Metadata Management for Data Lakes

Franck Ravat[1] and Yan Zhao[1,2](✉)

[1] Institut de Recherche en Informatique de Toulouse, IRIT-CNRS (UMR 5505),
Université Toulouse 1 Capitole, Toulouse, France
{Franck.Ravat,Yan.Zhao}@irit.fr
[2] Centre Hospitalier Universitaire (CHU) de Toulouse, Toulouse, France

Abstract. To prevent data lakes from being invisible and inaccessible to users, an efficient metadata management system is necessary. In this paper, we propose a such system based on a generic and extensible classification of metadata. A metadata conceptual schema which considers different types (structured, semi-structured and unstructured) of raw or processed data is presented. This schema is implemented in two DBMSs (relational and graph) to validate our proposal.

Keywords: Data lake · Metadata management · Metadata classification

1 Introduction

The concept of Data Lake (DL) was created by Dixon [4] and extended by various authors [5,8,20]. DL allows to ingest raw data from various sources, store data in their native format, process data upon usage, ensure the availability of data and provide accesses to data scientists, analysts and BI professionals, govern data to insure the data quality, security and data life cycle.

DLs facilitate different types of analysis such as machine learning algorithms, statistics, data visualisation... (unlike Data Warehouses (DW) [16]). The main characteristic of DL is 'schema-on-read' [5], data are only processed upon usage. Compared to DWs, which are structured data repositories dedicated to predetermined analyses, DLs have great flexibility and can avoid losing information.

However, a data lake that contains a great amount of structured, semi-structured and unstructured data without explicit schema or description can easily turn into a data swamp which is invisible, inaccessible and unreliable to users [18]. To prevent data lakes from turning into data swamps, metadata management is essential [1,8,20]. Metadata can help users find data that correspond to their needs, accelerate data accesses, verify data origin and processing history to gain confidence and find relevant data to enrich their analyses [1,14].

Nevertheless, many papers are focused on a single zone (especially ingestion zone) or a single data type of data lakes. Therefore, the goal of this paper is to propose a metadata management system dedicated to data lakes and applied to the whole life-cycle (multiple zones) of data. The set of the paper is as follows:

T. Welzer et al. (Eds.): ADBIS 2019, CCIS 1064, pp. 37–44, 2019.
https://doi.org/10.1007/978-3-030-30278-8_5

the second section introduces related work on metadata. In the third section, we propose our metadata conceptual schema with a classification. The fourth section describes the implementation of a metadata management system.

2 Related Work

DL metadata, inspired by the DW classifications [6,7], are classified into two ways. A first classification includes three categories [12,14]: *Technical* metadata concern data type, format and structure (schema). *Operational* metadata concern data processing information and *Business* metadata concern business objects and descriptions. A second classification includes not only the information of each dataset (intra-metadata) but also the relationships between datasets (inter-metadata). Intra-metadata are classified into data characteristics, definitional, navigational, activity, lineage, rating and assessment [2,6,19]. Inter-metadata describe relationships between datasets, they are classified into dataset containment, provenance, logical cluster and content similarity [9].

Compared to the first classification, the second one is more specific. Nevertheless, the second classification can be improved. Some sub-categories are not adapted to data lakes. For instance, the *rating* subcategory that concerns user preferences [19] needs to be removed. Because in data lakes, datasets can be processed and analysed by different users [5], a dataset that makes no sense to BI professionals can be of great value to data scientists. What's more, this classification can be extended with more sub-categories. For instance, data sensitivity and accessibility also need to be controlled in data lakes.

Concerning metadata management, various solutions for data lakes are presented with different emphases [1,9,15,17]. Regarding all the solutions of metadata management, authors mainly focused on a few points. Firstly, the detection of relationships between different datasets is always presented [1,9,15]. Relationships between datasets can help users find as many relevant datasets as possible to enrich data analysis. While we want to find a metadata model that shows not only the relationships between datasets but also the information of each single dataset. Secondly, authors often focused on unstructured data (mostly textual data) [15,17] for the difficulty of extracting information. However, in a data lake, there are various types of data (images, pdf files...). Thirdly, data ingestion is the most considered phase to extract metadata [1,9,17]. Nevertheless, the information that is produced during process and access phases has value too [6,17].

Until now, there isn't a generic metadata management system that works on both structured and unstructured data for the whole data life-cycle in data lakes. The objective of this paper is to define a metadata management system that addresses these weaknesses.

