PER model - Semantic data models developing

Hristo Tujarov, Ivelin Mihailov

Abstract: *The paper treats the developing of the semantic data models. On this basis the place and the function of PER model (processes, entity, relations) are traced.*

Key words: *Information systems, database, data modeling, semantic models.*

INTRODUCTION

Data modeling plays an important role in the process of the information system (IS) developing.

A well-developed model leads to:

- Development of an optimum IS architecture;
- Minimizing the abnormalities when saving, deleting and changing the data;
- Designing of an open structure that can undergo further changes;
- Effective user interface;
- Time saving when data processing.

The 30-year history of modeling is related to the development of a great number of models each aiming at:

- Better reflecting the meaning of the data in the model;
- Making the study of the model easier and making the model suitable for using it in the CASE tools;
- Making the transfer of things from the real to the computer world more detailed and closer to reality.

Data modeling

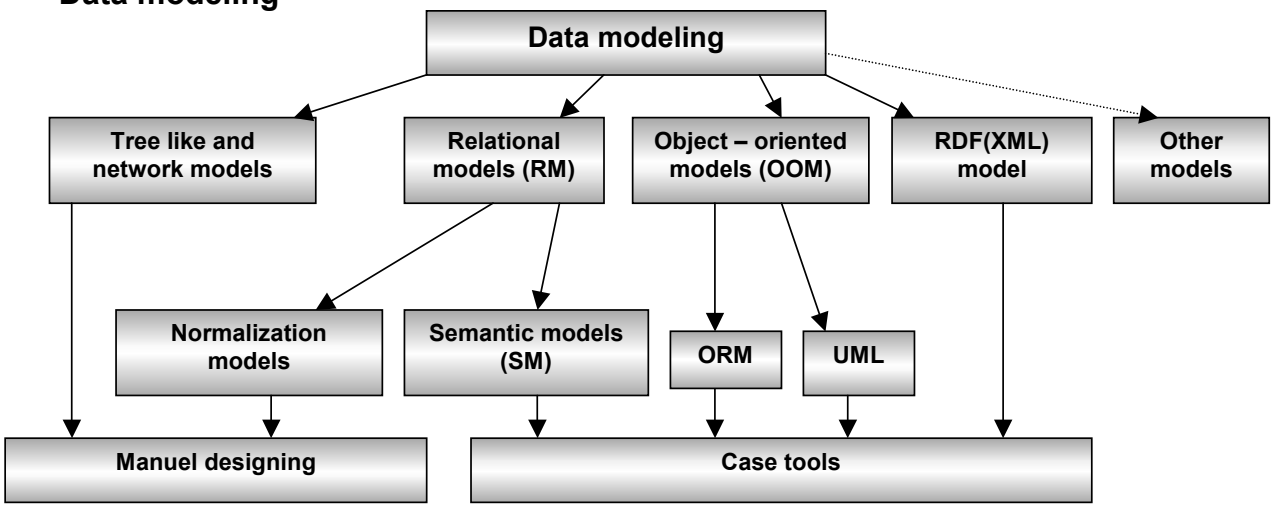


fig. 1 General data models

GENERAL DATA MODELS

Network and hierarchy (tree-like) models – They are almost replaced by the relational, the object-oriented models etc. Basic drawbacks of the models when they are used for constructing a database:

- Links of the "n-m" type are not allowed;
- The user is not granted a logic look at the database;
- The user has to be well grounded in the internal organization of the database;
- The allowed structures are tree-like;

- When deleting a record there is a possibility of many other records to be deleted. This leads to data losses.

Relational model – it is considered that the beginning of the contemporary database is connected with PhD Codd's research works. According to Date [4] he played a great role: Date commented that before approximately 30 years Codd had started working upon the matter witch afterwards had become relational data model. Date said that in 1969 Codd had published his first series of ingenious articles, describing the matter – articles which had made the world as they knew it. He claimed he was sure that after 100 years database systems would be based on the relational foundation of Codd.

The relational model concentrates on the logic description of the data from the user viewpoint. It is not connected with a concrete physical representation of the data.

