Semantic Data Lake - Semantic Data Integration

A. Zaid, M. Fallou, T. Claas, M. Noman, S. Hoseini
Supervised by Prof. Dr. Quix and Dr. Lange-Bever

RWTH AACHEN UNIVERSITY

Fraunhofer FIT

ABSTRACT

Although big data has been discussed for some years, it still has many research challenges. It poses a huge difficulty to efficiently integrate, access, and query the large volume of diverse data in information silos with traditional 'schema-on-write' approaches such as data warehouses. Data lakes have been proposed as a solution to this problem as their vision is based on a generic and extensible architecture with a unified data model, facilitating the ingestion, storage and metadata management over heterogeneous data sources. The goal of this lab project was to further develop a prototype of a data lake system based on a four-layered architecture, where at the time the prototype was handed over, most of the functionality served the interaction and storage layer. The work of this group focused mostly on the interaction and transformation layer. The functionality was extended to incorporate the annotation of datasets with instances of an ontology to enrich stored data with additional metadata. In addition, a UI was added to process data graphically via Apache Spark.

PROJECT SUMMARY

The variety of data poses a huge difficulty to efficiently integrate, access, and query the large volume of diverse data in information silos with traditional 'schema-on-write' approaches such as data warehouses. Data lakes have been proposed as a solution to this problem. "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 provides query processing and data analytics in an on-the-fly manner." [1]

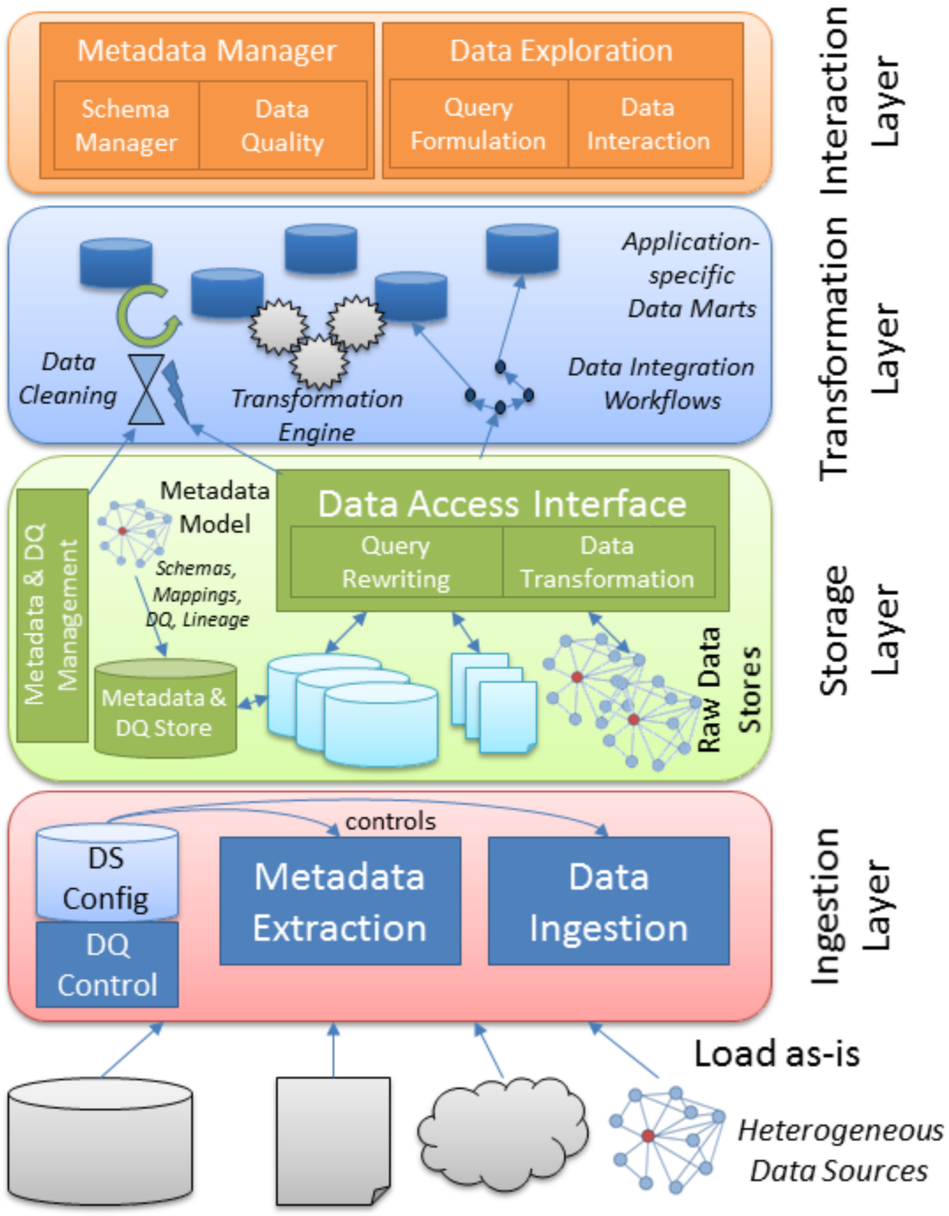

Figure 1: Four-layered Data Lake Architecture [2]

