SPECIALIZED SHELLS OF INTELLIGENT SYSTEMS FOR DOMAINS WITH COMPLICATED STRUCTURES

Irene Artemieva

Abstract: The paper covers the properties of domains with complicated structures and describes information and program components of shells of intelligent systems for such domains, defines all the properties of shell components. It is noted that the maintenance subsystem that adds new program components to the shell is an important component.

Keywords: Specialized Shell of Intelligent System, Program System Based on Ontology, Multilevel Domain Ontology

ACM Classification Keywords: I 2.5 – Expert system tools and techniques.

Conference: The paper is selected from XVth International Conference "Knowledge-Dialogue-Solution" KDS 2009, Varna, Bulgaria, June-July 2009

Introduction

Several problems are to be solved when a knowledge based intelligent system is being designed: how to represent the domain knowledge, how to use it to solve tasks, how to maintain the knowledge base. The more domain is complicatedly structured, the more difficult it is to develop such a system and to maintain its knowledge base. Universal and specialized shells are tools simplifying the intelligent system design process. Universal shells are based on a certain universal language used to represent knowledge. Many of existing commercial universal shells are rule-based systems [Commercial shells]. To represent knowledge, specialized shells use a domain-specific scheme defined by an ontology of the area the shell is designed for. This allows a domain expert to develop a knowledge base without the participation of a knowledge engineer which is an intermediary.

In science-associated domains with complicated structures, not only knowledge may change, but ontologies may also change leading to the changes in a set of classes of tasks to solve. However, methods for development of specialized shells do not embrace such properties of domains with complicated structures. The aim of this paper is to describe the features of specialized shells of intelligent systems for such domains.

Properties of domains with complicated structures

First, let us examine the properties of domains with complicated structures that define the properties of the specialized shells for such domains.

Each domain is characterized by many quantities and sets of mathematical terms to define operations and relations within the elements of these quantities [Artemieva et al, 1997]. The type of quantities used in domains is defined by the way of getting data about values and also by the way of verbal representation of information about domain objects. The elements of verbal representation can specify the information about the values of domain objects properties, the structure of objects and the (functional and nonfunctional) relations between domain objects. Each quantity is characterized by a number of elements of this quantity and also by a set of functions, operations and relations.

Quantities can be either standard (e.g. dimension quantities, scalar and structural quantities, quantities of sets and mappings [Artemieva et al, 1997]) or nonstandard. Each domain comprises its own nonstandard quantities. Nonstandard quantities for the area of chemistry are structural formulas of compounds and spectra.

Knowledge, which is defined as a set of statements about the properties of the reality situations, characterizes domains. Knowledge can be considered structured if it can be represented as a set of pairs <term, term value>. Otherwise, knowledge is not structured and represented by a set of statements on a certain logical language. There are domains in where only a part of knowledge is verbally represented. Such feature is characteristic of chemistry, where the structured part of knowledge describes the properties of various domain objects, while not structured part of knowledge contains the laws of this domain.

If knowledge of a domain or a part of it is structured, then ontology of knowledge is considered to be a property of domain [Kleshchev and Artemjeva, 2007] that defines notions used in verbal representation of knowledge and ontological agreements that specify constraints for sets of values for such terms. For domains with not structured knowledge, the ontology of knowledge is empty, and all domain laws are represented with the usage of terms of ontology of reality consisting of the descriptions of terms and ontological agreements that set the integrity constraints for the reality.

If domain knowledge is structured, then the set of ontological agreements except integrity constraints for the reality and knowledge contains another group of agreements that set the links between the knowledge and the reality of a domain.

Ontologies of domains with complicated structures are multilevel and modular [Artemieva, 2007]. The level with maximum n number represents the ontology of a domain with complicated structure. It contains terms which define the next level ontology (an ontology of a domain section). Transitions to a level with a lower number implies the setting of the ontology terms (of an ontology of domain section or subsection), as well as the setting of the ontological agreements for this level. To determine these ontological agreements it is necessary to set term values of a previous level ontology or to set the agreements on a language to represent these agreements.

Let us call the ontology with the knowledge system the ontology of level 1. Such an ontology defines the properties of a set of situations of the domain reality and therefore the values for the terms of ontology of the reality are not matched. By specifying the values of the terms of ontology of the reality we receive the description of the concrete situations of a domain (or the 0 level "ontology").

