VNU Journal of Science, Mathematics - Physics 23 (2007) 183-188

Approach to deductive database

Tuan DoTrung*, Tuan DuongAnh

Department of Mathematic, Mechanics, Informatics
College of Science, VNU, 334 Nguyen Trai, Hanoi, Vietnam

Received 15 November 2006; received in revised form 12 October 2007

Abstract. Researches on deductive database systems are not new ones, but propose of data model allowing manipulating data and knowledge on a particular framework is not easy but interesting goal. The paper aims at a model for knowledge in educational and training environment, beside of a presenting the achievement in the existing systems concerning knowledge. Certain data manipulation techniques for knowledge acquisition are proposed in the model, as knowledge discovering techniques in deductive database systems.

Keywords: Deductive database, data model, query language, datamining.

1. Introduction

Knowledge was studied in artificial intelligence from 1956, concerns to much research domains in some years. Human has reached key stones in data and knowledge manipulation. The third revolution of human society is attached to knowledge representation and manipulation. In [1, 2], there were some principal stones :

- Printing technique, 1440;
- Tacit knowledge, 1964;
- Knowledge society, 1973;
- The third wave, 1980;
- Information society, 1982;
- Internet, 1991;
- New economy, 1996.

Nonaka and Takeuk [2], 1964, summarize that the creation of knowledge is the result of a continuous cycle of 4 integrated processes. They are processes of externalization, internalization, combination, and socialization. These four knowledge conversion mechanisms are mutually complementary, but keeping interdependent that change according to the demands of context and sequence.

* Corresponding author. E-mail: tuandt@vnu.edu.vn

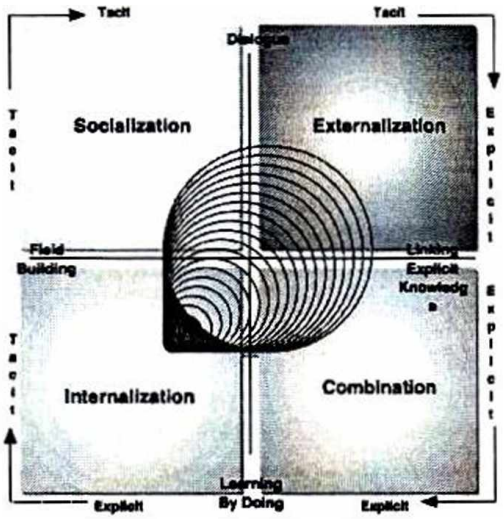

Fig. 1. Conversation of knowledge [11]

Knowledge activities focus to tacit knowledge, connect knowledge activity to knowledge conversion :

- Process of socialization allows to convert tacit knowledge to other tacit one. The process shares knowledge via practical activities;
- Process of internalization allows to convert explicit knowledge to tacit one. Here they have learning activities;
- Process of externalization allows to convert tacit knowledge to explicit one. It uses models, hypotheses;
- Process of combination allows to convert explicit knowledge to other explicit one. There are activities of combination, arrangement.

There are different directions of research; a lot of problems were solved. Techniques of knowledge representation are satisfied researchers. Kinds of knowledge are (i) declarative; (ii) procedural; (iii) meta; (iv) heuristics; and (v) structured. Some known techniques for knowledge representation are (i) attribute-object-value; (ii) rule; (iii) semantic net; (iv) frame; (v) dependency concept; (vi) logic... A lot of applied artificial systems, expert systems were referenced.

Concerning database systems, there are some data models : the hierarchical model, 1960; the network model, 1968; and the relational model, 1970. These models allows to describe and to manipulate data. Database management systems have functions permitting to design, to build database applications. After the year of 1990, advanced data models such as (i) distributed model; (ii) object oriented model; and (iii) deductive model, respond the need practical problems. Deductive model is for knowledge manipulation in database systems.

In the years of 80's, 90's, it existed proposals for linking database systems to artificial systems. A couple of expert systems and relational databases is known [3, 4, 5]. Each strict couple or loose couple allows to apply a mechanism focusing to knowledge activities.

The results on logic programming permit to prove theories automatically, to determine the relationship between facts and reasoning. The proving theory and model theory are used for knowledge manipulation. The proving theory uses the specification of reasoning process on premise; the model theory uses interpretability of clauses on facts. The Prolog language uses Horn clause and backward-chaining inference, bases on the semantic of the proving theory and model theory. Prolog

may present facts, inference, recursive function, query, updating integrity constraints by same way. The XBS [4] is known as successful one with the language HiLog that is developed on Prolog.

Database researches show resemble among the relational database and logic programming. Conversion predicate clauses to relational algebra clauses is not complicated. However the are problems (i) the effect ness when a big amount of knowledge is used; (ii) query languages in relational model can not inference. Besides the relational language is week for exhibiting all application data; it demands traditional programming language. Some deductive model focused simultaneously on Prolog language and relational data manipulation functions. Datalog is a restricted form of the Prolog language, accepting the negation, is used in deductive models in the years of 90's.