3 Metadata Model

Considering the diversity of data structural type and different processes that applied on datasets, our solution is based on intra- and inter-metadata.

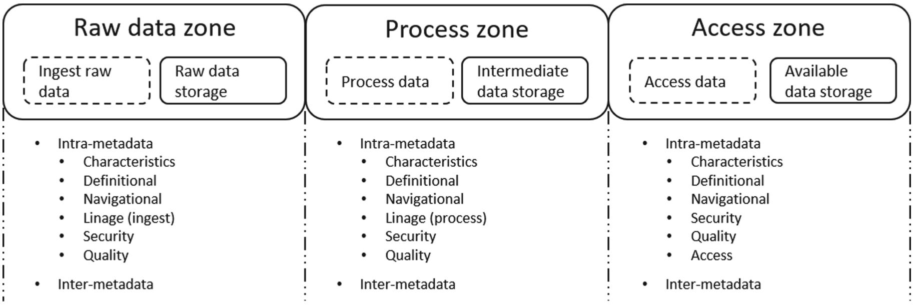


Fig. 1. Meta data classification

3.1 Metadata Classification

Our metadata classification has the advantage of integrating both intra-metadata and inter-metadata for all datasets. Intra-metadata allow users to understand datasets with their characteristics, meaning, quality and security level [2,19]. Inter-metadata help users find relevant datasets that can answer their requirements to make their data discovery easier [9,17].

- *Inter-metadata.* We complete the classification of [9] and obtain 5 subcategories. *Dataset containment* signifies that a dataset is contained in other datasets. *Partial overlap* signifies that some attributes with corresponding data in some datasets overlap. For instance, in a hospital, health care and billing databases contain the same attributes and data about patients, prescriptions and stays. But these two databases also contain their own specific data. *Provenance* signifies that one dataset is the source of another dataset. *Logical clusters* signifies that some datasets are in the same domain. For example, different versions, duplication of the same logical dataset. *Content similarity* signifies that different datasets share the same attributes.
- *Intra-metadata.* We extend the classification of [2,19] to include access, quality and security.
  - *Data characteristics* consist of information such as identification, name, size, structural type and creation date of datasets. This information helps users to have a general idea of a dataset.
  - *Definitional metadata* specifies datasets' meanings. In the original taxonomy, there are vocabulary and schema subcategories. We classify definitional metadata into *semantic* and *schematic* metadata. Structured and unstructured datasets can be semantically described by a text or by some keywords (vocabularies). Schematically, a structured dataset can be presented by a database schema.
  - *Navigational metadata* concerns the location of datasets, for instance, file paths and database connection URLs.
  - *Lineage* presents data life-cycle. It consists of the original source of datasets and the processing history. Information on datasets sources and process history makes datasets more reliable.

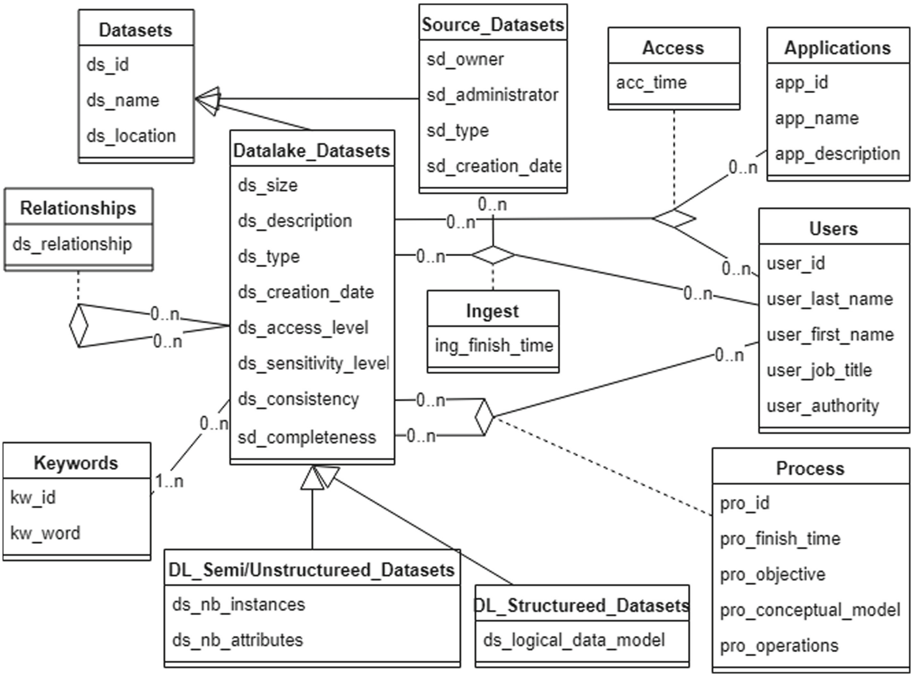
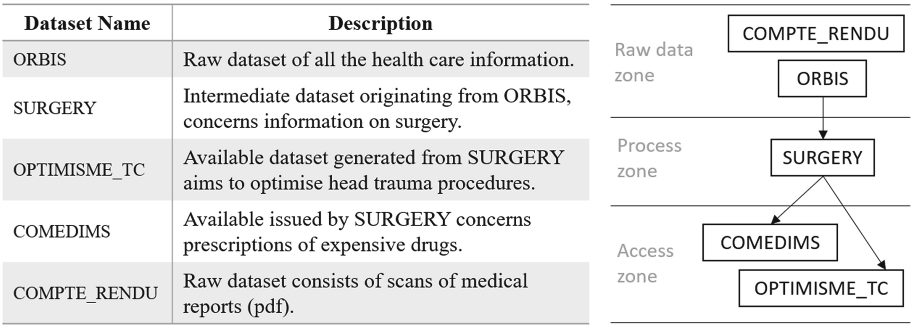


