A Prototype Model for Data Warehouse Security Based on Metadata

N. Katic[1] G. Quirchmayr[2] J. Schiefer[1] M. Stolba[1] A M. Tjoa[1]

[1]*Institute of Software Technology (E188)*
Vienna University of Technology
Resselgasse 3/188, A-1040 Vienna
Austria
{katic, stolba, js, tjoa} @ifs.tuwien.ac.at

[2]*Institute of Applied Computer Science and Information Systems*
University of Vienna
Liebiggasse 4, A-1010 Vienna
Austria

Abstract

The aim of this paper is to give an overview of security relevant aspects of existing OLAP/Data Warehouse solutions, an area which has seen rather little interest from product developers and is only beginning to be discussed in the research community. Following this description of the current situation, a metadata driven approach implemented as part of the WWW-EIS-DWH project is presented in detail. The prototype focuses on the technical realisation and is intended not to be open for use in different security policies.

1. Introduction

For a wide range of companies, in both private and public sectors, competitiveness and effectiveness depend on the quality of decision making; so it is of no surprise that many are looking to improve the quality of their decisions by learning from past business transactions and decisions. Accumulated data represent a priceless business asset.

Providing analysts with wide access to the mass of corporate data in a data warehouse requires the organisation and integration of heterogeneous data in a heterogeneous environment. Furthermore the production of derived aggregated data of the data warehouse requires the continuous maintenance of integrity.

These requirements imply several security issues to ensure that it is just authorised users who benefit from relevant data and that no unauthorised sources are used. Under many jurisdictions it is illegal to merge personal data unless anonymity can be ensured and especially in Europe stricter legislation has been proposed and is under review based on OECD and EC recommendations.

The aim of this paper is to discuss requirements and impacts on the selection of an adequate security model for a data warehouse environment. This security model should support such features as controlled access to individual data items, selective encryption and patented security processes. For the right choice of the security model it is important to pay attention to the metadata of the data warehouse. They contain security information such as access rules, classifications of security objects or clearances of security subjects.

This paper is composed of two parts: a theoretical part which deals with security in data warehouses in general and part 2, a description of an implementation of a security model prototype for a data warehouse environment based on metadata.

2. Data Warehouse & Security

A data warehouse is a collection of integrated databases designed to support managerial decision-making and problem-solving functions. It contains both highly detailed and summarised historical data relating to various categories, subjects, or areas [4]. All units of data are relevant to appropriate time horizons. The data warehouse is an integral part of the enterprise-wide decision support system and does not ordinarily involve data updating. It empowers end-users to perform data access and analysis. It also gives an organisation certain competitive advantages, such as fostering a culture of information sharing, enabling employees to effectively and efficiently solve dynamic organisational problems, minimising operating costs and maximising revenue, attracting and maintaining market shares, and minimising the impact of employee turnovers.

The security requirements of the data warehouse environment are similar to those of other distributed computing systems [3]. Thus, having an internal control mechanism to ensure the confidentiality, integrity and availability of data in a distributed environment is of paramount importance. *Confidentiality* denotes the protection of information from unauthorised disclosure either by direct retrieval or by indirect logical inference. *Integrity* requires data to be protected from malicious or accidental modification, including the insertion of false data, the contamination of data, and the destruction of data. *Availability* is the characteristic that ensures data being available to authorised users when they need them. Availability is closely related to integrity. It includes 'denial of service' of a system, i. e. a system is not functioning in accordance with its intended purpose [2].

2.1. Security Restrictions

A data warehouse by nature is an open, accessible system. The aim of a data warehouse generally is to make large amounts of data easily accessible to users, thereby enabling them to extract information about the business as a whole. Any security restrictions can be seen as obstacles to that goal, and they become constraints on the design of the warehouse.

There may be sound business reasons for any security restrictions applied to the data warehouse, but it is worth noting that they may lead to a potential loss of information. If analysts have restricted access to data in the data warehouse it may be impossible for them to get a complete picture of the trends within the analysed area.

