Ontology-based Dataset Discovery in the BUILDSPACE Data Management Platform

Iason Sotiropoulos[1], Ioannis Karvelas[1], Stamatia Rizou[1], Vangelis Marinakis[2] and Edlira Vakaj[3]

[1]SingularLogic, Cholargos, Greece
[2]National Technical University of Athens, Athens, Greece
[3]Birmingham City University, Birmingham, UK

Abstract

This position paper demonstrates the approach followed to implement and extend the BUILDSPACE Core Platform, a data management platform focusing on buildings and implemented in the context of the BUILDSPACE Horizon Europe project, to allow for data discovery and ontology-based metadata tagging. The paper provides the architectural overview of the platform, including technical details for its implemented components, and the conceptual overview for the Discoverability Layer that will enable the efficient discovery of datasets in the building domain by allowing data providers to tag their data. The demonstrated conceptual and technical approaches provide a novel interchange between metadata tagging and data discovery mechanisms through an ontology-based framework targeted towards the energy efficiency for buildings domain.

Keywords

data discovery, data integration, data handling, ontology, semantic technology, software architecture

1. Introduction

Data-driven services and operations related to data receive substantial focus in multiple domains. Data are not only regarded as a resource for deriving conclusions on scientifically studied phenomena, but as a valued asset that allows the extraction of knowledge for predicting outcomes (e.g., human behavior [1]). From simple data-driven inferences in research experiments to large-scale Machine Learning operations on Big Data in large enterprises, the role of data in modern technological societies is central (to the point where the terms "knowledge" and "data" can be used interchangeably sometimes [1]).

Data platforms is an essential part of workflows that include data processing mechanisms [2]. Depending on the workflow requirements, the supported functionalities as well as the overall architecture of a data platform can vary. Design decisions can alter the way the data reside within (or outside) the platform using centralized or decentralized approaches [3], address interoperability requirements [4], and enable or even enhance the discovery of the data by the user [5]. The present position paper demonstrates an ontology-based approach to the development of a data platform that incorporates mechanisms for data storage and management, interoperability, and discovery. Our intention is to build a semantic discovery system that allows users to discover data of interest by search and filtering. We are currently utilizing the building efficiency domain as a testbed for initial experimentation. The novelty of this work stems from the application of semantic approaches in heterogeneous data assets in the built environment.

NLP4KGC: 3rd International Workshop on Natural Language Processing for Knowledge Graph Creation, September 17, 2024, Amsterdam, Netherlands.

isotiropoulos@singularlogic.eu (I. Sotiropoulos); ikarvelas@singularlogic.eu (I. Karvelas); srizou@singularlogic.eu (S. Rizou); vmarinakis@epu.ntua.gr (V. Marinakis); edlira.vakaj@bcu.ac.uk (E. Vakaj)
0000-0001-6157-1593 (I. Sotiropoulos); 0000-0003-2502-2903 (I. Karvelas); 0000-0002-3683-060X (S. Rizou); 0000-0001-5488-4006 (V. Marinakis); 0000-0002-0712-0959 (E. Vakaj)

1.1. Data Interoperability and Ontologies

With the advent of advanced data processing and analytics strategies, along with the constantly increasing availability of powerful hardware, solutions that target more complex real-world problems emerge. Such complex data analytics services very often rely on heterogeneous data provisioning - multiple sources of data that provide different kinds of information about a specific domain. In these cases, a formally structured representation of the domain data, along with a uniform conceptual model for the interpretation of the labels situated within the data is crucial [6]. Furthermore, there are cases where inter-platform data integration is required, even with data that conceptually reside in diverse domains. Solutions to the challenges in the interoperability of systems have been proposed in multiple fields and through a plethora of different paradigms [7]. A prominent approach to address interoperability-related issues is the use of ontology-based mapping and tagging of the data [8]. Briefly, ontologies are formalized sets of associations between different concepts within a specific domain [9]. By ensuring global data compliance with a given ontology, interoperability across different data sources can be achieved by design, at least at a conceptual level. Heterogeneous data sources within a data platform, especially when the data target a specific problem-domain, necessitates consistency within the data at multiple levels: (1) the syntactic level which refers to how the data files are formatted and how the data is structured within the files, and (2) the semantic level which refers to the conceptual interpretation (i.e., the meaning) of the labels within the data.

