

A knowledge Conversion Tool for Expert Systems

Jin S. Kim¹

School of Business Administration, Jeonju University
303 Cheonjam-Ro, Wansan-Gu, Jeonju, Jeonbuk 560-759, South Korea

Abstract

Most of expert systems use the text-oriented knowledge bases. However, knowledge management using the knowledge bases is considered as a huge burden to the knowledge workers because it includes some troublesome works. It includes chasing and/or checking activities on *Consistency, Redundancy, Circulation, and Refinement* of the knowledge. In those cases, we consider that they could reduce the burdens by using relational database management systems-based knowledge management infrastructure and convert the knowledge into one of easy forms human can understand. Furthermore they could concentrate on the knowledge itself with the support of the systems. To meet the expectations, in this study, we have tried to develop a general-purposed knowledge conversion tool for expert systems. Especially, this study is focused on the knowledge conversions among text-oriented knowledge base, relational database knowledge base, and decision tree.

Key Words: Decision tree, Expert system, Inference, Intelligent system, Knowledge base, Relational database.

1. Introduction

The two core components in developing an expert system (ES) are the inference engine (IE)/mechanism and knowledge base (KB) construction. Where, the inference mechanism was the fundamental technology to improve the total performance of the ES. Therefore, various inference mechanisms were proposed by the researchers after the proposition of the MYCIN [3].

However, the speed of inference of IE and efficient management of the KB were still remained as a tackling point of ES development [1, 2]. In addition, the acquisitions and managing of knowledge were regarded as a huge burden to the ES managers and knowledge workers because the processes include various complex routines. The routines include *Checking of Consistency / Redundancy / Unlimited Circulation of the inference (rules), Refinement of the rules, Conversions of the rules*, and etc. To reduce the burdens, this study is focused on the *Conversions of the rules* and the *Checking of Unlimited Circulation of the Inference Rules*.

Since the ES have been widely used in the domains where mathematical models were could not be easily built [1, 2], human expert's knowledge was stored frequently in the KB as a form of inference rule. Then the rules and knowledge were frequently expressed by the OAV (Object-Attribute-Value) typed IF-THEN rule. However, the human expert's knowledge was sometimes conflict, redundant, and linked in an unlimited circulation.

To minimize the errors may come from the human expert knowledge this study proposes a knowledge conversion tool to transform the traditional *IF-THEN rules* into a *Decision Tree* (DT). By using the DT, ES managers may reduce their burdens

in managing KB. At the first step, the relational database management system (RDBMS) was used to construct KB and transform/manage it. The advantages of using RDBMS-based KB are could be summarized as follows:

- In anytime and anywhere, by using the communication network, the ES managers or knowledge workers can share the knowledge stored in the RDBMS-based KB [4].
- Knowledge management in RDBMS is more efficient than the text-based such works. ES managers can execute SQL-based various knowledge management routines such as *Create, Search, Retrieve, Update, Sort, Refinement, View/Group by using the Keywords or Logical Conditions and etc.* with the RDBMS-based KB [2].

The remainder of this paper is organized as follow: The research methodology was proposed in Section 2. The prototype system and the results of implementation were presented in Section 3. The conclusions and future works are finally given in Section 4.

2. Research Methodology

2.1 Knowledge Elicitation

To construct the KB, a knowledge elicitation routine is required. The routine is commonly regarded as a major obstacle in the process of ES development. Generally, the knowledge used in a KB construction could be acquired through one of two ways: either *Manual* or *Automatic acquisition* [5].

Traditionally, knowledge workers gathered the knowledge through the interactions with the human experts. However, the method was turn out to be a bottleneck, causing delays in the ES development [7].

One of the efficient knowledge acquisition and/or extraction mechanisms to overcome the limitations, Data mining (DM) was introduced. The DM is motivated by the need of new

techniques to help analyzing, understanding or visualizing the huge amount of data gathered from business and scientific applications [6]. This study is based on those two knowledge acquisition routes.

