Conceptual model of knowledge base system

L V Naykhanova, I V Naykhanova

East Siberia State University of Technology and Management, 40V, Klyuchevskaya Street, Ulan-Ude, 670013, Russia

E-mail: obeka_nlv@mail.ru

Abstract. In the article, the conceptual model of the knowledge based system by the type of the production system is provided. The production system is intended for automation of problems, which solution is rigidly conditioned by the legislation. A core component of the system is a knowledge base. The knowledge base consists of a facts set, a rules set, the cognitive map and ontology. The cognitive map is developed for implementation of a control strategy, ontology - the explanation mechanism. Knowledge representation about recognition of a situation in the form of rules allows describing knowledge of the pension legislation. This approach provides the flexibility, originality and scalability of the system. In the case of changing legislation, it is necessary to change the rules set. This means that the change of the legislation would not be a big problem. The main advantage of the system is that there is an opportunity to be adapted easily to changes of the legislation.

1. Introduction

A legal instrument regulates organization activity. The instrument can define not only composition of tasks, but also an order and the principles of their decision. The pension legislation can be an example. In such cases, business processes automation needs to be executed in strict accordance with the existing legislation. Now the software is developed on the basis of hard algorithms. This means that the change of the legislation requires development of the new software.

To decrease an outlay in maintenance of the software, the authors offer to use an artificial intelligence technology in this case and to develop the knowledge-based system.

A knowledge-based system (KBS) is a computer program that reasons and uses a knowledge base to solve complex problems. All knowledge based systems are an attempt to represent knowledge explicitly via tools, such as ontologies and rules, rather than implicitly via code, the way that a conventional computer program does, using hard algorithms. A knowledge based system has three types of sub-systems: a knowledge base, a user interface and an inference engine. The knowledge base represents facts about the world, often in some form of subsumption ontology. The inference engine represents logical assertions and conditions about the world, usually represented via IF-THEN rules [1].

In this paper, the knowledge-based system model for automation of problems, which solution is rigidly caused by the legislation, is considered.

2. General information about the system

The knowledge-based system is developed as a production system. The production system is artificial intelligence software which basis is the behavior rule set. Rules are called production rules or production. They are applied in automated planning, expert systems, action selection, etc. The
production system provides the mechanism that is necessary to make productions to achieve a definite goal for solution of the problem.

A production consists of two parts: the description of a situation and an action, i.e. production is a pair of a situation-action. The left part represents the list of the conditions describing a situation, and the right part - the list of actions which need to be made. This formalism is well-suited for problems associated with the analysis of situations. Such tasks include the tasks of the pension fund. The solution of the problem in each specific occurrence during the calculation of pensions depends on correct recognition of situations. It should be noted that the representation of knowledge about situation recognition in the form of products provides the flexibility, originality and scalability of the system.

3. The production system architecture
A production system consists of the following basic components: the knowledge acquisition, the interface, a knowledge base, the working memory loader, the reasoning machine, the working memory, the explanations mechanism [1 - 4]. Figure 1 shows the architecture of a production system.

![Figure 1. A conceptual framework of a production system.](image)

It should be noted that in all production systems, the knowledge base forms before the beginning of technology operations of software.

3.1. Knowledge acquisition
The main component of the subsystem "Knowledge acquisition" is the knowledge base editor. The knowledge base needs to be correct; therefore, this subsystem also includes the module "Testing of knowledge base". The main objective of testing consists in checking the condition of the rule for uniqueness. In addition, the generator of interpreters of rules belongs to this subsystem.

3.2. Interface
The interface subsystem includes two modules: a UI expert and UI. The interface expert and the knowledge base editor need a mechanism to build and modify efficiently the knowledge base. In the user interface, databases requests, which are necessary for the decision of the current task, are automatically created and executed. A user shall realize the check of their execution.
3.3. Knowledge base

The knowledge base consists of the facts base, a production rules set and ontology. The facts base contains the actual information on parameters of the current task. The ontology describes compliance of rules and situations and also contains knowledge about the legislation.

