DYNAMIC DATABASE DESIGN FOR SECURITY SYSTEM EVALUATION

Karol GRONDŽÁK

University of Žilina, Faculty of Management Science and Informatics, Slovak Republic

e-mail: Karol.Grondzak@fri.uniza.sk

Monika VÁCLAVKOVÁ

University of Žilina, Faculty of Management Science and Informatics, Slovak Republic

e-mail: Monika.Vaclavkova@fri.uniza.sk

Abstract

In this paper the process of security system database design will be described. This task is a part of the complex software for Integrated Security System evaluation design process. To represent the different types of data needed for the software design, Entity Attribute Value database model was applied. The advantages of this approach are demonstrated and emphasized in the paper.

Because of the complexity of the designed application, the layered architecture was used. The variety of the represented data required the application of polymorphism paradigm at the application layer. It resulted in the request for dynamic database model at the database layer.

Keywords**:** *integrated security system, dynamic database model, Entity Attribute Value database model, relational database*

1 INTRODUCTION

Protection of persons and property is becoming one of the prominent goals of the society recently. It must be taken into consideration when new protected object is being designed or when the security of existing object is being evaluated. Protected object security is realized by installing a security system. Basic components of the security system comprise:

- Mechanical, electronic and procedural access control
- Intrusion detection (with appropriate response procedures)
- Personnel Identification (authentication)

All components together form an Integrated Security System (ISS). ISS security level is characterized by the time needed for the intruder to reach his goal (t_R). This time depends on the properties of the ISS components. In this paper we will consider only mechanical access control components [1].

The most important property of the mechanical access control components is their breakthrough resistance. The breakthrough resistance characterizes the resistance of a material of the mechanical access control component against the agency of various tools. It is expressed by the time, needed to break through the component [2].

Recently, the ISS evaluation is performed by the security managers manually. The goal of VEGA project is to develop software system to automate this time consuming process. New software system has to allow the security manager to model the evaluated protected object. It includes the topology of the object, location of the security components, algorithm for time t_R estimation and visualization of the object and its security components.

The main advantage of the automated evaluation system is that it can determine with higher accuracy the characteristics of the evaluated ISS. This can lead to cost reduction when designing the ISS for protected object.

2 PROBLEM ANALYSIS

One of the modern approaches to develop a complex application is multi layered architecture. It generally consists of minimum three layers:

- Presentation layer, responsible for communication with users,
- Application layer, implementing the business logic,
- Database layer, storing the data.

The main advantage of this architecture is the flexibility of design. Each layer can be designed and developed independently. In the case of necessity it is possible to independently replace and of the layers with another implementation. If the application grows, it is easy to scale it according the customers' requirements. In this paper we will consider mainly the database layer.

To allow computer support for the evaluation of the ISS by security manager, it is necessary to be able to model the evaluated object including ISS components in a computer. Nowadays (according authors' knowledge) there is not any available specification of the database structure to store the object topology and ISS components properties [3].

The database structure is the basic part of the ISS evaluation application. Properly designed structure allows efficiently storing and retrieving the necessary data. The mostly used database system concept nowadays is relational databases.

A relational database stores data in a form of collection of data items, which are organized as a set of tables. In a relational database model a set of tables is related by relationships. A relational integrity of data is ensured by foreign keys [4].

During the process of the problem analysis and architecture design the general structure of the database was proposed. We identified two main parts of the data, which are to be stored in the database. The first part is topological data describing the evaluated object including all rooms, corridors, elevators, staircases, etc. The second part is security components data and their properties, e.g. the location of the security component in the evaluated object, its breakthrough resistance, etc.

3 DYNAMIC DATABASE DESIGN

Database design depends on the data access mode. In our case the application layer must be able to manipulate with variety of security components characterized by different groups of attributes. Application layer is using polymorphic approach to handle this situation. To reflect such a design in the database layer, we proposed dynamic database model sometimes referred as Entity Attribute Value (EAV) database model [5].

