On the Differences of Relational and Dimensional Data Model

Mladen Varga, **Ph. D.**
University of Zagreb
Faculty of Economics, Zagreb
mladen.varga@efzg.hr

Abstract: *Relational modelling is used for the modelling of transactional data stored in relational databases while dimensional modelling is prevalently used for the modelling of dimensional data stored in data warehouses. The modelling approach used is quite different. In relational modelling the focus is on identification of fundamental or strong entities involved in the execution of business transactions, while in dimensional modelling the focus is on identification of associative entities that carry business measures. In the paper some equivalence problems of two models are discussed. Also, the main steps of transformation of relational into dimensional model are described.*

Keywords: *relational data model, dimensional data model, relational modelling, dimensional modelling*

1. INTRODUCTION

Today we are witnessing two main database models. Relational model is widely used in relational databases in transactional systems, while dimensional model is prevalently used in data warehouses or decision support systems.

In this section, some general facts of relational and dimensional model are mentioned. In the section 2 some similarities and differences of both models are discussed. A small hospital example of both models is shown in the section 3.

The transactional systems that lean on the relational model may be used as a starting point in the process of modelling the data warehouses. The main steps of transformation of the relational in dimensional model are described in the section 4. In the same section some equivalence problems of the two models are discussed.

1.1 RELATIONAL MODEL

A relational database consists of a set of *relations*. A relation schema which is used to describe a relation r, denoted by $R(A_1, A_2, \ldots, A_n)$ is made up of a relation named R and a list of attributes $A_1, A_2, \ldots, A_n$. Each attribute A_i is the name of a role played by some domain D in the relation R. A relation r of the relation schema $R(A_1, A_2, \ldots, A_n)$ is a set of tuples $r=\{t_1, t_2, \ldots, t_m\}$. A tuple usually contains data of an entity (object) while a relation contains data of all entities of the same type.

If for any two distinct tuples t_1 and t_2 in the relation r of R there exists an attribute (set of attributes) K such that $t_1[K] \neq t_2[K]$ then such an attribute (set of attributes) is called the key. The chosen *primary key* is used to identify tuples in the relation.

A *foreign key* implements the relationship between two entities:

- The attributes in the foreign key have the same domain as the primary key of another relation schema R_2. The foreign key are said to reference the relation R_2.
- The value of the foreign key in a tuple t_1 of R_1 either occurs as a value of primary key for some tuple t_2 in R_2 or is null.

1.2 DIMENSIONAL MODEL

Let us suppose that a hospital database contains the admission data (such as the number of days and the results) of the patient, the date of admission, and the diagnosis. The admission data is determined by attributes PATIENT, DIAGNOSIS and TIME. They are referred to as *dimensions* while the admission data is referred to as *measures* or *facts* or *fact attributes*. The measures are mostly numerical, preferably continuously valued and additive. They vary over time. In the statistical database field [5] the dimension corresponds to *category attribute*, and measure to *summary attribute*. There is no a priori distinction between dimensions and measures while any attribute can play either role [3]. There is no formal way to decide which attributes are dimensions and which attributes are measures. This decision has to be solved during database design.

The structure of the *dimensional model* can be represented by the *star join schema* [4]. The centre of the schema is the *fact table*, which is the only table in the schema with multiple joins connecting it to other dimension tables. The fact table is where the measures of the business are stored [4] such as NUMBER_OF_DAYS and RESULT in the hospital database. The other tables are the *dimension tables*. *Dimension attributes* describe the item in the dimension, and are virtually constant over time. The primary key of the fact table is composite or concatenated key, which is the combination of as many foreign keys as many dimensions there are in the schema. Each component of the composite key is a foreign key referencing the primary key of a dimension table. In other words, every fact table represents a many-to-many relationship. It contains as many foreign keys as many dimensions there are in the schema. A multidimensional database consists of any number of star join schemas with some dimension tables overlapping.

The dimensions are usually organized into *hierarchies* that specify aggregation level and hence granularity of viewing dimensional data. The hierarchy of a dimension is defined by a sequence of functional dependencies $D_1 \rightarrow D_2, ..., D_{k-1} \rightarrow D_k$ over a set of dimensional attributes $D=\{D_1, D_2, ..., D_{k-1}, D_k\}$. For example Time={Day, Week, Month, Year} and its hierarchy is defined by functional dependencies Day→Week→Month→Year.