1.2. Data Discovery

The availability of data along with their capacity to be easily discovered on the Web are central factors in data services pipelines [10]. There are two distinct aspects when considering the availability of a data source on the Web from an end-user perspective: (1) the manner through which the data get published by the data providers, and (2) the manner which data consumers discover the data on the Web. Enriching data by providing additional information (i.e., metadata tagging) [11] is a way to enable data discovery through mechanisms that transform the provided information and make it available for searching and filtering.

2. Methodology

The architectural design of the Core Platform addresses the challenges presented in Section 1, namely data interoperability and discovery, on top of the complexities of contemporary data storage and management. The platform was initially designed and implemented in the context of the BUILDSPACE Horizon Europe project. [1] The Core Platform is designed to ensure a secure and organization-centric approach to data sovereignty, in addition to an ontological-based discoverability of the data. The platform follows a four-tier layering architecture, and it is presented in Figure 1. The conceptual implementation along with several technical components are described in Sections 2.1- 2.4.

2.1. Identity Management Layer

The Identity Management Layer establishes a centralized point for user identity management, by performing both user identification (User Authentication) and verification of the content that users are authorized to access (Group Authorization). Group Authorization ensures that all users within the same organization have access to the same data. Furthermore, this layer allows all services and users within the BUILDSPACE platform's ecosystem to integrate, leveraging robust authentication and authorization functionalities while promoting a consistent and secure user experience across the entire ecosystem.

User authentication is achieved through OpenID Connect (OIDC) [12], [2] providing a secure and standardized method for users to log in using their email and password. OIDC issues identity tokens in

[1] https://buildspaceproject.eu/
[2] https://openid.net/connect/

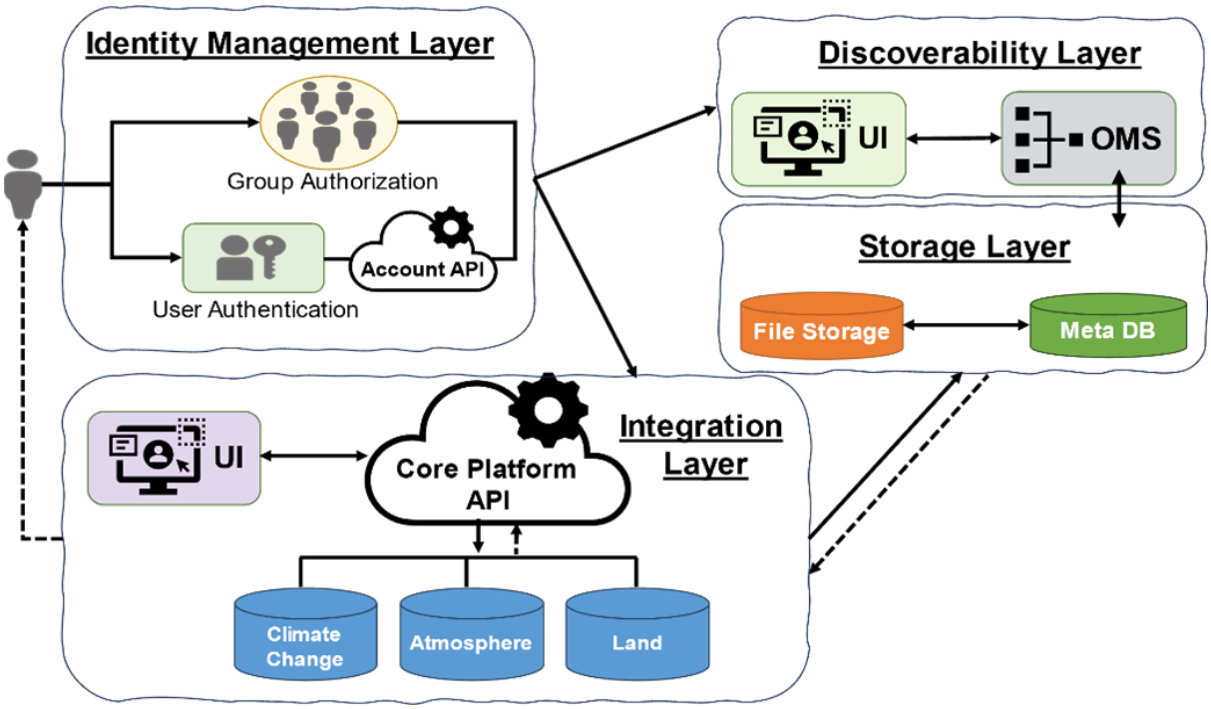

Figure 1: Visualization of the BUILDSPACE Core Platform's high-level architecture.

