1-1-1997

On the design of a secure data warehouse

Rihui Wang

University of Nevada, Las Vegas

Follow this and additional works at: https://digitalscholarship.unlv.edu/rtds

Repository Citation

https://digitalscholarship.unlv.edu/rtds/3390

This Thesis is brought to you for free and open access by Digital Scholarship@UNLV. It has been accepted for inclusion in UNLV Retrospective Theses & Dissertations by an authorized administrator of Digital Scholarship@UNLV. For more information, please contact digitalscholarship@unlv.edu.
INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps. Each original is also photographed in one exposure and is included in reduced form at the back of the book.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

UMI

A Bell & Howell Information Company
300 North Zeeb Road, Ann Arbor MI 48106-1346 USA
313/761-4700 800/521-0600

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
ON THE DESIGN OF
A SECURE DATA
WAREHOUSE

by

Rihui Wang
Bachelor of Science
Hunan University, China
1991

A thesis submitted in partial fulfillment
of the requirements for the degree of

Master of Science
in
Computer Science

Department of Computer Science
University of Nevada, Las Vegas
December 1997
The Thesis of Rihui Wang for the Degree of Master of Science in Computer Science is Approved.

Roy Hironshi Ogawa 8/29/97
Chairperson, Roy Ogawa, Ph.D.

John T. Minor
Examining Committee Member, John Minor, Ph.D.

E. A. Yfantis, Ph.D.
Examining Committee Member.

Samir Moujaes, Ph.D.
Graduate Faculty Representative.

Ronald W. Smith, Ph.D.
Dean of the Graduate College.

University of Nevada, Las Vegas
December 1997
ABSTRACT

On the Design of a Secure Data Warehouse

by

Rihui Wang

Dr. Roy Ogawa, Examination Committee Chair
Professor of Computer Science
University of Nevada, Las Vegas

The data warehouse is the combination of subject-oriented, authoritative, integrated databases designed to support the DSS (decision support). Data mining means to extract useful information which is previously unknown from information sources. The topic of data warehousing and data mining encompasses architectures, algorithms and tools for collecting selected data from multiple databases or other information sources into a single repository known as a data warehouse which is structured to facilitate query or analysis in supporting decision making.

In this thesis, we will discuss the principles, architecture, and implementation of data warehouses. We will also discuss construction of data mining and some algorithms applied to data mining. We will do some research on the security of data warehouses.
## TABLE OF CONTENTS

- ABSTRACT ......................................................................................................................... iii
- LIST OF FIGURES ................................................................................................................ vi
- ACKNOWLEDGMENTS .......................................................................................................... vii
- CHAPTER 1 INTRODUCTION ............................................................................................... 1
  - 1.1 Overview ...................................................................................................................... 1
  - 1.2 Thesis Organization .................................................................................................... 2
- CHAPTER 2 A STUDY OF DATA WAREHOUSES ..........................................................3
  - 2.1 Introduction to Data Warehouses ............................................................................. 3
  - 2.2 Data Warehouse Principles ........................................................................................ 3
  - 2.3 Data Warehouse Implementation ............................................................................... 5
  - 2.4 Metadata ..................................................................................................................... 7
  - 2.5 Architecture of Data Warehouse ................................................................................. 8
    - 2.5.1 Basic Architecture of Data Warehouse .......................................................... 8
    - 2.5.2 Client/Server Data Warehouse Architecture .................................................. 10
  - 2.6 Distributed Data Warehouses ..................................................................................... 10
    - 2.6.1 Introduction .......................................................................................................... 10
    - 2.6.2 Structure of Distributed Data Warehouse .................................................. 13
    - 2.6.3 Summary for Distributed Data Warehouses ................................................ 13
  - 2.7 Examples of Data Warehouses ................................................................................... 13
  - 2.8 Data Warehouse Conclusions ..................................................................................... 15
- CHAPTER 3 A STUDY OF DATA MINING ...................................................................... 16
  - 3.1 Introduction .........................................................................................................................16
  - 3.2 Data Mining Algorithms .................................................................................................17
    - 3.2.1 Introduction ........................................................................................................... 17
    - 3.2.2 The Functions of the Model ............................................................................. 18
    - 3.2.3 Model Representation ......................................................................................... 19
    - 3.2.4 Search algorithm .................................................................................................. 19
  - 3.3 A Construction of Data Mining System ....................................................................21
  - 3.4 Data Mining Conclusions ...............................................................................................23
- CHAPTER 4 SECURITY ISSUES OF DATA WAREHOUSES ........................................ 24
  - 4.1 Introduction .......................................................................................................................24
  - 4.2 Security Policies and Mechanisms .............................................................................. 24
  - 4.3 Security Issues ..................................................................................................................25
  - 4.4 Security Requirements ....................................................................................................27
  - 4.5 Protection Mechanisms ..................................................................................................28
- CHAPTER 5 SECURITY MODELS FOR DATA WAREHOUSES .................................... 29
  - 5.1 Access Control Model .....................................................................................................29
    - 5.1.1 Access Control .................................................................................................... 29
    - 5.1.2 Security Model ................................................................................................... 30
5.2 Security-Level Control Model ......................................................... 31
   5.2.1 Security Model ................................................................. 31
   5.2.2 A Simplified Case Study on Security Model ......................... 38

CHAPTER 6 CONCLUSIONS .......................................................... 45
   6.1 Summary on Data Warehouses and Their Security .................. 45
   6.2 Future Work ........................................................................ 45

REFERENCES .................................................................................. 47
LIST OF FIGURES

Figure 2-1 The Issue of Time Variance ................................................................. 6
Figure 2-2 A Basic Data Warehouse Architecture .............................................. 9
Figure 2-3 A Simple Client/Server Data Warehouse Architecture .................... 11
Figure 2-4 A Centrally Controlled, Centrally Housed Data Warehouse ............. 12
Figure 2-5 A Typical Distributed Data Warehouse ........................................... 14
Figure 4-1 The Secure Data Warehouse Model .................................................. 32
Figure 4-2 An Authorization Model .................................................................... 33
Figure 4-3 Security Hierarchy in Data Warehouse ............................................. 35
Figure 4-4 A Data Warehouse Security Model ................................................... 36
Figure 4-5 A Sample Case for Data Warehouse Security ................................. 39
Figure 4-6 An Example of the Role Lattice ....................................................... 40
Figure 4-7 Example of the Authorization Type Lattice ....................................... 41
Figure 4-8 Views for the Airline and the Travel Agent ....................................... 43
ACKNOWLEDGMENTS

I would like to acknowledge many people for their roles in making this thesis a reality.

