The Journal of Open Source Software

Discovery: Empowering Access and Reusability of RDF Graphs with a Programming Query Builder

Olivier Filangi [1], **Nils Paulhe** [2], **Clément Frainay** [3], **and Franck Giacomoni** [2]

1 IGEPP, INRAE, Institut Agro, Université de Rennes, Domaine de la Motte, Le Rheu 35653, France **2** Université Clermont Auvergne, INRAE, UNH, Plateforme d'Exploration du Métabolisme, MetaboHUB Clermont, Clermont-Ferrand, France **3** Toxalim (Research Center in Food Toxicology), Université de Toulouse, INRAE, ENVT, INP-Purpan, UPS, Toulouse 31300, France

DOI: 10.xxxxxx/draft

Software

- Review
- Repository
- Archive

Editor: Open Journals

Reviewers:

- @openjournals

Submitted: 01 January 1970
Published: unpublished

Summary

Linked data is increasingly available on the web and has been widely adopted by the bioinformatics community. However, it is not common to find APIs that enable the direct use of semantic information in web interfaces. This often leads web application designers to incorporate this information into relational databases, as they can benefit from the query builder and object-relational mapping features that are widely used in this community.

We have developed Discovery, a free software library designed to easily build intuitive and interactive user interfaces to exploit RDF data in graphical form. The API provides a dedicated query language to create and maintain complex queries to be used in a client or server side web development environment. We used Discovery to implement functionality in web decision support applications within the MetaboHUB consortium (French national Metabolomics and Fluxomics infrastructure) : FORUM(Delmas et al., 2021) (Metabolism Knowledge Network Portal) and PeakForest(Paulhe et al., 2022) (The Metabolomics spectral database web portal).

Statement of need

MetaboHUB is a French national infrastructure dedicated to research in metabolomics and fluxomics, with the aim of providing an integrated platform for the study of metabolic pathways and networks. This initiative brings together a wide range of academic and industrial partners, including experts in analytical chemistry and bioinformatics, to develop cutting-edge technologies and methodologies for metabolomics research. One of the key objectives of MetaboHUB is to ensure data and software interoperability within the consortium. In this context, our working group "Creating FAIR resources for knowledge mining" aims to organize data and metadata in Resource Description Framework (RDF) format, which is a graph-based representation format for data publishing and interchange on the Web developed by the W3C. Additionally, we seek to structure consortium software products into web components, enabling better reuse and integration of resources within the scientific community.

Presently, this has led to the establishment of a specialized infrastructure aimed at harnessing knowledge bases. Within these resources, we provide the metabolic community access to a knowledge graph that delineates connections between chemical compounds and the scientific literature(Delmas et al., 2021). Additionally, we have introduced an expanded knowledge graph using a Bayesian framework, encompassing overlooked metabolites lacking annotated literature(Delmas et al., 2023).

The Journal of Open Source Software

Bioinformatics Linked Open Data

Nowaday, the use of semantic web technologies into bioinformatics has become ubiquitous across all domains of life sciences(Wu & Yamaguchi, 2014). Many bioinformatics resources is now organized according to the FAIR (Findable, Accessible, Interoperable, and Reusable) principles(Wilkinson et al., 2016), enabling efficient management and reuse of data in both research and industrial settings. This implementation was made possible by the standardized languages and protocols defined by the World Wide Web Consortium (W3C) such as the RDF which provides a versatile framework for representing data and knowledge in a machine-readable format and the SPARQL query language to exploit these data known as knowledge graphs.

Bioinformatics communities are encouraged to develop ontologies that adhere to the principles of the Basic Formal Ontology(Otte et al., 2022) and the Open Biological and Biomedical Ontology Foundry(Otte et al., 2022). These ontologies aim to structure the modelling of knowledge in a common conceptual framework and allow the reuse of existing ontologies, favouring collaboration between different research communities. The datasets, now structured, use controlled vocabularies and taxonomies to use unambiguous standard terms.

Effective tools (BioPortal(Noy et al., 2009), EMBL-EBI Ontology Lookup Service(Côté et al., 2006) and AgroPortal(Jonquet et al., 2018)) exist to access ontologies and datasets. In addition, these resources can be imported into RDF data store, also known as triplet store, to be exploited using the SPARQL query language. In conclusion, semantic web technologies have greatly facilitated the integration and exploitation of bioinformatics data, allowing the efficient management of large and complex datasets.

