Conceptual Model of a Federated Data Lake

Pedro Guimarães
Centro de Computação Gráfica
Guimarães, Portugal *pedro.guimaraes@ccg.pt*

Diogo Rodrigues
Centro de Computação Gráfica
Guimarães, Portugal *diogo.rodrigues@ccg.pt*

Mariana Almeida
Centro de Computação Gráfica
Guimarães, Portugal *mariana.almeida@ccg.pt*

Mafalda Oliveira
Centro de Computação Gráfica
Guimarães, Portugal *mafalda.oliveira@ccg.pt*

Paulo Barbosa
Centro de Computação Gráfica
Guimarães, Portugal *paulo.barbosa@ccg.pt*

Daniela Barros
Centro de Computação Gráfica
Guimarães, Portugal *daniela.barros@ccg.pt*

Joana Ribeiro
Centro de Computação Gráfica
Guimarães, Portugal *joana.ribeiro@ccg.pt*

Maribel Yasmina Santos
ALGORITMI Research Centre - University of Minho
Guimarães, Portugal *maribel@dsi.uminho.pt*

Abstract

Valuable insights are frequently only available after combining and analysing data from multiple sources. This paper presents a Conceptual Model of a Federated Data Lake, as a contribution to formalize the required components and their relationships, in order to identify and address them in the implementation of a comprehensive system that supports on-the-fly query processing over multiple heterogeneous sources and provides an adequate data management by highlighting the concepts of a Data Lake and focusing on the Metadata Management domain as an engine to the integration of several Data Lakes.
Keywords: Data Lake, Federated Data Lake, Metadata Management, Data Management

1. Introduction

In a system where the data sources are federated, these need to be linked - in an integrated view - into a unified system. A Data Lake is a flexible, scalable data storage and management system, which ingests and stores raw data from heterogeneous sources in their original format, and can provide query processing on-the-fly [1]. Due to its high flexibility and scalability, the

data sources can form information silos without being integrated. However, valuable insights are frequently only available after combining and analysing data from these data silos [2]. Data integration is a challenge faced when querying and combining multiple heterogeneous data sources and providing unified data access for users [3].

When integrating multiple Data Lakes, several challenges such as selecting relevant data sources for a specific query, creating an efficient query execution plan considering the data source types, and combining partial results obtained from these sources [4] are faced. Assuming a large scale of sources in a Data Lake, we should take into account some techniques to deal with the heterogeneity of sources with regard to data models and schemas such as schema matching, schema mapping, query reformulation, entity linkage, among others [1].

In order to address these challenges, and considering several crucial contributions of related works presented below, we propose a conceptual model for Data Lake federation highlighting the concepts of a Data Lake and focusing on the Metadata Management domain as an engine to the Data Lake integration. In this component, the model starts by separating Functional Metadata, regarding different Metadata's areas (Business, Operational and Technical), from Structural Metadata, focusing in Objects properties. Having a Metadata Management model is critical for data analysis, query processing and data quality.

Without any metadata, the Data Lake is barely unusable as the structure and semantics of the data are not known, which quickly transforms a Data Lake into a 'data swamp' [2]. Nevertheless, current proposals for Data Lake implementations are not clear about the Metadata Management requirements, features and methods for efficient integrated query processing. In this paper we combine different perspectives in the Conceptual Model of a Federated Data Lake.

This paper is organized as follows. Section 2 addresses related work. Section 3 presents and describes the proposed conceptual model. Section 4 concludes with some remarks and proposals for future work.

2. Related Work

Data federation addresses the problem of uniformly accessing multiple, possibly heterogeneous data sources, by mapping them into a unified schema and by supporting the execution of queries.

The authors of [5] argue that the query answering process should compute an execution plan of the partitioned sub-queries based on the metadata catalogue and returns the decomposed sub-queries over the corresponding data sources via the mappings and the metadata catalogue. This study helps end-users to understand which system best suits their application requirements and supports decision making by improving currently available solutions and designing more powerful federation systems. It mainly focuses on the metadata catalog, source selection, query partition and optimization. However, it does not address the issue of data integration in a federated system.

The authors in [6] propose a 'lakehouse' architecture, which replaces traditional data storage architectures. They argue that to implement a Lake House, the main key is to apply a metadata layer to the data that is stored, in order to be able to organise it in a better way. Furthermore, they argue that a metadata layer is required to implement data quality control and data provenance features. In addition, metadata layers are ideal for implementing data governance features, such as access controls and audit logging. In order to improve query performance, the authors propose a set of techniques, such as the data layout. They argue that data layout plays an important role when it comes to performance. Finally, they present an efficient data access approach for advanced analytics that relies on providing a declarative version of the Data Frame APIs used, which maps the computation of data preparation into Spark SQL query plans. This research does not address other relevant issues such as data storage and the formats that this storage may have in the metadata layer, as well as issues related with data access and extraction.