First, and definitely foremost, is Dr. Kia Makki. Not only did Dr. Makki spend a lot of time guiding me through the research, but he also gave me valuable ideas, and made many technical contributions. Despite his busy schedule, he was always there to help. His excellent suggestions and criticisms were very helpful in the development of this work. Although Dr. Makki was not able to formally finish being my thesis advisor, he has acted as such up to the final steps.

I would like to express my deepest gratitude and appreciation to Dr. Nikki Pissinou from the Center for Advanced Computer Studies at the University of Southwestern Louisiana. Her valuable advice has had a great impact on this document.

I would also mention Dr. Roy Ogawa, who took over the formal steps finishing the requirements for the MS degree, a smaller role than Dr. Makki's but certainly necessary. I am deeply indebted to him for personal and moral support as well as guidance in research and ethics. I also would like to thank other members of my thesis committee Dr. John Minor and Dr. Evangelos Yfantis for reading my thesis and providing me with useful suggestions, comments and criticisms for the work and Dr. Moujaes, the external member for his time. I also thank Dr. Larmore for his help during my graduate studies.
1.1 Overview

Satisfying both the operations and the information needs of a company (those information needs probably include decision support systems [DSS] or executive information systems [EIS]) can be difficult. Databases structured to meet operations needs are often incapable of adequately providing information analysis, while attempts to provide information often hamper the performance and throughput of operations. Data warehouses, some claim, will be increasingly important in information management throughout the 1990s. Varying degrees of summarization of operational data, maintenance of historical information, consolidation of a variety of information sources, and other capabilities may move forwards solving the problem of managing data for information applications. More importantly, data warehouse trends, specifically in the area of distributing information with DSS/EIS applications that can access a variety of "mini-warehouses", will affect the use of data warehouse technology over the next few years [15].

The amount of information stored in databases is growing explosively so large amounts of data need to be summarized or reduced to only the descriptions of the forms. In recent years, there has been an emerging research area called data mining or knowledge discovery, that addresses the problems in finding implicit, previously unknown, and potentially useful patterns from large databases [13].

Maximizing the efforts of warehousing and subsequently mining data helps us to achieve enormous benefits. These benefits includes customer service improvement, decision making support and gaining ultimate advantages.
1.2 THESIS ORGANIZATION

The remaining chapters of this thesis are organized as follows: Chapter 2 discusses the principles, architecture, and implementation of data warehouses and focuses on the topic of data mining and its construction followed by some algorithms applied to data mining.

Chapter 3 and 4 cover some research on the security of data warehouses. Chapter 5 presents the conclusions.
CHAPTER 2

A STUDY OF DATA WAREHOUSES

Our research draws from two diverse yet significantly important areas: data warehouses and data mining. In this chapter, we provide an overview of data warehousing.

2.1 Introduction to Data Warehouses

One of the most frustrating problems facing many corporations today is the inability to receive online, timely, consolidated and flexible reports reflecting corporate data. It is the data warehouse approach that addresses this problem and provides the framework for decision support systems (DSS) and executive information systems (EIS). The importance of data warehousing stems from the need for enterprises to gather specific portions of their information into a single place for in-depth analysis, and the desire to decouple such analytical processing from the on-line transaction systems. Since decision support is often the goal of data warehousing, data warehouses may be tuned for decision support and vice-versa [17]. This data warehousing is the process whereby organizations extract value from their informational assets through the use of special stores called data warehouses [1].

A data warehouse is not a database, though data warehouses are implemented as Database Management systems (DBMS). The purpose of the data warehouse is not to support an organization’s operations or transactions but to centrally gather data extracted from operational applications and databases [7].

2.2 Data Warehouse Principles

Data is extracted according to principles selected according to the application requirements and is loaded into the data warehouse. The following four principles are guidances
for the formulation and organization of the data warehouse [14]:

1) Subject orientation: In operational databases and information systems, there are a lot of subject areas that are excellent foci for grouping within the data warehouse. The data warehouse, once designed, is organized by subject area. Typical subject areas are:

- customers
- products
- sales
- accounts
- incomes
- shipment

For example, a food store could have the following subject areas:

- Customers
- Foods
- Employees
- Suppliers

2) Integration capability: Identical conceptions of data may have dramatically varying representation among various operational databases and applications, and therefore integration becomes an important consideration. For a simple example, a database in a U.S. bank may contain the a numerical field PRICE; this field which can hold the price is expressed in U.S. dollars. Whereas a Canadian-based division of the same bank may
have the same database; however the same field PRICE may represent Canadian dollars. These types of mismatches can be resolved by converting all data as they are moved into the data warehouse.

3) **Non-Volatile data:** In the traditional operation of relational databases, inserts and modifications frequently occur. The data warehouse is “mass-loaded” from a centralized database or is “stocked” from distributed databases and information sources. In general, the data in a warehouse is not subjected to frequent changes.

4) **Time variance:** Data in a data warehouse can be viewed as a “sophisticated series of snapshots” of the main databases, there is always a temporal or time-oriented aspect to its contents. The time horizon for the data warehouse is much longer than that of databases. In operational databases, detailed information are kept for 6 months, but in a data warehouse, 10 years of summarized data are maintained. Figure 2-1 [7] illustrates the issue of time variance.