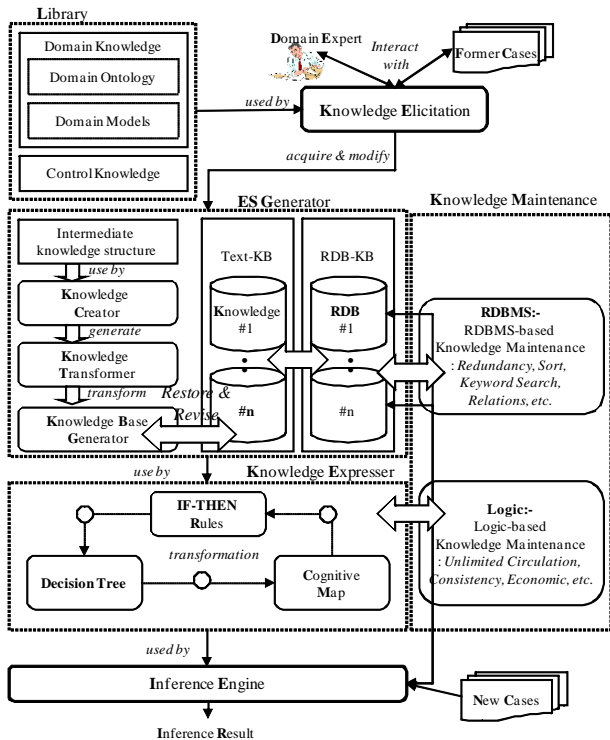


Fig. 1 Research methodology

Entire processes of our proposed methodology were graphically presented in Fig. 1. It includes six main components namely: *Knowledge Elicitation*, *Library*, *ES Generator*, *Knowledge Expresser*, *Inference Engine*, and *Knowledge Maintenance*. The former version of the methodology was proposed by Kim [1] and Rafea et al. [4]. However, the Fig. 1 shows the expanded/revised components and architecture. The detailed activities of the components are as follows:

- *Library*: Library has both reusable domain knowledge and control knowledge such as domain ontology, domain models, and control knowledge. It is used to create a new knowledge and refine the old KB with it.
- *Knowledge Elicitation*: The main functions of knowledge elicitor are to *create*, *maintain*, and *restore* knowledge elicited from the external input, *fetch* the relevant knowledge components from the library, and *transform* the knowledge into an appropriate knowledge structure. Especially, the detailed maintenance is executed in the *Knowledge Maintenance* component.
- *ES Generator*: Automatically, it generates an executable knowledge, which corresponds to the intermediate knowledge generated above. It contains knowledge creator, knowledge transformer, and knowledge base generator. During the knowledge conversion, ES Generator uses the RDBMS to restore and revise her KBs. The

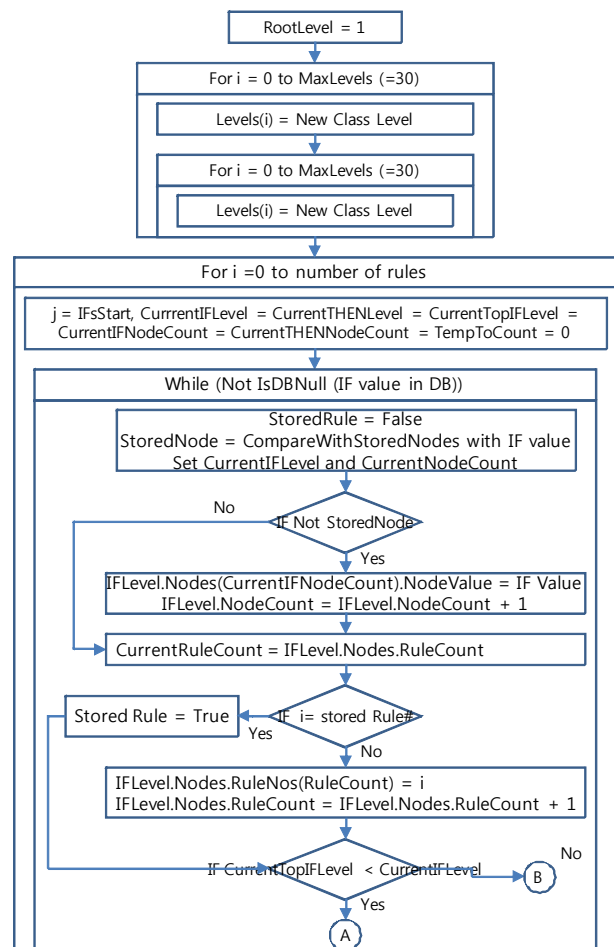
text-oriented knowledge was stored in Text-KB and RDB-oriented transformed knowledge was stored in RDB-KB.

