1. Introduction

Modern database schemata are very large and complex. Even modest schemata often have hundreds of relations; the largest have thousands. As a tool to manage this complexity, the notion of a *view* — a "window" schema which provides partial information about the database — has arisen. Views are used not only to provide simplified and manageable interfaces to the user; they are also essential tools for the study of many issues in modelling and managing the complexity of modern database systems.

The research proposed here includes three projects on the foundations of database systems, each involving views in an essential way. The first extends the foundational work on view structure and the support of updates through views which has been a cornerstone of the research of the author for many years. The other two are new efforts for the author, both begun in 2005, with colleagues, one in Umeå and the other in Kiel, Germany.

2. Three Research Projects Based upon Database Views

2.1 View Updates and the Complexity of View Axiomatization

The support of updates on database views has long been known to be a very difficult task. Ad hoc approaches typically result in difficult update anomalies, and in any case require the user of the view to understand not only that view, but the entire database schema. These problems are addressed elegantly in *closed update strategies*, first forwarded in the classical paper of Bancilhon and Spyratos [BS81]. With such strategies, the entire family of view updates must be well behaved as a unit, in that no knowledge of the main schema beyond the view is required of decide the admissibility of or effect of an update to the view. Closed update strategies continue to find application in support of the latest technology, such as data synchronization on mobile devices [FGM*05].

A major limitation of this approach is that the way in which view updates are reflected to the main schema is not unique; rather, it depends upon the choice of a so-called complementary view. In a recent series of papers, the author has addressed this shortcoming systematically. In [Heg04], it is shown that by using the natural order structure on databases, it is possible to show that the there is only one way to reflect an *order-based* view update to the main schema under a closed strategy, regardless of the choice of complement. In [Heg06], it is shown that in a stronger but nonetheless realistic context, all updates defined by so-called *atomic strategies*, order-based or not, have unique realizations in the main schema. In [Heg05], the closely related question of the relative complexity of view axiomatization is examined in detail.

In the proposed research, several new topics will be investigated. First of all, the possibility of relaxing the axioms for a closed strategy somewhat, along the lines of so-called "decreasing complements" [GPZ88], will be investigated in the context of providing unique update strategies. Second, the extension of the existing results to views with built-in operations, as modelled in [Tha05], will be investigated. Finally, the integration of the existing results with modern predicate-based formulations of database updates, such as that studied in [Lin02], will be pursued.

2.2 Managing Schema Complexity via Database Componentware

This research is a joint effort with Bernhard Thalheim of Christian-Albrechts-Universität zu Kiel in Germany, who has already laid the foundations with his pioneering work on database componentware [Tha03], [Tha05]. The goal of this work is to describe the structure of complex database schemata as the interconnection of relatively simple *components*, in an effort to understand the behavior of such systems and to manage the inherent complexity.

From April through June of 2005, the author visited Kiel, during which the foundational ideas of how the theory of views could be used in this investigation were discussed thoroughly, and an approach to obtaining initial results was established. The short-term research plan involves a two-week return visit by the author to Kiel during October 2005, at which time a publication-quality paper describing the results which have been obtained will be completed. These results, which are set within the context of the classical relational model, include a remarkable mapping of the abstract notion of component developed in [Tha05] to the projection views involved in an acyclic decomposition. They also provide insight into the crucial rôle which keys play in the formulation of components.

The longer-term research plan involves several steps. First, the results which have already been obtained will be extended to the more general HERM data model [Tha00], which will provide needed insights into how components pan out in the context of the entity-relationship model of data. Second, the question of how the initial results extend to contexts in which the components have explicit update operations, as modelled in [Tha05], will be pursued. Finally, the question of how a more general notion of acyclic decomposition into general views, such as that developed by the author in [Heg93], can be used to formulate an abstract notion of component will be examined. It is anticipated that the author will visit Kiel once again during 2006 for three to four months, in order to carry out this research in close collaboration with Professor Thalheim and his research group.

