LODHub - A Platform for Sharing and Analyzing large-scale Linked Open Data

Stefan Hagedorn and Kai-Uwe Sattler

Database & Information Systems Group,
Technische Universität Ilmenau, Germany
{first.last}@tu-ilmenau.de

Abstract. In this demo paper we describe the current prototype of our new platform LODHub that allows users to publish and share linked datasets. The platform further allows to run SPARQL queries and execute Pig scripts on these datasets to support users in their data processing and analysis tasks.

Keywords: Linked Open Data, data processing, data analysis, web platform

1 Introduction

Over the last years, the (Linked) Open Data movement has received a growing popularity. More and more public and government agencies as well as organizations publish data of common interest allowing to make use of it and building interesting applications. This is fostered by various hubs such as datahub.io, data.gov etc., which represent central registries for data sources.

However, creating added value from the published data requires typically to preprocess and clean the data, combine it with other data, and to create new datasets or build models. Results of such transformation and analysis tasks are twofold: first, the new datasets or models might be useful for other users in the form of curated datasets and second, the tasks could be reused for other datasets, too. Particularly, recent developments on big data technologies provide numerous useful building blocks for such an environment, e.g. scalable platforms like Hadoop or Spark or higher-level programming models and data flow languages like Pig or Jaql.

In [2] we argue that there is a need for a platform addressing these requirements by combining functionalities of Open Data management frameworks with an infrastructure for exploration, processing, and analytics over large collections of (linked) data sets while respecting access and privacy restrictions of the datasets.

In this demo we plan to show our initial results of building such a platform. Our LODHub platform provides services for uploading and sharing (RDF) datasets. Furthermore, it contains facilities to explore, query, and analyze datasets by integrating a SPARQL query processor as well as a visual data flow tool for designing and executing Pig scripts on the hosted datasets.

2 LodHub Services

The platform has several features that let users work with their datasets. However, it is currently only a prototype and some features are not completed yet.

2.1 Managing datasets

Upload Users can upload their datasets via the website. During the upload process, the user has the possibility to enter the dataset name, tags, and a short description. The tags can later be used to search in the users collection of datasets. An import feature that lets users upload datasets that are not in the linked data format (e.g., CSV files, Excel sheets, etc.) is not finished yet, but it is a work in progress.

Update During time, the contents of a dataset may change. These changes can be updated values for some particular statements, new additional statements, or removed statements. If the set of changes is small, one might consider it a new version of a dataset instead of a completely new one. To reflect this issue in LodHub, the platform supports versioning. This means it is possible to explicitly upload a new version of a dataset. By default, users will use the most recent version of a dataset. However, it is also possible to switch to an older version and, e.g., run queries on that version.

Collaboration The idea of LodHub is to allow users to work together on potentially large datasets. When a user uploads a new dataset, she or he becomes the owner of it and hence, has the permission to perform all operations on it. To also allow other users to work with this dataset, one can share the dataset to other users. With sharing we mean that the other users can see this dataset in their collection so that they are able to work with it. When sharing a dataset, the owner can choose what permission the other users should have on this dataset: *Read* allows to query the dataset, *Write* is like Read access plus the permission to upload new versions, *Share* is the permission to share the dataset to other users.

2.2 Querying datasets

Working with datasets means to run queries on them to find the needed information. However, there are different types of queries that users need to execute. On the one hand, there are the rather short ad-hoc queries that can be run directly on the datasets. On the other hand there are the data-intensive transformation and analysis tasks. LodHub supports both types of workload by providing two ways to formulate the queries.

Ad-hoc queries Since LodHub was designed around linked data, the main query language used on the platform is SPARQL. SPARQL queries can be used to instantly formulate a query on one dataset or even the union of many datasets. The user is presented a text input area where she or he can enter the query. The query is then directly executed by the underlying RDF framework (in our case Jena) and the result triples are presented on the website.