Checking security restrictions will of course have its price by affecting the performance of the data warehouse environment, because further security checks require additional CPU cycles and time to perform.

2.2. Security Requirements

Security requirements describe all security conditions that have to be considered in the data warehouse environment.

It is important to determine in an early stage any security requirements that will be enforced in the data warehouse, because they can seriously impair the organisation and design of the warehouse. It is very difficult to add security restrictions after a data warehouse has gone live. So it is important to capture the ultimate security requirements at the beginning and make them part of the system design.

The first step for the definition of security requirements is to classify the security objects and security subjects of the data warehouse environment. Security objects can be classified in different ways. Which solution is suitable depends on the security level which should be achieved. Qualified classifications would be classification by sensitivity (public, confidential, top secret) or according to job functions (accounting data, personnel data). As with security objects, there is a number of ways in which security subjects can be classified. We can follow a top-down company view, with users classified by department, section, group, and so on. Another possible classification is role based, with people grouped across departments based on their role. This approach would classify all analysts as one group, irrespective of their department. If each department genuinely accesses different data, it is probably better to design the security access for each department separately.

2.2.1. Legal Requirements. It is vital to consider all legal requirements on the data being stored in the data warehouse. If individual customer data are being held, such as account details in a banking data warehouse, it may be required by law to enforce certain restrictions.

In this context the following issues are to be clarified:

- Which arrangements have to be made for being allowed to hold legally sensitive data?
- Which data are subjected to legal restrictions?
- Which separate handling does this data require concerning storage, access and maintenance?
- Which analyses may be performed on this data?
- If data held online is used for trend analysis, and is therefore held in summarised rather than detailed form, do any legal restrictions apply?
- Which data may be used only for the companies own purposes and which data may be passed on third parties?
- Can the analysis of legally sensitive data be limited in a way that no legal restrictions apply?

These issues explain why the administrator of a data warehouse must have special know-how about the legal and business field in order to identify legally sensitive data and to accordingly limit the access to this data.

2.2.2. Audit Requirements. Resulting audit information is the basis for further reviews and examinations in order to test the adequacy of system controls and to recommend any changes in the security policy.

Auditing is a security feature that is often mandated by organisation. Given the high volume of data involved in a data warehouse, auditing can cause an extremely heavy overhead on the system. To make up for this overhead more hardware will be needed. Basically the following activities are interesting for auditing:

- Connections
- Disconnections
- Access to data
- Change of data
- Deletion of data

For each of these activities it may be necessary to audit success, failure or both. For security reasons the auditing of failures can be particularly important, since it can highlight any attempted unauthorised or fraudulent access.

If data access is to be audited, it has to be established whether each access is to be audited separately, or whether it is sufficient to audit the fact that a user has accessed specific tables during a session. This has impacts on the audit information that needs to be held and implicitly avoids both space and I/O overhead.

If data changes are being audited, it has to be determined whether it is sufficient to audit the fact that a change occurred, or whether it is required to capture the actual change that was made.

2.2.3. Network Requirements. Network requirements are a further important part of security requirements. For the transfer of data from the source system (usually an operational system) into a data warehouse they must mostly be transmitted over a network. For such a data transfer precautions must be taken, in order to retain the confidentiality and integrity of the data.

It must be clarified and proved whether data have to be encoded before transmission into the data warehouse to prevent a manipulation during the transfer. If the data are transmitted for example over a public network, a secure connection between source system and data warehouse has to be constructed to transmit data in encoded form. The expenditure for data encryption and decryption can be very high regarding processing speed and delay. Particularly with large quantities of data this factor can affect the system performance of the source system as well as of the data warehouse system negatively.

