University of Zurich^UZH

Department of Informatics

University of Zürich
Department of Informatics
Binzmühlestr. 14
CH-8050 Zürich
Phone. +41 44 635 43 11
Fax +41 44 635 68 09
www.ifi.uzh.ch/dbtg

UZH, Dept. of Informatics, Binzmühlestr. 14, CH-8050 Zürich

Martin Noack

Prof. Dr. Michael Böhlen
Professor
Phone +41 44 635 43 33
Fax +41 44 635 68 09
boehlen@ifi.uzh.ch

Zürich, October 30, 2012

BSc Thesis
Datenbanktechnologie

Topic: Data Lineage and Meta Data Analysis in Data Warehouse Environments

Data is one of the most important assets of a bank. This asset becomes hard to handle if the bank is operating on a global scale and is itself a result of a series of mergers and acquisitions. Consequently, there are many flows of data throughout the company, supporting various needs such as bank operations, regulatory reporting or risk assessment. These data flows were usually built in several iterations over the years and mostly by not touching the existing parts but by further extending them.

The organic growth of data flows and the data transformations they contain led to a complex data processing infrastructure, which can be considered as a directed and attributed graph. Data consumers typically take data from a peripheral node of such a graph, where data has been modified and or filtered and aggregated in a way that its original source (the so called golden source) cannot be determined anymore.

The purpose of this assignment is to elaborate new ways of data lineage to overcome this situation and enable data consumers to backtrack the data they requested to its golden source(s). The setting consists of a set of data flows, represented through a graph, which is available as a triple set in the RDF (Resource Description Framework) format. The main edges of this graph are called mappings as they represent an atomic part of a data flow from one to many source data entities to a target data entity; including all the data transformations and operations in between.

As multiple sources and targets can be linked together each mapping poses a potential n-m relationship. Following such paths from last targets to first sources (data lineage) results in having to follow an exponentially growing number of paths. This project aims at using the logic in the mappings, i.e. the data transformation descriptions, to restrict path traversals. Therefore, the mapping's logic is described using a context free grammar. Now the system to be built is capable of distributing a query across the grammar rules. Thus all internals of a mapping can be queried once the parse tree has been established.

This approach can eliminate paths to be traversed in the data flow graph as the query can filter out mappings that do not meet a certain criterion, e.g., only paths shall be inspected where particular value ranges are included, such as high-volume orders or particular data formats, such as encrypted customer master data.

Imagine the situation where the bank needs to find the local branches that sold a special financial product to their customers. Considering as a given that there exists a system T that fetches various information from many systems and also the aggregated value of the product in question sold worldwide, the task is to find out the source systems that hold the unmodified information. The obvious solution is to start from system T and trace the paths back to the systems that hold the desired information. This includes following potentially unnecessary paths that do not lead to the desired information – potentially the number of paths to follow grows exponentially. A way to ignore some paths is to query the information transforming mappings if the next source back in the chain of mappings does actually provide the desired information (or aggregate). If this is not the case, this source and the attached sub-graph can be ignored.

Goals

The expected results of the bachelor thesis include as a minimum:

- Problem statement including data lineage use case description
- Elaboration of sample data lineage queries with the existing grammar against the data flow graph
- Analysis and description of the resulting parse trees
- Definition of graph traversal patterns based on the sample data lineage queries as well as their parse trees
- Explanation of the limitations with special regard to potential extensions of the given grammar

Further results could comprise of:

- Specification of the query engine
- Implementation of a prototype on top of the existing Credit Suisse infrastructure

References

[1] Buneman, P., Wang-Chiew, T. (2007). Provenance in databases. ACM SIGMOD international conference on Management of data (pp. 1171-1173). Beijing, China: ACM.

[2] Cheney, J., Chiticariu, L., Wang-Chiew , T. (2009). Provenance in Databases: Why, How, and Where. In Provenance in Databases (pp. 1-95). Hanover, MA, USA: now Publishers Inc.

[3] Christopeit, D., Böhlen, M., Kanne, C.-C., Mazeika, A. (2011). Querying Versioned Software Repositories. Adbis (pp. 42-55). Vienna, A: Springer.

[4] Glavic, B., Dittrich, K. R. (2007). Data provenance: A categorization of existing approaches. BTW, Germany. Aachen: BTW.

Supervisor:	Claudio Jossen and Dietrich Christopeit
Start date:	15.10.2012
End date:	14.4.2013

University of Zürich
Department of Informatics

Prof. Dr. Michael Böhlen