PRACE NAUKOWE AKADEMII EKONOMICZNEJ WE WROCŁAWIU
Nr 1064 2005
Pozyskiwanie wiedzy i zarządzanie wiedzą

Grzegorz J. Nalepa, Antoni Ligęza
Institute of Automatics, AGH University of Science and Technology

KNOWLEDGE MANAGEMENT ISSUES IN RULE-BASED SYSTEMS INTEGRATED DESIGN PROCESS[1]

1. Introduction

Knowledge Management requires efficient knowledge representation methods as well as flexible and powerful knowledge processing algorithms. Although there exist numerous paradigms attempting at solving these problems, it seems that no single approach will be capable to assure all the required features. However, the rule-based approach seems to be much more sophisticated and powerful than it could be expected on the basis of the lecture of classical handbooks.

Rule-Based Systems (RBS) constitute a powerful tool for specification of knowledge in design and implementation of knowledge-based systems (KBS) in applied Artificial Intelligence and Knowledge Engineering. They provide also a universal programming paradigm for domains such as system monitoring, intelligent control, decision support, situation classification, system diagnosis and operational knowledge encoding. part from off-line expert systems and deductive data-bases, one of the most useful and successful applications consists in development of wide spectrum of control and decision support systems. Some features of modern rule-based systems decisive for success in sophisticated applications include:

- possibility of defining complex preconditions and conclusions (depending on the language in use),
- ability to specify dynamic shaping of knowlde in the knowledge-base (with use of the *retract* and *assert* predicates),
- incorporation of arbitrarily complex inference control mechanism,

[1] The paper is supported from KBN research Project No. 4 T11C 035 24

- possibility of hierarchical knowledge encoding and operation, and last but not least,
- capability of automated verification of knowledge specification.

Further, although the rule-based programming paradigm seems relatively conceptually simple, in case of realistic systems it is a hard and tedious task to design and implement a rulebased system that works in a correct way. Problems occurs as the number of rules exceeds even relatively very low quantities. It is hard to keep the rules consistent, to cover all operation variants and to make the system work according to desired algorithm.

Practical design of non-trivial rule-based systems requires a systematic, structured and consistent approach. Such an approach is usually referred to as a *design methodology*. The basic elements distinguishing one methodology from the other are the internal design process structure i.e. the way structuring of the design process and the components of the process. The structure can be linear, linear with loops, hierarchical (top-down, bottom-up), etc. The components may be various procedures, techniques, tools and documentation aids to support and facilitate the process of design.

The paper outlines main problems encountered in practical RBS design. Then a complete, integrated RBS design methodology is presented. This methodology is supported by a visual CASE tool called Mirella. Main knowledge management issues concerning the methodology are discussed. Finally concluding remarks are given.

2. Problems of RBS Design

Rule-based systems are used extensively in practical applications, especially in domains such as automatic control, system monitoring, technical and medical diagnosis, etc. Many modern applications in various domains are reported in. The rule-base technology is useful and efficient in numerous areas requiring symbolic processing of knowledge. Its success is mostly due to extremely powerful but simultaneously straightforward and transparent operational knowledge specification.

However, although the rule-based programming paradigm seems relatively conceptually simple, in case of realistic systems it is a hard and tedious task to design and implement a rulebased system that works in a correct way. Problems occurs as the number of rules exceeds even relatively very low quantities. It is hard to keep the rules consistent, to cover all operation variants and to make the system work according to desired algorithm.

Particular problems concern the selection of knowledge representation formalism as well as design of an appropriate rule-base and knowledge acquisition. Further problems include building the inference engine and developing control strategy. Last but not least there are problems with verification, validation and testing of the knowledge-based systems [1,2,3]. Building a non-trivial system requires solving several methodological issues and use of specific software tools.

The main problems encountered in the development of knowledge-based systems in general and expert systems or rule-based systems in particular are located in the following areas:

- knowledge representation, which consists in selecting a proper language for encoding the acquired knowledge,
- knowledge acquisition, which is the process of extracting domain knowledge possessed by human expert,
- developing inference mechanism, which concerns design and implementation of an interpreter capable of rule execution,
- developing reasoning control, which concerns a meta-level knowledge for organizing the search and order of rules during reasoning so as to perform in an efficient way and avoid exponential explosion, dead-ends or infinite loops, etc.
- knowledge verification, which consists in checking certain characteristics and correctness of the system knowledge base,
- explaining solutions, which concerns human/computer interaction, presenting solutions to the user, as well as explanations why and how the solutions have been found,
- developing man-machine interfaces, which concerns human/computer interaction during design, knowledge acquisition, inference and verification.

