Proceedings of KFIS Fall Conference 2004 Volume 14, Number 2

동적지식도와 데이터베이스관리시스템 기반의 전문가시스템 개발

Development of Expert Systems based on Dynamic Knowledge Map and DBMS

김 진 성
전주대학교 경영학부

Jin Sung Kim

School of Business Administration
Jeonju University, Jeonju, Jeonbuk, 560-759, Korea
E-mail: kimjs@jj.ac.kr

Abstract

In this study, we propose an efficient expert system (ES) construction mechanism by using dynamic knowledge map (DKM) and database management systems (DBMS). Generally, traditional ES and ES developing tools has some limitations such as, 1) a lot of time to extend the knowledge base (KB), 2) too difficult to change the inference path, 3) inflexible use of inference functions and operators. First, to overcome these limitations, we use DKM in extracting the complex relationships and causal rules from human expert and other knowledge resources. Then, relation database (RDB) and its management systems will help to transform the relationships from diagram to relational table. Therefore, our mechanism can help the ES or KBS (Knowledge-Based Systems) developers in several ways efficiently. In the experiment section, we used medical data to show the efficiency of our mechanism. Experimental results with various disease show that the mechanism is superior in terms of extension ability and flexible inference.

Keywords: Database management systems, Dynamic knowledge map, Expert systems, Knowledge base, Knowledge-based systems, Relational database.

1. Introduction

Recently, directors and managers in organizations started seriously discussing the knowledge asset of organizations. Therefore, system designers are trying to develop a knowledge-based ES (Expert Systems) or KBS (Knowledge-Based Systems) to store and reuse the expert's knowledge efficiently. In other areas, there was a growing effort to develop ES ontology or conceptualization mechanism that will help to clarify the area of applied knowledge (Gordon, 2000). In this study, we focus on the development of ES by using efficient knowledge representation and inference tools. From this point of view, the majority of S/W tools for building expert systems seem to fall into four broad categories (Jackson, 1999):

(1) *Expert system shell*, which are essentially abstractions over one or more applications programs.
(2) *High-level programming languages*, which to some extent conceal their implementation details, thereby freeing the programmer from low-level considerations of efficiency in the storage, access and manipulation of data.
(3) *Multiple-paradigm programming environments*, which provide a set of S/W modules that allow the user to mix a number of different styles of artificial intelligence programming.
(4) *Additional modules* for performing specific tasks within a problem solving architecture.

Unfortunately, however, 'traditional ES and KBS construction mechanisms have several problems. First, traditional KBS were non-applicable because of the conversion form tacit knowledge to explicit documented knowledge was very difficult. Second, it is often difficult to extend and enhance a KBS with additional expert knowledge once the system is fielded. Third, within the context of rapidly changing technologies and processes, an existing KBS might no longer seem capable of meeting the increasingly complex knowledge demands in the industry (Woo et al., 2004).

In this study, we tried to overcome most of the above mentioned pitfalls. To this purpose, we use the dynamic knowledge map (DKM) and relational database (RDB) metaphor and its management systems (DBMS). First, DKM could help an expert, who wants to transform his tacit knowledge into explicit knowledge. Then, DBMS could assist a knowledge base manager to link distributed knowledge with causal relationships. Third, DBMS could help the ES managers to develop an efficient inference engine based on OLAP (Online Analytical Processing) concept. Therefore, the developed mechanism enables an interactive navigation and inference by using DKM, RDB, and DBMS.

2. Methodology

KBS, ES, and artificial intelligence (AI) mechanisms have made an important contribution to our understanding of expert knowledge. Especially, many researchers in ES field tried to develop a rigorous representation for expert knowledge so that the knowledge could be brought to life in a computer program (Shortliffe, 1976).