Analytical tasks Writing SPARQL queries can be cumbersome and requires the users to learn the language. Furthermore, there may be complex tasks that are too difficult to express in SPARQL, e.g., data transformation steps, or complex operators that are not even available in SPARQL. In this case, it is easier to formulate the query in a script language where data is processed in a streaming fashion to achieve high throughput.

To achieve a low entrance barrier, we provide a graphical editor that lets users create queries via drag and drop. Users can choose between several predefined operators which they can then connect to model the data flow between the operators. For each operator, its parameters like filter conditions, join attributes, or projection columns can be set individually. Thus, users intuitively build an operator tree, without having to care about language specific rules.

Each operator is translated into a Pig script statement. Pig, being a framework on top of Apache Hadoop, allows a distributed execution of the query and to load the data from a distributed file system (e.g., HDFS). Hence, this approach does not use the generated indexes from the used RDF framework. However, the data flow approach allows a high parallelization in a cluster environment.

Currently, the graphical editor produces Pig scripts only. However, the editor was designed so that it will be possible to also generate other languages from the graph. Thus, in a future version the editor will be able to generate traditional SPARQL queries and maybe other languages, too.

Data exploration To help people to understand the content of datasets and to find useful datasets that may contribute to the users question, LodHub allows to visualize how datasets are interlinked with each other, i.e., how many objects of a dataset occur as subjects in another dataset (and vice versa).

3 Architecture

The platform was written using the Play! framework[1]. Play's integration of Akka allows to easily run the application in a cluster environment. However, in the current development phase we concentrate on a single machine, but plan to distribute the application in a cluster to achieve better load distribution.

To store the datasets, we use the Jena TDB framework[2]. The SPARQL queries on these datasets are directly passed to the Jena library which then

[1] `http://www.playframework.com/`
[2] `http://jena.apache.org/documentation/tdb/`

evaluates the query. The Pig scripts are currently executed in `local` mode, i.e., they are not distributed to a cluster. However, this is just a configuration step, so that the application can be run in a cluster environment without having to change a lot of code.

The modular design of the application will enable us to easily replace the RDF store with another one or even to install a new store along with the others. Thus, we could use the for example our fast CameLOD [1] for analytical SPARQL queries that have to read a massive amount of data, while transactional queries are still handled by the Jena TDB store.

4 Demo

In our demo we will show the features that were described in the previous section. Users will be able to upload datasets, update existing ones, and to execute queries on the datasets. For the queries they can either type in traditional SPARQL queries or use our graphical editor to define a data flow and then generate a Pig script from the created graph.

A short demonstration of the current status of the platform can be found in this video:

`http://youtu.be/m4kKiBrw2m4`

In this demonstration we used several datasets which contain information about events, their location, date, and an URL for more information, as well as datasets containing information about cities and the country they belong to.

After an introduction of the dashboard that is the user's entry point to all actions, we run a SPARQL query with a `GROUP BY` and `HAVING` clause to find the subjects that have the same predicate two or more times.

Next, we show how to create more complex analytical data processing tasks using the graphical editor to create the Pig scripts. In this editor we create a data flow using a `LOAD`, `GROUP BY`, and a `PROJECTION` operator. For each operator, we enter the necessary parameters. `LOAD`: the path of the file to load and the schema, `GROUP BY`: the grouping column, and `PROJECTION`: the column names to project. After uploading a new dataset we create a second Pig script that uses a special `MATERIALIZE` operator. This operator allows to materialize the results of a Pig script into a new dataset that is immediately available to the user and can be used just like a normal dataset.

At the end of the demo video we show how to visualize the interlinks between selected datasets.

References

1. Hagedorn, S., Sattler, K.U.: Efficient parallel processing of analytical queries on linked data. In: OTM, pp. 452–469 (Sept 2013)
2. Hagedorn, S., Sattler, K.U.: Lodhub - a platform for sharing and integrated processing of linked open data. In: In proceeding of: 5th International Workshop on Data Engineering Meets the Semantic Web. pp. 260–262. IEEE (March 2014)