A Methodology of Constructing Canonical Form Database Schemas in a Multiple Heterogeneous Database Environment

JeongSeok Lim
University of Connecticut, USA

Dong-Guk Shin
University of Connecticut, USA

The Query Clearing House (QCH) model aims at achieving an ideal heterogeneous multiple database environment in which users can submit queries without concern for the location of the data or the specifics of the relevant database schemas. One important prerequisite for building such an environment is to organize local database schemas in a cohesive way so that a systematic method of determining data relevancy can be developed. In this paper, we propose a method for converting multiple local relational schemas into canonical form expressions. The resulting canonical form schema expressions include much more descriptive data semantics than what is offered by the original database schemas. These expressions also include information for mapping between the terms used in the canonical form expressions and their counterparts implemented in local databases. We include a number of examples illustrating how the canonical form schema expressions are produced.

Databases are usually developed independently by each group or organization to meet their own individual requirements. However, as interactions between groups and organizations become increasingly common, many applications end up requiring data not from one single database but from several related databases. Thus, users have to deal with databases whose designs have never been coordinated. Interoperability between these multiple heterogeneous databases has been a hot research topic.

Previously in Lim et al. (1997), we proposed the Query Clearing House (QCH) model, which aims at assisting users in finding information in multiple heterogeneous databases. QCH plays the role of mediating between the end users’ data needs and the local databases’ data publication needs.

The objective of this paper is to propose a uniform schema conversion method with which one can transform local relational database schemas into one cohesive format—i.e., into canonical form schema expressions. These canonical form expressions are then used as the basis for constructing two important resources for the QCH model: meta-data and mapping libraries. The meta-data includes not only the schematic information for the local databases, but also additional semantics that are not available in the database schemas themselves. The mapping library for each database has the mapping information between the general terms used in the meta-data and the specific terms used in the database schema so that a user’s query can be systematically transformed into a database-specific query expression.

We use $L_k$, which was initially presented in Shin (1991, 1994), as the representation formalism for expressing canonical form schemas and the meta-data. $L_k$ is known for its flexible, descriptive power, which facilitates expressing general concepts at the desired level of granularity. The language is also known for its versatile association mechanism, which
allows dynamic ways of linking related concepts on the fly.

The rest of the paper is organized as follows. First, we review related works. The subsequent section includes a description of the QCH model, an overview of LK, and an introduction to the procedure used for meta-data construction. A detailed method for transforming the relational database schemas into canonical forms is given. Finally, the conclusion is presented.

Related Works

Two well-known families of database integration approaches are the global schema approaches (Larson et al., 1989; Reddy et al., 1994) and the multidatabase query language approaches (Kim et al., 1993). The key ideas of the global schema approaches (Batini et al., 1986) are identifying the schemas’ semantics and potential conflicts, and constructing a global schema by merging related schemas and resolving identified schema conflicts. On the other hand, the multidatabase query language approaches aim at producing one powerful query language with which end-users can query local heterogeneous databases by following one uniform query syntax.

A few recent works have focused on developing a common data model with which local database schemas could be reorganized to form a global schema. Bright et al. (1994) proposed the Summary Schemas Model (SSM), which is designed to represent a concise, abstract description of the semantic contents of a group of input local schemas. Garcia-Solaco et al. (1995) presented the BLOOM model, which is a semantic extension of object-oriented models. The BLOOM model includes features, such as metaclasses, metametaclasses, abstractions between objects, object aggregation, and object dependency description, that the object-oriented models do not include. Sciore et al. (1994) used meta-attributes to specify the context associated with attribute values. They proposed a semantic-value schema, a semantic-value specification, and C-SQL (Context-SQL). The newly proposed features are to be used to resolve data conflicts in multiple databases.

All of the above-mentioned global schema approaches are limited in their ability to capture and represent the contextual knowledge surrounding the database schemas. For example, the SSM model expresses semantics associated with table names and attribute names, but fails to express even those relationships that are commonly found in the ER model. The BLOOM model includes the well-understood relationships, such as super- and sub-class relationships, but it is unclear how such relationships could be exploited in semantics-oriented processing. The C-SQL approach lacks the ability to explicitly express (i) the relationships associated with foreign keys; and (ii) many-to-many correspondence relationships between relations in different databases.

We believe that a better model for representing a global schema should include a way of expressing and exploiting contextual knowledge that is beyond what we normally call the data semantics. The level of semantics that should be captured and expressed should be close to the level of semantics that would normally be used in natural language processing.

The Query Clearing House Model

The Architecture

The Query Clearing House (QCH) model aims at building an ideal heterogeneous multiple database environment in which a user can submit queries without knowing which database(s) might include the data he/she is looking for and in what format such data is stored. Figure 1 shows the architecture of the QCH model, which consists of three components: User Interface, Clearing House, and Agents. The User Interface allows the user to formulate queries. The interface subsequently submits the query object to the clearing house along with the user’s profile. This front end is also responsible for interacting with each local database that has a query result to send to the user.

The middle unit, Clearing House (CH), conducts the main tasks of the QCH model—interpreting the input query, determining which local database(s) is/are relevant to the query, and sending the interpreted queries to their corresponding database(s). Each local database that receives a query sends the query results directly to the user. The middle unit CH contains three major components: meta-data, mapping libraries, and processing module. Meta-data and mapping libraries
Related Content

Concept-Oriented Query Language for Data Modeling and Analysis
[www.igi-global.com/chapter/concept-oriented-query-language-data/52297?camid=4v1a](www.igi-global.com/chapter/concept-oriented-query-language-data/52297?camid=4v1a)

Retrieval Optimization for Server-Based Repositories in Location-Based Mobile Commerce
[www.igi-global.com/chapter/retrieval-optimization-server-based-repositories/4302?camid=4v1a](www.igi-global.com/chapter/retrieval-optimization-server-based-repositories/4302?camid=4v1a)

Big Data Warehouse Automatic Design Methodology
[www.igi-global.com/chapter/big-data-warehouse-automatic-design-methodology/85453?camid=4v1a](www.igi-global.com/chapter/big-data-warehouse-automatic-design-methodology/85453?camid=4v1a)

Extended Spatiotemporal UML: Motivations, Requirements and Constructs
[www.igi-global.com/article/extended-spatiotemporal-uml/3255?camid=4v1a](www.igi-global.com/article/extended-spatiotemporal-uml/3255?camid=4v1a)