In numerous knowledge-based systems the basic, core knowledge representation consists in using a rule-based system specified in appropriate logic--based language. The basic language is normally one being equivalent to attribute logic or some more elaborated versions of it. With respect to inference mechanism usually forward chaining is used in monitoring and control systems, and backward chaining in diagnostic and some decision support systems. Examples of some well-known solutions, both with respect to basic theory and practical examples of such systems, are presented in the domain literature.

However, a practical implementation of rule-based systems encounters two main problems. The first one is known as the *knowledge acquisition bottleneck* and it consists in the well-known difficulties with obtaining a precise knowledge specification. Using specific knowledge representation structures and abstract knowledge representation may be used to help to overcome this problem. This is so thanks to systematic guiding the development of the rulebase by introducing structure ordering the design.

The second problem concerns the analysis, verification and validation of knowledge. By making the analysis of the knowledge base a part of ongoing knowledge acquisition and review process, expert system developers can minimize the time and resources devoted to development of such systems. In order to assure safe, reliable and efficient performance, analysis and verification of selected qualitative properties should be carried out. Those properties include features such as, completeness, consistency and determinism [1,2]. However, verification of them after the design of a rule-based system is both costly and late [5,6]. The verification may be complex, so in most of practical applications building, debugging and maintaining the rulebase are the most costly activities.

In order to address these problems a new knowledge representation method has been developed.

3. XTT-based RBS Integrated Design Process

The existing RBS design methods and tools have limitations mostly in the following three areas:

- knowledge representation methods,
- framework for analysis and verification of system properties,
- integrated computer tools supporting the design process.

To overcome these limitations a new approach, Extended Tabular Trees (XTT) has been proposed and developed. It was discussed in detail in [12]. The idea is based on integrating the so-called Psi-trees [5,6] and the tabular knowledge representation scheme [7]. An early proposal of this approach was presented in [4].

The XTT approach was invented with the goals to integrate the system design and verification stages; further, it supports the implementation through introducing the possibility of automatic code generation. Basing on it, an integrated rule-based system design and implementation process, supported by a computer tool, is discussed in the following paragraphs.

The main idea behind new visual knowledge representation language called *Extended Tabular-Trees* aims at combining some of the existing approaches such as decision-tables and decision-trees by building a special a hierarchy of Object-Attribute-Tables. This hierarchy is based on the so-called Psi-tree structure [5,6].

The new language has some unique features such as: simplicity and transparency, due to an intuitive way of graphical knowledge specification, hierarchical, tree-like knowledge representation, highly efficient way of visualization with high data density using tables similar to Relational Database tables, power of the decision table representation, flexibility with respect to knowledge manipulation, analogies to the RDB data representation scheme, direct knowledge representation mapping into Prolog schemes and rule-based systems.

The language plays a key role in the new approach to RBS design method, on which the Mirella tool is based. It also serves as a foundation for a new approach to practical RBS design.

The proposed approach follows the structural methodology for design of information systems. It is simultaneously a top-down approach, which allows for incorporating hierarchical design - in fact, any tabular component can be split into a network of more detailed components, and a network of components can be grouped together to form a more abstract component. The approach covers the stages of conceptual, logical and physical design. The principles of the integrated design process are based on selected existing approaches to system design.

The approach proposed herein does not aim at covering the whole life-cycle, as for example the one that can be found in classical software engineering handbooks. However, it does aim at including all phases of the system life-cycle from the design to implementation phase.

The following three design phases are identified:

1. ***Conceptual design***, in which the basic structure of the system is identified, along with data and control flow, as well as main operating contexts, objects and their attributes; this allows for further defining the headers of XTT tables,

2. ***Logical design***, which involves building table rows (corresponding to rules), connecting tables; the XTT structure can be incrementally built, analyzed, and possibly verified and even optimized on-line,

3. ***Physical design***, in which a preliminary implementation is done by building a Prolog code (or any other target language since the approach is of generic character), which can be executed, compiled, debugged and possibly translated to system-specific representation.