2. SIMILARITIES AND DIFFERENCES OF RELATIONAL AND DIMENSIONAL MODEL

2.1 PURPOSE OF MODEL

Relational model

The relational model is used in transactional systems where many transactions are executed, most of them concurrently. A transaction inserts, updates or in any other way processes data in a database. In many occasions a transaction is an integral part of the business process. As an example, a bank transaction is a business transaction and at the same time a transaction of its transactional information system.

The relational model must serve the transactional system in the best way. Since the transactional system executes many transactions, the relational model has to trace the execution of each transaction in the system. In other words, each transaction has to be traceable throughout the data model. In some way such a data model is a *process flow model* because *the data in the flow of business process is modelled*.

Dimensional model

The dimensional model is used in the decision support systems or the data warehousing systems. The data in the decision support systems represents the effects of the business process. They are the effects of many transactions executed in the transactional system. As opposed to the transactional system in the decision support system the trace of the transactions is not visible. Only the effects of their execution have to be visible.

The dimensional model is therefore an *informational model* because *the data of the effects of the business process is modelled*.

2.2 TYPE OF ANALYSIS

Relational model

In the relational *business process flow modelling* the execution of each transaction is traced. Many entities that are involved in the execution of the transaction are incorporated in the model. All entities are fundamental (strong, independent) - sometimes with weak (dependent) entities paired to them. For example, a purchase order is a fundamental entity. The purchase item is the weak entity paired to the purchase order entity.

The structure of the relational model shows the relationships of entities that are important *in the execution of the transactions of the business process*.

Dimensional model

In the dimensional *process effects modelling* we are interested in the modelling of the effects of many transactions executed in the business process. These effects are various business measurements. They are usually numeric and are taken from individual transactions, such as the purchase item order or customer withdrawal at an automatic teller machine. The most important attributes are the numeric measures (facts) about business. In statistical terminology they are summary attributes.

The structure of the dimensional model shows the relationships of entities (dimensions) that are important *in modelling of business measures in the business process*.

2.3 DETAILS OF ANALYSIS

Both in relational and dimensional modelling the data modelled is of one transaction. The transaction is always connected to an elementary business process, such as the withdrawal at an automatic teller machine.

Relational model

In the relational modelling the execution of the transactions is considered. In this *process flow modelling* it is very useful to find one or more business entities, such as the patient in the example shown in chapter 3, and follow them through their entity life cycle. Through the life cycle of the business entity many informational entities may be found. For example, the product is an important business entity. In its life cycle the (product) purchase order and the purchase item are the informational entities found.

The structure of the relational model shows *the relationships between the fundamental (strong) entities in the course of the execution of one or more transactions*.

Dimensional model

In the dimensional modelling the effects of the transactions are considered. The effects are shown in the attributes of associative entities or in the attributes of many-to-many binary relationships or n-ary relationships. These attributes are business measures. They are stored in fact tables that are implementations of associative entities or many-to-many binary or n-ary relationships.

The structure of the dimensional model shows *the relationships of associative entities connected to the fundamental dimensional entities*.

3. EXAMPLE

The relational and the dimensional modelling are shown in the following small example of a typical hospital information system. The decomposition diagram of the Hospital process is shown in Figure 1. The Hospital process is decomposed into the Patient admission, the Patient treatment and the Patient discharge. The Patient admission is further decomposed into the Patient registration, the Patient examination and the Patient transfer to ward. These processes are transactions executed within the Hospital information system.

Hospital
- **Patient admission → *Diagnostics, Accommodations***
  - **Patient registration**
  - **Patient examination**
  - **Patient transfer to ward**
- **Patient treatment → *Treatments, Medicines***
- **Patient discharge → *Costs***

Figure 1: The Hospital process (information system) decomposition

In Figure 2 the relational process flow model of the Patient admission is shown. The Patient registration entities with their relationships are PATIENT, TOWN, ADMISSION and DOCTOR (admission doctor), the Patient examination entities are PATIENT, ADMISSION, DOCTOR (diagnosis doctor), DIAGNOSIS and ADMISSION_DIAGNOSIS, and the Patient transfer to ward entities are PATIENT, ADMISSION, DOCTOR (doctor in cure), WARD and ADMISSION_WARD.