A further substantial fact is the reliability of the data communication. It should be guaranteed that data are transferred error free from the source system into the data warehouse. Connection interruptions should be prevented as far as possible, since incomplete transfers threaten the integrity of data in the data warehouse. Therefore measures must be taken, which make possible a complete rollback of the entire transfer process in case of an incomplete data transmission.

2.2.4. Increasing comfortability of use through access authorisation. The primary goal of security restrictions and access authorisation in data warehouses is the prevention of disclosure of protected data.

However, authorisations in data warehouses have in contrast to operational systems a second important function: Users should and *want* to see mainly only “their“ (relevant) data of the information system, which are based on a data warehouse, since they then penetrate more directly to the information important for their daily work. The work with the data warehouse system thus becomes more comfortable by the fact that only data are offered, which are important for a certain user.

2.3. Metadata & Security

An essential part of a data warehouse are metadata - data about the data contained within the data warehouse. Without metadata, locating information contained in the data warehouse becomes a daunting task, akin to searching for a person's telephone number without the aid of a telephone directory. Metadata not only describe the contents of the data warehouse but also provide the user with information useful in judging the quality of the content [1]. They might describe each fact contained within the warehouse in terms of when it was last updated, the source of the fact and how it is derived from an organisation's operational systems. Metadata may also identify the hierarchies within dimensions (for example, identifying the sales territories that fall into each region).

The role of metadata is rapidly expanding as organisations develop a data warehouse strategy that may result in the creation of operational data stores, integrated data warehouses and multiple data marts. The data warehouse is increasingly multiple databases that have common elements but serve different functions. Metadata must describe the enterprise warehouse even if it is no longer a single database residing on one server. The role of metadata is being redefined as providing data about distributed information resources. Metadata must isolate the user from the complexities of accessing distributed information resources, while facilitating the currency and synchronisation of multiple databases. Failing this, users are confronted with precisely the problems that data warehousing was intended to solve. Different answers to the same question and the resulting lack of confidence in the information obtained are just one example.

Metadata management is provided via a metadata repository and accompanying software. Metadata repository management software can be used to map the source data to the target database, generate code for data transformations, integrate and transform the data, and control moving data to the warehouse. This software enables users to specify how the data should be transformed, such as data mapping, conversion, and summarisation.

Metadata can also describe security mechanisms in a data warehouse environment. In this case access rules with corresponding information about security objects and subjects are stored as metadata. Security subjects are responsible for changes in the data warehouse and cause information to flow within different objects and subjects. When a user accesses data of the data warehouse, the secure query management layer has to check whether this access is allowed or not. To ensure that, it verifies the corresponding access authorisations by analysing the security metadata.

In the following section a prototype is introduced which implements a security model based on metadata. It is an implementation of a security manager for administration, definition and description of users and user groups and a secure query management layer (SQML) which has the task to filter user queries by checking if they are allowed to be performed.

The prototype is integrated into the WWW-EIS-DWH information system and meets all discussed security requirements which were placed by the WWW-EIS-DWH project.

3. Prototype

3.1. Introduction

The realisation of this metadata structured security model was developed within the framework of the WWW-EIS-DWH project. Since the structure and realisation of this prototype model does not depend on the WWW-EIS-DWH project, this prototype model is also suitable for other, similar, data warehouse solutions. The metadata description of the content of data warehouse is one especially important feature for the realisation of the prototype model of this information system. This security model is advantageous for those kinds of data warehouse solutions, which content can be described by metadata. The data of this data warehouse are represented by means of a simple structured description language (MQL)[10] in the form of a tree structure. This description data (metadata) may further be used for security relevant data management.

The main goal of the proposed security model is to reduce user's queries only to those data, which are to be seen by that user.

For each user (or for each user group), the Security Manager defines data areas, which can be accessed by this user (user group). This definition happens only once, by the registration of the user. The Security Manager is the only authority, which can redefine or delete user's access rights at any time.