- *Knowledge Expresser*: It supports the three knowledge expression methods such as, IF-THEN Rules, DT and Cognitive Map. It also could help end-users to understand the structure of knowledge.
- *Knowledge Maintenance*: It has two main procedures RDBMS-based knowledge maintenance and Logic-based knowledge maintenance. RDBMS-based knowledge maintenance supports the DB-oriented maintenance activities such as Duplication Check, Sort, Keyword Search, Checking Relationships and etc. Then the Logic-based knowledge maintenance support the activities: Checking of Unlimited Circulation, Consistency, Economics of the Rules and etc.

The Fig. 1 shows all the components used in our proposed research methodology.

2.2 Knowledge Conversion

This study supports three knowledge conversion and/or transformation methods. In the Fig. 2, we can see the procedures used in the conversions from IF-THEN rules to DT. The detailed knowledge inference mechanism and other procedures were introduced in Kim's study [2].



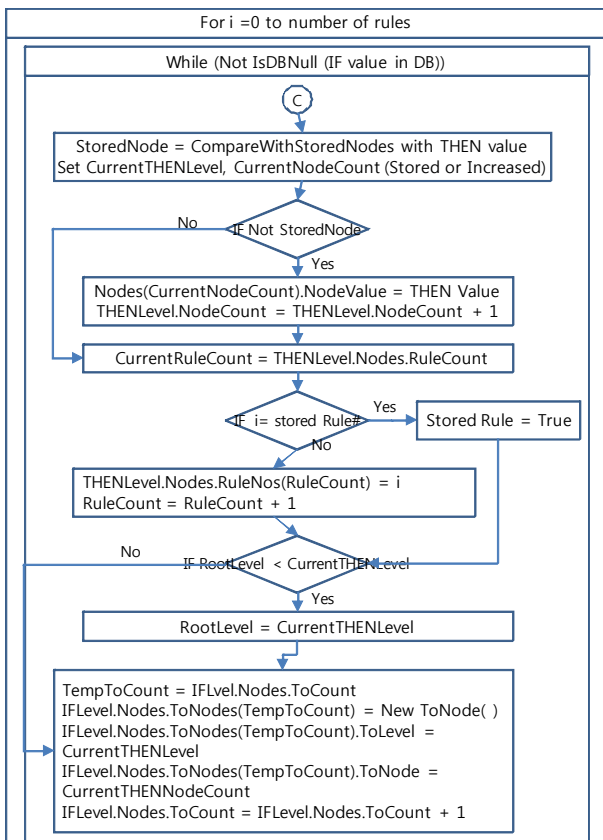
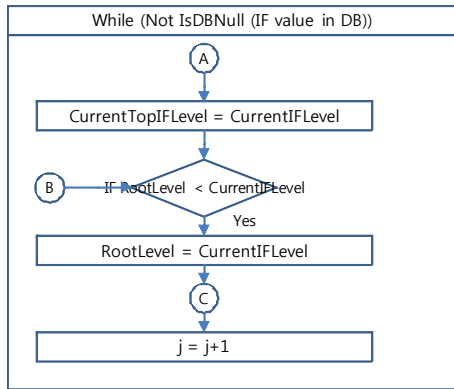


Fig. 2 Conversions from IF-THEN rules to DT

2.3 Checking of Unlimited Circulation of Inference Rules

Through the *Knowledge Conversion* process all the inference rules were stratified. Using the DT, after the process, we can check the unlimited circulation of the rules. To find out the circulation, we just need to check the Level number and Node numbers of every rule. Table 1 shows the pseudo code for the mechanism.

The checking mechanism is so simple because the previous procedures offered important information required in checking circulation.

Table 1 Pseudo code for the checking of circulation

```

For i=0 to Root/Top Level
    Current Level = i
    For j = 0 To Node count in Levels i
        For k = 0 To Count of connection
            If Level of target node < Current level Then
                Circulation = True
            End If
        Next k
    Next j
Next i
If Not Circulation Then There's no circulation
End
    
```

3. Implementation and Results

3.1 Open Knowledge Base

To implement our proposed mechanism, we developed a prototype ES development tool K-Expert (ver. 1.0). The tool was developed by using the Visual Studio 2010 in a Windows 7 environment. The main KBs were constructed by MS-Access database. The first sample data set used in the implementation is the *Classification of Animals*. The Fig. 3 shows the main window of the K-Expert and process of Open knowledge base.