This data representation is similar to space-efficient methods of storing a sparse matrix, where only non-empty values are stored. It is typical, that EAV tables have a lot of rows with a limited amount of columns.

Using static database model to represent values of security components, each component would be modeled by separate table. This table would contain one column for each security component attribute. Such a model is difficult to maintain and extend, if a new security component should be added. It would also require modification of the code of application layer. Such approach is not acceptable, in terms of software engineering. It requires additional maintenance costs and is error prone.

In an EAV data model, each security component attribute-value pair is stored in a row of a single common table. Such a model allows easy to maintenance and extending. It does not require modification of the code of application layer, when new security component is added. Such approach is preferable, because it minimizes additional maintenance costs.

General description of this model is as follows [6]. Data is recorded as three columns:

- The entity: the item being described.
- The attribute or parameter: a foreign key into a table of attribute definitions. At the very least, the attribute definitions table would contain the following columns: an attribute ID, attribute name, description, data

type, and columns assisting input validation, e.g., maximum string length and regular expression, set of permissible values, etc.

- The value of the attribute.

3.1 Dynamic database structure

Above mentioned general dynamic database concept has been adopted for our database model. In our case the entities represent the security component type (e.g. security door, security lock, etc.). They are stored in the table TypPrvku. Each row describes unique security component by its name (column nazovTypu) and description (column popis).

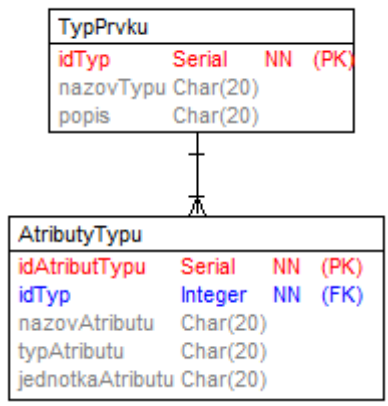

Figure 1 Core structure of security components

Each unique security component is characterized by variety of different attributes of different data types (time, length, text, etc.). Attributes of security component are stored in AtributyTypu table. These two tables are related using primary key of the TypPrvku table. Table AtributyTypu stores for each attribute the following values: name of the attribute (column nazovAtributu), data type of the attribute (column typAtributu) and measure unit of the attribute [7] (column jednotkaAtributu) (Figure 1).

The value of the column typAtributu determines the method of operation on the attribute value in the application layer.

The concrete value of the attribute of some security component is stored in HodnotyAtrBP table (Figure 2). The attributes producer and material of the security component are exception of this principle. They are common to all the components, so it is more convenient to store them in separate tables (Material and Vyrobca, see Figure 2).

This approach allows flexibility in managing the values of the security component attributes. It is possible to easily add new attributes to the security component. If for some security component instance the value of some attribute is not relevant, it will not be stored. This will lead to the saving of space in the database.

Another advantage of this approach is the support of polymorphic concept in application layer. It is realized by the table BezpecnostnyBod. This table contains particular security components with their respective attributes.