Object - oriented models (OOM) - The rise of Object-Oriented Data Base (OODB) was determined by the need of a complex IS to be developed. IS which were not completely satisfied by the relational database. The semantic data modeling approach strongly influences the concept of OODB. The general features of OOM are as follows:

- Each object comprises a state (a composition of meanings of its attributes) and behavior (a composition of methods /program code/ operating on a level over the state of the object);
- The interaction between the objects is effected through message exchange and the implementation of the correspondent methods;
- The objects are able to inherit another object's attributes and they can also add attributes of their own.

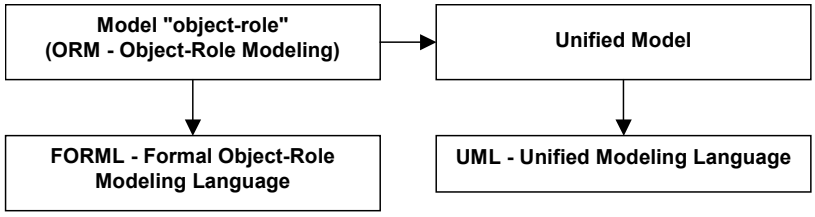


Fig. 2 OOM progress

"Object–Role" model (ORM). The Formal Object Role Modeling Language (FORML) language is created on its base. This model makes the effectiveness of the developing, higher at the expense of the combined modeling of data and processes. This leads to flexibility and intelligibility. Some of its advantages are as follows:

- The representation of the facts and rules is in English where intuitive graphic objects are used;
- It represents most of the business rules in graphics (it is more expressive) etc.;
- It can be used together with other modeling approaches.

UML – A popular language used for documenting the lifecycle, for representing and developing program products [9]. It is applicable to business modeling and other non-software systems. UML comprises the best engineering practices, which have successfully proved theirselves when modeling large systems.

RDF (XML) – In fact RDF and XML complement one another. RDF is a model of metadata and is frequently used when saving and transferring parts of the document related to its symbolic multitude, language etc. In this case (and not only), RDF relies on the XML standard.

Data normalization – the process of database developing in accordance with the rules defined by Codd is called data normalization. The normalization is carried out by stages. It is a technique of data structuring so that it allows avoiding problems when the da-

tabase is further used and refreshed. The normalization leads to logically stable structure of the records, which is easy to comprehend and maintain. Different levels of normalization could appear. The criteria, which define the levels of normalization, are called "normal forms".

The figure below shows that the types of objects and their correspondent tables should meet the following requirements in order to be normalized:

- They should be flat: that means they shouldn't include reoccurring groups of attributes;
- The data in all the non-key attributes should completely depend on the meaning of the primary key;
- Attributes shouldn't depend on each other.

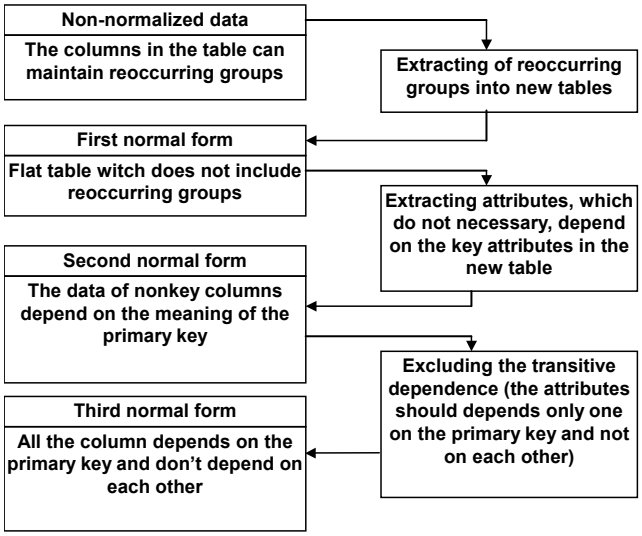


Fig.3 Normalization rules

Semantic models: The common use of the relational database in a variety of spheres proves that the relational model successfully copes with the modeling of object spheres. But the developing of relational database when using normalization tends to be a complicated and inconvenient design process. The reasons are as follows:

- Lack of tools for showing the meaning of the data;
- Representing the object sphere in one table only and its further normalization is difficult for the developers;
- The process of normalization cannot to be automated.