For data reasoning, query processing, and data quality management metadata management is crucial, because without it, the data lake is hardly usable as the structure and semantics of the data are not known, which turns a data lake quickly into a 'data swamp'. If sufficient metadata cannot be extracted from the sources automatically, a human expert has to provide additional information about the data source. [1] One of the two major additions of this lab aimed to solve the aforementioned problem and provides functionality to interlink datasets with instances of a user-defined ontology to enrich the available metadata using Apache Jena-Fuseki as a triple-store. The second major addition is a Workflow diagram to enable the user to define end-to-end data transformation pipelines graphically. In the following we list all of this labs additions to the project, which can mostly be associated to the interaction and transformation layer (see Figure 1):

1. Annotate Data with Semantic information
2. Graphical WorkFlow for Data Transformations written with ReactFlow
3. Integrated Apache Zeppelin to the System
4. Workspace Abstraction
5. Datamart Deletion
6. Complete refactoring of the projects code
7. ER-Diagram and backend Documentation using PyDocs

ARCHITECTURE

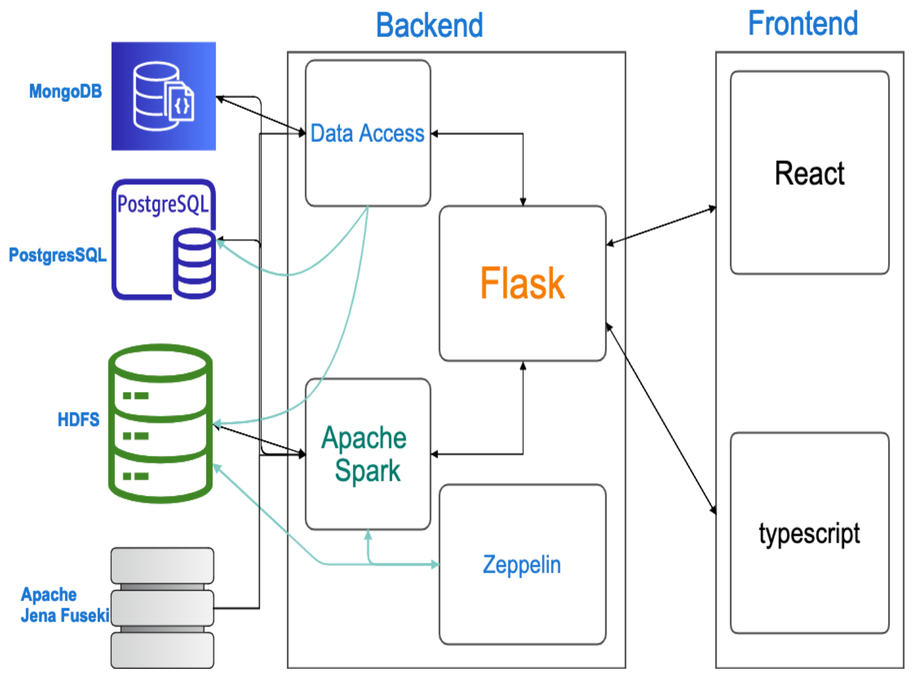

Figure 2: Architecture of the System. The left side displays databases and the underlying HDFS filesystem. The backend runs a Flask server with various data access function to store and process large-scale data using the Apache Spark analytics engine. The interaction between the frontend written in React is based on RESTful APIs. Figure created by Abdullah Zaid.

FUNCTIONALITIES

1. The WorkFlow diagram enables the user to define Data Transformation pipelines which are executed upon data sources in a single Spark Session.