Chemistry can be seen as an example of a domain with complicated structure [Artemieva et al, 2007], the multilevel ontology of which is shown on Fig. 1.

2. Specialized shell components

The following groups of users of the specialized shell of intelligent systems for domains with complicated structures can be distinguished: support programmer, knowledge engineer, expert, domain specialist.

Support programmer ensures program components development, knowledge engineer and expert provide development of information components, while domain specialist uses complete program and information components to solve applied problems.

The basic components of a specialized shell are the following: editor of multilevel ontology, ontology-controlled knowledge editor and systems for solving tasks of different classes.

Specialized editor of multilevel ontology is used by knowledge engineers. It provides the opportunity to develop and edit a new module of an ontology of level i+1 and allows the reusability of the existing modules of the every level ontologies.

Specialized knowledge editors, controlled by an ontology, are used by experts in a domain with complicated structure. They provide the opportunity to develop and edit a new module of knowledge controlled by the knowledge ontology module.

Specialized solvers for classes of applied tasks are used by domain specialists to specify input data, to get the results of solutions and their descriptions in terms of the ontology of the reality.

An important component of a specialized shell for the developing domains is a support subsystem that is used by a support programmer. It makes it possible to change the structure of the program components of a shell.

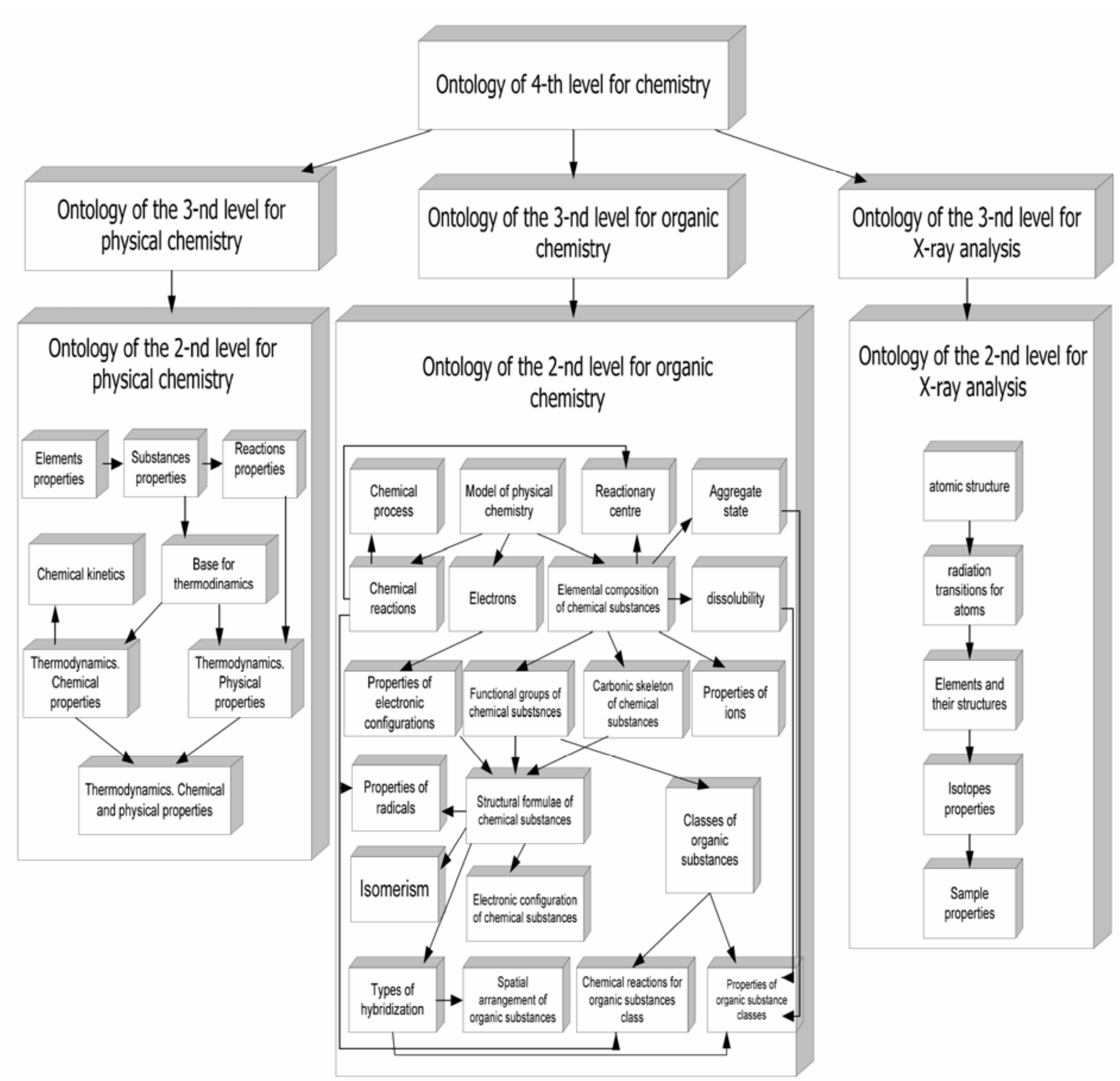

Fig. 1. A multilevel ontology for chemistry.

3. Properties of information components editors

Multilevel module ontology and module knowledge base are information components of a specialized shell for a domain with complicated structure. Development and editing of information components is carried out by a multilevel ontology editor and editor of knowledge, the development of which is based on the n level ontology.

Multilevel ontology editors and knowledge editors are to provide the development and editing of module ontologies and knowledge and to ensure the reusability of the modules for the development of ontologies and knowledge for new sections and subsections of a domain. Thus the development and editing of a module for the ontology of i-1 level are to be controlled by the ontology of i level, while that of a knowledge module is to be controlled by the ontology of level 2.

