Topic1: Combine sql and nosql to improve efficiency

Topic2: An algorithm to predict data feature and choose proper database

Topic3: An algorithm to collect useful business data through Facebook, GoogleMap or Twitter

注：git里新加了几个paper, 如果合适可以再看下paper的reference,

另外还有以下链接关于提高MySQL的查询效率

SQL数据库如何加快查询速度: <http://blog.csdn.net/jking1989/article/details/6430603>

数据库大数据访问的解决方法: <http://www.studyofnet.com/news/379.html>

提高数据库处理查询速度: <http://www.cnblogs.com/surge/archive/2012/09/21/2696882.html>

Topic1

**Background**:

Use case is online inquiry platform INKIDU for user-designed questionnaires

The **popularity** of NoSQL databases keeps growing and more companies have been moving away from relational databases to nonrelational NoSQL databases. In this paper, the **partitioning** of a relational data model of an inquiry system into semi-structured and relational data for storage in both SQL and NoSQL databases is shown. Furthermore, the **CAP theorem** will be applied to categorize the storing of the correct parts of the model into their corresponding databases. As a result of these reorganizations and the introduction of additional histogram data, overall **performance improvements** of about 93% to 98% are achieved.

**Problem**:

1 companies and organizations have started to **collect huge amounts of data**, several types of nonrelational data stores are preferred over the relational ones

2 In most projects there is either **no time for developing a connection driver** or it is unsafe to rely on community projects.

3 The effective structure and amount of answers of an answer set is finally determined by the questions, which the user decided to give an answer for. Hence, **there is no way to predict the exact amount and structure of data.** Only a rough estimate of the structure is known: A list of key-value-pairs. The current way of **organizing answer sets in relational data model** requires more effort such as data redundancies and index data and produces a structure which is easy to read for machines and humans as well.

4 There are parts of the data model which do require ACID transactions and strong consistency as well as parts which do not require this functionality.

**Solution:**

1 **NoSQL** databases store and organize data as collections of documents, rather than as structured tables with uniformsized fields for each record. With these databases, users can add any number of fields of any length to a document

2 The presence of a proprietary connection driver can be a very important factor for choosing the appropriate NoSQL database. Many of them use Representational State Transfer (REST) [8,14] where often a community-developed connection driver has to be used (Redis, Apache CouchDB), but some also come with a proprietary connection driver (**MongoDB**).

3 The approach of reorganizing the above data structure is that **grouping all answer sets into a single record.** Partitioning data model into Semi-structured and relational data for SQL and NoSQL

4 Parallel usage of a traditional SQL database and a NoSQL document store.

**Reference:**

Ebel, Marius, and Martin Hulin. "Combining relational and semi-structured databases for an inquiry application." *Multidisciplinary Research and Practice for Information Systems*. Springer Berlin Heidelberg, 2012. 73-84.