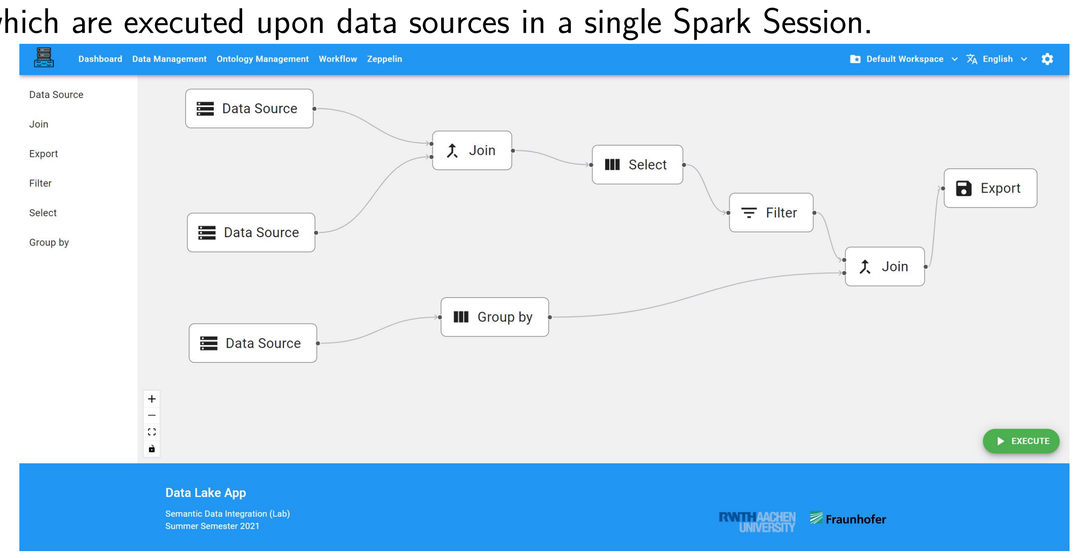

Figure 3: Snapshot of the Workflow UI

2. Datasets can be enriched by metadata using instances of user-defined ontologies. In the figure below the *Date* column of a dataset is interlinked with an ontology instance with label *primary key*.

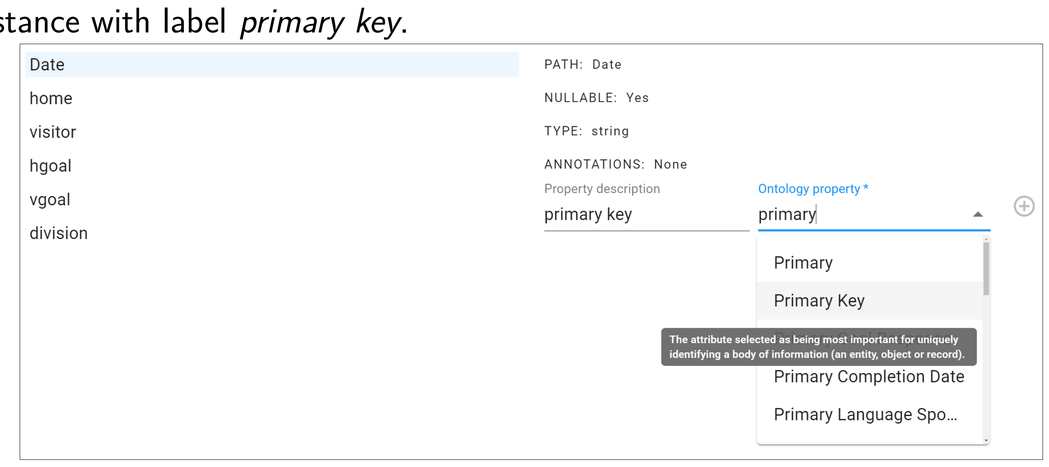

Figure 4: Interface for the annotation of datasets with ontology instances. The application displays metadata, allows to search instances of an ontology via keyword as well as entering a comment and displays the description provided by the ontology.

3. Apache Zeppelin is incorporated into the system's UI to perform data-driven, interactive collaborative data analytics with pyspark.

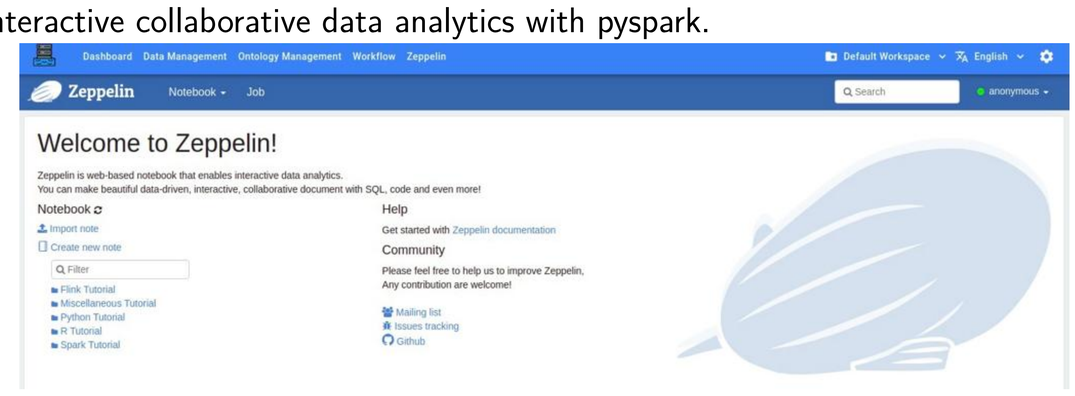

Figure 5: Apache Zeppelin's UI integrated into the application

REFERENCES

[1] R. Hai, C. Quix, M. Jarke, "Data lake concept and systems: a survey"

[2] M. Jarke and C. Quix. "On warehouses, lakes, and spaces – the changing role of conceptual modeling for data integration."