# DEDUCTIVE MODEL FOR ACTIVE DATABASE SYSTEMS

**Do TrungTuan**

*College of Science, Vietnam National University, Hanoi*

**Nguyen Thi ThanhDuyen**

*Graduated Student, Vietnam National University, Hanoi*

**Abstract.** Researches on deductive database systems are not new ones, but propose of data model allowing to manipulate 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. An active database is proper for applying such technique on rule-based knowledge.

Index Terms—Rule, active database, deductive, datamining

## 1. Introduction

A rtificial intelligence has knowledge as an importance object. Understanding and using knowledge is obtained goal of human society and scientific research. Since 1964, tacit knowledge and other kind of knowledge is examined in the relationship to tacit knowledge. In the research on database system, knowledge is represented in systems. Among proposed different kinds of knowledge, rules are preferred.

Beginning with file management systems in 1960, a lot of application systems based on hierarchical model, network model (1960, 1968), relational model (1970) have remarkable role in the economy society. After 1990, advanced model, such as distributed model, object oriented model and deductive model has studied. The systems based on these advanced models did not distinguished from ones of the second generation of database management systems.

In the years 80's, some works approached knowledge manipulation in information systems. A solution for lose gap or strict gap between artificial intelligence and database theory has some results for knowledge manipulation [5, 11].

Researches on logic programming allows to prove theorems automatically, to set a relationship between events and deduction by (i) proving theory; and (ii) model theory. The proving theory presents specifications of true reasoning after premises the model theory presents assertions after events. Prolog language uses Horn clauses and backward chaining inference with semantics of both the proving theory and the model theory. This language supplies capacities of (i) event representation; (ii) inference; (iii) recursive reasoning; (iv) updating integrity constraints. The system XBS [7] is a success with the HiLog, a development on Prolog.

At the point of database research, there is a similar of relational database systems and logic programming. Transferring a predicate clause to a relational clause is not difficult. However it exists problems, such as (i) effectiveness of organizing a great amount of knowledge; (ii) reasoning capacity of query language in the relational model. The relational query language can not exhibit whole application; sometime it needs application programs. Some deductive models focus on Prolog and technique for relational data manipulation. Datalog is a restrict Prolog, using negation and is a approach in deductive model of year 90's.

Concerning to the aspect of end-user interface in deductive databases, it exists problems [5] (i) deductive databases use flat structure that is not appropriate to exhibit complex objects; (ii) capacity of modeling objects; (iii) data schema at the user level; (iv) updating knowledge. In analyzing these problems, there are solutions as follows :

1. Deductive language for complex objects. Datalog is developed in order to specify complex data structure, nested data. It obtains (i) LDL; (ii) COL; (iii) HiLog; and (iv) Relationalog;
2. Object oriented deductive language. This approach examines (i) objects; (ii) complex data; (iii) methods; (iv) class; (v) heritage; and (vi) encapsulate. Some results are (i) O-logic; (ii) F-logic; (iii) ROL; and (iv) IQL;
3. Data schema at user level. The second generation of database management systems with relational data model has not been clear on schema specification when users want to introduce data structure, integrity constraints. The first order predicate logic and higher tools may be used for solving the problem. The language allowing to specify user schema (i) HiLog; (ii) L2; (iii) F-logic; and (iv) ROL;
4. Updating knowledge. Database applications need an interactive interface for updating meta data. Update objects are (i) extensive databases and intention databases; (ii) undefined data; (iii) collection, set; and (iv) update after conditions.

## 2. Architecture of deductive model with active databases

### *Related work*

Some systems focus on integrity constraints in database management systems for manipulate knowledge in database. Meta data in the data dictionary composed of design rules are a kind of knowledge in database. The systems in [6, 7] are prototype for manipulating knowledge on in database. Besides meta data, the relationship among data let the systems to discover association rule [1, 2, 4]. Information about the data relationship are accumulated as knowledge.

Active databases [8, 9, 10] use mechanisms of (i) event; (ii) condition; and (iii) actions to introduce integrity constraints, policies under the form of knowledge. The system arrange a detector and a trigger for realizing actions when the conditions are satisfied.

## *Proposal for deductive model with an active database*

Data model for deductive databases has not yet separated from relational model and artificial systems. Throughout, it is necessary to construct an independent deductive model.

The model consists of :

1. External level allows to specify system requirements, to query to deductive database (goal). The process of input data and knowledge is realized in this level. In fact, it needs interactive interface for transferring the user demands to appropriate query to the (data/ knowledge) management systems;
2. Management level composes of data management and knowledge management. A approach using Prolog inference engine is good; whatever it needs a correspondent development to scale of deductive model, with negation and higher level specification...;
3. Physical level is correspondent to organization of events and rules. Rule based knowledge is proper within inference engine in management level. Some other kinds of knowledge are transferred to rules.