Fig. 2. Class diagram of metadata conceptual model

- *Access metadata* present access information, for example, name of the users who accessed datasets and the access tools. This information helps users to find relevant datasets by accessed users and to trust data by other users' access histories.
- *Quality metadata* consist of data consistency and completeness [10] to ensure datasets' reliability.
- *Security metadata* consist of data sensitivity and access level. Data lakes store datasets from various sources. Some datasets may contain sensitive information that can only be accessed by certain users. Security metadata can support the verification of access. This information ensures the safety of sensitive data.

3.2 Metadata Conceptual Schema

From the functional architecture point of view [5,11,13,20], a data lake contains four essential zones. A *raw data zone* allows to ingest data without processing and stores raw data in their native format. A *process zone* allows to process raw data upon usage and provides intermediate storage areas. The *access zone* stores refined data and ensures data availability. And a *governance zone* is in charge of insuring data quality, security and data life-cycle.



Dataset Name	Description
ORBIS	Raw dataset of all the health care information.
SURGERY	Intermediate dataset originating from ORBIS, concerns information on surgery.
OPTIMISME_TC	Available dataset generated from SURGERY aims to optimise head trauma procedures.
COMEDIMS	Available issued by SURGERY concerns prescriptions of expensive drugs.
COMPTE_RENDU	Raw dataset consists of scans of medical reports (pdf).

Fig. 3. List of datasets

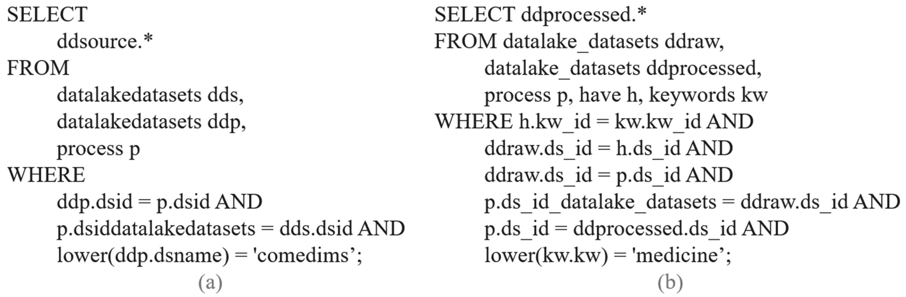
Our metadata classification is applied on the multi-zones functional architecture of data lakes (see Fig. 1). Based on this classification, we propose a metadata conceptual schema (see Fig. 2). A dataset, structured or unstructured, is ingested from one or plural sources by one or more users. Datasets can be processed by users to transform to new datasets. Users can access datasets with some tools for their analyses. Datasets that are stored in a data lake can have relationships.

This metadata conceptual schema has several advantages: (i) Data sources (external datasets) are recorded. (ii) Both structured and unstructured datasets are considered. (iii) All the work (ingest, process, access) that has been done in a data lake is registered. (iv) Information of each single dataset and relationships between different datasets are stored. And (v) datasets' quality, sensibility and access level are controlled.