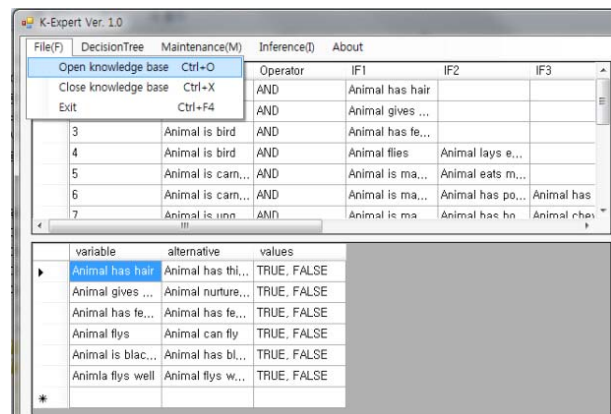


Fig. 3 The knowledge base stored in a MS-Access DB table

The above table shows the inference rules and variables, and the below table contains the names of the variable and the alternative questions for the inference rules shown in the above table. The alternative questions and values can replace the variables when the rule is executed.

3.2 Knowledge Conversion

In the text box at the bottom of Fig. 4, we can see the information of Decision Tree (level #, Node#, Rule#, and Target (to) Node#).

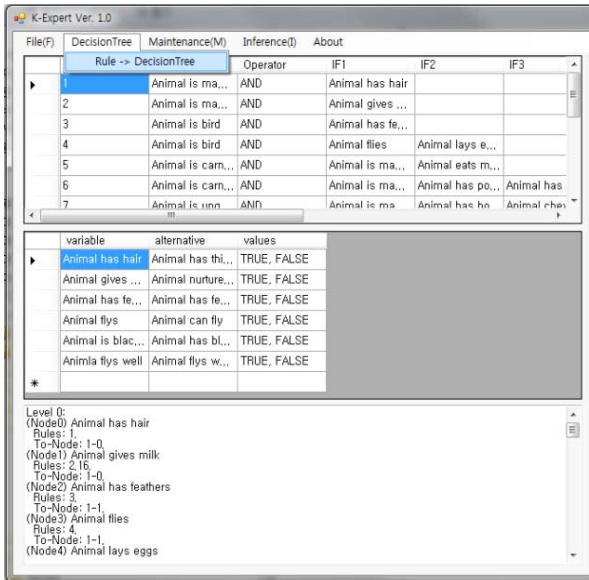


Fig. 4 The Levels and Nodes in the Decision Tree

The tool has drawn the DT information into a graphical expression. Table 2 shows the descriptions of the information.

Table 2 The information used to draw the DT

| | |
|-----------------------------|--|
| Level 0: | " The number of the Level = 0 |
| (Node0) Animal has hair | " The node number = 0 |
| | " The node value is 'Animal has hair' |
| Rules: 1. | " The rule number = 1 |
| To-Node: 1-0. | " The target (to) node number = 0 in Level 1 |
| (Node1) Animal gives milk | |
| Rules: 2.16. | |
| To-Node: 1-0. | |
| (Node2) Animal has feathers | |
| Rules: 3. | |
| To-Node: 1-1. | |
| (Node3) Animal flies | |
| Rules: 4. | |
| To-Node: 1-1. | |
| (Node4) Animal lays eggs | |
| | |

Fig. 5 shows the whole structure of the decision tree. On that figure, the level 0 is shown at the top of the tree and level 4 is shown at the bottom of it. Where, the users could change the positions of each node and the figure shows the changed shape.

3.3 Checking of Unlimited Circulation

Fig. 6 shows the process of *Checking of Unlimited Circulation*. The result of the process is shown in the textbox at the bottom of the window. The sentence below shows that the system found one circulation.

Circulation

Level# = 1, Node# = 0, Animal is mammal

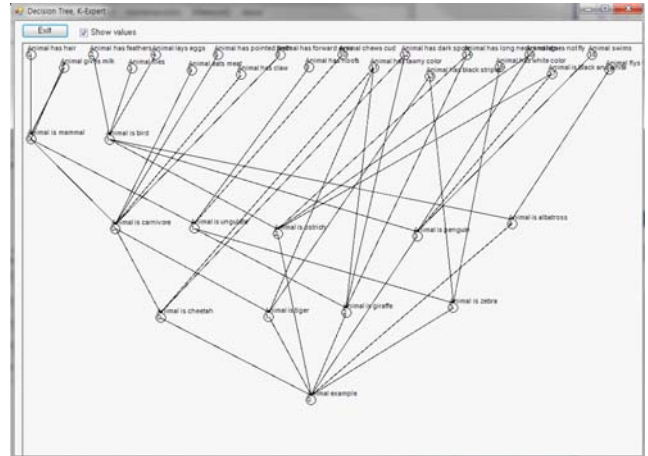


Fig. 5 A Graphical representation of the Decision Tree

The meaning of the sentence above is as follows:

"One circulation is found at the Node #1 in the Level #1. Then the value of the node is 'Animal is mammal'."