The relational model is depicted in the form of the entity-relationship diagram. The relational schema, not shown here, may be constructed by well-known transformation rules for transformation of entity-relationship model into the relational schema.

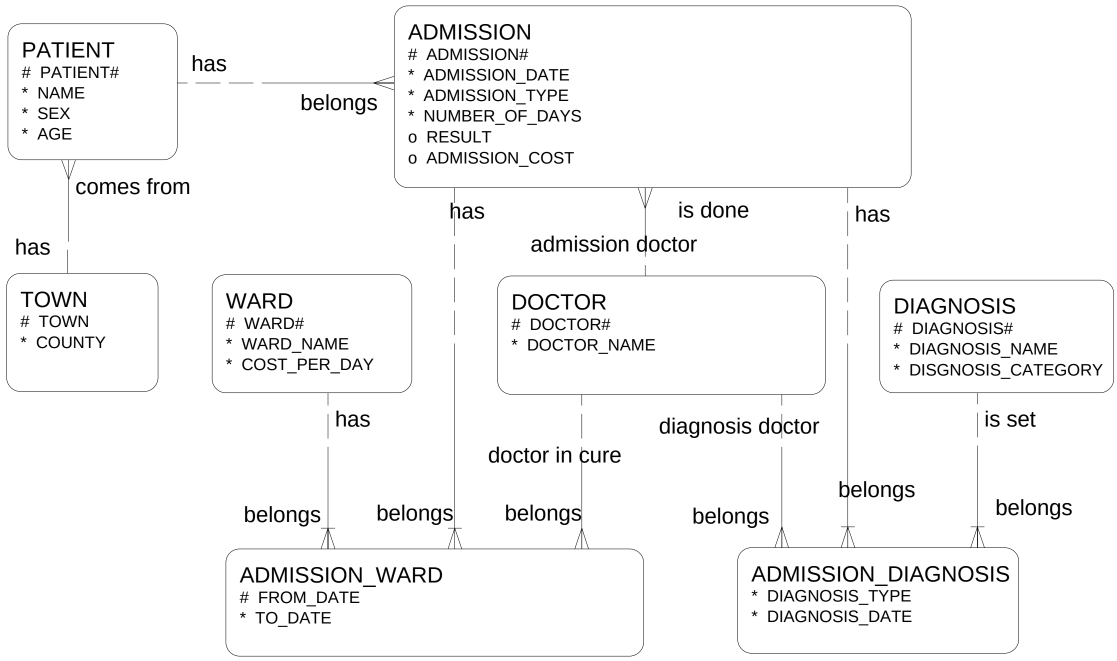

Figure 2: The relational model of the Patient admission process

The two effects of the Patient admission process are Diagnostics and Accommodations. Figure 3 shows the Accommodation effect. The dimensional model is also shown in the form of the entity-relationship diagram. The central entity is ACCOMMODATION that is the associative entity connected to a series of dimensional entities: AGE, SEX, PATIENT, WARD, ADMISSION, DOCTOR (admission doctor, doctor in cure) and TIME (admission date, accommodation date).