There are different levels of data within the data warehouse. Some data is very detailed, for example, sales of a particular brand of shoes. Other data is summarized, for example, specific total sales figures from food stores. There exists another component of the data warehouse known as “metadata”, which is information about data. Metadata is a card-catalog as to what the contents of the data warehouse are and where the contents came from. It is also used to store data models and definitions of data elements.

### 2.3 Data Warehouse Implementation

According to Cass Squire of Prism Solutions. Inc. [15], there are six major steps involved in implementing a data warehouse:

1) building the data warehouse data model

2) defining the system of record, or "best data" for the warehouse
Operational Data Warehouse

- time horizon--
current to 60-90 days
- update of record
- key structure may/may not contain an element of time

- time horizon--
5-10 years
- sophisticated snapshots of data
- key structure contains an element of time

Figure 2-1: The issue of time variance [7]
3) designing the physical data warehouse
4) creating the integration and transformation programs for building the warehouse.
5) loading and maintaining the data warehouse
6) building and maintaining the directory of metadata

Building a data warehouse first requires data specification and the record creation since it is first a database. Then, data must be loaded from an operational environment to the data warehouse by passing through a set of integration and transformation programs. These programs contain a lot of functions which include logical data selection, data summarization and merging, modification of key structures, and reformatting and recalculation performing on the data [15].

The first most important aspect of the design of a data warehouse is that of granularity. Granularity refers to the level of detail of the data contained in the data warehouse. The less detail there is, the coarser the level of granularity.

Another major design issue of data in the warehouse (after that of granularity) is the issue of partitioning. Partitioning of data is the breakup of data into separate physical units that can be handled independently. In the data warehouse, the issues surrounding partitioning do not focus on whether it should be done but on how it should be done.

All other aspects of the data warehouse design and implementation will become easy as long as granularity and partitioning are handled properly. If granularity is not done properly or if partitioning is not designed and implemented carefully, then any other aspect of design does not really matter. For example, if the levels of granularity are not chosen properly, then centers around managing the volume of data and storing data at such a high level granularity that detailed usage of the data cannot be made [7].

2.4 Metadata
Two levels of metadata exist in the warehouse: the developer level and the user level. Developer metadata is used to manage and control the development and maintenance processes. End user data resides on the data warehouse platform itself and is available to the end user as a regular part of the access and analysis of the warehouse [10. p. 43].

A metadata repository is used to store data models, the definitions of data elements for consistent reuse, business transformation rules, and source and target database descriptions [1].

One reason why metadata management is mandatory for the data warehouse is due to the size of the warehouse. Metadata serves as a “card catalog” of the warehouse, helping users to navigate the warehouse and to find relevant information. Metadata is additionally important due to the span of time that is managed in the warehouse. It is typical for 5 to 10 years of data to be stored, and one of the implications of managing such a lengthy period is that the structure of the data could possibly change over time. Metadata stores the context of these changes and their history [1].

2.5 Architecture of Data Warehouse

2.5.1 Basic Architecture of Data Warehouse

The data warehouse has a structure which is shown in Figure 2-2 [7].

Figure 2-2 shows that there are different levels of detail in the data warehouse. There are four levels of data exists: a level of highly summarized data, a level of lightly summarized data, a level of old detailed data (usually an alternate bulk storage), and a level of current detailed data. It is from the operational environment that data flows into the data warehouse. When data is passed from the operational level to the data warehouse level, usually we need to make transformation of data.

Once the data ages, it passes from current detail to older detail. As the data is
highly summarized monthly sales by product line 1981-1992

lightly summarized weekly sales by sub-product line 1984-1992

Meta-data

operational transformation

current detail

old detail

sales detail 1990-1991

sales detail 1984-1989

Figure 2-2: A Basic Data Warehouse Architecture [7]
summarized, it passes from current detail to lightly summarized data, then from lightly summarized data to highly summarized data [7].

2.5.2 Client/Server Data Warehouse Architecture

Client/server data warehouses are a relatively minor extension of the base concepts to data warehouses. Figure 2-3 [17] shows an alternative data warehouse architecture which is based on client/server models. In Figure 2-3, EIS (executive information system) applications written in the Xbase language access a centralized SQL-based warehouse via the Open Database Connectivity (ODBC) Application Programming Interface (API). In this environment, it's not difficult to implement a basic client/server model with a single server supporting multiple clients [17].

2.6 Distributed Data Warehouses

2.6.1 Introduction

Most organizations like to build and maintain a single centralized data warehouse system. There are many reasons why a single centralized data warehouse system makes sense [17].

- The data in the warehouse is integrated from across the corporation and only at headquarters an integrated view is used.

- The volume of data in the data warehouse makes a single centralized repository of data possible.

- Even if data could be integrated on demand, if that data is dispersed across multiple local sites, it would be a time and bandwidth burden to access.

Figure 2-4 shows the structure of a centralized data warehouse.
Figure 2.3: A Simple Client/Server Data Warehouse Architecture[17]
Figure 2-4: A centrally controlled, centrally housed data warehouse [7]
But there can be a need for a distributed data warehouse in some special cases. For example, the spread of a company over multiple locations favors a distributed data warehouse.

2.6.2 Structure of Distributed Data Warehouse

Figure 2-5 [7] shows what a typical distributed data warehouse might look like.

The scope of the global data warehouse is the corporation or the enterprise, while every local data warehouse has as its scope only the local site that it serves. The global data warehouse has historical data and it is also true for the local data warehouses. An interesting issue is about the commonality of data among the different local data warehouses for the structure and content of the local data warehouses may be very different [7].

2.6.3 Summary for Distributed Data Warehouses

Most systems are based on a single centralized data warehouse. But in some circumstances, there can be local data warehouses that house data unique to and of interest only to the local operating site, so a local distributed data warehouse fits this case. Meanwhile, a globally distributed data warehouse is possible too. The structure and content of the distributed global data warehouse should be decided centrally, but the mapping of data into the global data warehouse can be decided locally.