The above mentioned predefined data areas are the base of the security model in this information system. Users produce their queries and these queries are „on the fly“ (by the SQML) reduced to the predefined set of data. This reduction happens transparent to the user. It simultaneously gives the impression, as if the content of the data warehouse, for the respective user, actually was everything that is to be seen. This is a very important security measure, because users don't feel restricted and don't get the idea of trying to reach the „forbidden“ data.

The hardware requirements for the realisation of this security model do not deviate from the hardware requirements for the Intranet based WWW-EIS-DWH information system and therefore also not from an usual data warehouse system. Since this software-based security model was implemented in “**JAVA**“ it is platform independent, meaning that the current version runs on Windows NT and UNIX.

3.2. Metadata Security Model

One of the most important parts of data warehouse are its metadata. Metadata influence all levels of a data warehouse, but exist and act in another way as the rest of warehouse data. Metadata, which are used by developers in order to manage and control the creation and maintenance of the data warehouse, are kept outside of the data warehouse. The metadata concerning data warehouse users are on the contrary a part of data warehouse. This data are used to control access to and analysis of data.

Consequently there are two types of metadata [11]:

- Structural metadata
- Access metadata

3.2.1. Structural metadata. Structural metadata are used for the creation and maintenance of the data warehouse. They completely describe the data warehouse - its structure and its content. The basic element of structural metadata is a model which describes their data subjects, their features and their intermediate relationships. [7]

3.2.2. Access metadata. Access metadata represent a dynamic relationship between the data warehouse and the end-user applications. They contain the measured values of the enterprise, user-defined names and aliases. These data hold the description of the data warehouse server, databases, tables, detailed data and summed up data with a description of the initial data sources and of the transformations being carried out.

Access metadata set up “drill-down“ and “roll-up“ rules as well as the views over dimensions and available hierarchies (such as products, markets and customers). This data can also contain the rules for user-defined calculations and queries. Such metadata allow the security implementation for an individual user, user groups or the whole enterprise, relating reading, modifying etc. of calculations, summed up data or analyses. [7]

3.2.3. WWW-EIS-DWH metadata. We will explain the structure of the metadata file of our data warehouse by means of an example. The data warehouse illustrated here is called „grocery“ and consists of one characteristic number (Sales) and four corresponding dimension tables (Time, Product, Promotion and Store) (Kimball: Grocery Data Warehouse [6]).

The general data are stated at first (program segment 1) such as the currency which is used in the data warehouse, the weight, the date of last update, the name of data warehouse and so forth

```
dwh = (
   description = (
   lastupdate="Fri,13-Jan-1998 10:54:00 GMT"
   currency = (
      USD = ("us dollar" "dollar")
      1
   )
   weight = (
      ("Once" "octane number")
      1
   )
   ini = (
      user = test
      pass=test
      database=grocery
   )
) // EOF description
```

Program segment 1

After this general description, the characteristic numbers (facts) are specified, together with related dimensions and the primary keys belonging to these dimensions (program segment 2). At this way, the main data warehouse structure is illustrated. The more detailed description follows.

```
facts = (
(
   id = "Sales_Fact"
   sql = "Sales_Fact"
   des = "Sales data"
   dim = (
   (
      sql = "TIME"
      key = "time_key"
   )
   (
      sql = "Product"
      key = "product_key"
   )
         …
   )
)
```

Program segment 2

Further are all attributes of the facts (attr) enumerated and described. To the description of an attribute belong:

- its identifier (id tag),
- its SQL description (sql tag),
- its format (format tag),
- its type (type tag) - this field usually has the values “additive“ or “derived“, which points out that the attribute is in summable form, or that its value could be derived from the values of other attributes. In case of a derived attribute an additional field (calculation) isneeded , in order to announce how the attribute value is to be computed.
- finally, there is one more description field used for the description of the attribute (des). This field consists of two entries: a long and a short description (program segment 33)

```
attr = (
   (
      id = "dollar_sales"
      sql = "dollar_sales"
      format = "$#,##0.00"
      type = additive
      des = ("Dollar Sales" "$ Sales")
   ) // dollar_sales
   …
)
```