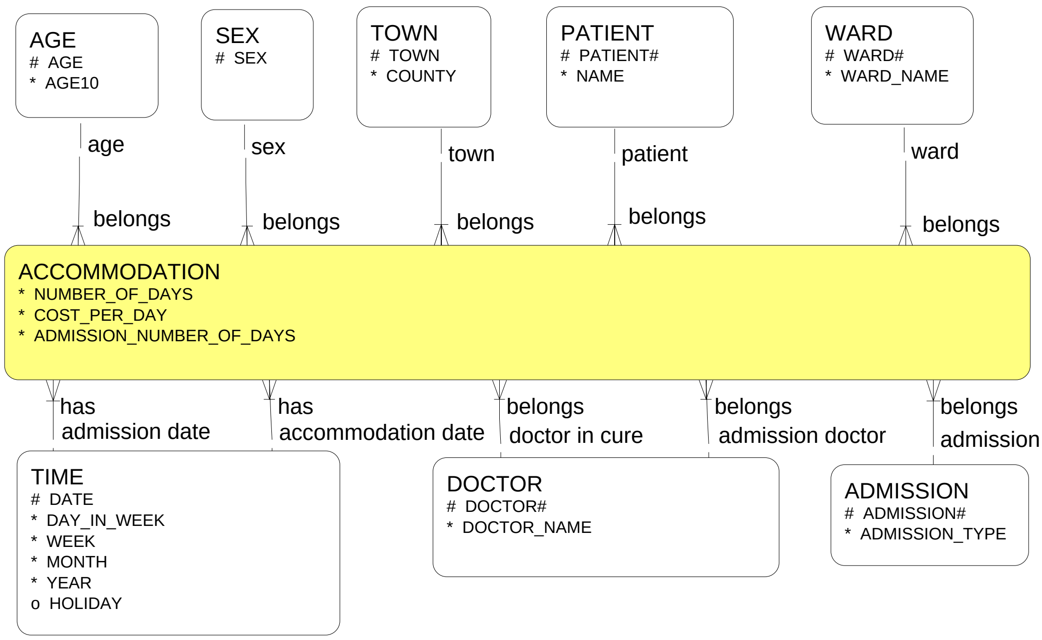

Figure 3: The dimensional model of the Accommodation effect

4. EQUIVALENCE AND TRANSFORMATION OF MODELS

4.1 EQUIVALENCE OF DIMENSIONAL AND RELATIONAL MODEL

A few questions may be raised: Can a dimensional model represent a relational model or vice versa? Is it possible to transform a relational model of the enterprise into an equivalent dimensional model? The answer depends on the relationships used in the relational model [1]. If there are associative entities, n-ary relationships or many-to-many binary relationships in the relational model then it is possible to define the corresponding dimensional model. Only the presence of the mentioned types of relationships makes the transformation of a relational into a dimensional model possible. But, if the relational model has no such relationships then there is no direct possibility of its transformation into a dimensional model. The next question is: Has a relational model with no associative entities, n-ary relationships or many-to-many binary relationships data suitable for successful transformation into a dimensional model? Probably no if there are no attributes that have characteristics of the measure attributes [4] of the dimensional model. Without such attributes, business performance can’t be evaluated and the primary purpose of the dimensional model used in the data warehouse can’t be fulfilled.

In the correct relational model the entities containing measure attributes can only be associative entities or many-to-many binary relationships or n-ary relationships. The measure attributes do not belong to any fundamental (strong) entity but are properties of associative entities or n-ary relationships or many-to-many binary relationships among fundamental entities.

An aspect that makes the transformation of the relational into the dimensional model more difficult is the rule used by many designers of relational models. This rule says that n-ary relationships are rare in the real world and that they have to be avoided. Nevertheless, dimensional modelling relies on modelling n-ary relationships.

In the correct relational model which is intended to be transformed into the dimensional model all entities must fulfil their roles entirely, that is every strong entity must be strong entity without any characteristics (attributes) of associative entity and every associative entity must be only associative entity without any characteristics (attributes) of strong entity. If this is not true, the measure attributes may be available somewhere in the relational model and have to be manually examined and transited to associate entities in order to be appropriate for the transformation into the dimensional model.

4.2 TRANSFORMATION OF RELATIONAL INTO DIMENSIONAL MODEL

The main steps for the transformation of a relational into a dimensional model presented below are similar to the approach described in [2]:

1. Identify associative entities, n-ary relationships or many-to-many binary relationships in the relational model.
2. For each many-to-many binary relationship, n-ary relationship or associative entity:
   a) Identify the measure attributes.
   b) Identify the corresponding dimensions, i.e. strong entities involved in the relationship. Each foreign key identifies one dimension. For each dimension identify dimensional attributes. Some attributes may become measure attributes.
   c) Identify the attributes of associative entities or n-ary relationships that define new dimensions. The time attribute (date, for example) is frequently used in the associate entity or in the n-ary relationship. Open the new dimension (time, for example) and define other dimensional attributes.
   d) Identify additional dimensions from distant entities. A distant entity is not directly involved in the many-to-many binary relationship and neither in the n-ary relationship, nor is it directly associated to the associative entity. It is reachable from the associative entity via the relationship of cardinality 1, showing that the associative entity functionally determines the distant entity. The distant entity can be reached by the same rule from the previously found distant entity. The attributes of each distant entity must be carefully examined while their attributes can apply both as dimensional attributes and as measure attributes.

