Thomas Brenner

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

Zürich, March 6, 2013

Vertiefung (3 ECTS)
Datenbanktechnologie

Topic: Constant Interval Extraction using Hadoop

Temporal databases are those that provide support for the time dimension. Using such databases allows to keep track of historical information and data changes. Each relation in a temporal relation has two additional columns: `TS` (time start) and `TE` (time end); defining the time interval during which a tuple is valid.

Apache Hadoop is a software framework that supports distributed applications with large amounts of data on large clusters of commodity hardware. PL/pgSQL[1] (Procedural Language/PostgreSQL) is a programming language that allows user to implement procedural funcations over databases.

To automatically evaluate such nontemporal functions over temporal data, it is required to determine the constant intervals during which the data in the database is constant (data is not modified). The aim of this project is to provide a Hadoop implementation, that connects to a PostgreSQL database, extracts time points and computs the constant intervals.

Tasks

1. Install Hadoop and get the `WordCount.java` example running[2].
2. Use a connection adaptor (Hadoop-PostgreSQL connector) to allow Hadoop to connect and retrieve data from a PostgreSQL database.
3. Implement a Hadoop application, that:
   - Extracts all time points from all relations of a temporal database and sort them in

[1] http://www.postgresql.org/docs/9.1/static/plpgsql.html

[2] http://hadoop.apache.org/docs/r1.1.1/mapred_tutorial.html

   ascending order

   - Create constant intervals from each two successive time points
   - Example:
     - extracted time points: $\{3, 1, 9, 5\}$
     - generated constant intervals: $\{[1, 3], [3, 5], [5, 9]\}$

4. Experiment the implementation with TPC-H benchmark, analyzing the effect of the following on the performance (time):
   - number of mappers
   - number of reducers
   - data size in PostgreSQL
5. Report (5-10) pages that discusses:
   - explanation of the implemented application
   - experimental results and analysis
   - at which point, does it become beneficial (faster) to compute the intervals using Hadoop instead of PostgreSQL
   - strengths and weaknesses of your implementation

Supervisor:	Amr Noureldin (noureldin@ifi.uzh.ch)
Start date:	18 February 2013
End date:	15 April 2013

University of Zürich
Department of Informatics

Prof. Dr. Michael Böhlen