Program segment 3

Following the facts tables description, all dimensions and their aggregations are described. First, the sql value of one dimension is stated (sql) and then its primary key (key) and its long-short description (des) (program segment 4).

```
dim = (
(
   sql = "Product"
   key = "product_key"
   des = ("Product" "Product")
```

Program segment 4

Further, the aggregations of each dimension are described (agg). In order to recognize the relevant hierarchy from this description, every aggregation possesses a predecessor and a successor field (prev, next). Program segment 5 shows the description of the aggregation “Sales District“ (dimension “Store“). On account of the fact that this aggregation stands in the hierarchical representation between “sales region“ and “city“, these two are stated as its predecessor and successor. To the specification of an aggregation, also belong its SQL - denotation (sql) and the long and the short description (des).

```
(
   prev = ( “sales_region" )
   next = ( "city" )
   sql = "sales_district"
   des = (  "Sales District", "Sales Dist." )
) // sales_district
```

Program segment 5

Metadata files of the presented structure can completely describe one data warehouse. But, different user groups should be able to see different data of the data warehouse. One possible way of doing this is the one we have applied in this security model. It reduces insight into whole data warehouse data For each user group one reduced (view of) data warehouse is defined and described by its own metadata file. These new metadata files are derived from the original metadata file.

The users of one specific group then have the impression that nothing does limit them. They may undisturbed browse through their data warehouse using “drill-down“ and “roll-up“, set up different queries or rotate and sum up data. The security implemented in this way spares the developer some of further inspection measures.

A “reduced“ data warehouse mostly looks different from the original. It can have its own hierarchies. If e.g. one user group in the dimension “Store“ instead of all four hierarchy levels (“Sales region“→ “Sales District“→ “city“→ “Store name“) may only see “Sales region“ and “city“, than the hierarchy of the “Store“ dimension for this user group is: “Sales region“→ “city“. The users of this group can no more access the Sales District and the lowest hierarchy level (“Store name“). They do not even know that these levels exist.

3.3. System structure

We consider the structure of the WWW-EIS-DWH [10] project. Its structure consists of three layers: extraction layer, R-OLAP layer and presentation layer. The two last layers are particularly important for the realisation of the security in our information system.

- **The R-OLAP layer:**

Among other components, which are important for the realisation of WWW-EIS-DWH information system, the R-OLAP layer includes the description data (metadata) of our data warehouse. This data play one of the most important roles in the realisation of the security concept. At the beginning of the bridging process, metadata are extracted from information sources and converted into a homogeneous format (extraction and transformation layer) in order to be finally implemented into the data warehouse. Changing the content of the data warehouse may, in a case of relevant modifications, cause the corresponding changes in its description data (metadata).

After the bridging process is successfully finished, we get a new physical database called data warehouse. This database includes data selected from different information sources according to the crucial decision supporting rules. The metadata, corresponding to this new database, describe the content of the data warehouse. In this phase the metadata include only a pure description of the data warehouse content. The security model extends the spectrum of the metadata by entries for security purpose.

- **The presentation layer:**
  The security aspect of the presentation layer considers:

- general access rights in the information system
- decoding of encoded queries
- encoding of results of a queries, which are not contained in cache
- syntactic analysis of queries received from the Internet
- registration and definition of user and user groups as well as their administration
- View definition and its administration

3.4. The M-View environment

We consider a simplified representation of our project in security related manner (figure 1).

The main components of this model are:

- The Security Manager
- The Secure Query Management Layer (SQML)

We will further enter into single main components. Apart from these components, we will deal with performance relevant cached EIS pages and OLAP tools important for result generation as well as with the EIS page generator which converts the supplied data into a representable form.