Overview of the General Design

Discovery enables the development and maintenance of sophisticated SPARQL queries within a web application. The library provides a component for configuring access to an RDF data source and a core building component called Query Builder (QB) for incrementally constructing queries, which are translated into SPARQL queries at the time of result retrieval. Additionally, Discovery incorporates a component for processing the results of the query generated by the QB. These components are serializable, facilitating the seamless transport of the query construction state within a web application. This serialization allows user interfaces to capture and integrate new elements specific to their functionality, ensuring flexibility and adaptability.

The library relies on the manipulation of immutable data structures, a fundamental tenet of functional programming. Once created, these structures persist unaltered throughout the application's execution, providing advantages such as improved code clarity and the avoidance of unintended side effects. Developers can effortlessly construct intricate SPARQL queries by combining merging immutable query fragments. This immutability is crucial for reducing bugs linked to unforeseen alterations in object state, thereby simplifying long-term code maintenance.

The Discovery API utilizes the Scala.js compiler to ensure compatibility with established JavaScript libraries, a critical aspect in the realm of web development. This functionality facilitates the smooth assimilation of widely-used JavaScript libraries, allowing for tasks like DOM manipulation and other UI-related functions within web components.

Furthermore, Scala.js produces optimized JavaScript code, a critical consideration in web applications where responsiveness and a seamless user interface are imperative. The synergy between functional programming in Scala and transpilation through Scala.js facilitates the manipulation of a high-level API, enabling developers to focus exclusively on the concepts dedicated to the construction of a query in the end.

We extensively leverage the open-source framework Comunica(Taelman et al., 2018), a knowledge graph querying framework for JavaScript that provides flexibility in using SPARQL and GraphQL over decentralized RDF on the Web. This utilization aims to efficiently handle RDF

data access and SPARQL query processing. Discovery's maintenance focus is directed toward the development and upkeep of Scala.js facades, abstracting away complexities associated with RDF manipulation.

Key Features

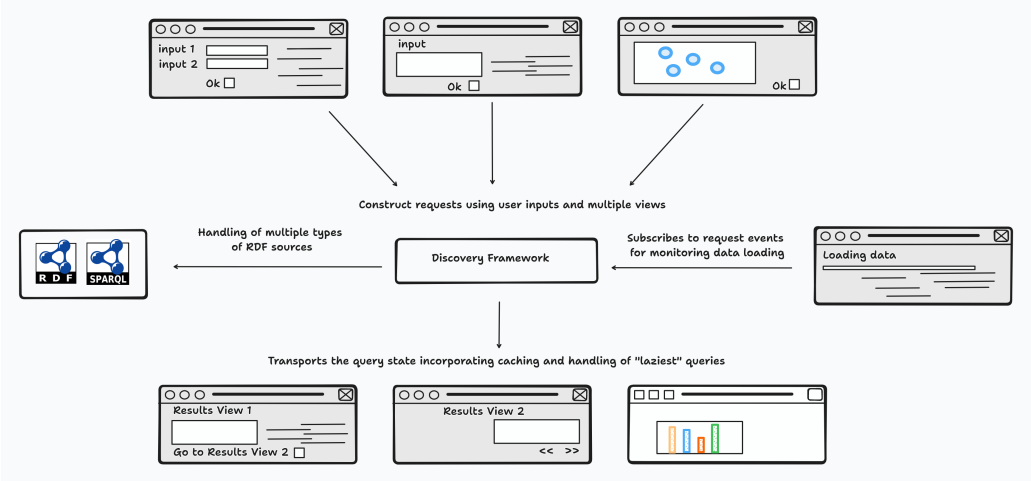

Figure 1: Interplay between the Discovery API and Web Components

Simplified Configuration and Versatile Access to RDF Resources

Elementary Building Blocks

A distinctive quality of the QB module is the categorization of construction elements, such as resources and qualifiers. Immutability is deliberately imposed, fortifying the security of the development process and simplifying debugging. This intentional structure promotes stability in query creation, a critical factor for precise and error-free development. Discovery allows the creation of an object that holds the construction state of a query. Subsequently, it enables the incremental addition of new elements based on this object, using a focus to contextualize the integration of a new element. At each construction step, a new instance of the QB module is instantiated, preserving all the stages of query construction.

Data Flow Management and Pagination

Addressing scalability concerns, the QB module incorporates intelligent pagination, particularly beneficial when crafting result lists with a significant number of elements. This optimization ensures the efficiency of queries and responses, enhancing the overall performance.

Les Datatypes properties (les attributs de type datatype des resources) sont traités autrements afin d'obtenir des performances

Request Transport via String Serialization in a Web Architecture

Tailored for web development, Discovery's Query Builder introduces features such as string transport, simplifying component communication. Additionally, developers can enhance user queries by embedding decoration metadata, providing contextual information within graphical representations for a more enriched user experience.