The need of more appropriate and powerful tools in object sphere modeling caused the rise of the semantic data models. The main advantage of semantic data models is the possibility of describing the semantics of the data. With the help of the semantic model, important for the object sphere type objects (entities) are defined, as well as their features (attributes) and the relationship between them are specified.

In [2] Cod defined the purpose of the semantic data modeling. He things that in fact the task to maintain the meaning of data is endless. The aim of the semantic modeling is of an exclusive important because even a tiny success could contribute to understanding and order in the sphere of database development.

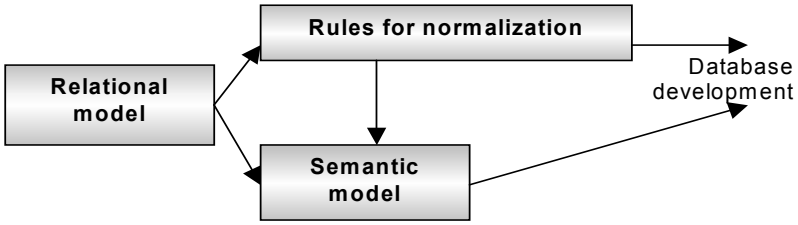


Fig. 4 Semantic models location

The semantic model progress:

The development of the semantic model took two courses, which were shaped in 70[ies] by Codd and Chen. Every new model leads to:

- Better reflecting the data meaning in the model;
- Making the study of the model easier and making the model suitable for using it in the CASE tools;
- Making the transfer of things from the real to the computer world more detailed and closer to reality.

Entity-Relationship [ER] model, which is the first semantic model, was developed by Phd. Chen in 1976 [10]. The object sphere modeling is based on the use of graphic diagrams that include a small number, various components – objects, attributes, and relationships between the objects. Chen's model is further developed by Barker [1]. Barker's method is used in ORACLE company's CASE tools. T. Ramey develops **IDEF1** method. It is based on Chen's model too, and its modernized version IDEF1X is used in a number of CASE tools (ERwin, Design/IDEF etc.). In 1979 Codd offers a new **RM/T** model. This model is the beginning of a new course in the semantic models. The objects are classified according to their role and place in the data model – cores, attributes and associations. C. Date [6] further develops Codd's model, defining the three basic object categories: cores, associations and attributes, and adding a subcategory: association objects – designations. He defines the relationships between them.

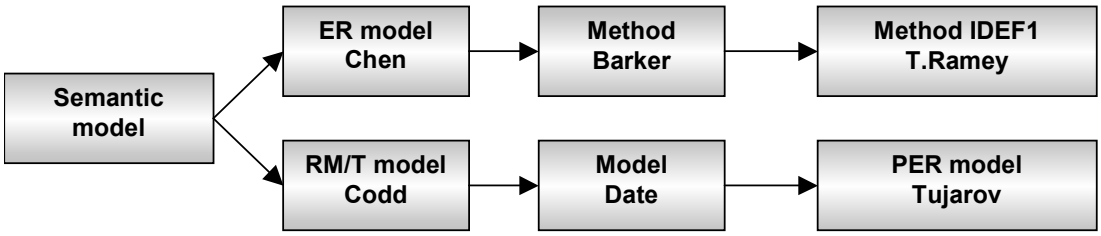


Fig. 5 The semantic models progress

PER (Process-Entity-Relationship) model (offered by PhD. Tujarov) [11], [12], [13] further develops RM/T in some directions:

- Offers a new object classification;
- Adds the processes is a new type of object;
- Puts on different levels the different types of objects, so that any ambiguity when defining the type of the relationship between the different objects is excluded.

Rules for creating a model: The primary key of a type object from a lower level is an outside key of a type object from a higher level; The relationship between the object from a lower level and the object from a higher level is '1–n"; The connection object owns a composite key consisting of primary keys of two or more objects from a lower level; If the rules are kept normalization of the data is not needed.