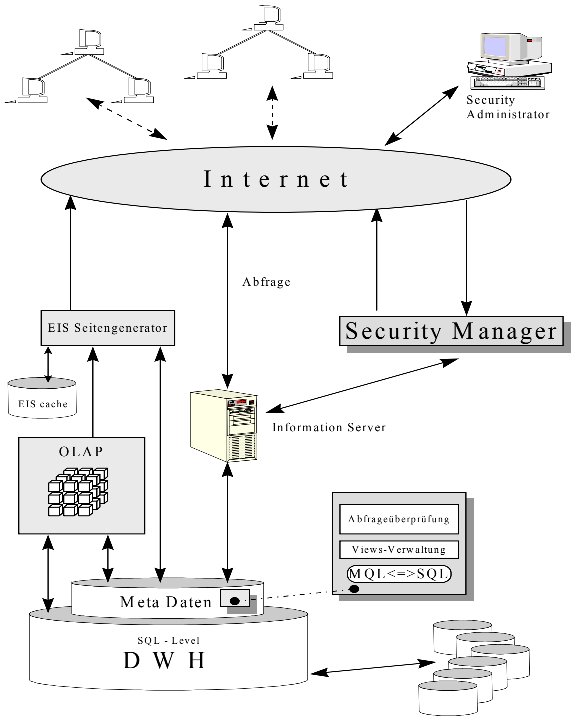


Figure 1: M-View structure [7]

3.4.1. Security Manager. The “Security Manager“ deals with following security administrating points:

- **The user groups and corresponding views: administration, definition and description**

Different user groups can be defined. While creating new user group we will precisely specify which data the users of this user group may access or rather what data are not to be disclosed . Therefore, all users of an user group have access to the same information, they have the same “view“ of data.

One of the tasks of the Security Administrator is to accomplish the user group administration, definition and assignment. Since these are very important tasks, one authorization check has always to be performed before one of the tasks is carried out.

Defining new user groups is only carried out by security administrators and managed as follows:

The security manager gets a dialogue input mask (see figure **Fehler! Keine gültige Verknüpfung.**), which contains facts, their attributes, dimensions with corresponding aggregations and values, relative to the considered MDDB.

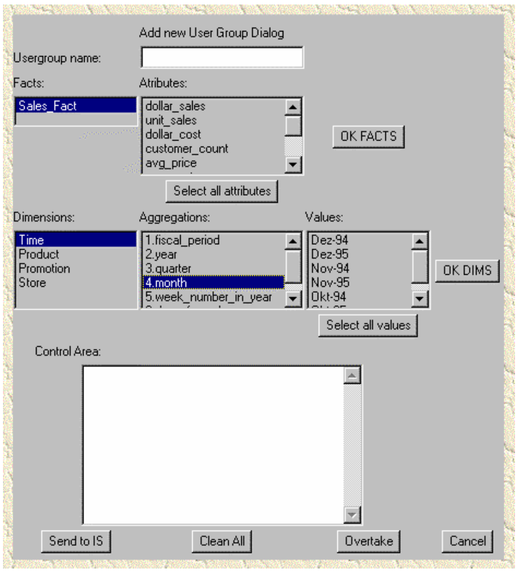


Figure 2: Add New User group dialogue window

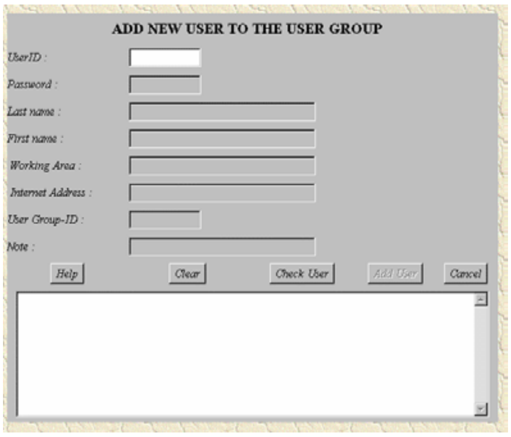
First, the security manager enters the name of the new user group. Then, he selects the first fact table from the list of existing fact tables, which is to be seen by this user group. After that, the list of corresponding attributes for this fact table appears in the „Attributes“ - list box. Security manager selects only those attributes, which should be accessible by this user group. After this, he (she) selects the dimensions and their aggregations with corresponding values, in the same way as before, for facts. Due to the easily understandable user interface design, security administrator can fulfil every action per mouse click.