It should be noted that the coordination and administration of the distributed data warehouse environment is much more complex than that of the centralized data warehouse.

2.7 Examples of Data Warehouses

There are two organizations which impact data warehouse design. One is the NSWCDD data warehouse and the other one is Harris semiconductor's data warehouse [1].

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Figure 2-5: A typical distributed data warehouse [7]
NSWCDD is the short for Naval Surface Warfare Center, Dahlgren Division. An organization specializes in surface warfare for the Navy. It developed a data warehouse in July, 1992. The data warehouse runs on a UNIX server which uses the ORACLE DBMS. Users of the data warehouse can access multiple "shared" applications which are supported by Oracle development tools. The data warehouse provides an on-line Data Dictionary, SQL user scripts, canned queries, etc..

Harris's integrated yield management (IYM) system provides a data warehouse which uses a client/server environment by separating UNIX servers into application servers, data base servers, user clients and so on. Users access to the data warehouse by the graphical user interface which is developed by Windows4GL and Ingres/net tools allows the x-terminal emulator package to enable users access the data warehouse.

2.8 Data Warehouse Conclusions

Data warehousing is a viable and, in some cases, superior alternative approach to traditional solutions for multiple, distributed, heterogeneous information sources. Data warehouse trends, specifically in the area of distributing information with DSS/EIS applications that can access a variety of "mini-warehouses", will affect the use of data warehouse technology over the next few years [17].
CHAPTER 3

A STUDY OF DATA MINING

Since the data warehouse must be mined, before we proceed with the discussion of a security data warehouse, we need to discuss data mining issues. This is because data mining can be used to extract, through inference and other means, secure data from the warehouse.

3.1 Introduction

Advances in database technologies and data collection techniques including remote sensing, satellite telemetry, etc., have accumulated huge amounts of data in large databases. This explosively growing data creates the necessity of knowledge/information discovery from data, which leads to the growth of a promising field called data mining or knowledge discovery in databases. Data mining represents the integration of several fields, including machine learning, database systems, data visualization, statistics, and information theory. It is the search for relationships and global patterns that exist in large databases but are hidden among the vast amounts of data. For example a relationship between a patient's data and their medical diagnosis. These relationships represent valuable knowledge about the database and the conceptual objects characterized by the database. If the database is a faithful mirror it represents valuable knowledge about the real world registered by that database [8].

The use of data mining intends to implement decision support systems by using a set of extremely complicated information and database tools. Decision support systems which use data mining enable users to generate trend information and forecast information. Data mining users can initiate the search process to attempt to find information and answers
they don't already have, which is different from what is possible for the decision support system users.

3.2 Data Mining Algorithms

3.2.1 Introduction

Data mining involves fitting models to, or determining patterns from observed data. The fitted models play the role of inferred or discovered knowledge if the fit is good. Most data mining algorithms can be viewed as composed from a few basic techniques and principles. In particular, data mining algorithms consist largely of some specific mix of three components [6]:

- The model. A model consists of the functions of the model and the representational form of the model. Some functions can be classification and clustering, while the Gaussian probability density function is an example of a model representation form.

- The preference criterion. The criterion is usually some form of function applied to the model relating to the given data. Perhaps it is tempered by a smoothing term to avoid over-fitting or generating a model with too many degrees of freedom to be constrained by the given data.

- The search algorithms. The specification of an algorithm for searching for well fitting models and parameters, given data, a model(or family of models), and a preference criterion, makes up the last component.

A particular data mining algorithm is usually an instantiation of the model, preference and search components (e.g., a classification model which is based on a decision tree representation, a model preference based on data likelihood and the algorithm is greedy search,
using a particular heuristic. Algorithms often differ largely in terms of the model representation, for example, linear and hierarchical, and model preference or search methods are often similar across different algorithms.

3.2.2 The Functions of the Model

The more common model functions in current data mining practice include [6]:

- **Classification**: maps (or classifies) a data item into one of several predefined categorical classes.

- **Regression**: maps a data item to a real-value prediction variable.

- **Clustering**: maps a data item into one of several categorical classes (or clusters) in which the classes must be determined from the data, unlike classification in which the classes are predefined. Clusters are defined by finding natural groupings of data items based on similarity metrics or probability density models.

- **Summarization**: provides a compact description for a subset of data. A simple example would be the mean and standard deviations for each record field. More sophisticated functions involve summary rules, multivariate visualization techniques, and functional relationships between variables. Summarization functions are often used in interactive exploratory data analysis and automated report generation.

- **Dependency modeling**: describes significant dependencies among variables. Dependency models exist at two levels: structured and quantitative. The structural level of the model specifies, often in graphical form, which variables are locally dependent: the quantitative level specifies the strengths of the dependencies using some numerical scale.
• Link analysis: determines relations between fields in the database, for example association rules [13] to describe which items are commonly purchased with other items in grocery stores. The focus is on deriving multi-field correlations satisfying support and confidence thresholds.

• Sequence analysis: models may exhibit sequential patterns, for example in data with time dependence, such as time-series analysis. The goal is to model the states of the process generating the sequence and to extract and report deviations in the sequences and trends over time.

3.2.3 Model Representation

Model representation determines both the flexibility of the model in presenting the data and the interpretability of the model in human terms [6]. Some well-known models include decision trees and rules, linear models, nonlinear models, for example, neural networks, example-based methods, for example, nearest-neighbor and case-based reasoning methods and probabilistic graphical dependency models, for example, relational attribute models.

3.2.4 Search algorithms

Search algorithms are used to traverse the search space. The search space is the set of descriptions, each description is evaluated with the quality function. The search space can be expanded or contracted by using some operations or functions, such as generalizing or specializing operations. For example, by applying a generalizing function to a description obtains a new description that covers more data objects. Often, we use specialization to extend the representation of a class of objects, while generalization is the inverse of specialization, to provide more compact representations. We discuss some search algorithms in the following.
• bottom up: One technique used in the DBLearn system is "bottom-up". This search technique forms all examples of the target class by starting with an initial set. It is also called "data driven". Using generalization to obtain a more general rule can make examples of the target class correctly.