JSON Web Token (JWT) format, ensuring the secure transmission of user information. Leveraging the OIDC standardized scopes to define access levels, the system allows users that share a common group affiliation (e.g., an organization) to also share access privileges, and thus streamlines data sharing among group members. In terms of deployment, the open-source free software Keycloak [13] [3] was selected. Acting as an administrative gateway, a minimal generic REpresentational State Transfer (REST) API, developed with OpenAPI standards [14] [4] in Python (v3.8.6) [12], facilitates organizational and user management tasks. [5]

2.2. Storage Layer

The Storage Layer consists of two components and is designed to ensure efficient and secure data management. The first component is a S3-compatible file system, [6] MinIO, [7] which provides a scalable and versatile solution for the actual file storage. MinIO ensures scalability, reliability, and high availability, aligning with the platform's commitment to data integrity. This is achieved by utilizing parallelization techniques and horizontal scalability, as well as various security features, including encryption in transit and at rest, and the support of secure communication protocols. Notably, the architecture prioritizes data privacy, with files stored as multiple parts, in an anonymized fashion to uphold confidentiality and security standards. The second component is a NoSQL database (MongoDB) that facilitates the metadata storage. MongoDB's schema-less approach accommodates various metadata structures efficiently, adapting to the evolving needs of the platform. In addition, its advanced query capabilities, coupled with the organization-centric approach, empower users to perform detailed searches based on metadata attributes [13].

[3] https://www.keycloak.org
[4] https://www.openapis.org
[5] The API is open-source and already available at: https://github.com/PROJECT-BUILDSPACE/account-manager-api. The Swagger documentation can be found at: https://account-buildspace.euinno.eu/swagger/.
[6] https://aws.amazon.com/s3/
[7] https://min.io

2.3. Integration Layer

Serving as the backbone of the Core Platform, a publicly exposed REST API (Core API) that follows OpenAPI standards orchestrates seamless communication and interaction across all essential functionalities. Built upon a foundation of standardized HTTP requests, the Core API manages the Core Platform's operations, including facilitation of the file management, creating MinIO buckets, establishing folders for organized content, and data sharing. The Core API interconnects with the other layers, ensuring a seamless flow of data. Files are stored with anonymity in MinIO, metadata are efficiently managed in MongoDB, while user access is authenticated and authorized through Keycloak, and data can potentially be discoverable easily. Furthermore, the Core API currently serves as the integrator of three Copernicus APIs related to Land, Climate Change, and Atmosphere services.

To conclude, the REST API serves as a crucial intermediary for all services within the BUILDSPACE platform's ecosystem, providing a standardized interface for data retrieval and manipulation. As the primary point of integration, it ensures that various services efficiently consume and contribute to the shared data ecosystem, providing a unified approach to data management and collaboration. In future iterations, the API will be complemented with an intuitive User Interface (UI) promoting a user-friendly experience.

2.4. Discoverability Layer

In the BUILDSPACE project's context, a user would be able to identify data on buildings that have specific structural properties (e.g., number of floors, specific floor areas, height, location coordinates, etc.). The Discoverability Layer is designed to enhance the visibility and accessibility of datasets within the system. This layer integrates a UI, supported by a robust metadata management framework, to ensure efficient and precise data discovery. A conceptual overview of this layer is presented in Figure 2.

The Discoverability Layer encompasses a comprehensive UI where users can fill out a form with metadata about their datasets. This form is designed with predefined fields, informed by a network of ontologies, which ensure semantic interoperability and data discoverability. Upon submission of the metadata form, the front-end application converts the input data into on ontology instance which is sent to the Ontology Management System (OMS) for storage and management. We have selected Apache Jena Fuseki as our OMS due to its powerful capabilities in handling TTL data and SPARQL queries. Fuseki offers robust support for semantic web technologies, enabling efficient storage, retrieval, and querying of the ontology instances. Its scalability and performance make it an ideal choice for managing the complex metadata structures that our platform requires [15].

In addition to metadata submission, the Discoverability Layer provides a search interface where users can fill out a search form. This form, also based on our network of ontologies, allows users to specify detailed criteria for their search. The UI translates the search inputs into a SPARQL query, which is then processed by the OMS. The result consists of a set of IRIs corresponding to the dataset IDs that match the search criteria. These IRIs provide references to datasets across the entire system, allowing users to discover not only their own data but also datasets from other organizations. This cross-organizational discoverability is a key feature of the platform, promoting data sharing and collaboration. In terms of the ontology that will be incorporated, multiple existing ontologies in the building domain, such as the Building Ontology Topology (BOT), [8] the Building Ontology (BO), [9] and the Brick Ontology, [10] will be utilized and extended in order to capture the specific requirements for the BUILDSPACE project.