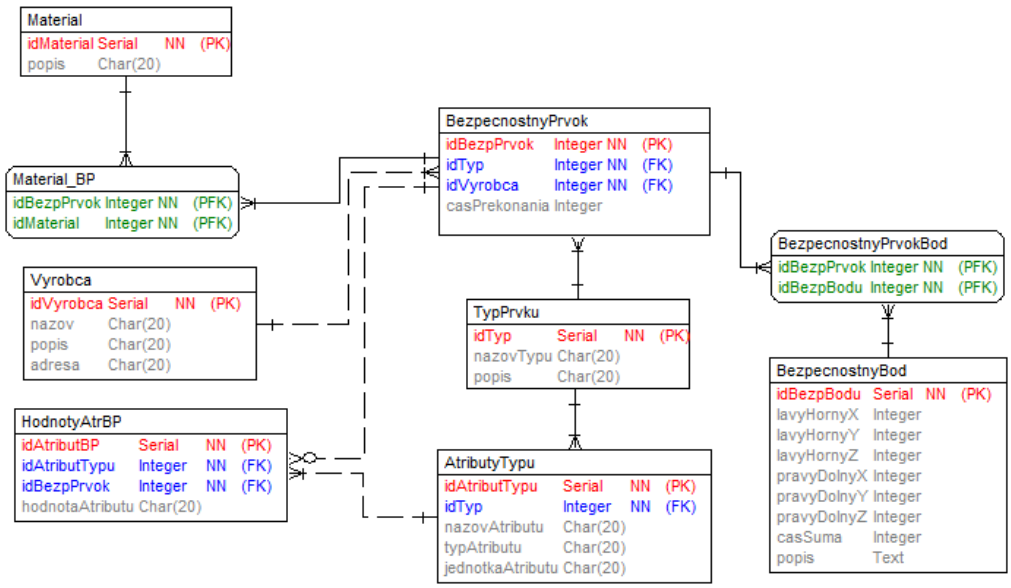

Figure 2 Dynamic part of the database model

4 CONCLUSIONS

In this paper the paradigm of dynamic database design was described (sometimes referred as Entity Attribute Value database model). This paradigm has a lot of advantages over the traditional approach, as has been demonstrated in the paper. On the other hand it requires different approach to the maintenance of stored data. It also influences the application layer design.

We applied this general paradigm to the problem of security components and their attributes representation. There are many different types of security components and each of them can be characterized by a very diverse set of attributes.

In traditional approach modeling such a diverse data would require a table with hundreds of columns with many of them not being used. Adding a new security component or its attribute would require change of the database model and consequently an application layer code.

Applying the dynamic database model we were able to solve these problems. This approach allows for reduction of maintenance costs on both database and also on application layer.

ACKNOWLEDGMENT

This research was supported by the project VEGA No. 1/0981/11.

REFERENCES

[1] BOC, K., VIDRIKOVÁ, D.: Návrh hodnotiacich parametrov mechanických zábranných prostriedkov na úseku plášťovej ochrany (stavebné prvky objektov). In: IBSES 2012: interdisciplinárny vedecký workshop o hodnotení účinnosti integrovaných bezpečnostných systémov pomocou expertných systémov, 22.3.2012, Slovenská republika. EDIS 2012. ISBN 978-80-554-0554-4

[2] REITŠPÍS, J., MESÁROŠ, M., BARTLOVÁ, I., ČAHOJOVÁ, Ľ., HOFREITER, Ľ., SELINGER, P., Manažérstvo bezpečnostných rizík. EDIS 2004. 289 pp. ISBN 78-80-8070-823-8

[3] RISTVEJ, J., ZAGORECKI, A.: Information system for crisis management - current applications and future directions. In: Communications, Scientific letters of the University of Žilina. No. 2 (2011), pp. 59-63. ISSN 1335-4205

[4] MATIAŠKO, K., VAJSOVÁ, M., ZÁBOVSKÝ, M., CHOCHLÍK, M.: Databázové systémy a technológie (in Slovak), Nakladateľstvo STU, Bratislava 2009. 693 pp. ISBN 978-80-227-3025-8

[5] NADKARNI, P. M., MARENCO, L., CHEN, R., SKOUFOS, E., SHEPHERD, G., MILLER, P.: Organization of Heterogeneous Scientific Data Using the EAV/CR Representation. Journal of the American Medical Informatics Association. 1999;6: pp. 478–493

[6] DINUA, V,. and NADKARNIA, P.: Guidelines for the Effective Use of Entity-Attribute-Value Modeling for Biomedical Databases. Int J Med Inform. 2007; 76(11-12): pp. 769–779

[7] CHOCHLÍK, M.: Representing physical quantities in a relational database system. Journal of Information, Control and Management Systems. Vol. 6, No. 1 (2008), pp. 57-66. ISSN 1336-1716