The steps applied to the relational model in Figure 2 to transform it into the dimensional model in Figure 3 are:

1. ADMISSION_WARD is the associative entity of the relational model.
2. For ADMISSION_WARD associative entity:
   a) No measure attribute exists directly but the measure attribute NUMBER_OF_DAYS, showing the number of days patient spent in the ward, can be computed from FROM_DATE and TO_DATE.
   b) Attributes that identify dimensions are WARD#, DOCTOR# and ADMISSION#. Their corresponding dimensions are WARD with the dimensional attributes WARD# and WARD_NAME, DOCTOR (in cure) with the dimensional attributes DOCTOR# and DOCTOR_NAME and ADMISSION with the dimensional attributes ADMISSION# and ADMISSION_TYPE. COST_PER_DAY from WARD becomes the measure attribute. The attribute NUMBER_OF_DAYS from the ADMISSION entity becomes the measure attribute as well. Since it is the second attribute with the same name it is renamed in ADMISSION_NUMBER_OF_DAYS. Also, a new dimension is identified by ADMISSION_DATE attribute. This is TIME (admission time) dimension.
   c) The new dimension identified from the attributes of the ADMISSION_WARD associative entity is TIME (accommodation time). The dimensional attribute is FROM_DATE. The TIME dimension is properly modelled with a set of dimensional attributes DAY_IN_WEEK, WEEK, MONTH, YEAR and HOLIDAY.
   d) The new dimensions identified from distant entities are as follows: A distant entity not directly connected to the ADMISSION_WARD associative entity is PATIENT. The ADMISSION entity, which is functionally dependent on ADMISSION_WARD, determines the PATIENT entity via “belongs” relationship where its ADMISSION# key functionally determines the PATIENT’s key PATIENT#. The attributes of PATIENT identify new dimensions PATIENT, SEX and AGE. The attribute NAME of the PATIENT entity is the dimensional attribute of the PATIENT dimension. Via “comes from” relationship a new

distant entity is identified. This is TOWN, because PATIENT# functionally determines its key attribute TOWN. The COUNTY attribute from the TOWN entity is the dimensional attribute that belongs to the TOWN dimension and defines the TOWN dimension hierarchy: TOWN→COUNTY. From the ADMISSION entity via “is done” relationship the admission DOCTOR is identified and the new DOCTOR (admission doctor) dimension is specified.

5. CONCLUSION

The main differences between the relational and the dimensional modelling are summarized in Table 1.

	Relational modelling	Dimensional modelling
Aim	Data modelling of transactional systems	Data modelling of decision support systems
Analysis subject	Execution of business process → *process flow modelling*	Effects of business process → *process effect modelling* or *informational modelling*
Analysis focus	Discovery of strong entities in the course of business process execution	Discovery of associative entities (relationships of strong entities) that represent the effects of business process
Analysis details	Definition of the strong entities attributes and the relationship between them	Definition of business measures – attributes of associative entities, definition of business dimensions

Table 1: Difference between relational and dimensional modelling

6. REFERENCES

[1] Firestone, J. (1998): *Dimensional Modeling and ER Modeling in the Data Warehouse*. http://www.dkms.com/dmerdw.html

[2] Golfarelli, M.; Maio, D.; Rizzi, S.(1998): *Conceptual Design of Data Warehouses from E/R Schemes*, Proceedings of the Hawaii International Conference on System Science, Kona.

[3] Gyssens, M.; Lakshmanan, L.(1996): *A Foundation for Multi-Dimensional Databases*, Proceedings of 22nd VLDB Conference, Mumbai (Bombay).

[4] Kimball, R. (1996): *The Data Warehouse Toolkit*, John Wiley, New York.

[5] Shoshani, A (1997): *OLAP and Statistical Databases: Similarities and Differences*. 16th ACK SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems.