BigDataBus: Towards a Big Data Aggregation and Exchange Platform for eScience

Ana-Maria Oprescu[1], Paola Grosso[1], Pedro Bello-Maldonado[2], Yuri Demchenko[1], Cees de Laat[1]

[1] Informatics Institute, University of Amsterdam, Amsterdam, The Netherlands
[2] Florida International University, Florida, USA
emails: {a.m.oprescu, p.grosso, y.demchenko, delaat}@uva.nl, pbell005@fiu.edu

Keywords: data-centric, data storage, BigData service bus

1. Introduction

Currently, scientists are able to generate huge amounts of data that grow exponentially every year as newer technologies emerge. However, we are still unable to successfully connect various big data sites that belong to different organizations or research areas.

Cross-disciplinary discoveries are brought by inter-disciplinary research and exchange of data/experiments. The OSDC [1] platform is a growing venue for such data. However, there is a lack of tools to discover data content and to allow data exchange, independent of the store format. In a typical scenario (see **Fig. 1**), lab A has experimental data stored in a MySQL database, while lab B has experimental data stored in a HBase database. Both lab A and B would benefit by using the data in other lab, but a) it is unaware of its existence and relevance to their own experiments and b) it is stored in an incompatible format. Data exchange requires schemas and metadata for discovering the content of the data that is being shared. This problem becomes even more difficult when the data is not structured and tagged in a useful way at ingestion.

Within the UvA - OSDC PIRE program, we initiated the development of the BigDataBus, a data service that allows users from different domains to access relevant data without prior knowledge of *how* or *where* the data is stored. The BigDataBus will provide two types of tools: one that deals with categorizing the data content, by using generic semantics about it (scientific domain, if it is an experiment, benchmark, measurement, simulation); and one that deals with non-functional aspects of the data, e.g. type of database, geo-location. Our prototype implementation offers a simple set of APIs: a) a data provider-centric API, b) an user-centric API and c) a data service -to- data service API. Our approach allows for a flexible implementation of inter-operable data services, as well as the addition of emerging database paradigms. We also present initial results of investigating different parallelization strategies to improve the scalability of the BigDataBus.

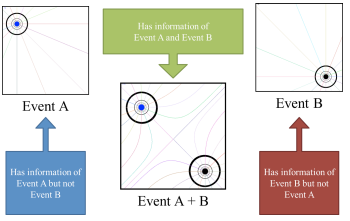

Fig. 1. A typical scenario for the Scientific Data Aggregation and Exchange.

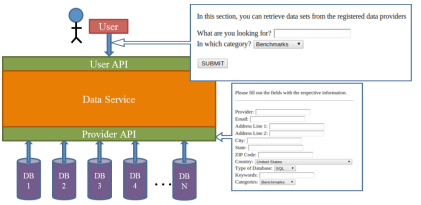

Fig. 2. BigDataBus Architecture

2. BigDataBus System Architecture

Fig. 2 shows the architecture of our BigDataBus, next to screenshots of the web interface of our prototype implementation.

- **The user-centric API -** Through the User API, the Data Service serves specific data to the user based only on (currently static) semantic queries. The user does not need to know where or how the data is stored. Our data service will translate the user request into database implementation- dependent queries and will forward these queries to known data providers. As the results of these queries arrive at the data service, the BigDataBus aggregates the data and presents it to the user.

- **The data provider API -** A data provider, i.e. the entity controlling the data acquisition, registers her data storage through the Provider API. The BigDataBus creates and maintains a list of registered databases along with their information, both functional (i.e. what type of data sets it contains) and non-functional (i.e. type of database, access points, data rights). Based on the functional information, the BigDataBus filters the data providers to which the database-dependent queries are forwarded, such that they fit the user-specified scientific domain (chemistry, physics), type of data source (experiment, measurement, simulation) and type of data (radiation, DNA).

- **The data service -to- data service API -** The BigDataBus service architecture allows for a completely decentralized deployment of a web of such data services. Through traditional service discovery techniques, different instances of the BigDataBus service could exchange information about their respective registered data providers. The specific data provider description may be enriched with information gathered while executing user requests.

3. Results and conclusions

We implemented the prototype on top of the Java EE [2] platform. We currently provide support for MySQL [3] and HBase [4] databases. Our initial evaluation shows that the overhead of the (currently) static semantic layer is negligible. The lightweight of the BigDataBus service renders it suitable for on-demand deployment, possibly in the cloud.

Data discovery and aggregation is very challenging, given that the same data might be represented differently in different scientific domains. An open question remains where dynamic semantic queries are concerned, that is replacing the current categories with dynamically extracted semantic information from the user request. Next, the semantics of the search engine must be further analyzed in order to reduce the complexity of the data aggregation.

References

1. OSDC – Open Science Data Cloud https://www.opensciencedatacloud.org/
2. Java EE - http://www.oracle.com/technetwork/java/javaee/overview/index.html
3. MySQL - http://www.mysql.com/
4. HBase - http://hbase.apache.org/