There are several accepted methods of knowledge representation that have been devised for AI-type applications. Some of these are also suitable for use and interpretation by humans and can form a bride between human knowledge and machine knowledge (Gordon, 2000). As one of useful methods of knowledge representation, in this study, we use dynamic knowledge map (DKM). DKM was originated from KM. KM is the name given (McCagg & Dansereau, 1991) to a type of mental diagram by means of which complex ideas can be easily and quickly set out in a logical order. KMs typically point to people as well as to documents and databases to enable a person to find an appropriate knowledge source (Devenport & Prusak, 1998). Conventional KMs locate the holders of knowledge when their expertise is needed rather than spending time with imperfect solutions or searching for explicitly documented knowledge. KMs are a graphic representation of the connections made by the brain in the process of understanding facts about something. They are built starting with the attribute that defines the problem to then develop a graphical diagram that sets out on paper the manner in which the mind comes up with ideas in the process of understanding (Gómez et al., 2000). However the static nature of most KMs is an obstacle to disseminating tacit knowledge. More recently, the role of knowledge mapping has been changed to expert locater, which allows users to search through a set of biographies for an expert on a particular knowledge domain (Devenport & Prusak, 1998). To overcome these limitations, Devenport and Prusak (1998) proposed the basis for dynamic knowledge map. However they didn't suggested technical and graphical representation of DKM.

In contrast with Davenport and Prusak's DKM (1998) this study propose a technical and graphically manageable DKM construction mechanism. To this purpose, we referred the details of Gómez et al.'s (2000) KM construction mechanism. A general-purpose 6-phased procedure for outputting a KM during the knowledge conceptualization process is given below (Gómez et al., 2000).

Phase 1: Identify the main goal of the system. Generally, the purpose of the KBS or ES is to make a decision on a concept and, more particularly, on a property or attribute of that concept, which we have termed the (main) goal property. Therefore, the above main goal should have already been decided, as it is essential for drawing up the KM. The attribute or goal property in a medical diagnostics system, for example, would be the disease suffered by the patient and a prescription presented by doctor.
Phase 2: Design the goal decision block. To extract a graphical representation of KM, in this phase, draw a rectangle around the property, specifying to which concept it belongs, using the property/concept form, and the possible values of that property. In the example of the medical diagnostics systems, the possible values would be the names of the diseases that are to be diagnosed by the system. Figure 1 shows an example of how to represent the properties in the KM.

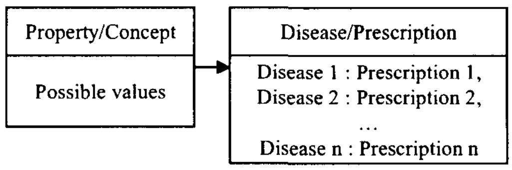

Figure 1 Properties of KM

Phase 3: Add the properties for inferring or calculating the goal decision. After the design of goal decision block, place the properties inside boxes around the goal decision. The relation with the goal property is expressed by means of an arrow that will start from the property used to infer or calculate the goal decision. The number of values of the source property of the arrow is specified on this arrow. If the number of values is 1, no specification is required. Each attribute around the goal decision must be involved in inferring the value of this decision. However, there is no need to infer the all attributes at the same time. Because of most human expert could deal with only a small number of attributes simultaneously. If a lot of attributes are used to infer the value of the goal decision, domain expert will probably calculate the value of some intermediate attributes in order to infer the goal decision. Then, the number of attributes around the goal decision could be reduced by introducing intermediate attributes.

Phase 4: Extend the KM. If there were more additional information to infer the decision goal, the properties used to infer the values of each property have to be added.

Phase 5: Repeat phase 4 until none of the *peripheral* properties are inferred, that is, they are taken from external sources such as user input, sensors, files, database, and other external or internal changes.

Phase 6: Check the knowledge reflected in the KM. These checks come under two categories. First, checks related to the validation of the knowledge reflected in the KM with domain expert. Second, checks related to the verification of the KM against the static and dynamic models generated during the synthesis stage.

To combine the KM with RDB, we extended the Gómez et al.'s (2000) process to 9-phased process. Then, we called this process as dynamic knowledge map (DKM) construction process. Additional processes for DKM are given below.

Phase 7: Frame-based RDB table construction. After the check for KM, transformed the KM into frame-based RDB table forms.

Phase 8: Add the inference rules into frame-based RDB table. This phase is critical difference with Gómez et al.'s (2000) KM construction. To infer the properties or find goal decision, we added inference rules into the frame-based RDB table.