The authors of [1] propose a Data Lake architecture that integrates a layer for the storage

systems, which can be based in file-based storage systems, like HDFS (Hadoop Distributed File System). They also present single data storage systems as an option for specific Data Lakes that deal with particular types of data. Polystore systems are also an example given by the authors of storage systems that can be implemented in this layer and these systems are ideal for heterogeneous data. [2] propose the Constance system, a Data Lake system with a sophisticated metadata layer applied to raw data extracted from heterogeneous data sources, which uses this type of storage system. Lastly, they present the option of cloud storage. Regarding data exploration, the authors propose two approaches to address queries in a Data Lake: discover the data lakes based on the relatedness of datasets or provide a unified query interface for heterogeneous data sources. According to [5], the key task of data federation systems is to support federated query answering, so that users have may querying multiple data sources and keep the data in the original storage. Our proposal complies the required components into a conceptual model of a Federated Data to promote and provide the access to multiple heterogeneous data sources taking in consideration the challenges described by different authors and our own Data Lake implementation experiences.

3. Conceptual Model of a Federated Data Lake

In this section we present the proposal for the Conceptual Model of a Federated Data Lake as a contribute to identify and describe the different components and relationships within a Federated Data Lake focusing in metadata management as a key factor for a successful data integration.

Our proposal (Fig. 1) includes the components that a Data Lake should support to promote the integration on a federation. Regarding the storage management needs, there is the need of a Physical Storage, for Structured, Semi-structured and Unstructured Data Types [1], and Storage System with different storage approaches.

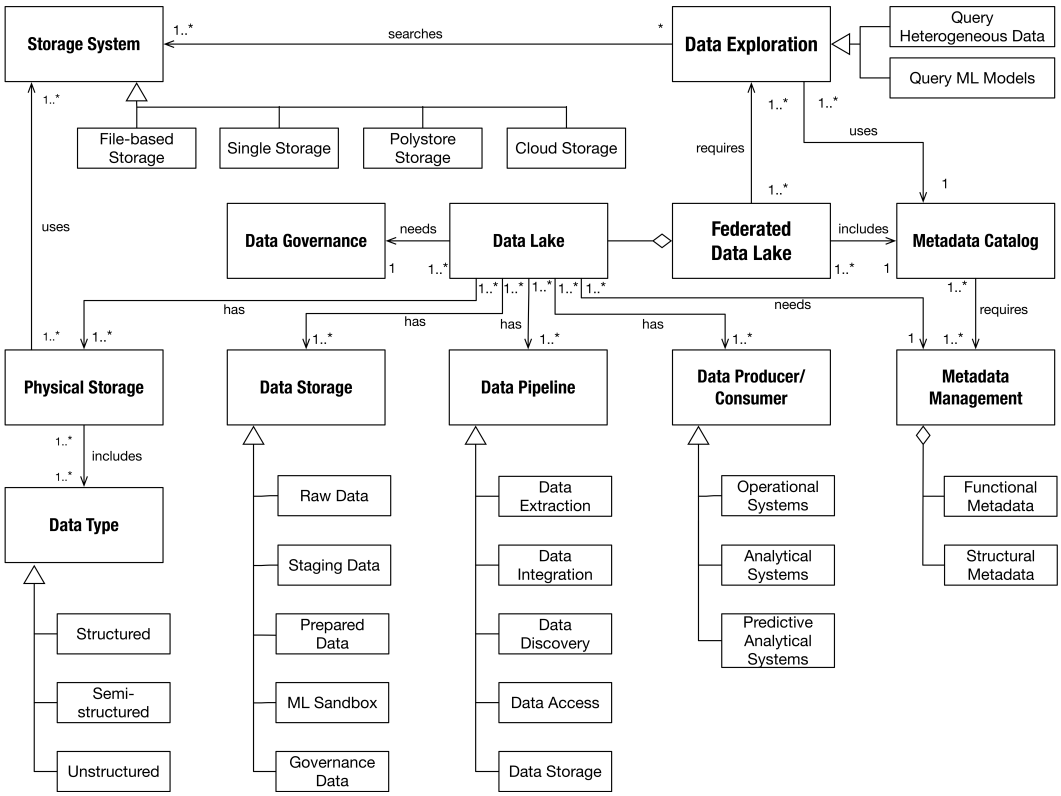

Fig. 1. Proposed Conceptual Model for a Federated Data Lake.