The Fig. 7 shows the circulation (unlimited loop) on the decision tree. Where the first link came from #1 Node (Animal gives milk) in the second link came from Level 0, and #0 Node (Animal is mammal) in the Level 1. Therefore, the relationship of the two nodes could be represented as a circulation or unlimited loop.

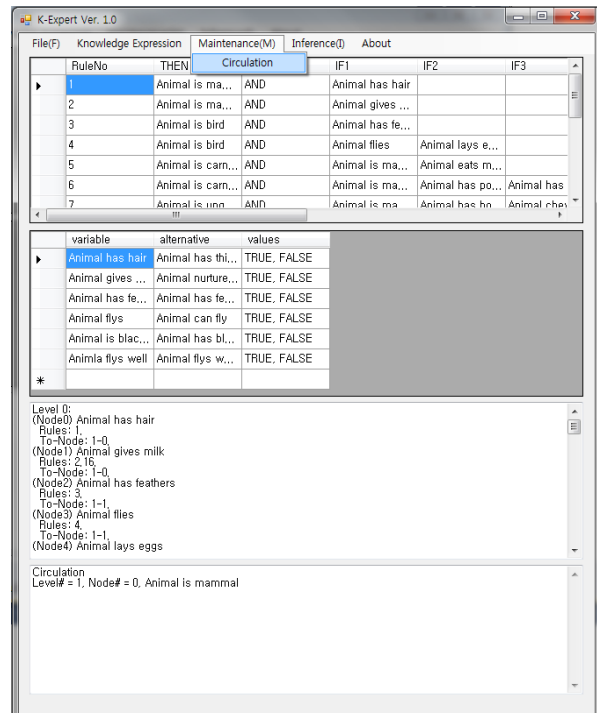


Fig. 6 Checking of Circulation

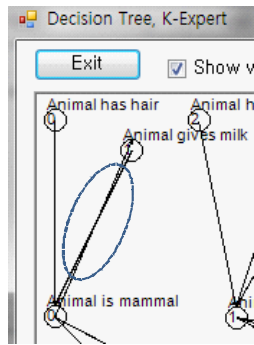


Fig. 7 The circulation on the decision tree

3.4 Application of Classification Problem

In this section, to verify the usefulness of our proposed expert systems development tool, we used the second sample data set titled as 'Blood Transfusion Data Set' gathered from the Blood Transfusion Service Center in Hsin-Chu City, Taiwan. This is a traditional classification problem [8]. The result of the classification was used as a target group in a marketing project. Table 3 shows the sample data set and brief descriptions.

Table 3 Blood Transfusion Data Set

*) Field descriptions

R (Recency): months since last donation

F (Frequency): total number of donation

M (Monetary): total blood donated in c.c.

T (Time): months since first donation

D (Donation): 1 stands for donating blood; 0 stands for not donating blood

| |
|-------------------|
| R, F, M, T, D |
| 4,4,1000,4 ,0 |
| 2 ,7,1750,14 ,1 |
| 1 ,12,3000,35 ,0 |
| 2 ,9,2250,22 ,1 |
| 5 ,46,11500,98 ,1 |
| 4 ,23,5750,58 ,0 |
| 0 ,3,750,4 ,0 |
| 2 ,10,2500,28 ,1 |
| 1 ,13,3250,47 ,0 |
| 2 ,6,1500,15 ,1 |
| ... |

The Fig. 8 shows the result of Knowledge Elicitation. In this study the SPSS Modeler 13 was used to support the procedure. Totally 11 inference rules were generated.