• top down: There are also "top-down" techniques which are used in Meta-dendral, BACON and the ID3-system. This kind of search begins with an initial description covering the entire universe. By applying both generalization and specialization operations repeatedly, its quality will exceed the target threshold, and therefore the description is transformed. The "top-down" search is also known as "model-driven".

• search graph: The aim of the search-graph technique is to find a description of sufficient quality, construct a graph which is made up of descriptions as nodes, as fast as possible. There are two kinds of search-graph strategies, one is the irrevocable search and the other is the tentative search.

  The irrevocable search strategy selects an operation and applies it irrevocably, but it doesn't make reconstruction later. It just evaluate a single sequences of operations, or path, is evaluated.

• tentative search: Unlike the irrevocable search strategy, the tentative search returns the result of the computation of the applicable operation. Backtracking and search-graph are two types of tentative strategies. Backtracking establishes a point when an operation is created. If a solution can not be found in subsequent computations, the computation will come back to the previous tracking point. The graph-search strategy stores the backtracking part of the graph that has been explored so far. This provides greater flexibility for we can generate new descriptions at any place in the graph, but the cost is more storage.
The above search strategies are similar: they generate many descriptions, so they can be uniformed. We should know that they are expensive too for each description requires some degree of computations. But by choosing the shortest path to a goal node during the operation can result in a reduction of generated nodes and the performance of the system would increase.

- heuristic search: An alternative class of search algorithms is heuristic searches. It is based on the information about the heuristic based on domain-knowledge, which is used to guide the search and operations by selecting the nodes on the shortest path to a goal node so it can reduce the search effort by carefully selecting operations, and finding a sufficient quality of fit very soon.

- genetic algorithms and simulated annealing algorithms: Other class of search algorithms are genetic algorithms and simulated annealing. These algorithms overcome the drawback of the heuristic approach and the traditional symbolic computational approaches. Genetic algorithms use population (a set of candidate descriptions) to construct new descriptions and reorganize parts of the best descriptions in the current population. The process keep going until descriptions of sufficient quality are found or no further improvement occurs. Simulated annealing uses a process to select each applicable operation which has a probability of being chosen. But genetic algorithms need a greater number of evaluations. Therefore in the case of learning tasks for smaller databases where no domain knowledge is available, genetic algorithms will be the best choice.

3.3 Construction of Data Mining System

The construction of a data mining system based on the above concepts for a particular
application consists of the following stages. Each stage is outlined with the most important design considerations and a description of various techniques used [9].

1) Define the mining task. First, users need to specify the relationship they need to discover in the data.

2) Selecting the data. After the mining task has been defined, the next step is the data selection. Usually the relevance of information cannot always be determined in advance, so selecting data reflects the user's expectations about hidden relationships, and therefore can restrict the relationships that can be found.

3) Knowledge representation. Choosing appropriate representations of data is a key to defining a data mining systems. The knowledge representation must fit the application domain, and thus be able to describe hidden relationships in the data.

4) Transformation operations. Transformation are made from one knowledge structure into another by these operations.

5) Quality function. With each knowledge structure, an associated quality that measures its information content quality, should be defined. This criterion can be defined in terms of the classification accuracy of the knowledge structure with respect to its class. The higher the accuracy, the higher the quality.

6) Search algorithm. It is necessary to choose suitable search algorithms to find the most preferred knowledge structure among all structures. Search algorithm selection is based on the quality of data and knowledge sought.

7) Heuristics. Heuristics are used to estimate the result the application of operations to the current knowledge structure. Any known heuristic information can improve the quality of the search results.

8) Noise and missing information. Data mining techniques should be enhanced to deal with corrupted or missing information.
3.4  Data Mining Conclusions

In conclusion, data mining is used to support DSS and ESS by providing another tool for preparing data on which decision can be made. It is a field of increasing interest combining databases, artificial intelligence, and machine learning. It has become increasingly more important as the amount of information stored in databases grows. Large corporations have vast amounts of information relating to their company, customers and product. Large corporate databases, coupled with falling computer prices, have enabled this area to prosper in recent years, and hence, we may expect the interest and application of data mining tools to continue to grow throughout the next several decades [11].
CHAPTER 4

SECURITY ISSUES AND SPECIFICATIONS OF DATA WAREHOUSES

4.1 Introduction

Because information is such a highly valuable asset for corporations and governments around the world, its security is certainly a critical need. Protection of corporate information system assets is no small task, and this is particularly true of data. Data has to be made secure in at least two ways. First, care must be taken that incorrect data cannot be entered. Second, it must be guaranteed that only authorized users can access data. This is done to prevent unauthorized users from deleting, mutilating or illegally copying data, intentionally or unintentionally. Since the early 1980s, concerted efforts have been made on database security issues and much progress has been made. As with SQL, we have the GRANT/REVOKE model, where owners of data grant permission for various read/write operations to other system users. The U.S. government has long been interested in database and information security models, and a number of efforts in the early to mid-1980s focused attention on this area. A great deal of research has been done in these areas over the past decade, and commercial products are appearing that incorporate new security technologies. Data in a warehouse database is protected from change and integrity if it is rigorously maintained [1]. Data warehousing involves extracting data from existing databases, and reorganizing it for business analysis just as with databases. Security is a very important issue. Since a data warehouse is implemented as a special kind of database, the discussion below is based on security for databases.

4.2 Security Policies and Mechanisms

Security policies define the rules that regulate an organization's security management.
Security mechanisms perform actions based on the security policies. That is to say, security policies state “what to do”, while security mechanisms determine “how to do”.