From the users part in deductive models, there are problems concerning the user interface [6]:

- deductive database uses flat structure which does not be displayed, does not support complex value data in new database applications;
- the integration of objects to database for modeling possibility increasing is net easy;
- deductive model needs the unified way for presenting data schema and data higher levels;
- updating in logic database.

For solving these problems, there are some research directions [6] :

1. Deductive language with complex valued data. The researches cited are data models with embbeded, complex valued data on relationship. It uses the first normal form relations; therefore Datalog can not present embedded n_tuple, sets. Using functions is solution for that problem. Datalog is developed; they achieved certain language, such as (i) LDL; (ii) COL: (iii) HiLog; and (iv) Relationalog;

2. Object oriented deductive language. The object oriented approach intends (i) objects; (ii) complex valued data; (iii) methods; (iv) classes; (v) heritage; (vi) encapsulation. Languages in this approach are (i) O-logic; (ii) F-logic; (iii) ROL; and (iv) IQL;

3. Data schema and data higher levels. In the second generation of database management systems, data schema permits to describe database structure, integrity constraints. But deductive model has not yet described data schema clearly. For reasoning on data schema. It is preferred to use schema variable, logic which is higher than the first order predicate logic. Languages permiting to manipulate data schema are : (i) HiLog; (ii) L^2; (iii) F-logic; and (iv) ROL;

4. Knowledge updating. The important aspects in database systems are (i) updating extension database and intention database; (ii) updating un-defined data; (iii) updating sets; and (iv) updating according to conditions.

2. Architecture of the proposed deductive database

2.1. The first idea

Some database systems use integrity constraints for introducing knowledge. Integrity constraint management leads to information management in database systems. Using data dictionary that composed of meta-data for database design is a kind of knowledge manipulation in database systems. The systems in [7, 8, 9, 10] suggest an idea for developing database systems on knowledge manipulation. Besides when manipulating data, describing the relationship among data and datamining [11, 6, 12], users can find new relationship in database. The relationship keeps space-temporal information.

2.2. The private contribution

The proposed deductive model has not yet detached from the relational model and artificial systems. However it is necessary to build a model that separates the relational model with the second generation database management systems.

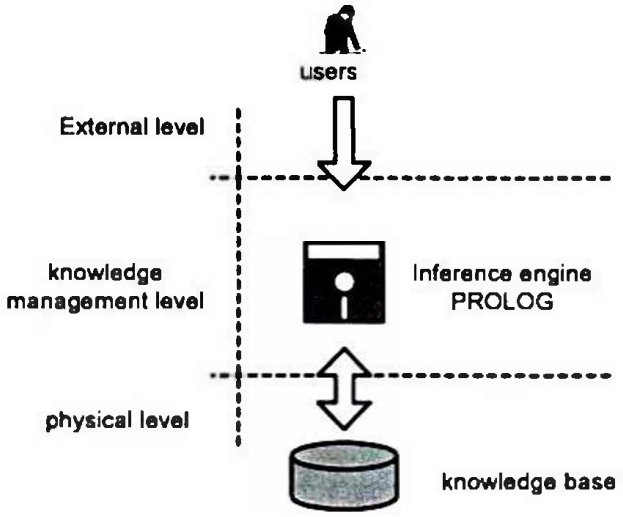

Fig. 2. Conceptual levels for the proposed deductive model.

Conceptual levels of the deductive model are as followings :

1. External level. The level permits to describe the system requirements, to query to deductive database. Action for inputting facts and knowledge is described here. It needs the end-user interactive interface to convert natural language clauses of users to formal form;

2. Knowledge management level. The level permits to manage data and to inference on data and knowledge. The variant of Prolog inference engine is feasible. Depending on the scale of the deductive model, the engine has to modified about (i) negation aspect; (ii) higher level logic; (iii) interactive interface...;

3. Physical level. The level permit to organize facts, rules. Knowledge under the form of rules is appropriate to the Prolog engine in the second level. Other kinds of knowledge representation may be transferred to rule representation.

2.3. Modules in system architecture

In the architecture of deductive systems below, each module is correspond to system function. The role of these modules is explained, for meeting the needs of deductive systems :

1. Language in user interactive interface. The language permits to describe events, facts and knowledge. In the interface, users can choose data structure, present integrity constraints. The user language does not obligate Prolog language. Demand and response in the interface may be examined by pre-processing. Then the user query is transferred to either the inference engine, or data management;

2. Data management. The management assures data retrieve and data manipulation in according to user query on data/ knowledge. The module needs data dictionary, application software in designing a database;

3. Database. Database is organized as knowledge base physically. The separation of data, events and knowledge is in the data management and the knowledge management;

4. Datamining methods. The methods permit to discover rules among data. The relationship between data is meta data, belongs to knowledge base.