2.3 Completeness of Answers to Database Queries

Traditional (non-deductive) databases consists of atoms of information (tuples in the relational model). Every such atom is taken to represent a true fact. If all such true facts are represented in the database, it is called *complete*. Generally, databases are not complete, although some components may be. For example, a database of courses given by the Faculty of Science and Technology may be incomplete in general, but complete for courses given by the Department of Computing Science. When providing the answer to a query, it is very useful for the system to be able to provide completeness information about that answer; that is, to indicate which components of the answer are guaranteed to be complete, and which are not. Although abstract formalizations of the notion of completeness for answers to queries have existed for some time [Mot86], [Hal00], there has been little concrete work on the problem of identifying components (or subqueries) of queries on standard relational databases which provide complete answers. The purpose of the work proposed here is to conduct such an investigation.

This research is a joint effort with Michael Minock of Umeå University, and is based upon his preliminary results reported in [Min04]. During the summer of 2005, when the collaboration began, Minock and the author re-worked the foundations of the approach entirely. The approach is to represent the complete component of the main database as a view, called the *completeness view*, and then to develop techniques which identify subqueries of the given query which can be factored through that view. Such subqueries, by construction, represent complete answers. Because the problem of deciding whether an arbitrary query (expressed in first-order logic) always provides complete answers is undecidable, it is necessary to investigate this question in limited domains.

The research plan is to begin with restrictive cases for which solutions are known, and to relax gradually the conditions. Specifically, it is known how to obtain solutions when neither the completeness view nor the queries contain negation. Initially, the completeness view will be assumed to be free of negation, since such constraints are very realistic. By systematically allowing specific forms of negation, coupled with judicious limitations on quantifier forms, it is anticipated that useful cases in which the completeness components of the query will be identified.

References

The papers listed below which are by the author of this proposal are available in PS.gz and PDF format at the web site `http://www.cs.umu.se/~hegner/Publications/`.

[BS81] Bancilhon, F. and Spyratos, N., "Update semantics of relational views," *ACM Trans. Database Systems*, **6**(1981), pp. 557–575.

[FGM*05] Foster, J. N., Greenwald, M. B., Moore, J. T., Pierce, B. C., and Schmitt, A., "Combinators for bi-directional tree transformations: a linguistic approach to the view update problem.," in: *Proceedings of the 32nd ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2005, Long Beach, California, USA, January 12-14, 2005*, pp. 233–246, 2005.

[GPZ88] Gottlob, G., Paolini, P., and Zicari, R., "Properties and update semantics of consistent views," *ACM Trans. Database Systems*, **13**(1988), pp. 486–524.

[Hal00] Halevy, A. Y., "Theory of answering queries using views," *SIGMOD Record*, **29**(2000), pp. 40–47.

[Heg93] Hegner, S. J., "Characterization of desirable properties of general database decompositions," *Ann. Math. Art. Intell.*, **7**(1993), pp. 129–195.

[Heg04] Hegner, S. J., "An order-based theory of updates for database views," *Ann. Math. Art. Intell.*, **40**(2004), pp. 63–125.

[Heg05] Hegner, S. J., "Uniqueness of constant-complement update strategies for database views," 2005, submitted for journal publication.

[Heg06] Hegner, S. J., "The complexity of embedded axiomatization for a class of closed database views," *Ann. Math. Art. Intell.*, **46**(2006), to appear.

[Lin02] Link, S., "Towards a tailored theory of consistency enforcement in databases," in: Eiter, T. and Schewe, K.-D., eds., *Foundations of Information and Knowledge Systems, Second International Symposium, FoIKS 2002 Salzau Castle, Germany, February 20-23, 2002, Proceedings*, pp. 160–177, Springer, 2002.

[Min04] Minock, M., "Managing database incompleteness with the guarded fragment," in: *Workshop on Guarded Logics, European Summer School in Logic Language and Information, (ESSLLI-04), Nancy, France, August 2004*, pp. 27–45, 2004.

[Mot86] Motro, A., "Completeness information and its application to query processing," in: *VLDB'86 Twelfth International Conference on Very Large Data Bases, August 25-28, 1986, Kyoto, Japan, Proceedings*, pp. 170–178, Morgan Kaufmann, 1986.

[Tha00] Thalheim, B., *Entity-Relationship Modeling*, Springer-Verlag, 2000.

[Tha03] Thalheim, B., "Database component ware," in: *Database Technologies 2003, Proceedings of the 14th Australasian Database Conference, ADC 2003, Adelaide, South Australia, February 2003*, pp. 13–26, Australian Computer Society, 2003.

[Tha05] Thalheim, B., "Component development and construction for database design," *Data Knowl. Eng.*, **54**(2005), pp. 77–95.