Phase 9: Relate the RDB tables. To confirm the relationships among each node in KM, connect the RDB tables with RDB relationship facilities. Figure 2 shows an example of how to represent the properties in our proposed DKM.

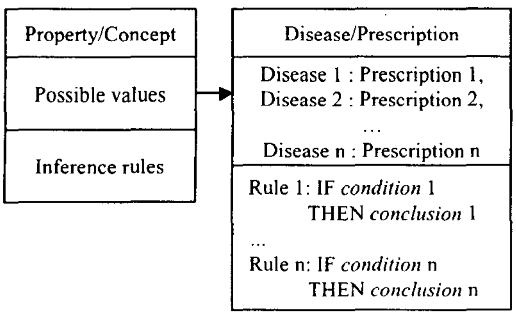

Figure 2 Properties of DKM

3. Example of DKM construction

To validate the performance of our proposed mechanism, in this section, we proposed a practical application. The example is a part of a real medical expert's knowledge and illustrates how the DKM is drawn up from the static and dynamic models. Figure 3 shows the example of our proposed DKM. In contrast to Gómez et al.'s (2000) KM construction, our proposed mechanism has several advantages.

First, each RDB table has its inference rules. Therefore, there is no need to construct a huge rule base independently.

Second, it is very easy to revise and extend the KB through the graphical user interface supposed by RDB management systems (RDBMS).

Third, on the basis of our proposed mechanism, ES has no need to have special inference engine. Because of every inference is performed by each knowledge module respectively.

Fourth, there are no conflicts among inference rules. Because of every DKM nodes possess his own inference rules, and its' decision depends on his own properties.

Fifth, DKM node has several inference rules within his block. Contrary to Gómez et al.'s (2000) KM, therefore, our mechanism could handle the multiple choices and inference rules.

4. Conclusion

In this research, we extended traditional KM construction process and combined these two different knowledge representation tools as a dynamic knowledge map (DKM). The method could support the organizations in several ways:

- It can improve the efficiency of knowledge inference and its application
- It is applicable to real world decision, because of the conversion form tacit knowledge to explicit documented knowledge is very easy.
- It is easy to extend and enhance a knowledge-based system with additional expert knowledge once the system is fielded.

The method also has advantages for the individual and for organizations specializing in education:

- It allows an individual to see and understand a conceptualization process of knowledge and its applications.
- It can be easily applied to the educational field such as ES development or decision support systems (DSS) construction.
- It will identify appropriate directions for the use of knowledge management systems.

Further research should be conducted in order to test the suitability of DKM in real-world application. First a Web application should be constructed to support the knowledge collection and management on the Web site. Second, a set of real-world experiments will prove the efficiency and robustness of DKM.