This is a top-down approach. The names of design phases are similar to Relational-Data Base design phases. However, the actual stages in each phase are different.

All of the stages discussed above are supported by ***Mirella***, an integrated CASE environment. In the following subsections the guidelines for each phase are given. An example schematically showing the subsequent design phases is presented in Fig. 1.

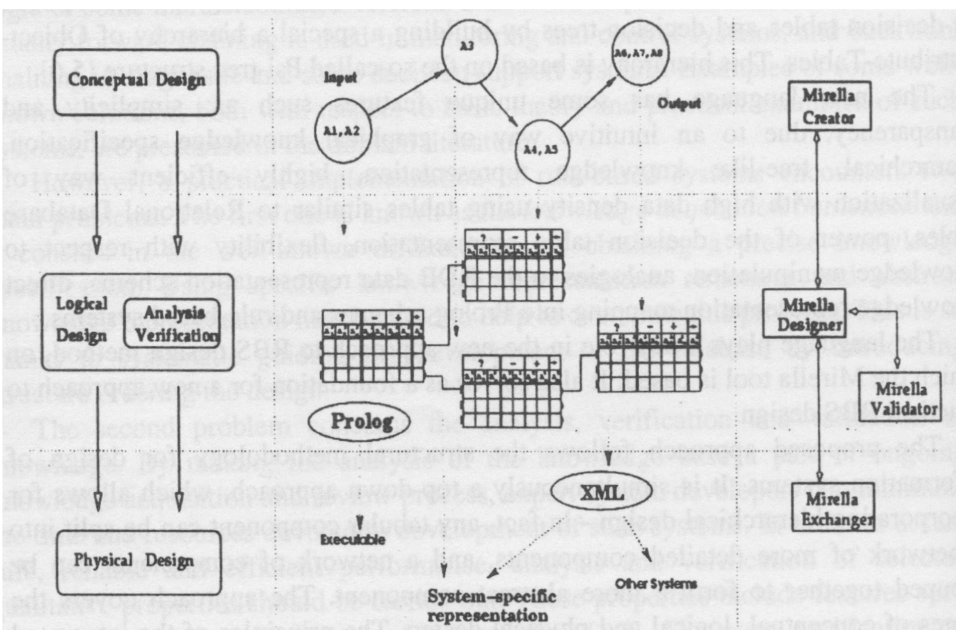

Fig. 1. Integrated RBS Process Scheme

One of the most important features of this approach is the separation of logical and physical design, which also allows for a transparent, hierarchical design process.

4. The Mirella Tool

Mirella [12] is an intelligent visual design tool supporting on-line verification of rulebased systems, based of the XTT knowledge representation. It is oriented towards designing reliable and safe rule-based systems in general. The main goal of the system is to move the design procedure to a more abstract, logical and graphical level, where knowledge specification is based on use of abstract rule representation. The designed graphical specification is automatically translated into a predefined XML (XTTML) knowledge format, so the designer can focus on logical specification of safety and reliability; simultaneously, practical code can be generated form a wide class of systems. On the other hand, formal aspects such as completeness, determinism, etc. may be automatically verified on-line during the design, so that it verifiable characteristics are preserved.

An example of a design session is given in Fig. 2. In the figure the XTT representation of a thermostat rule-based system is shown. Mirella and the XTT approach has been previously discussed in several papers, namely [12] and [13,14,15,16].