The knowledge base core is the productions set. A rule has the structure IF-THEN. After the word “IF”, a certain pattern that describes the situation through the subconditions list (antecedents) follows. After THAT, the conclusion follows, which describes the list of actions that should be carried out.

The rule base consists of a set of production rules tables. Each table represents the production rules system. Let us call it the atomic system of the production rules. This system analyzes all situations, in case of which the value of one parameters is set. Each rule has the list of actions which need to be executed if the rule is applicable. For each atomic system of the production rules, the interpreter of rules is generated. Figure 2 shows interaction between components "Knowledge acquisition" and "Inference machine" during creation of the system.

![Figure 2. Interaction between components (Knowledge acquisition and Inference machine) during creation of system.](image)

3.4. Working memory

A working memory loader is intended for loading a start state of a working memory in a random access memory. The working memory is presented by data structures, in which the facts are stored. The facts are created from the information obtained from the user interface and different databases (The database containing reference information and information on regulations, Population Register, etc.).

3.5. Inference machine

Principal components of a subsystem "Inference machine" are the control strategy and the rules interpreter. A control strategy [1, 2]:

- determines the order in which the rules are applied to a current state of a working memory;
- provides a way of resolving any conflicts that can arise when several rules match at once. A process of rules application to a current state of a working memory (recognition of the current situation) is executed by the rules interpreter. This process is executed in a cycle;
• defines achievement of a final state. If the final state is reached, then a computation process comes to the end.

The cognitive map, which specifies what rule system needs to be called for resolving this or that situation, is developed for implementation of the Control strategy.

The rules interpreter has three stages:
1. Matching the condition/premise patterns in the rules against the elements in the working memory to identify the set of applicable rules. The working memory is located in a random access memory.
2. If there is more than one rule that can be ‘fired’ (i.e. that can be applied), then a Conflict Resolution strategy is used to choose which one to apply.
3. Applying the chosen rule may result in calculation of formulas, adding new items to the working memory, or in deleting old items. Computation of formulas is laid to the formulas interpreter.

The basis of the rules interpreter is an algorithm of a pattern-matching search.

3.6. Explanation mechanism

Any practical production system shall have an explanatory facility [1]. It is important that the production system was able to explain how the decision is obtained.

The results of production system are:
• the calculated values of required parameters;
• the decision in the form of the applied rules sequence.

The found rules sequence allows creating a necessary explanation. The explanation subsystem uses ontology, in which the description of a situation and the article of the legislation are connected to each rule.

4. Results and Discussion

The testing based on real data confirmed operability of the system. When recognising a situation, there is always only one rule. This fact shows that the technique of implementation of the rule base and the control strategy is developed correctly.

5. Conclusion

In the article, the production system architecture, which is intended to solve problems, whose solution is rigidly caused by the legislation, is provided. Unlike traditional production systems, for each atomic rule system, the rules interpreter is generated. It allows storing rules not in a declarative representation and in procedure representation. This adjustment of the rules interpreter to the specific rules system and their procedure representation increases considerably the system performance.

In case of changing legislation, it is necessary to change the appropriate atomic rules system and to generate their rules interpreter. This means that the change of the legislation would not be a big problem. Thus, the main advantage of the system is the ability to adapt to changes of the legislation.

6. References

[1] Bullinaria J M 2005 IAI: Production Systems p 20 (Electronic Materials: http://www.cs.bham.ac.uk/~jxb/IAI/w7.pdf)
[2] Eisenberg M, Fischer G 1997 Rules and Rules Chaining – Winston, Chapter 7 - L3D p 19
[3] Sze Grosan C, Abraham A. 2011 Intelligent Systems: Rule-Based Expert Systems, ISRL 17, pp 149-185 (Springer-Veglar Berlin Heidelberg)
[4] Nilsson N 1998 Artificial Intelligence: A New Synthesis (San Francisco: Morgan Kaufmann) p 513