Event Management for Dialog Box Notifications and User Interactions

Asynchronous Results and Error Handling

The QB module places a premium on asynchronous result reception, ensuring the responsiveness of web applications. Developers can subscribe to events, staying abreast of specific interactions or changes and fostering a dynamic and interactive web development environment.

In essence, Discovery, as the Query Builder, serves as a pivotal guide in the intricate realm of SPARQL query generation. From streamlined configuration to categorization, scalability, and web-specific functionalities, this module empowers developers to navigate the complexities of web-based RDF data manipulation with precision and efficiency.

Illustrative Outcomes

The FORUM Metabolism Knowledge Network Portal and PeakForest (The Metabolomics spectral database web portal)

Acknowledgements

References

Côté, R. G., Jones, P., Apweiler, R., & Hermjakob, H. (2006). The Ontology Lookup Service, a lightweight cross-platform tool for controlled vocabulary queries. *BMC Bioinformatics*, *7*, 97. https://doi.org/10.1186/1471-2105-7-97

Delmas, M., Filangi, O., Duperier, C., Paulhe, N., Vinson, F., Rodriguez-Mier, P., Giacomoni, F., Jourdan, F., & Frainay, C. (2023). Suggesting disease associations for overlooked metabolites using literature from metabolic neighbors. *GigaScience*, *12*, giad065. https://doi.org/10.1093/gigascience/giad065

Delmas, M., Filangi, O., Paulhe, N., Vinson, F., Duperier, C., Garrier, W., Saunier, P.-E., Pitarch, Y., Jourdan, F., Giacomoni, F., & Frainay, C. (2021). FORUM: Building a knowledge graph from public databases and scientific literature to extract associations between chemicals and diseases. *Bioinformatics*, *37*(21), 3896–3904. https://doi.org/10.1093/bioinformatics/btab627

Jonquet, C., Toulet, A., Arnaud, E., Aubin, S., Dzale-Yeumo, E., Emonet, V., Graybeal, J., Laporte, M.-A., Musen, M. A., Pesce, V., & Larmande, P. (2018). AgroPortal: A vocabulary and ontology repository for agronomy. *Computers and Electronics in Agriculture*. https://doi.org/10.1016/j.compag.2017.10.012

Noy, N. F., Shah, N. H., Whetzel, P. L., Dai, B., Dorf, M., Griffith, N., Jonquet, C., Rubin, D. L., Storey, M.-A., Chute, C. G., & Musen, M. A. (2009). BioPortal: Ontologies and integrated data resources at the click of a mouse. *Nucleic Acids Research*, *37*(Web Server issue), W170–173. https://doi.org/10.1093/nar/gkp440

Otte, J. N., Beverley, J., Ruttenberg, A., Borgo, S., Galton, A., & Kutz, O. (2022). BFO: Basic formal Ontology1. *Appl. Ontol.*, *17*(1), 17–43. https://doi.org/10.3233/AO-220262

Paulhe, N., Canlet, C., Damont, A., Peyriga, L., Durand, S., Deborde, C., Alves, S., Bernillon, S., Berton, T., Bir, R., Bouville, A., Cahoreau, E., Centeno, D., Costantino, R., Debrauwer, L., Delabrière, A., Duperier, C., Emery, S., Flandin, A., … Giacomoni, F. (2022). PeakForest: A multi-platform digital infrastructure for interoper-

able metabolite spectral data and metadata management. *Metabolomics*, *18*(6), 40. https://doi.org/10.1007/s11306-022-01899-3

Taelman, R., Van Herwegen, J., Vander Sande, M., & Verborgh, R. (2018, October). Comunica: A modular SPARQL query engine for the web. *Proceedings of the 17th International Semantic Web Conference*. https://comunica.github.io/Article-ISWC2018-Resource/

Wilkinson, M. D., Dumontier, M., Aalbersberg, Ij. J., Appleton, G., Axton, M., Baak, A., Blomberg, N., Boiten, J.-W., Silva Santos, L. B. da, Bourne, P. E., Bouwman, J., Brookes, A. J., Clark, T., Crosas, M., Dillo, I., Dumon, O., Edmunds, S., Evelo, C. T., Finkers, R., … Mons, B. (2016). The FAIR guiding principles for scientific data management and stewardship. *Scientific Data*, *3*(1), 160018. https://doi.org/10.1038/sdata.2016.18

Wu, H., & Yamaguchi, A. (2014). Semantic web technologies for the big data in life sciences. *Bioscience Trends*, *8*(4), 192–201. https://doi.org/10.5582/bst.2014.01048