4 Metadata Implementation

The University Hospital Center (UHC) of Toulouse is the largest hospital center in the south of France. All medical, financial and administrative data are stored in the information system of this center. The UHC of Toulouse plans to launch a data lake through an iterative process. The aim of this project is to ingest (i) all the internal relational databases and medical e-documents (including scans of hand written medical reports), (ii) external data coming from other French UHCs and (iii) some public medical data. The objective of this data lake is to combine these different data sources in order to allow data analysts and BI professionals to analyse available data to improve medical treatments. The first step of this project concerns 5 datasets (4 structured and 1 unstructured datasets). These 5 datasets are in different functional zones of a data lake (see Fig. 3).

We have implemented two proofs of concept in the UHC of Toulouse in order to validate our proposal. Regarding metadata management systems, there are metadata stored in key-value [9], XML documents [15,17], relational databases [17] or by ontology [1]. We have chosen to implement a relational database and a graph database for the fallowing reasons: relational databases have a standard



```
SELECT
        ddsource.*
FROM
        datalakedatasets dds,
        datalakedatasets ddp,
        process p
WHERE
        ddp.dsid = p.dsid AND
        p.dsiddatalakedatasets = dds.dsid AND
        lower(ddp.dsname) = 'comedims';
```

(a)

```
SELECT ddprocessed.*
FROM datalake_datasets ddraw,
        datalake_datasets ddprocessed,
        process p, have h, keywords kw
WHERE h.kw_id = kw.kw_id AND
        ddraw.ds_id = h.ds_id AND
        ddraw.ds_id = p.ds_id AND
        p.ds_id_datalake_datasets = ddraw.ds_id AND
        p.ds_id = ddprocessed.ds_id AND
        lower(kw.kw) = 'medicine';
```

(b)

Fig. 4. Logical data model

query language (SQL) and a high security level insured by many RDBMSs (Relational Database Management System); graph databases ensure scalability and flexibility. Moreover, these systems are currently used in the UHC of Toulouse.

4.1 Relational Database

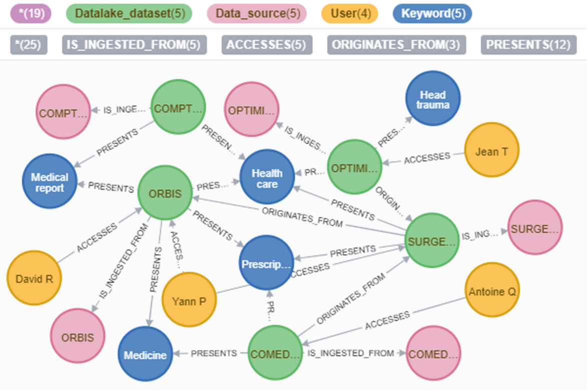
We firstly implemented the conceptual schema of metadata on a relational DBMS. After the implantation, we collected the needs of data scientists from a metadata point of view. The first questions were about data trust and data lineage analysis. To validate our proposal, we have written several queries to compare the feasibility and usability of different environments. In the following paragraphs, you will find two examples.

(i) When a user works on a dataset, he may wants to know where does the data come from to have more confidence on the dataset. There is an example to find the original dataset of ’COMEDIMS’ (see Fig. 4 (a)). (ii) Besides finding the origin of one dataset, users may also want to find relevant datasets that come from the same origin of the dataset. For example, users want to find out all the datasets that used the data of the original dataset of COMEDIMS (see Fig. 4 (b)).

4.2 Graph Database

The second solution of implementation is graph database. We firstly introduce a Neo4j model for the 5 datasets of UHC of Toulouse. In addition, 2 queries that answer the same questions in the last subsection will be executed.

We extended the mapping from UML class diagram to property graphs that proposed by [3] to Neo4j Cypher query language. Based on this mapping, we implemented a graph database with neo4j (Fig. 5). To test the implementation, we also answered the same questions than the relational database.



1.

```
$ MATCH
(ds:Datalake_dataset{ds_name:'COM
EDIMS'})-[ORIGINATES_FROM]-
>(origds) RETURN origds, ds
```

2.

```
$ MATCH
(ds:Datalake_dataset{ds_name:'COM
EDIMS'})-[ORIGINATES_FROM]-
>(origds)-[r]-(b) RETURN origds, ds,
r,b
```

Fig. 5. Neo4j data model

5 Conclusion and Future Work

To prevent a data lake from turning into a data swamp, metadata management is recommended. In this paper, we firstly proposed a generic and extensible classification of metadata based on a multi-zones of data lake functional architecture. The classification considers not only the metadata on each dataset (intra-metadata) but also the relationships between datasets (inter-metadata). Based on the classification, we presented a metadata conceptual schema for data lakes. What's more, for validating the conceptual schema, we implemented a graph DBMS and a relational DBMS for metadata management system in UHC of Toulouse.