The Data Storage needs to foresee the distribution of data into several stages of maturity, such as Raw Data as the location to store data in its raw/original format, Staging Data supporting the ETL process, Prepared Data for data already prepared and available for data analytics, ML Sandbox where data can be accessed by data scientists for training machine learning models, and Governance Data to manage, monitor and govern metadata, data quality, data catalogs and security [8].

After data ingestion (from Data Producers such as Operational Systems), several data pipelines are needed to extract data, integrate data with existent data structures or define different access levels for data storage, preparing it for Data Consumers such as Analytical Systems or Predictive Analytical Systems. Data Producers and Data Consumers are mainly associated to systems that collect or intend to support operational systems and analytical systems of different types. A Data Lake has a great level of complexity due to its flexibility and mostly its heterogeneous data. Multiple data sources require data governance policies and tools to ensure the consistency and trustworthiness of the data, demanding for the use of a Metadata Catalog to create and provide an unified view of a Federated Data Lake and support data quality.

This Metadata Catalog is crucial to explore the data of the federated data lake and, for that reason, the Metadata Catalog of the Federated Data Lake has to be used in all the required Data Explorations. Each Data Exploration only uses one Metadata Catalog, depending on the Federated Data Lake that is being analysed. We divided that exploration into two types, a unified querying interface for heterogeneous data [1] and the querying of machine learning models. This data exploration is accomplished by searching the Storage System. This storage system can be of four types, identified by how the ingested data is stored in the data lake, a file-based storage, being HDFS the most popular one, a single storage, for use cases that aim at specific types of data, a polystore storage, which has integrated access to a configuration of multiple data stores for heterogeneous data, and a cloud storage, vastly used in commercial data lakes [1].

Detailing the concept associated with the Metadata Management, and based in the works of [7] and [8], Fig. 2 details the two main types of metadata relying on data lakes, the Functional Metadata and the Structural Metadata. These two typologies of metadata focus on basic information and characteristics of data, as data size and its formats, data semantics (e.g., tags, descriptions, etc.) and data history. The key difference between these two typologies is that Structural Metadata, unlike Functional Metadata, also concerns about data linkage and user interactions with the data lakes. A Metadata Catalog (for a federated Data Lake) requires Metadata Management, that is, the administration of both Functional and Structural Metadata about the Data Lake's data. The Metadata Management component is responsible for preventing a Data Lake from becoming an inoperable data swamp, since data ingested in data lakes have no explicit schema [8].

Functional Metadata considers the way that metadata is assembled and it is organized into three categories: Business Metadata, Operational Metadata and Technical Metadata. The Business Metadata includes the definition of the business rules that, for instance, determine integrity constraints for a better understanding of the data. In Operational Metadata, information is automatically generated during data processing. In this case, it is important to have a description of the main data and its source, thus evaluating its quality and provenance. The Technical Metadata describes how data is represented, taking into account, for example, its format and schema [7].

Structural Metadata provides a description of the structure and the schema of data. The Intra-Object Metadata category includes a general definition of the object and its properties, a pre-visualization of what is the structure or content of the object, its version (obtained from data updates) and annotations. The Inter-Object Metadata category describes the links between two or more objects, shows groups of organized objects, the similarity links, their strength and parenthood that keeps the data lineage by recording the process of creating new objects. At last, there is the global metadata category heading a context layer in order to process and analyse data