After the Security Administrator has defined a new user group, the access rights of this new user group are saved. The access restrictions of a new user group are stored in the form of an MQL statements.

- **The users: administration, definition and description**

Only the security administrator can register new users. In this case, it is only a matter of entering user’s data into already existing user groups. To perform this task, the security administrator gets a “Add New User To The User Group“ dialogue window. (figure 4). Here is where he (she) enters the necessary user data: user-group-ID, first and last name, working area, user login, user password and so forth.

There are similar dialogue windows for deletion, update and search of users. The “Delete User“ window allows the security administrator to strike one user off the user list of the information system. After a user was deleted, he can no more access the data in data warehouse.



ADD NEW USER TO THE USER GROUP

UserID :

Password :

Last name :

First name :

Working Area :

Internet Address :

User Group-ID :

Note :

Figure 3: To The User group Add New user

The security administrator is also the only person who can change user entries (working area, password and so forth). The change of user-ID and user-group-ID entries is intentionally disabled here.

3.4.2. Secure Query Management Layer. The Secure Query Management Layer (SQML) is implemented on information server component. Its sphere of activity extends from the metadata layer to the information server component. The main task of this layer is to receive user queries from Internet and to check if they are allowed to be performed.

As soon as the information server receives encoded data from the Internet, it attempts to decode them and to convert them into am MQL query. SQML is responsible, among other tasks, for the syntactic analysis of received query. After a successful syntactic analysis, SQML tries, on the basis of user login and password, to assign this query to one of the predefined user groups. As soon as the evaluation of the user group and the assignment of the access restrictions belonging to it are carried out, the user query will be submitted for additional query scope testing analysis. In this inspection phase it will be checked, whether the domain of the user query is within the scope of allowed data area. Concerning the corresponding user group limitations, the user query will be modified, replaced or retyped, so that no unauthorised data disclosures may occur. [8]

For every valid user query directed to the information system the access rules and filtered metadata will be assigned on the fly. These access rules and user(group) dependant view are represented through the user group dependant content of metadata. Both of them are written in well structured MQL description language. With the help of MQL, the existing and on user identification dependant fact tables, dimensions, aggregations and attributes are structurally described for each user group depending on current security restrictions.

Since the access rules and view description are well structured, one can put this data (in the form of nodes and leaves) into a data structure suitable for further processing.

An example of the entire present data structure of the WWW-EIS-DWH is represented by a star scheme (figure 4).

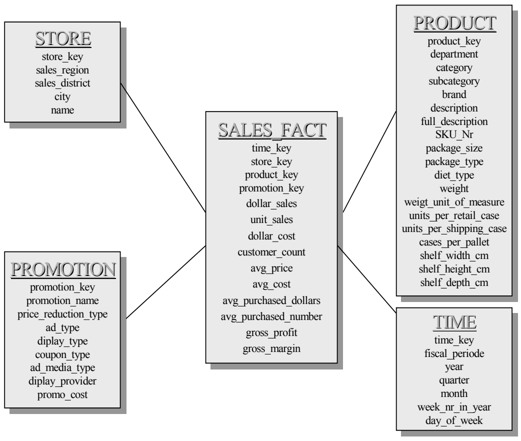


Figure 4: Star scheme “WWW-EIS-DWH“ [8]

3.5. Mode of operation of M-View

The working method of M-View can be organised into two great working phases: off-line stage (Security manager) and online stage (Security Query Management layer). [8]