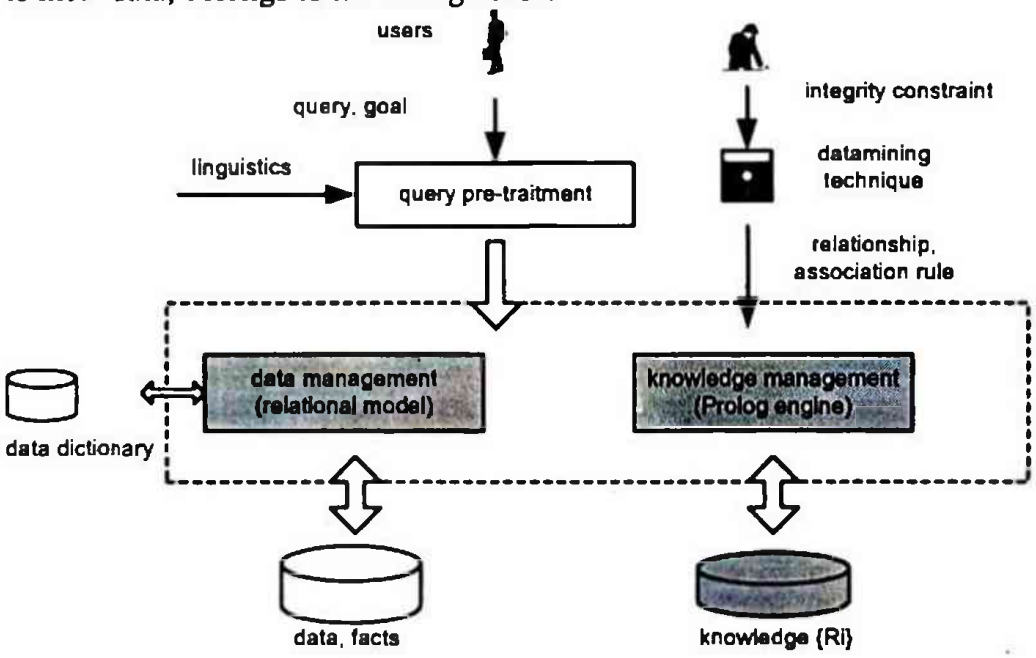

Fig. 3. Architecture of deductive systems.

Furthermore, the architecture has formal specification. Syntax of user language has to been described carefully. These aspects are examined in next step.

3. Conclusion remark

In the proposed data model for knowledge and reasoning, logic and the Prolog inference engine are focused. Aspect on data/ knowledge manipulation mechanism is rather general, depending on possibility of Prolog. The proposed architecture combines Prolog engine and integrity constraint processing for metabase. Metabase is composed of database and knowledge base.

The systems need to space-temporal aspect on data, datamining techniques. These aspects supply knowledge on the systems.

Acknowledgements. This paper is based on the talk given at the Conference on Mathematics, Mechanics, and Informatics, Hanoi, 7/10/2006, on the occasion of 50th Anniversary of Department of Mathematics, Mechanics and Informatics, Vietnam National University, Hanoi.

References

[1] Maja van der Velden, Knowledge facts, knowledge fiction, *Journal of International Development*, J. Wiley & son, (2002) 15.

[2] www.nwlink.com/~donclark/knowledge/knowledge.html, 2006

[3] B. Yang et al., Deductive synthesis of workflows for e-science, *IEEE inter. Symposium on cluster computing and the grid* (2005) 168.

[4] K. Sagonas, T. Swift, D.S. Warren, XSB as an efficient deductive database engine, *Proc. Of SIGMOD* (1994) 442.

[5] M. Liu, Deductive database languages : problems and solutions, *ACM computing surveys* vol. 31, (1999) 27.

[6] D. Botzer, O. Etzion, Self-tuning of relationships among rule's components in active databases systems, *IEEE tran. On knowledge and data engineering*, vol. 16 (2004) 375.

[7] C. Zaniolo, H. Wang, Logic-based user-defined aggregates for the next generation of database systems, in "*The logic programming paradigm: a 25 year perspective*", Ed. Springer.

[8] C. Zaniolo, Mining databases and data streams with query languages and rules, *IEEE tran.* (2005) 14.

[9] Y. Law, H. Wang, C. Zaniolo, Query languages and data models for database sequences and data streams, *Proc. of the 30th VLDB conference* (2004) 12.

[10] M. Liu, Design and implementation of the ROL system, *Journal of intelligent information systems*, N. 14 (2000) 1.

[11] C. Lee, M. Chen, Progressive partition miner: an efficient algorithm for mining general temporal association rules, *IEEE tran. On knowledge and data engineering*, vol. 15 (2003) 1004.

[12] S. Flesca, S. Greco, Declarative semantics for active rules, *IEEE tran.* (2003) 27.