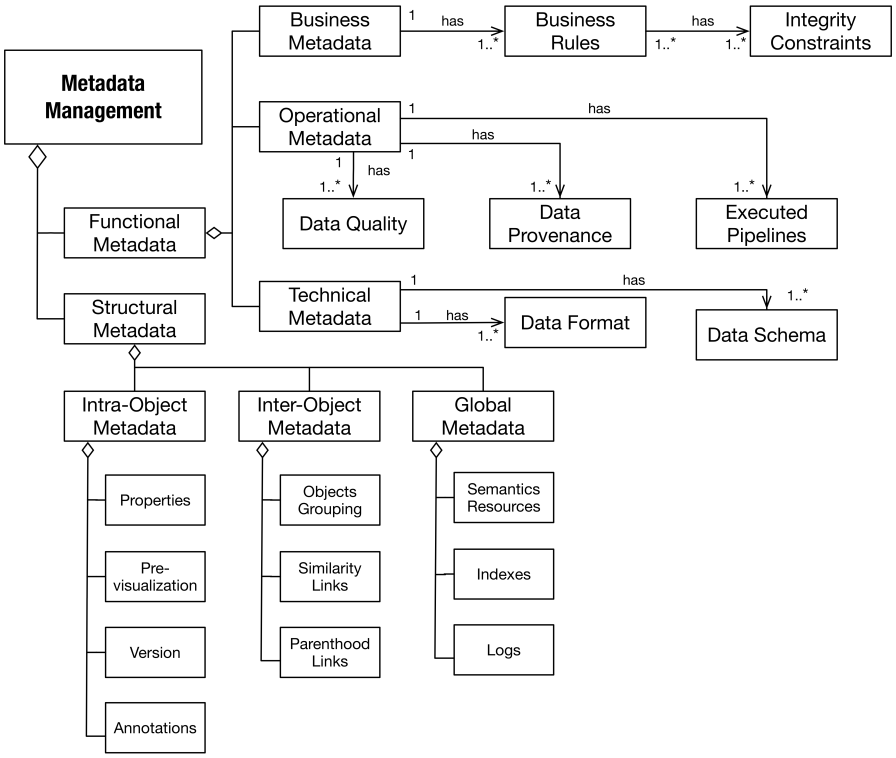

Fig. 2. Detailed View of the Metadata Management Concept.

in a more simplified way by using semantic resources and indexes that will provide improved data retrieval based on terms or patterns [8].

4. Discussion and Conclusions

Twenty years after discussing Data Warehouses integration, now arises the challenge of applying the same rationale to Data Lakes. This is a complex challenge due to the very peculiar characteristics of a Data Lake, such as its schema-less and multiple heterogeneous data sources. This complexity prompts interest in both academia and industry, where different approaches and models have been proposed, suggesting a set of components and relationships between them in order to facilitate the integration of multiple Data Lakes. However, these contributions often focused only on the technological part and left unexplored the conceptual and metadata management part. In the proposed approach, based on the literature and on several Data Lakes' implementations, we provide a broader view of the components and the working areas that a Federated Data Lake should explore, from Data Exploration, Metadata Management, Query Heterogeneous Data, or Query ML Models. The concept of Federated Data Lake is complex and is not limited to the components here proposed, as this proposal will continue to evolve to address new challenges and new perspectives that will emerge as a driver for the improvement of this conceptual vision. Also, we foresee the implementation of a demonstration case to show how this conceptual model can be used in real contexts.

5. Acknowledgements

This work has been supported by *FCT – Fundação para a Ciência e Tecnologia* within the R&D Units Project Scope: UIDB/00319/2020, and was carried out within the project "City-Catalyst" reference POCI/LISBOA-01-0247-FEDER-046119, co-funded by *Fundo Europeu de Desenvolvimento Regional* (FEDER), through Portugal 2020 (P2020).

References

1. Hai, R., Quix, C., Jarke, M.: Data lake concept and systems: a survey. http://arxiv.org/abs/2106.09592 (2021)
2. Hai, R., Geisler, S., Quix, C.: Constance: An intelligent data lake system. Proceedings of the ACM SIGMOD International Conference on Management of Data, 26-June-2016(June), 2097–2100. https://doi.org/10.1145/2882903.2899389 (2016)
3. Doan, A., Halevy, A., Ives, Z.: Principles of Data Integration, Morgan Kaufmann, Pages xvii-xviii, ISBN 9780124160446, https://doi.org/10.1016/B978-0-12-416044-6.00025-9 (2012)
4. Endris, K., Rohde, P., Vidal, M.-E., Auer, S.: Ontario: Federated Query Processing against a Semantic Data Lake (2019)
5. Gu, Z., Corcoglioniti, F., Lanti, D., Mosca, A., Xiao, G.: A systematic overview of data federation systems (2022)
6. Armbrust, M., Das, T., Sun, L., Yavuz, B., Zhu, S., Murthy, M., Torres, J., van Hovell, H., Ionescu, A., Łuszczak, A., Świtakowski, M., Szafrański, M., Li, X., Ueshin, T., Mokhtar, M., Boncz, P., Ghodsi, A., Paranjpye, S., Senster, P., ... Zaharia, M.: Delta lake: High-performance ACID table storage over cloud object stores. Proceedings of the VLDB Endowment, 13(12), 3411–3424. https://doi.org/10.14778/3415478.3415560 (2020)
7. Diamantini, C., Giudice, P. L., Musarella, L., Potena, D., Storti, E., Ursino, D.: A New Metadata Model to Uniformly Handle Heterogeneous Data Lake Sources. In Benczúr, A., Thalheim, B., Horváth, T., Chiusano, S., Cerquitelli, T., Sidló, C., Revesz, P.Z. (Eds.), New Trends in Databases and Information Systems (Vol. 909, pp. 165–177). Springer International Publishing. https://doi.org/10.1007/978-3-030-00063-9_17 (2018)
8. Sawadogo, P., Darmont, J.: On data lake architectures and metadata management. Journal of Intelligent Information Systems, 56(1), 97–120. https://doi.org/10.1007/s10844-020-00608-7 (2021)