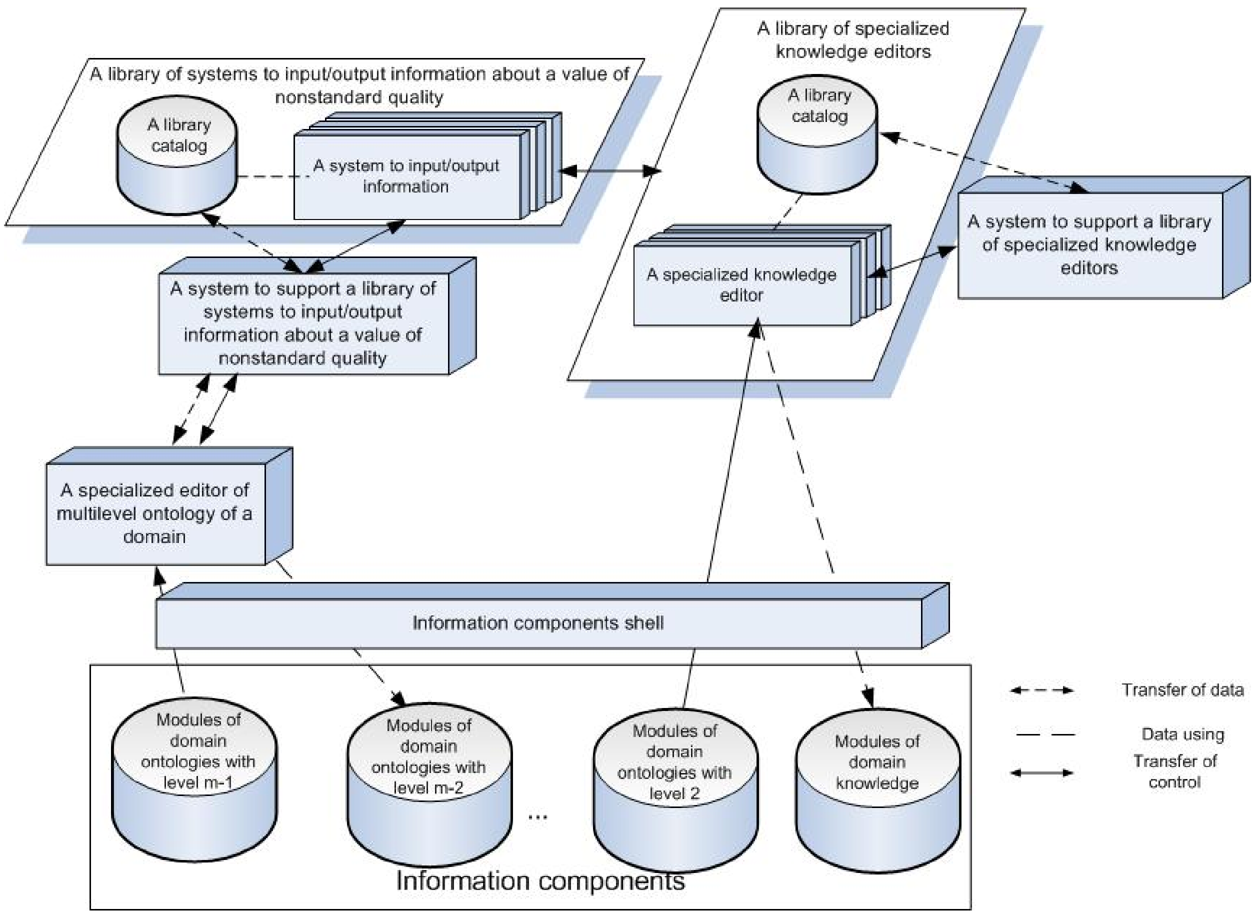

Fig. 2. Information editors for a specialized shell.

Ontology editor is to provide the opportunity to choose that of the existing modules of the ontology of level i that controls the editing of a module under development. The same pattern is used in the editing of a knowledge module, when an opportunity to choose the "controlling" module of the ontology of level 2 is to be provided.

Editors of ontology and knowledge are to provide the opportunity to specify the structured and not structured parts of the ontology and the structured and not structured parts of knowledge, i.e. a program component for these editors would be a specialized statement editor that allows to input the ontological agreements and laws of a domain.

Knowledge editor is to provide the input/output values of nonstandard quantities. There can be a method of graphic representation for the values of nonstandard quantities. For instance, in chemistry [Artemieva et al, 2008] there can be a graphic representation of a short structural formula or structural formula of a chemical compound. Therefore, while developing and editing knowledge, knowledge editor is to provide the opportunity to use the method of graphic representation of the values of nonstandard quantities accepted in a domain. The quantity, to which corresponds a value of a certain property, is specified by the ontology of level 2. Therefore knowledge editor is to provide an automatic choice (controlled by the ontology of level 2) of the means for graphic representation of values of nonstandard quantities.

Ontology editor interprets the ontology of level i while developing a module of the ontology of level i-1. Knowledge editor interprets the ontology of level 2 while developing knowledge module. The same ontology can be interpreted differently by different knowledge editors. Knowledge editors can vary by the ways of knowledge interpretation and by interface as well. It is obvious that for an editor aimed to interpret one ontology, not the whole class of ontologies, more easy-to-use interface and more comprehensible way of interpretation for an expert can be developed. Therefore a specialized shell should allow the application of editors supporting different ways of interpretation of a module of the ontology of level 2, as well as providing an expert with the opportunity to choose the editor needed.

Thus, information components editors are to comprise (Fig.2) specialized editor of multilevel module ontology, specialized knowledge editors controlled by ontologies of level 2, systems to input/output values of nonstandard quantities, statement editor allowing to specify laws and ontological agreements of a domain. 4. Properties of systems for solving applied tasks

