some advanced temporal databases allow the analysis and querying of data at different points in time from two perspectives: how the data appeared in the real world and how it evolved within the database. However, by their nature, such databases are limited to a linear history and cannot be directly used to store a dataset with a branching history.

3 CONTRIBUTIONS PERSPECTIVES

Our motivation is to find a method for retrieving knowledge from a set of urban data versions stored in RDF format[4]. Resource Description Framework (RDF) offers a flexible and standardized format for representing the state of the city. RDF's graph-based structure allows the integration of diverse data sources, enabling a comprehensive view of the city's attributes and relationships. By versioning the city dataset, we can systematically track and record changes, modifications, and additions over time. This comprehensive version of history provides a foundation for analyzing the city's evolution, identifying trends, and extracting valuable knowledge. For example, if we have an urban dataset, we can identify the following problem: *How to analyze a set of versions of a city to produce additional knowledge?*

From a semantics point of view, a versioned graph can be assimilated to a collection of graphs, that is one graph for each version. Together with the `GRAPH` statement in SPARQL, this provides a way to query multiple versions at the same time. For example, the accessibility status of a given metro (**Q1**) or the height of the building (**Q2**). However separately storing each version would cost too much space and would probably lead to unefficient query processing.

Borrowing from historicized databases, one can associate version metadata to RDF triples. However, while a tuple in historicized database can be associated with a validity time interval, the branching nature of versioning history requires a different representation. Using provenance techniques [3] , this information could then be used at the query engine level to compute partial answers for several versions at once. We aim at implementing such a query engine and compare its efficiency with solutions using existing approaches with version checkout. We also aim at comparing the efficiency of different representations of version metadata associated to triples (for example representing the set of versions in extension or by a set of version intervals).

Note that this representation is independent from version metadata, we can thus reuse representation of version graph such as in [1] to trace the origin and lineage of data, for example to reference the code used to produce the data. It helps to understand its authenticity and assess its trustworthiness.

ACKNOWLEDGEMENTS

This work *Knowledge Hub for Evolving Urban Cities* is supported and funded by the IADoc@UDL (Université de Lyon, Université de Lyon 1) and LIRIS UMR 5205. We also acknowledge the BD team and the Virtual City Project[1] members for their invaluable advice and assistance.

REFERENCES

[1] Natanael Arndt. 2020. *Distributed Collaboration on Versioned Decentralized RDF Knowledge Bases.* Ph. D. Dissertation. Universität Leipzig. https://doi.org/10.33968/9783966270205-00

[2] John Samuel, Sylvie Servigne, and Gilles Gesquière. 2020. Representation of concurrent points of view of urban changes for city models. *Journal of Geographical Systems* (Feb. 2020). https://doi.org/10.1007/s10109-020-00319-1

[3] Leslie F. Sikos and Dean Philp. 2020. Provenance-Aware Knowledge Representation: A Survey of Data Models and Contextualized Knowledge Graphs. *Data Science and Engineering* 5, 3 (Sept. 2020), 293–316.

[4] Diego Vinasco-Alvarez, John Samuel, Sylvie Servigne, and Gilles Gesquière. 2021. Towards a semantic web representation from a 3D geospatial urban data model. In *SAGEO 2021, 16ème Conférence Internationale de la Géomatique, de l'Analyse Spatiale et des Sciences de l'Information Géographique.* 227–238. https://hal.science/hal-03240567/file/SAGEO_2021.pdf

Received June 7, 2023; revised August 21, 2023

[1] https://projet.liris.cnrs.fr/vcity/