Then the Table 4 shows the text-oriented KB preserving the inference rules shown in the Fig. 8.

| | | | |
|-----------------|---------------------|------|-------|
| donated = 0.000 | Time > 51.500 | 9.0 | 100.0 |
| | Frequency < 6.500 | | |
| donated = 0.000 | Recency > 9.500 | 46.8 | 88.89 |
| donated = 1.000 | Monetary > 4625.000 | 0.6 | 100.0 |
| | Frequency < 21.000 | | |
| donated = 1.000 | Monetary > 4625.000 | 0.6 | 100.0 |
| | Recency < 5.500 | | |
| donated = 1.000 | Frequency < 21.000 | 0.6 | 100.0 |
| | Monetary > 4625.000 | | |
| | Recency < 5.500 | | |
| | Recency > 3.500 | | |
| donated = 1.000 | Time > 67.000 | 0.6 | 100.0 |
| | Frequency > 18.500 | | |
| | Frequency < 21.000 | | |
| donated = 1.000 | Frequency > 18.500 | 0.6 | 100.0 |
| | Recency < 5.500 | | |
| | Frequency < 21.000 | | |
| donated = 1.000 | Frequency > 18.500 | 0.6 | 100.0 |
| | Recency < 5.500 | | |
| | Recency > 3.500 | | |
| donated = 1.000 | Time > 67.000 | 1.4 | 71.43 |
| | Monetary > 4625.000 | | |
| | Recency < 5.500 | | |
| donated = 1.000 | Frequency > 18.500 | 1.4 | 71.43 |
| | Recency < 5.500 | | |
| donated = 1.000 | Frequency > 18.500 | 2.0 | 50.0 |

Fig. 8 Knowledge Elicitation

Table 4 Text-oriented KB (inference rules)

| | |
|-----------------|---|
| Inference rules | |
| Rule 1 | if Time > 51.500 and Frequency < 6.500 then 0.000 |
| Rule 2 | if Recency > 9.500 then 0.000 |
| Rule 3 | if Monetary > 4625.000 and Frequency < 21.000 then 1.000 |
| Rule 4 | if Monetary > 4625.000 and Recency < 5.500 and Frequency < 21.000 then 1.000 |
| ... | |
| Rule 10 | if Frequency > 18.500 and Recency < 5.500 then 1.000 |
| Rule 11 | if Frequency > 18.500 then 1.000 |

Fig. 9 shows the RDB converted from the inference rules in Table 4, and Fig. 10 shows its graphical representation of the decision tree.

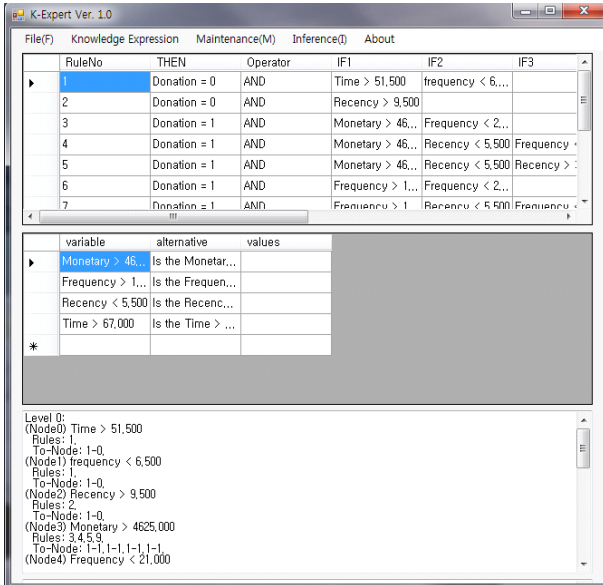


Fig. 9 The RDB-KB

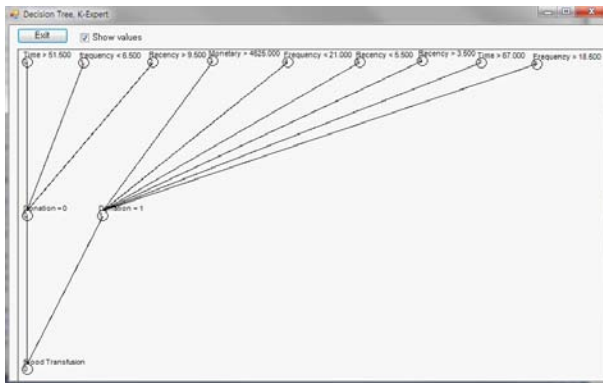


Fig. 10 Decision Tree of the Blood Fusion data set

The Fig. 11 shows the *Economics of the Rules*. The inference rules #6, #7, #8, #9, and #10 would be replaced by the rule #11 because the shortest rule #11 can replace the other rules (#6-#11).