The values of nonstandard quantities are used to edit knowledge and enter input data. The method of graphic representation of input data is more convenient for a domain specialist since in this case it is unnecessary to apply cumbersome verbal data representation [5]. Graphic representation of the results of a solution proves to be a more easy-to-see method of representation. Therefore the shell is to provide input/output of the values of nonstandard quantities while specifying input data of the tasks and ensure the use of the method of graphic representation of the values of nonstandard quantities accepted in a domain while entering input data of the tasks and outputting of the results of their solutions.

It has been noted that the quantity to which a value of a certain property pertains is specified by the ontology of level 2. Shell is to provide an automatic choice (controlled by an ontology) of the means for graphic representation of the values of nonstandard quantities while specifying input data of tasks.

Each section (and subsection) of a domain with complicated structure is characterized by its own set of classes of applied tasks though different sets can comprise both common classes of tasks and classes specific for a section (subsection). Solver can be intended for solving classes of tasks of one section (thus it uses the ontology and knowledge of this section) or for classes of tasks of different sections (in this case it uses different ontologies and knowledge). In the first case, the ontology used by a solver is defined by the class of tasks. In the second case, an additional indication what ontology and knowledge are to be used while solving tasks is required. Specialized shell of intelligent systems for domains with complicated structures is to ensure the solution of different classes of tasks with the user being able to specify the module of an ontology and the module of knowledge that are to be used while solving tasks. While solving tasks of one class various systems with different specifications of complexity can be used. These specifications are determined by input data and the statement of a problem. The systems for solving of one class of tasks can also vary by the interface, use different methods of solving of tasks of this class. For example, different laws of a domain can be used while calculating a value of a certain property of a physical-chemical process.