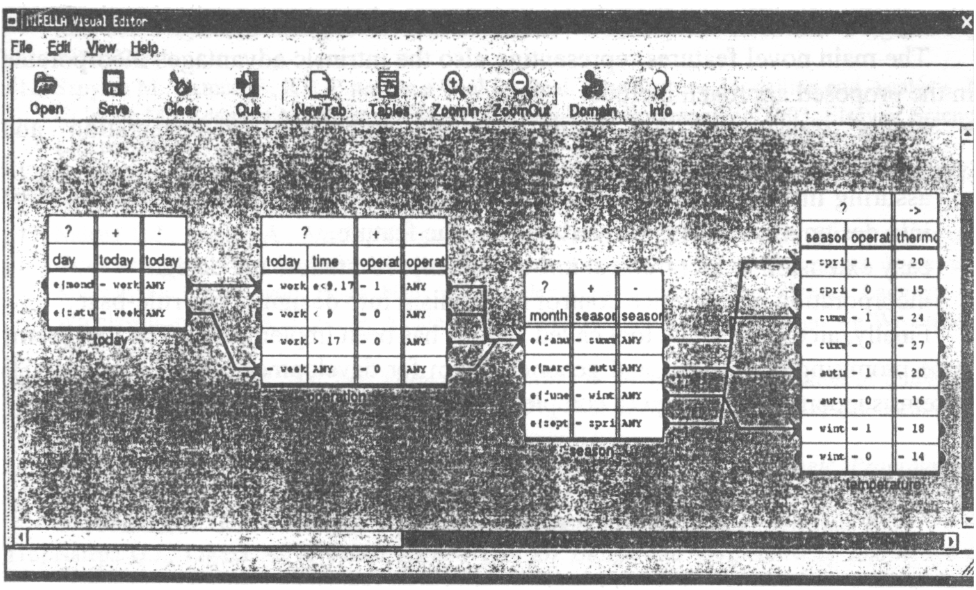

Fig. 2. Mirella Session, Thermostat System

5. Knowledge Management Issues

The integrated design process is based on knowledge about the system. In this process the most important knowledge menagement issues are:

- knowledge acquisition,
- knowledge representation,
- knowledge transformation, and
- knowledge verification.

Each of these issues is addressed in the process and in the Mirella CASE tool.

The knowledge acquisition process takes place mainly during the conceptual and logical design phase. It is supported by the Mirella Creator and Designer modules. The Designer allows for both visual and structural knowledge representation using the XTT. Apart from the visual representation, its logical formulation is also preserved.

The knowledge about the system being designed is transformed by the Mirella Exchanger module from the XTT to markup-based XTTML language as well as Prolog-based rule-based system prototype. Using the logical system representation the Mirella Validator module performs knowledge verification using Prolog-based procedures.

6. Concluding Remarks

The paper discusses rule-based systems design problems. It also gives some practical solutions to these problems, including the XTT knowledge representation method, as well as the integrated RBS design process. Selected knowledge management issues concerning these solutions are also presented.

The main novel features representing also the intrinsic advantages incorporated in the proposed approach include:

- highly transparent and intuitive visula knowledge representation and manipulation (XTT schemes),
- assuring integrated deisgn process through incorporating the verification stage into design as an intrinsic, available on-line feature,
- easy extension to hierarchical knowledge representation and design,
- incorporation of inference control mechanism into design of the rule-base.

Finally, an experimental tool called Mirella has been implemented. The tool seems to be promising, innovative and generic system for development of industrial CASE systems supporting the design of complex knowledge-bases in rule-based technology.

References

[1] Ligęza A., *Intelligent Data and Knowledge Analysis and Verification; Towards a Taxonomy of Specific Problems*, Validation and Verification of Knowledge Based Systems: Theory, Tools and Practice (1999) 313-325.

[2] Andert E.P., *Integrated Knowledge-Based System Design and Validation for Solving Problems in Uncertain Environments*, Int. J. of Man-Machine Studies 36 (1992) 357-373.

[3] Van Harmelen F., *Applying Rule-based Anomalies to Kads Inference Structures*, ECAI'96 Workshop on Validation, Verification and Refinement of Knowledge-Based Systems (1996) 41-46.

[4] Ligęza A., Wojnicki I., Nalepa G., *Tab-Trees: a Case Tool for Design of Extended Tabular Systems*, [in:] Database and Expert Systems Applications. Volume LNCS 2113 of Lecture Notes in Computer Sciences. Springer-Verlag, Berlin (2001) 422-431.

[5] Ligęza, A., *Towards Design of Complete Rule-based Control Systems*, [in:] J. Kocijan, R.K., ed., IFAC/IMACS International Workshop on Artificial Intelligence in Real-Time Control. IFAC, Bled, Slovenia (1995) 189-194.

[6] Ligęza A., *Logical Support for Design of Rule-based Systems. Reliability and Quality Issues*, [in:] Rousset, M., ed.: ECAI-96 Workshop on Validation, Verification and Refinment of Knowledge-based Systems. Volume W2. ECAI'96, Budapest1996 28-34.

