VIRTUAL MERGING OF DATABASES

AMIHAI MOTRO

A DISSERTATION

in

COMPUTER AND INFORMATION SCIENCE

Presented to the Graduate Faculties of the

University of Pennsylvania

in Partial Fulfillment of the Requirements

for the Degree of Doctor of Philosophy.

1981

P. Buneman

Supervisor of Dissertation

A. K. Joshi

Graduate Group Chairperson

ACKNOWLEDGEMENTS

I wish to express my gratitude to Peter Buneman, the supervisor of this dissertation, for introducing me into the world of databases and for providing me with continuous inspiration (mostly during Tuesday lunches). I am indebted to Jean Gallier, the Chairman of my Dissertation committee for the careful review of this manuscript, and for his generous help in many other small ways. The other members of this Committee, Eric Clemons, Aravind Joshi and Howard Morgan, have offered criticism and suggestions, and for these I am thankful. To my friend Chaim Broit, for the long hours of stimulating (and sometimes heated) discussions, which have helped me gain insight into complex problems. And to Amir Pnueli, for his valuable comments and for making me aware of the passage of time through his (much appreciated) seasonal visits to the Penn campus. Finally, I am indebted to my wife Pnina, for her love and encouragement, and for providing us with essential distractions (chiefly in the form of movies). This work is dedicated to my parents, Rahamim and Elisheva Motro, as a token of appreciation for their commitment to my education.

TABLE OF CONTENTS

Chapter 1 - Introduction

1.1 Physical and Virtual Restructuring..... 1

1.2 Sub-views and Super-views..... 2

1.3 Ad-hoc vs Systematic Merge..... 5

1.4 Research Goals..... 6

1.5 Outline of Research..... 9

Chapter 2 - Review of Related Research

2.1 Introduction..... 10

2.2 Database Restructuring..... 11

2.3 Distributed Database Management Systems..... 16

2.4 Sub-views..... 21

2.5 Incremental Modeling..... 24

Chapter 3 - The Database Model

3.1 Introduction..... 28

3.2 Objects and Attributes..... 31

3.3 Database Schema..... 33

3.4 Database Assignment..... 38

3.5 The Abstract Database Model..... 45

3.6 Some Modeling Examples..... 51

Chapter 4 - The Union of Two Databases

4.1 Introduction..... 56

4.2 Base and Span..... 58

4.3 Consistency, Containment and Equivalence..... 61

4.4 Database Union..... 62

4.5 Some Union Examples..... 68

Chapter 5 - Automatic Merge

5.1 Introduction..... 72

5.2 Abstract Databases: Equivalent Definition.... 73

5.3 Database Transformations..... 77

5.4 A Merging Algorithm..... 85

5.5 Storage Considerations..... 90

5.6 Normal Form Databases..... 92

Chapter 6 - Structural Transformations

6.1 Introduction..... 95

6.2 Add and Aggregate..... 96

6.3 Delete and Telescope.....106

6.4 Meet, Join and Fold.....111

Chapter 7 - Cooperative Merge

7.1 Introduction.....114

7.2 Retraction.....116

7.3 Normalization.....120

7.4 A Detailed Example.....124

7.5 A Cooperative Algorithm.....129

Chapter 8 - Query Decomposition

8.1 - Introduction.....133

8.2 - A Basic Query Language.....135

8.3 - Recording the Restructuring History.....139

8.4 - Single Step Translations.....142

8.5 - A Decomposition Algorithm.....152

8.6 - A Detailed Example.....160

Chapter 9 - Implementation Issues

9.1 - Introduction.....164

9.2 - Identifying Objects.....165

9.3 - A Simple Implementation.....167

9.4 - Outline for a More Sophisticated System.....172

Chapter 10 - Other Data Models

10.1 - Introduction.....175

10.2 - CODASYL Databases.....177

10.3 - Relational Databases.....183

Chapter 11 - Conclusion

11.1 - Summary of Contributions.....186

11.2 - Directions for Further Research.....191

References.....195

Index.....202

ABSTRACT

Even with the increased use of large and complex databases, it often happens that the information required for a specific application extends over two or more physically independent databases. The writing of such applications is considerably simplified if the databases appear to the program as a single integrated database. However, the cost of performing any physical restructuring may be prohibitive and may impose unnecessary constraints on the structure and content of the database as it is viewed by the original users.

A method was developed that performs a virtual merge of existing independent databases: it presents the user with a larger conceptual structure that may be queried without compromising the independence of the existing databases. This process is in some sense the inverse of constructing "user views": given two or more logical schemas, what larger schema, or superview, has these schemas as user views?

Among the issues involved: what is a suitable data model (a semantic data model is introduced and investigated) and can the method be adjusted for other data models; can a

superview be inferred "automatically" from given schemas; and how to use schema mappings to translate superview queries into queries against the actual databases (and translate back the answers). An experimental Virtual DBMS was constructed that enables the user to construct and query such superviews.

Since the method may be extended to other types of database restructuring, it suggests the general concept of a <u>virtual database</u>: instead of the customary database intention (schema) and extension (data), a virtual database consists of an intention and a mapping.