Our next plan concerns the automatic extraction of metadata. For this automatic extraction, we plan to adapt to the context of existing works such as automatic detection of relationships between datasets [1] and automatic extraction of data structure, metadata proprieties and semantic data [1]. Nevertheless, there isn't a system which can extract automatically inter-metadata, intra-metadata from different types (structured, semi-structured, unstructured) of datasets.

Our long term goal is to accomplish a metadata management system which integrates automatic extraction of data, effective researches of metadata, automatic generation of dashboards or other analyses.

References

1. Alserafi, A., Abelló, A., Romero, O., Calders, T.: Towards information profiling: data lake content metadata management. In: 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW), pp. 178–185. IEEE (2016)
2. Bilalli, B., Abelló, A., Aluja-Banet, T., Wrembel, R.: Towards intelligent data analysis: the metadata challenge. In: Proceedings of the International Conference on Internet of Things and Big Data, Rome, Italy, pp. 331–338 (2016)

3. Delfosse, V., Billen, R., Leclercq, P.: Uml as a schema candidate for graph databases. NoSql Matters 2012 (2012)
4. Dixon, J.: Pentaho, Hadoop, and Data Lakes, October 2010. https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes/
5. Fang, H.: Managing data lakes in big data era: what's a data lake and why has it became popular in data management ecosystem. In: 2015 IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER), pp. 820–824. IEEE (2015)
6. Foshay, N., Mukherjee, A., Taylor, A.: Does data warehouse end-user metadata add value? Commun. ACM **50**(11), 70–77 (2007)
7. Gabriel, R., Hoppe, T., Pastwa, A.: Classification of metadata categories in data warehousing-a generic approach. In: AMCIS, p. 133 (2010)
8. Hai, R., Geisler, S., Quix, C.: Constance: an intelligent data lake system. In: Proceedings of the 2016 International Conference on Management of Data, pp. 2097–2100. ACM (2016)
9. Halevy, A.Y., et al.: Managing Google's data lake: an overview of the Goods system. IEEE Data Eng. Bull. **39**(3), 5–14 (2016)
10. Kwon, O., Lee, N., Shin, B.: Data quality management, data usage experience and acquisition intention of big data analytics. Int. J. Inf. Manag. **34**(3), 387–394 (2014)
11. LaPlante, A., Sharma, B.: Architecting Data Lakes, March 2016
12. Lopez Pino, J.L.: Metadata in Business Intelligence, January 2014. https://www.slideshare.net/jlpino/metadata-in-business-intelligence
13. Menon, P.: Demystifying Data Lake Architecture, July 2017. https://medium.com/@rpradeepmenon/demystifying-data-lake-architecture-30cf4ac8aa07
14. Oram, A.: Managing the Data Lake. OReilly Media, Inc., Sebastopol (2015)
15. Quix, C., Hai, R., Vatov, I.: Metadata extraction and management in data lakes With GEMMS. Complex Syst. Inf. Model. Q. **9**, 67–83 (2016)
16. Ravat, F., Song, J.: A unified approach to multisource data analyses. Fundam. Inf. **162**(4), 311–359 (2018)
17. Sawadogo, P., Kibata, T., Darmont, J.: Metadata management for textual documents in data lakes. In: 21st International Conference on Enterprise Information Systems (ICEIS 2019) (2019)
18. Thor, O.: 3 keys to keeping your data lake from becoming a data swamp, June 2017. https://www.cio.com/article/3199994/3-keys-to-keep-your-data-lake-from-becoming-a-data-swamp.html
19. Varga, J., Romero, O., Pedersen, T.B., Thomsen, C.: Towards next generation BI systems: the analytical metadata challenge. In: Bellatreche, L., Mohania, M.K. (eds.) DaWaK 2014. LNCS, vol. 8646, pp. 89–101. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10160-6_9
20. Walker, C., Alrehamy, H.: Personal data lake with data gravity pull. In: 2015 IEEE Fifth International Conference on Big Data and Cloud Computing, pp. 160–167. IEEE (2015)