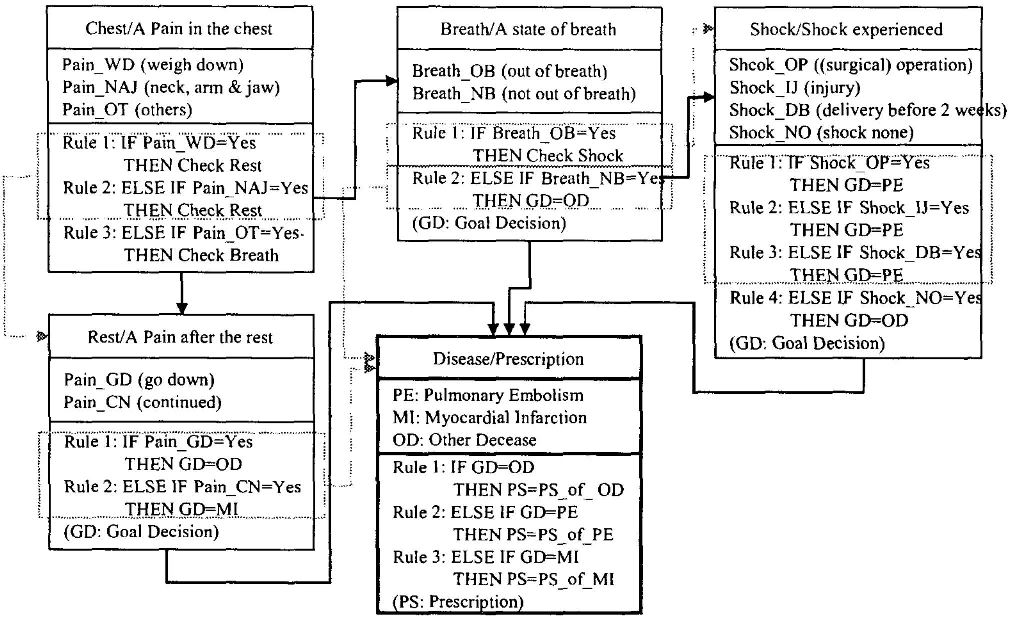

Figure 3 Example of DKM (sub-problems)

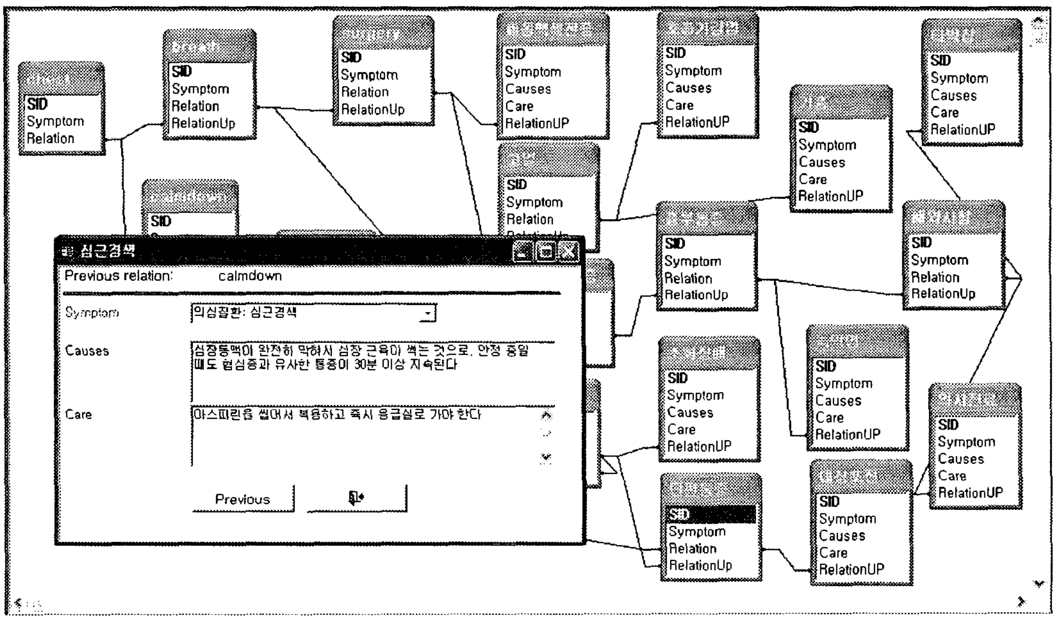

Figure 4 Another representation of DKM and inference window

References

[1] Beier, J. and Tesche, T. (2001), Navigation and interaction in medical knowledge spaces using topic maps, *International Congress Series*, 1230, 384-388.

[2] Davenport, T. and Prusak, L. (1988), *Working knowledge: How organizations manage what they know*, Harvard Business School Press, Boston.

[3] Gómez, A., Moreno, A., Pazos, J., and Sierra-Alonso, A. (2000), Knowledge maps: An essential technique for conceptualization, *Data & Knowledge Engineering*, 33, 169-190.

[4] Jackson, P. (1999), *Introduction to Expert Systems* (3rd edition), Addison Wesley Longman Limited.

[5] McCagg, E.C. and Dansereau, D.F., A convergence paradigm for examining knowledge mapping as a learning strategy, *Journal of Educational Research*, 84(6).

[6] Polanyi, M. (1966), *The tacit dimension*, Dobuleday, Garden City, 1966.

[7] Shortliffe, E.H., *Computer based medical consultations: MYCIN*, Elsevier, New York, 1976.

[8] Woo, J., Clayton, M.J., Johnson, R.E., Flores, B.E., and Ellis, C. (2004), Dynamic knowledge map: reusing experts' tacit knowledge in the AEC industry, *Automation in Construction*, 13, 203-207.