Mac (Mandatory Access Control) and Dac (Discretionary Access Control) are examples of security policies. In Mandatory Access Control, the decision about what information should be accessed and who can access the resources is independent on the owner of the data, it is the central authority that makes the policy. Unlike Mandatory Access Control, Discretionary Access Control gives the owner of the resources authority for access control. We also need to know that policies maybe changed at different times and different places [12].

4.3 Security Issues

Data base security is concerned with the ability of the system to enforce a security policy governing the disclosure, modification, or destruction of information. A data base security policy states [4]:

1. which type of sub-channels between (groups of) users can be established.
2. the requirements for the availability of certain facilities of the sub channels.
3. the requirements on the separation and non-interference of sub channels.

The basic problems we have to consider for the security of a data warehouse is similar to that of a database, the concerns are listed as follows [12]:

1. **Data Reliability**

   To ensure data reliability, we use the following two-phase update procedure.

   The first phase has two steps, which are the intent step and the committing step, where the first step gathers the information and other resources it needs to perform the update but makes no change. The second step involves writing a commit flag to the data warehouse. After committing, the data warehouse begins making permanent changes.
The second phase is making the permanent changes. After the second phase has been completed, the data warehouse is again completed.

2. **Confidentiality**

There are two issues we consider for confidentiality.

One issue is sensitive data which is not granted to the public. Another issue concerns multi-security levels. Two different users operating at two different levels of security might get two different answers to the same query in the multi-security level case.

Two methods for maintaining confidentiality are:

1. **Sensitivity Lock.** A sensitivity lock is a combination of an identifier (for example, a record number can be an identifier) and the sensitivity level. Because the identifier can be made unique, each sensitivity lock pertains to one particular record.

2. **Partitioning.** Partitioning is used for multilevel security. The data warehouse is divided into separate ones, each has its own level of sensitivity. This can be viewed as maintaining separate file cabinets.

3. **Inference Problem**

   (1) An inference problem is caused by inferring a result based on one or more statistical results such as using SUM, or COUNT to obtain sensitive data.

   (2) We can use trackers, security logs, conditional accesses, etc. to prevent data from inference problems.

Some other suggestions are:

- Don't provide sensitive data values, but reject the query which would result in the output of sensitive data by not responding.

- Provide a value close to, but not exactly, the actual value to protect sensitive data.

4. **Protecting Metadata**
Metadata is an important component of a data warehouse. The metadata stored in the catalog is often granted to the public. The end-user metadata helps users create their queries and interpret the results. When unauthorized users use queries to access the table, it is better just as above, that they get a message that indicates that the table does not exist than one that indicates a lack of authorization. This latter message tells the user that the object exists, and then the user may check another way to obtain the data [3].

4.4 Security Requirements

The following is a list of requirements for the security of a data warehouse [12]:

- Physical Protection. This ensures that the data of a data warehouse is immune to physical attacks such as forced disclosure of a password or theft of the physical storage devices. The data warehouse can be reconstructed in case it is destroyed.

- Logical integrity. This ensures to preserve the logical structure of the data warehouse. For example, a modification of the value of one field will not make changes to other fields with logical integrity.

- Auditability. This is the ability to track who has accessed to the data in the data warehouse.

- Data Confidentiality. This makes sure that unauthorized users are unaware of the existence of the protected data.

- Access control. This allows a user to access only authorized data; as a result, different users have different access modes (such as READ or MODIFY).

- User authentication. It is also called user sign-on or login. It ensures that every user is positively identified [3].
• Accountability. This means that any user action can be traced to only one individual user, both for the audit trail and for permission to access certain data [3].

• Availability. Meaning that users can access general data as well as the data which they are authorized to.

4.5 Protection Mechanisms

1. Views as Protection Mechanism

A view or subschema is an abstract model of a portion of the conceptual database. For example, in an airline reservation system, the travel agency does not need to know the assignment of pilots to the flight. The dispatcher may need to know about aircraft, the aspects of the personnel files like which pilots are qualified to fly a 747, but doesn’t need to know about the passengers booked on a flight. So, there may be one view for reservations and another for the dispatcher’s office. A view can serve as a convenient protection facility. There are two operations on a view, one is called read-only in which the user has the privilege to read the data but not to modify the data. Another type permits both reading and writing of the objects that are part of the view [16].

2. The Use of Query Languages to Define Rights

The data manipulation language can be used to define the privileges each user has for accessing the data warehouse. We can write a database selection and database projection to be included automatically with every query posed by designated users about a designated relation. The selection and the projection have the effect of making certain values invisible [16]. For example, if a user query to the bank database system to project the MEMBER relation onto NAME and ADDRESS, whether or not he specifies that project, then the user cannot see balances.

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
5.1 Access Control Model

5.1.1 Access Control

There are many different aspects to security in organizations, from the legal, social and ethical issues to how often passwords should be changed. It is required that owners of personal information about individuals to take reasonable precautions to ensure that unauthorized people do not have access to it. The basic security issue is control of access to data objects. For example, a user may be allowed to read two relations but only to update one of them. Access control ensures that users and processes acting on their behalf are prevented from gaining access to information or resources that they are not authorized to access or have no need to access. This includes the administration (that is, the granting and revocation) of access rights and their verification [2].

Four main types of controls can be identified [3]:

- Content independent
- Content dependent
- Statistical control
- Context dependent

Under content-independent control, a user is allowed/not allowed to access a particular data object without considering the content of the data object. An example of a content-independent access control would be: any user is allowed read access to the student relation.
Checking for content independent access control can be done at compile time, since the actual value of the data object does not have to be examined.

Under content dependent control, access to a data object by an individual user with respect to the content of the data object, therefore it should be determined at run-time. "Manager x is allowed to modify the price field of shoes provided the current price is less than $50" is an example of a content dependent access control.

Under statistical control the user's access privileges is limited on the performance of statistical operations on the data without the right to access individual records.