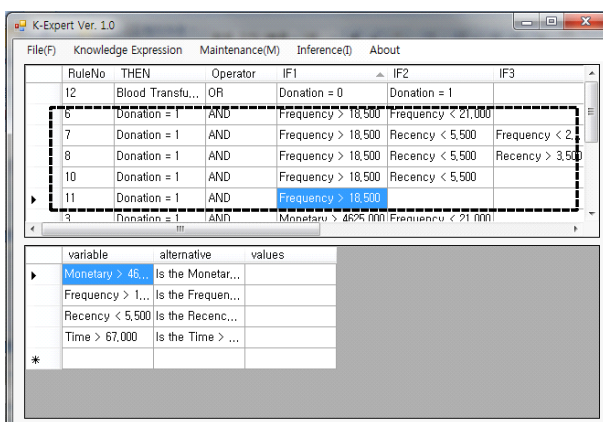


Fig. 11 The Economics of the rules

If there is a *Circulation*, it would be discovered by our

proposed mechanism. However, there is no circulation because the rules in Fig. 8 were automatically generated by the system. After the construction of the initial KB, the human expert could engage into the revision of the KB. In that case, our proposed mechanism could help the expert to revise the KB without the concerns with unexpected errors.

4. Conclusions

In this study, we proposed a RDBMS-based knowledge conversion tool for ES development. The two main functions we have focused on this study are the *Conversions of the rules* and the *Checking of Unlimited Circulation of the Inference Rules*. Including those functions, the tool has six main components *Library, Knowledge Elicitation, ES Generator, Knowledge Expresser, Inference Engine, and Knowledge Maintenance*. To show the processes executed by the tool, we developed the prototype system K-Expert. The tool was developed by using the Visual Studio 2010 and MS-Access Database.

Through the study, it is expected that our proposed mechanism would have significant effects on the knowledge management in using and/or development of ES. Especially, it could help to reduce the burden of knowledge management.

The further research topics still remain as follows. First, to help the knowledge managers efficiently, we need to develop the other reasonable *Knowledge Maintenance* routines. Second, the other intelligent decision support mechanisms such as fuzzy logic, cognitive map, and neural networks would be improve the reasoning ability of ES.

References

- [1] J. S. Kim, "Prediction of User's Preference by using Fuzzy Rule & RDB Inference: A Cosmetic Brand Selection," *International Journal of Fuzzy Logic and Intelligent Systems*, vol.5, no.4, pp.353-359, 2005
- [2] J. S. Kim, "RDB-based Automatic Knowledge Acquisition and Forward Inference Mechanism for Self-Evolving Expert Systems," *Journal of Fuzzy Logic and Intelligent Systems*, vol.13, no.6, pp.743-748, 2003.
- [3] B. G. Buchanan and E. H. Shortliffe, *Rule-Based Expert Systems: The MYCIN Experiments of the Stanford Heuristic Programming Projects*, Addison-Wesley, Reading, MA, 1984.
- [4] H. Yan, Y. Jiang, J. Zheng, B. Fu, S. Xiao, and C. Peng, "The Internet-based Knowledge Acquisition and Management Method to Construct Large-scale Distributed Medical Expert Systems," *Computer Methods and Programs in Biomedicine*, vol.74, no.1, pp.1-10, 2004.
- [5] A. Rafea, H. Hassen, and M. Hazman, "Automatic Knowledge Acquisition Tool for Irrigation and Fertilization Expert Systems," *Expert Systems with Applications*, vol.24, no.1,

pp.49-57, 2003.

- [6] M.S. Chen, J. Han and P.S. Yu, "Data mining: An Overview from a Database Perspective," *IEEE Transactions on Knowledge and Data Engineering*, vol.8, no.6, pp.866-883, 1996.
 - [7] I. Hatzilygeroudis and J. Prentzas, "Integrating (rules, neural networks) and Cases for Knowledge Representation and Reasoning in Expert Systems," *Expert Systems with Applications*, vol.27, no.1, pp.63-75, 2004.
 - [8] I-C Yeh, K-J Yang, and T-M Ting, "Knowledge Discovery on RFM Model using Bernoulli Sequence," *Expert Systems with Applications*, vol.36, no.3, pp.5866-5871, 2009.
-



Jin S. Kim

Associate Professor of MIS in Jeonju University

Research Areas: Fuzzy theory, artificial intelligence, intelligent decision support, expert systems, and semantic web

E-mail : kimjs@jj.ac.kr