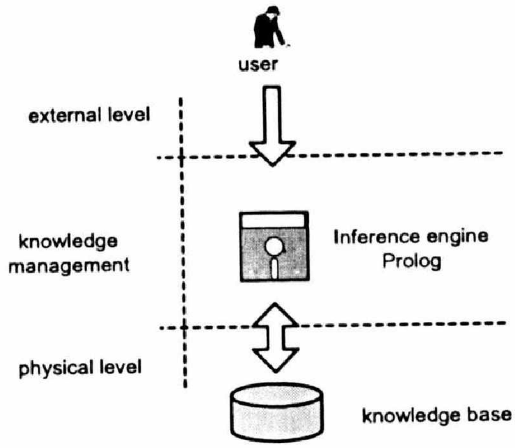


*Figure 1.* Deductive model with 3 levels

## *Modules in the architecture*

Followings are some modules in the architecture of deductive model in active database systems.

1. Query module. It allows to describe events, data and knowledge. In the end users interface, data structures and knowledge, integrity constraints may be introduced;
2. Data management module. This module supplies tools for searching and manipulating data via query language. Query concerning knowledge is transferred to predicates in Prolog. Data management needs a data dictionary, software package for design purposes;
3. Query language module. The existence of this module is for knowledge specification. User demands for reasoning on event, knowledge did not yet exist in relational language. Therefore, it is examined by this module, then transferred to a Prolog goal. Matching relational algebra to logic predicates is proved;
4. Metabase. Database is traditional one. Knowledge base is organized physically as database. Whatever a unique metabase for both is preferred. At the level of management, data and knowledge are distinguished;
5. Knowledge discovering module. In this module, datamining algorithms, association rule discovering methods... are proposed. With the help of event-condition-action, new relationship among data is added to knowledge base.

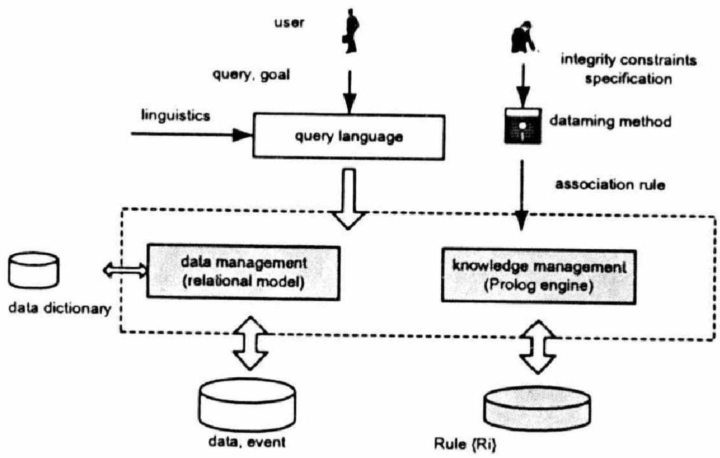


*Figure 2.* Architecture of the deductive model with active databases

Besides modules, it is necessary to implement formal specification, complete notation, syntax and semantic aspects specification. A box for event-condition-action may be either (i) an external application; or (ii) a function embedded in database management system.

## 3. Conclusions

The paper presents the principal modules of the architecture of deductive model in active databases. The database management system is relational one the knowledge manipulation based on Prolog inference engine.

Knowledge is in integrity constraints, association rules among data, events at external level. In order to acquisition knowledge, datamining algorithms, association rule discovering are applied.

## References


1. Botzer D., O. Etzion, Self-tuning of relationships among rule's components in active databases systems, *IEEE tran*. On knowledge and data engineering, V. 16, N.**3**(2004), p. 375- 379.
2. Flesca S., S. Greco, *Declarative semantics for active rules*, 2003, 27 p.
3. Law Y., H. Wang, C. Zaniolo, Query languages and data models for database sequences and data streams, *Proc. of the 30th VLDB conference,* 2004, 12 p.
4. Lee C., M. Chen, Progressive partition miner : an efficient algorithm for mining general temporal association rules, *IEEE tran*. On knowledge and data engineering, V. 15, N. **4**(2003), 1004- 1017.
5. Liu M., Deductive database languages: problems and solutions, *ACM computing surveys*, V. 31, N.**1**(1999), p. 27-59.
6. Liu M., Design and implementation of the ROL system, *Journal of intelligent information systems*, N.**14**(2000), p.1-21.
7. Sagonas K., T. Swift, D. S. Warren, XSB as an efficient deductive database engine, *Proc. Of SIGMOD*, 1994, p. 442- 453.
8. Sudermier A., Dietrich Suzannne W., Shah V., *An active database approach to integrating black-box software components*, Arizona state University, 1998.
9. Urban S. D. et al., An evaluation of distributed computing options for a rule-based approach to black-box software component integration, *Proc. of 3rd int. workshop advanced issues of e-commerce and web-based information systems* (WECWIS'01), 2001.
10. Urban S. D. et al., Delta Abstractions: a technique for managing database states in runtime debugging of active database rules, *IEEE tran.* On knowledge and data engineering, V. 15, N.**3**(2003), p. 597- 612.
11. Yang B. et al., Deductive synthesis of workflows for e-science, *IEEE inter. Symposium on cluster computing and the grid*, 2005, p. 168- 175.
12. Zaniolo C., 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*, p. 401-426
13. Zaniolo C., *Mining databases and data streams with query languages and rules*, 14 p., 2005.