Order for creating: defining the process objects (OP), if any; defining the main objects (MO); taking part in the process object; defining the dependent objects (DO), if necessary; defining the reference object (OR) for all the other objects; defining the link objects (LO). Example:

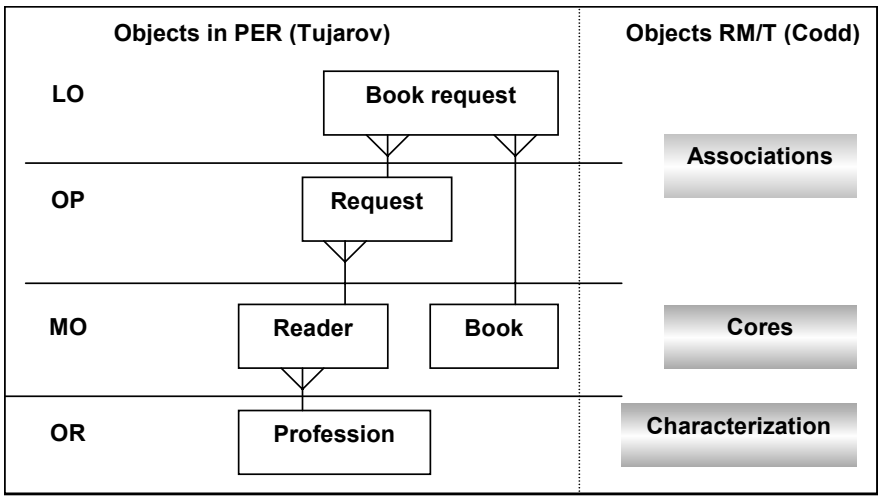


fig. 6 Juxtaposition between PER model and RM/T

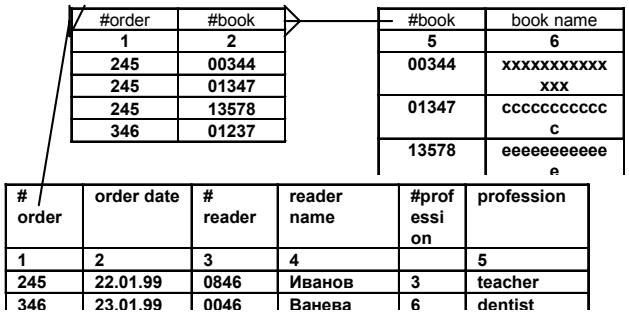
Non-normalized relation

#order	order date	#reader	profession	reader name	#book	book name
1	2	3	4	5	6	7
245	22.01.99	0846	teacher	Ivanov	00344 01347 13578	xxxxxxxxxxxxx ccccccccccc eeeeeeeeeee
346	2502.99	0048	dentist	Vaneva	01237	yyyyyyyyyy

Firs normalization form

#order	order date	#reader	profession	reader name	#book	book name
1	2	3	4	5	6	7
245	22.01.99	0846	teacher	Ivanov	00344	xxxxxxxxxxxxx
245	22.01.99	0846	teacher	Ivanov	01347	ccccccccccc
245	22.01.99	0846	teacher	Ivanov	13578	eeeeeeeeeee
346	25.02.99	0048	dentist	Vaneva	01237	yyyyyyyyyy

Second normalization form

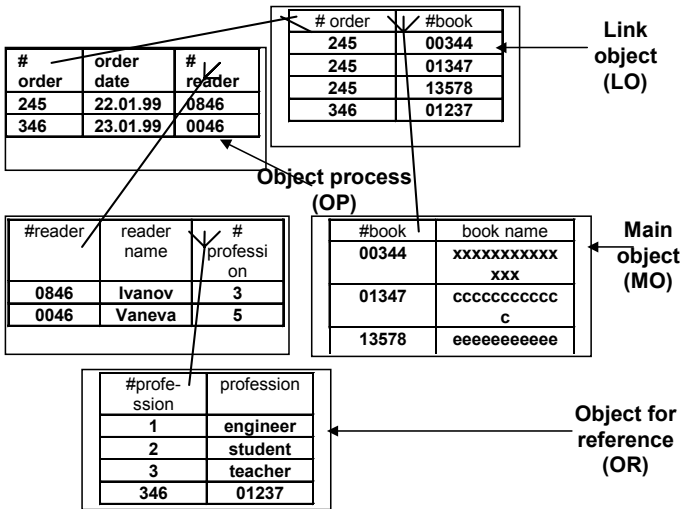


#order	#book
1	2
245	00344
245	01347
245	13578
346	01237

#book	book name
5	6
00344	xxxxxxxxxxxxx
01347	ccccccccccc
13578	eeeeeeeeeee

# order	order date	# reader	reader name	#profession	profession
1	2	3	4		5
245	22.01.99	0846	Иванов	3	teacher
346	23.01.99	0046	Ванева	6	dentist