3. Discussion and Future Work

In this paper, the conceptual overview of the BUILDSPACE Core Data Platform was presented, along with details on its technical implementation. The Identity Management, Storage, and Integration layers

[8] https://w3id.org/bot#
[9] http://ontology.eil.utoronto.ca/icity/Building#
[10] https://brickschema.org/ontology/1.3

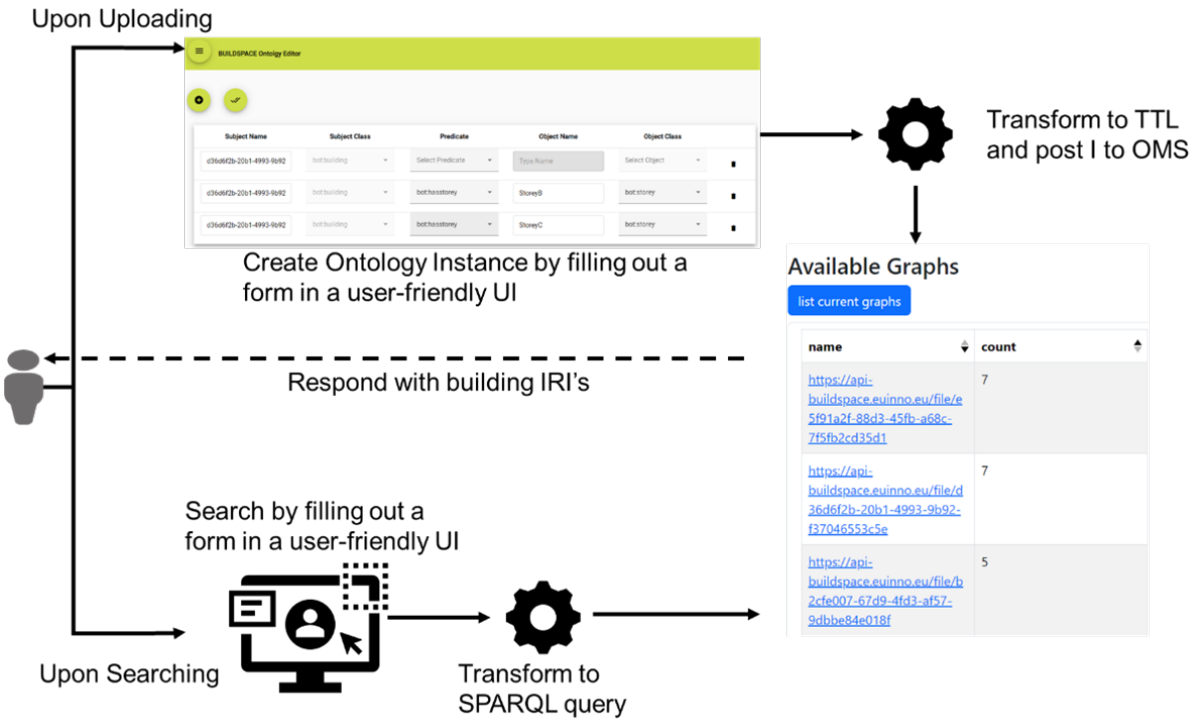

Figure 2: Discoverability Layer conceptual architecture.

are already implemented (except for the front-end component of the Integration layer), while the Discoverability layer is currently in the design phase, with the implementation currently ongoing. The Discoverability layer (see Section 2.4) aims to allow users to provide metadata in the form of semantic triples. The intended UI will provide a dedicated view that will allow users to select concepts, their properties and relations from a drop-down menu, and provide names and values to create instances of the ontology that will be used to populate the metadata. The ontology will be designed using consolidated ontology engineering methodologies and following best practices in reusing existing ontologies related to the building energy efficiency domain. The semantic model will cover aspects that describe the geometry of the building, along with more general information (e.g., building type, location, etc).

The demonstrated design approach provides two main benefits: (1) efficiency in identifying suitable data on the subject domain, and (2) semantic interoperability with data in compliance with existing ontologies. Beyond the implementation and the testing of the platform, our future work will focus on expanding the platform using IDS-compliant conceptual models and building blocks, while also focusing on more abstract implementations and working towards solutions that enable the domain-agnostic integration of the platform.