Under context dependent control, a user has access rights to a data object according to the context in which the request he made. For example, a user may only be allowed to modify a student's grade in a course if the user is the course lecturer.

5.1.2 Security Model

There are a wide variety of alternative techniques available for front-end data access in the data warehouse, which include general-purpose relational query tools, multidimensional views of relational data, special-purpose DSS/EIS user interfaces, and custom-built user interfaces developed with client/server tools [1].

In Figure 4-1, we introduce a security model for access control to data warehouse. The lowest level has the following functions:

- enforces access controls and licensing policies.

- Interacts file performance.

- check authenticity of file transaction requested by user processes.

- Ensures the integrity of conveyed data and ensures that the data is accessible only to authorized roles. Roles are classified and arranged on the basis of the specific
task they have within an organization.

- Pass data to higher levels.

The second level is security transfer. It processes the data from the lower level by security mechanisms. It also checks the classification of data against the security level of user. It stores all of the information on security levels for the roles. It creates an array which has all possible index values by computation functions as one axis, while the other axis contains all rows in the bases table. This level makes sure that the user doesn't access values beyond the permitted amount. The third level are the DSS/EIS tools used for high-end multi-dimensional data analysis, helping end-users for their specific requirements such as specialized data manipulation and drill-down functions and a wide range of customer reports. It assists end-users in making decisions about the business of the business. Therefore this level provides the user interface for normal operations.

Figure 4-2 illustrates an authorization model as a matrix. In this security matrix, all users and processes are matched against all objects within the enterprise system. Each intersection point on the matrix contains some number of operations that the user or process is allowed to perform against the specific object.

5.2 The Security-Level Control Model

5.2.1 The Security Model

As we discussed earlier, the kinds of information contained in the data warehouse include the following [3]:

1. Highly summarized data.
2. Lightly summarized data.
Figure 4-1: The Secure Data Warehouse Model
<table>
<thead>
<tr>
<th></th>
<th>A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>User3</td>
<td></td>
<td>R,C,U</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User4</td>
<td>R,U,C,D</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User5</td>
<td>R,U,C,D</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>User7</td>
<td>R,C,U</td>
<td></td>
<td></td>
<td>R,U,C,D</td>
<td></td>
</tr>
<tr>
<td>User8</td>
<td></td>
<td>C,U,I</td>
<td>M,I,U</td>
<td></td>
<td></td>
</tr>
<tr>
<td>User9</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

R=Read   D=Delete
C=Create  I=Insert
U=Update  M=Modify

Figure 4-2: An Authorization Model
3. Old detail data.


Figure 4-3 shows a data hierarchy in the data warehouse. It forms a security hierarchy. On the top, it has less sensitive data, while on the bottom it has the most sensitive data. Each layer has a different security level, the bottom has the highest security level, while the top has the lowest one. The higher security level can have access the lower one, but not the reverse. To gain access to each lower layer we need an additional password to gain access to it. Users access the data via a user view which is also called logical schema, the user view then is mapped into the global logical schema, which is then mapped onto an internal security level. Since users access the data only through views and hence are prevented from accessing data which is not contained in their views. Here we show the different roles and their privileges to the access of the data.

According to the structure of the data warehouse, we introduce a security model in which a user who wants to access the data has to pass through some layers. The model is shown in Fig 4-4.

1. The first layer identifies the users to be of two types: general users and security administrator users. The general users can be business users, or any front-end users. The security administrator users are security officers, analysts, administrators. The general user has to pass through the second and third layer to access the data, but the security administrator users can directly access certain data by their status. Security services provide the user interface to the user. For example, it asks for user identification such as name and password, organization name, security type. With this information, the users are characterized as having different roles and are assigned certain privileges according to their specific functions. The privileges allow them to see a view which is predefined by the security administrators according to the security policies, security mechanisms and
Fig 4-3: security hierarchy in data warehouse
Fig 4-4:  A Data Warehouse Security Model
protocols. The view information is provided by the higher layers.

2. The second layer consists of security mechanisms, security protocols and security level control. Their functions are described as follows:

- **Security mechanisms** provide tools to implement the individual functions of the security system such as security services for authenticity and integrity etc.. Each mechanism may serve as a tool for the implementation of several security services. Security mechanisms can be multilevel security mechanisms and discretionary security mechanisms, for example. Multilevel security mechanisms classify security levels as Top-secret, Secret, Confidential and Unclassified. This classification is widely used in the military security. Discretionary security mechanisms leave users flexible access rights. This is used in commercial environments with DBMSs.

- **Security protocols** contain special policies based on the decision of a security officer or, according to the organization's regulations for security requirements, they may combine security mechanisms and security services to make protocols. Security protocols can be direct access protocol, ODMG'93 protocol. Direct access protocol uses a data definition language for the interface, while the latter uses an object definition language for the interface.

- The **security-level control** stores all information about the level of objects and subjects in a category. The user's access privileges depend on his role and security policies, that is to say the specific user will get just the part of information that he needs to perform his task. Meta-data in the data warehouse repository is a best way to document security rules. By creating the relationship between a set of tables and a set of access rules, each application area will gets its own role and all the valid authorizations for the users.
3. The third layer is *security management*. It includes all functions for security management, performs any required analysis, and is used for recovery, tuning and to audit the data.

5.2.2 A Simplified Case Study of a Security Model

In Figure 4-5, the travel data warehouse contains three mini databases of the information for Southwest Airlines, TWA and Triple A travel agents. Suppose there is a query: give the fare from Las Vegas to New York. We will discuss security with this sample case.

1. **Authentication**

   First we check the user identification to the data warehouse. For each mini-database, we also have a local security login. If you want to access to Southwest Airline’s data, you need a password to get in, so we have double authentication.

2. **Authorization Subjects**

   Authorization subjects are roles. Users are grouped together according to role status. The concept of role helps to reduce the number of authorization subjects. Figure 4-6 is an example of a role lattice in which roles are arranged. Each name is a node which represents a role, and the arrow indicates authorization. Super-users can be security officers. Their authorizations are greater than those of Department Managers.