After one user had identified him(her)self, one predefined “default“-EIS page, with public accessible data, appears. The user can move freely in this data set, using roll-up or drill-down navigation through the data. If the customer does not find fitting answers in this data area, he or she ought to design an additional query with the help of the “ad-hoc“ query-generator and send it to the information server. This means that, in case that the data on the publicly accessible EIS page are not sufficient to give an answer to the users question, the user (after the inevitable identification) forms his (her) query with the aid of “ad-hoc“ navigator. Already in this stage (on account of the user identification), the corresponding view of the data warehouse for this user is going to be made. Afterwards, the user sends the newly designed question to the information server in the form of an encoded query.

The information server first decodes the encoded query. It analyses the received query, passes it through the general correctness inspection and delivers an answer to the client browser. In case the query does not correspond to the security regulations defined in SQML, an error message is generated and sent to the client in form of a report. Otherwise, the query will be passed on for further inspection together with user ID and password . On the basis of the user identification (login and password) the customer role is determined and the corresponding view is assigned. In order to send the query results to the client, the query is transmitted to the OLAP engine. The OLAP engine supplies results under the consideration of the evaluated user view. . Results gained in such a way are converted into an EIS page and sent back via net to the client browser. Some often requested EIS pages are cached and they are, instead of OLAP calculations, immediately returned as a result of user query.

4. Conclusion

Through the integration of a security model in a data warehouse, new requirements need to be integrated.

Security aspects should already be considered in the design phase of the data warehouse to better match the security requirements and to avoid later fundamental, cost-intensive adaptations. For the identification of security requirements legal, audit, network and other issues have to be considered.

In the second part of this paper a security model prototype was introduced. This prototype implements a security model based on metadata. This model assigns one view of reduced data warehouse to the each single user group. Users of this user group can freely navigate through the reduced data of the data warehouse.

This security model restricts the scope of user queries to data that do not threat the security regulations of the information system. Due the Secure Query Management Layer (SQML) the user access area is on-the-fly limited to the predefined amount of data. This software solution provides insight only into data areas, which are predefined by the respective security measures. It simultaneously creates the impression that the content of the data warehouse, for respective user, actually is not more, than what is to be seen. That is why it is essential that all data transformations and format conversions are carried out transparently for the end-user.

5. References

[1] Alex Berson, Stephen j. Smith, *Data Warehousing, Data Mining & OLAP*, McGraw-Hill Series on Data Warehousing and Data Management, 1997

[2] [Pernul, 1994] Günther Pernul, *Database Security*, Advances in Computers, Vol. 38, p. 1-72, 1994

[3] Slemo Warigon, *Data Warehouse Control and Security*, Association of College and University Auditors LEDGER, Vol. 41, No. 2, April 1997; pp. 3-7

[4] Sam Anahory, Dennis Murray, *Data Warehouse*, Addison-Wesley, 1997

[5] Andreas Mangold, *Zugriffsschutzverfahren für Data Warehouse*

[6] R.Kimball, *The Data Warehouse Toolkit - Practical Techniques For Building Dimension Data Warehouse*, John Wiley & Sons

[7] N.Katic, *Ein Metadaten-basiertes Sicherheitsmodell für OLAP Datenbanken*, MSC. Thesis, Vienna University of Technology, 1997

[8] M.Stolba, *Sicherheitsmodellierung in Data Warehouse Systemen*, Vienna University of Technology, 1997

[9] N.Katic, R.Kirgöze, M.Stolba, A M. Tjoa, "A Security Concept For OLAP", Dexa '97, IEEE White Paper, 1997

[10] A.Kurz, A M. Tjoa, "Data Warehousing within Intranet: Prototype of a Web-based Executive Information System", IEEE White paper, 1997

[11] A. Perkins, *Developing a Data Warehouse, the Enterprise Engineering Approach*, Visible Systems Corporation, 1995-96