Acknowledgments

This research was supported by the European Union through the BUILDSPACE Horizon Europe Project under Grant no. 101082575.

References

[1] C. Cappiello, A. Gal, M. Jarke, J. Rehof, Data Ecosystems: Sovereign Data Exchange among Organizations (Dagstuhl Seminar 19391), Dagstuhl Reports 9 (2020) 66–134. URL: https://drops.dagstuhl.de/entities/document/10.4230/DagRep.9.9.66. doi:10.4230/DagRep.9.9.66.

[2] R. C. Amorim, J. A. Castro, J. Rocha da Silva, C. Ribeiro, A comparison of research data management platforms: architecture, flexible metadata and interoperability, Universal Access in the Information Society 16 (2017) 851–862. doi:10.1007/s10209-016-0475-y.

[3] J. R. Cedeno Jimenez, P. Zhao, A. Mansourian, M. A. Brovelli, Geospatial blockchain: review of decentralized geospatial data sharing systems, AGILE: GIScience Series 3 (2022) 29. URL: https://agile-giss.copernicus.org/articles/3/29/2022/. doi:10.5194/agile-giss-3-29-2022.

[4] A. Martikkala, A. Lobov, M. Lanz, I. F. Ituarte, Towards the interoperability of iot platforms: A case study for data collection and data storage, IFAC-PapersOnLine 54 (2021) 1138–1143. doi:https://doi.org/10.1016/j.ifacol.2021.08.134, 17th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2021.

[5] M. Wu, F. Psomopoulos, S. J. Khalsa, A. de Waard, Data discovery paradigms: User requirements and recommendations for data repositories, Data Science Journal (2019). doi:10.5334/dsj-2019-003.

[6] M. Andresel, V. Siska, R. David, S. Schlarb, A. Weißenfeld, Adapting ontology-based data access for data spaces, in: The Second International Workshop on Semantics in Dataspaces, co-located with the Extended Semantic Web Conference, May 26 – 27, 2024, Hersonissos, Greece, 2024.

[7] M. Singh, W. Wu, S. Rizou, E. Vakaj, Data information interoperability model for iot-enabled smart water networks, in: 2022 IEEE 16th International Conference on Semantic Computing (ICSC), 2022, pp. 179–186. doi:10.1109/ICSC52841.2022.00038.

[8] H. Masoumi, B. Farahani, F. S. Aliee, An ontology-based open data interoperability approach for cross-domain government data services, in: 2020 25th International Computer Conference, Computer Society of Iran (CSICC), 2020, pp. 1–8. doi:10.1109/CSICC49403.2020.9050079.

[9] G. De Giacomo, D. Lembo, M. Lenzerini, A. Poggi, R. Rosati, Using Ontologies for Semantic Data Integration, Springer International Publishing, Cham, 2018, pp. 187–202. URL: https://doi.org/10.1007/978-3-319-61893-7_11. doi:10.1007/978-3-319-61893-7_11.

[10] A. Helal, M. Helali, K. Ammar, E. Mansour, A demonstration of kglac: a data discovery and enrichment platform for data science, Proc. VLDB Endow. 14 (2021) 2675–2678. doi:10.14778/3476311.3476317.

[11] C. Lu, J.-r. Park, X. Hu, I.-Y. Song, Metadata effectiveness: A comparison between user-created social tags and author-provided metadata, in: 2010 43rd Hawaii International Conference on System Sciences, 2010, pp. 1–10. doi:10.1109/HICSS.2010.273.

[12] G. Van Rossum, F. L. Drake, Python 3 Reference Manual, CreateSpace, Scotts Valley, CA, 2009.

[13] A. Kamsky, Adapting tpc-c benchmark to measure performance of multi-document transactions in mongodb, Proc. VLDB Endow. 12 (2019) 2254–2262. URL: https://doi.org/10.14778/3352063.3352140. doi:10.14778/3352063.3352140.

[14] The go programming language. go 1.20 release notes, 2023. URL: https://go.dev/doc/devel/release#go1.20.

[15] J. J. Carroll, I. Dickinson, C. Dollin, D. Reynolds, A. Seaborne, K. Wilkinson, Jena: implementing the semantic web recommendations, in: Proceedings of the 13th International World Wide Web Conference on Alternate Track Papers & Posters, WWW Alt. '04, Association for Computing Machinery, New York, NY, USA, 2004, p. 74–83. URL: https://doi.org/10.1145/1013367.1013381. doi:10.1145/1013367.1013381.