Third normalization form



# order	#book
245	00344
245	01347
245	13578
346	01237

# order	order date	# reader
245	22.01.99	0846
346	23.01.99	0046

#reader	reader name	# profession
0846	Ivanov	3
0046	Vaneva	5

#book	book name
00344	xxxxxxxxxxxxx
01347	ccccccccccc
13578	eeeeeeeeeee

#profe-ssion	profession
1	engineer
2	student
3	teacher
346	01237

Table 1. Comparison between used objects in most type of models:

Objects by Codd	*Description*	*Objects by Chen*	*Objects by PER*	*Description*
Association	It presents a "m-n" connections	Relation	Link object	A logical connection between two or more type objects. Can be used in relation "n-m". Expresses by expression – book selling, student examination...
			Object process	Includes data for the process, which is modeling. Expresses by verb - sell, buy, order…
Core	Object that possess independent existing	Entity	Main object	Includes basic data for objects which directly or indirectly participates in the process. Expresses by noun – customers, books…
			Dependent object	Represents a detail data for the Primary object. Expresses by the name of the primary object and explanation.
Characterization	The most function of this object is to make a characterization to other object (core, characterization or association).	Attribute	Object for reference	Data with permanently character, which explains attributes of some objects.

Objects by Cod can be linked with theirs analogous in E/R model in this way: Core correspondences with "Entity" E/R, Characterization with "Attribute" E/R; Association with "Relation" E/R (just for relation from"m-n" type) [5].

CONCLUSIONS

The progress of base data models is represented. In the differentiated two courses of the semantic models the place of PER model is located.

REFERENCES

[1] Barker R., CASE*Method. Entity-Relationship Modeling. Copyright Oracle Corporation UK Limited, Addison-Wesley Publishing Co., 1990.

[2] Codd, E.F., Extending the Relational Database Model to Capture More Meaning. IBM Research Report RJ2599 (August 6th, 1979).

[3] Codd, E.F., Relational Database: A Practical Foundation for Productivity. IBM Research Report RJ3339 (December 21st, 1981. Republished in CACM 25(2), February 1982.

[4] Date C.J., The Birth of the Relational Model, Intelligent Enterprise, Volume 1, Number 1, October 1998

[5] Date C.J., When's an extension not an extension?, Intelligent Enterprise, June 1, 1999, Volume 2, Number 8

[6] Date, C.J., The Extended Relational Model RM/T. In C.J.Date, Relational Database Writings 1991-1994. Reading, Mass.: Addison-Wesley, 1995.

[7] http://orm.net/overview.html

[8] http://www.cetus-links.org/oo_ooa_ood_methods.html

[9] http://www.rational.com

[10] Pin-Shan Chen, The Entity-Relationship Model - Toward a Unified View of Data". ACM Transactions on Database Systems 1(1), March 1976. Republished in Michhael Stonebraker (ed.): Readings in Database Systems (2nd edition), San Mateo, Calif.: Morgan Kaufmann, 1994.

[11] Tujarov Hr., Tree Like Object Data Model. CompSysTech 2001, p.II12.1-II12.6

[12] Тужаров, Хр. Access - Разработка на приложения. ПИК, В. Търново, 2001

[13] Тужаров, Хр., Ст. Калчев. Информационни системи. ПИК, В. Търново, 2000.

ABOUT THE AUTHORS

Assoc. Prof. Hristo Tujarov, PhD, Department of Computer Systems and Technologies, University of V.Turnovo, Phone: +359 62 499 66, E-mail: Tujarov@uni-vt.bg

Eng. Ivelin Mihailov, PhD student, Department of Computer Systems and Technologies, University of Veliko Turnovo, Phone: +359 6122250, E-mail: ivelin@press.bg