[7] Ligęza A., *Toward Logical Analysis of Tabular Rule-based Sstems*, International Journal of Intelligent Systems 16 (2001) 333- 360.

[8] Nalepa G.J., Ligęza A., *Graphical Case Tools for Integrated Design and Verification of Rule-based Ssystems*, [in:] T. Burczynski, W. Cholewa, W.M., ed., AI-METH 2002 Methods of Artificial Intelligence, 13-15 November, Gliwice, Poland, Gliwice, Poland, Silesian University of Technology. SUT DSMCM DFMD, Polish Association for Computational Mechanics (2002) 307-313.

[9] Nalepa G.J., Ligęza A., *Designing Reliable Web Security Systems Using Rule-based Systems Approach*, [in:] Menasalvas, E., Segovia, J., Szczepaniak, P.S., eds., *Advances in Web Intelligence*. First International Atlantic Web Intelligence Conference AWIC 2003, Madrid, Spain, May 5-6, 2003. Volume LNAI 2663 of Lecture Notes in Artificial Intelligence., Berlin, Heidelberg, New York, Springer-Verlag (2003) 124-133.

[10] Szpyrka M., Nalepa G.J., *On Two Formal Approaches to Design and Verification of Embedded Rule-based Systems*, accepted to 28th IFAC/IFIP Workshop on Real-Time Programming, WRTP2004.

[11] Szpyrka M., Szmuc T., RTCP-nets as a Tool for Real-Time Systems Modelling and Analysis. In Colnaric M., Adamski M., Węgrzyn M. (Eds.) Real-Time Programming 2003, Elsevier Science Ltd. pp. 21-26.

[12] Nalepa G.J., *Meta-Level Approach to Integrated Process of Design and Implementation of Rule-based Systems*, Rozprawa doktorska, AGH, Kraków, 2004.

[13] Nalepa G.J., Ligęza A., *Markup-languages-based Approach to Knowledge Management and Representation*, Pozyskiwanie Wiedzy i Zarzadzanie Wiedza, eds. Malgorzata Nycz and Mieczyslaw Lech Owoc, Prace Naukowe Akademii Ekonomicznej im. Oskara Langego we Wrocławiu, nr 975, ss. 332-339, Akademia Ekonomiczna im. Oskara Langego we Wrocławiu, Wrocław, 2004.

[14] Nalepa G.J., Ligęza A., *On Two Formal Approaches to Design and Verification of Embedded Rule-based Systems*, Proceedings of WRTP2004, 28th IFAC/IFIP Workshop on Real-Time Programming, Elsevier, 2004.

[15] Nalepa G.J., Ligęza A., *A Graphical Tabular Model for Rule-based Logic Programming and Verification*, Proceedings of XV International Conference on Systems Science, Eds.: Z. Bubnicki, A. Grzech, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2004.

[16] Nalepa G.J., Ligęza A., *A Visual Edition Tool for Design and Verification of Knowledge in Rule-based Systems*, Proceedings of XV International Conference on Systems Science, Eds.: Z. Bubnicki, A.Grzech, Oficyna Wydawnicza Politechniki Wrocławskiej, Wrocław 2004.

PROBLEMATYKA ZARZĄDZANIA WIEDZĄ W ZINTEGROWANYM PROCESIE PROJEKTOWANIA SYSTEMÓW REGUŁOWYCH

Streszczenie

Systemy regułowe stanowią uniwersalny paradygmat programowania i reprezentacji wiedzy. Znalazły liczne zastosowania w systemach inteligentnych, np. systemach ekspertowych. Pomimo prostoty samego formalizmu regułowego, implementacja tych systemów jest problematyczna. Występują problemy z pozyskiwaniem, reprezentacją i weryfikacją wiedzy.

Artykuł omawia wybrane aspekty nowej, oryginalnej metody projektowania i analizy systemów regułowych, opartej na reprezentacji XTT. W metodzie tej wyróżnia się 3 fazy projektu: koncepcyjną, logiczną i fizyczną. Proces projektowania jest wspierany przez wizulane narzędzie projektowe. W artykule uwypuklone są główne aspekty zarządzania wiedza w procesie projektowania: pozyskiwanie, reprezentacja, przetwarzanie i weryfikacja wiedzy.