3. **Authorization Types**

   Authorization types includes R(read), W(modify), C(create), S(select). If the user is not authorized to execute any operation, we can return the message that the data doesn’t exist, so the user won’t attempt to access the data by other ways. The authorization type forms a lattice as shown in Figure 4-7.

4. **Authorization Inference Rules**

   By Armstrong’s axioms from database management theory, we have a set of inference
Fig 4-5: A Sample Case for Data Warehouse Security

Data Warehouse Security Check

local security check

DB  DB  DB
Southwest  TWA  AAA travel agent

Reproduced with permission of the copyright owner. Further reproduction prohibited without permission.
Fig 4-6: An Example of the Role lattice
Fig 4-7: Example of the Authorization Type Lattice

W = write
R = read
C = create
S = select
rules as follows [16]:

- A1: If $Y \subseteq X \subseteq U$, then $X \rightarrow Y$ is logically implied by $F$. $F$ is a set of functional dependencies, $U$ is a set of attributes.

- A2: If $X \rightarrow Y$ holds, and $Z \subseteq U$, then $X \cup Z \rightarrow Y \cup Z$.

- A3: If $X \rightarrow Y$ and $Y \rightarrow Z$ hold, then $X \rightarrow Z$ holds.

We can derive some new rules from the above very basic inference rules. New derivations of authorization rules are based on implication relations among authorization subjects, objects, and access modes. For example, in Figure 4-6, the department manager can access the information the supervisor can access; the authorization to read an object implies authorization to read all attributes of the object. In Figure 4-7 the authorization to write an object implies authorization to create the object.

5. Views

If the user writes a query to give me the fare from Las Vegas to New York, then the views for airline and travel agent are different as shown in Figure 4-8. The airline needs to know flight number, departure time, arrival time, origin, destination, pilot, price, and type of flight (passenger or cargo), but the travel agent, whose view is different, for example, does not need to know the information about which pilot is able to fly which flight.

6. Security-Level Control for Subjects and Objects

We can assign each subject (group of user) three security levels such as [4]:

- Absolute security level: it is given to the subject from its creation and the duration is the whole life cycle of the subject.

- Read Security Level: means the subject is allowed to read data.
Airline's View

<table>
<thead>
<tr>
<th>FLT#</th>
<th>ORG</th>
<th>DES</th>
<th>TYPE</th>
<th>PILOT</th>
<th>DEP</th>
<th>ARV</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>752</td>
<td>LV</td>
<td>NY</td>
<td>Passenger</td>
<td>Bill</td>
<td>0820</td>
<td>1250</td>
<td>$400</td>
</tr>
<tr>
<td>147</td>
<td>LA</td>
<td>Denver</td>
<td>Cargo</td>
<td>Ted</td>
<td>1020</td>
<td>0130</td>
<td>$300</td>
</tr>
</tbody>
</table>

Travel Agent’s View

<table>
<thead>
<tr>
<th>FLT#</th>
<th>ORG</th>
<th>DES</th>
<th>DEP</th>
<th>ARV</th>
<th>PRICE</th>
</tr>
</thead>
<tbody>
<tr>
<td>752</td>
<td>LV</td>
<td>NY</td>
<td>0820</td>
<td>1250</td>
<td>$400</td>
</tr>
<tr>
<td>147</td>
<td>LA</td>
<td>Denver</td>
<td>1020</td>
<td>0130</td>
<td>$300</td>
</tr>
</tbody>
</table>

Fig 4-8: Views for the Airline and the Travel Agent
• Write Security Level: means the subject is allowed to write.

Meanwhile, each object (data storage entity) has three security levels:

• Absolute security level: the security level that data contained in the object.

• Flow Security Level: it shows the security level of which data to flow.

• Correlation Security Level: it is the security level from which data may flow into the object.

Then a query by the subject to establish a connection between objects is granted if:

• The subject’s read security level dominates the object’s absolute security level, and

• The subject’s write security level is dominated by the object’s absolute security level.
CHAPTER 6

CONCLUSIONS

6.1 Summary on Data Warehouses and Their Security

Data warehousing is an emerging technology whereby organizations extract value from their informational assets through the use of special stores called data warehouses. A data warehouse contains transformed disparate data into information that is consistent across the whole organization [1]. A data warehouse is a special kind of database. Security of data warehouses is therefore like that of databases. We have discussed many different aspects to the security of data warehouses such as the topics of security mechanisms, security policies and security requirements for data warehouses. We also introduce two models for data warehouse security: the access control model and the security level control model.

The access control model uses authorization mechanisms to provide security to the access of data from the front-end users.

The security level control model is designed by the nature of contents of data in the data warehouse. We have not discovered any new conceptions for data warehouse security. This would be work for the future.

6.2 Future Work

Security for distributed data warehouses is more complex than for central data warehouses because the data warehouses in different locations have to keep their sensitivity level of data in its own way; therefore the design of access control is not easy to meet with many different levels of sensitivity: we need to customize the security of each database in the data warehouse. For the security-level control model, we need to check local security...
automatically.

There is also an interesting topic which is the formalism of the object security requirements of a model. It is used for information security [5], but we can apply it also to the security of data warehouses. This can be done in the future.
REFERENCES


VITA

Graduate College
University of Nevada, Las Vegas

Rihui Wang

Home Address:
10005 Pratt St.
Omaha, NE 68134

Degree(s):
Bachelor of Science, Computer Communication, 1991
Hunan University, China

Thesis Title: On the Design of a Secure Data Warehouse

Thesis Examination Committee:
Chairperson, Dr. Roy Ogawa, Ph. D.
Committee Member, Dr. John Minor, Ph. D.
Committee Member, Dr. E. Yfantis, Ph.D.
Graduate Faculty Representative, Dr. Samir Moujaes, Ph. D.