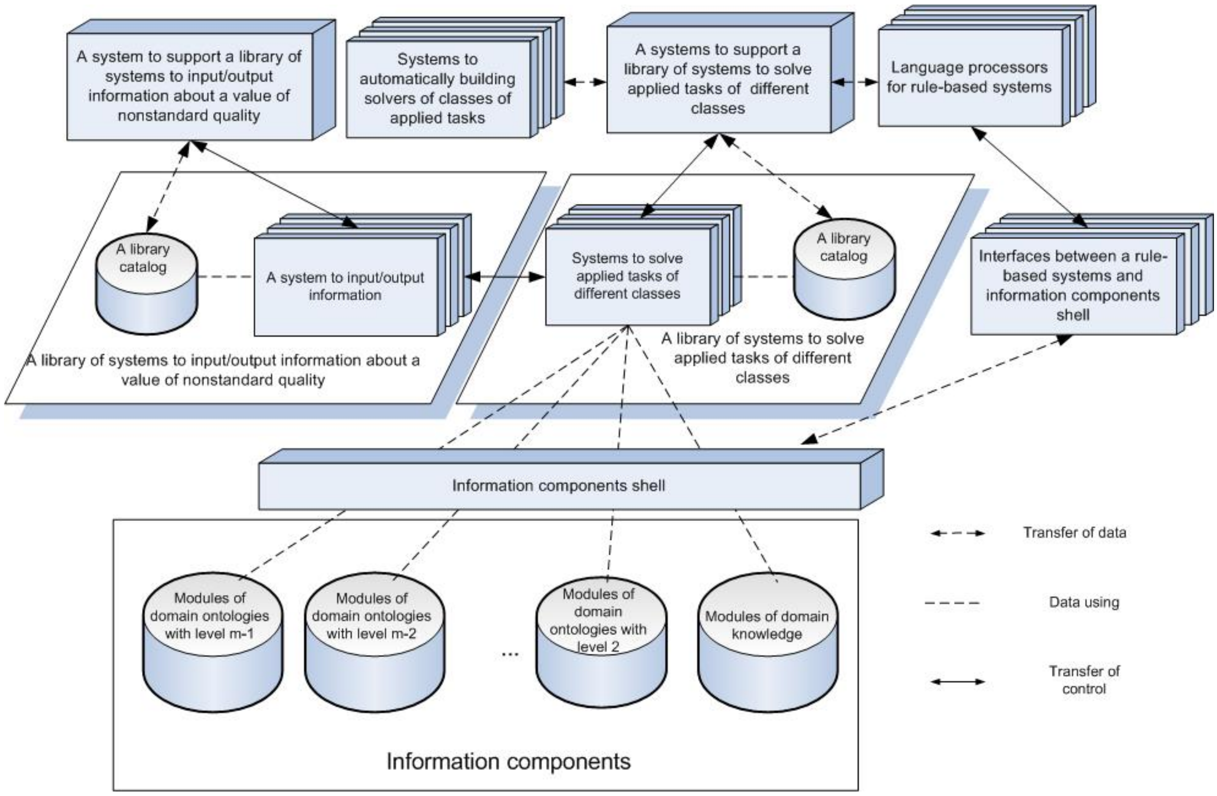

Fig. 3. Systems to solve applied tasks.

Thus, specialized shell is to contain extensible system libraries for solving tasks of different classes, systems for automatic building of methods of solving tasks according to their specifications (Fig.3). A task solving method can be represented in a form of an algorithm or as a set of rules of productions system. In the first case, an algorithmic language processor is applied to develop a solver, in the second case a rule-based language processor [] that is one of program components of specialized shell is used.

Conclusion

If a domain is developing, its ontology and knowledge are changing. Changes in ontology lead to change in a set of classes of tasks solved. Thus, the development of a domain with complicated structure implies the emergence of new information and program components of an intelligent system. Alteration of information components is supported by the editors of these components. Program components development implies the existence of a support subsystem for this process.

The maintenance subsystem for a specialized shell is to make it possible to add new subsystems to input/output values of nonstandard quantities, in particular for those quantities supporting the method of graphic representation of these values, new classes of tasks and subsystems for their solution. It is also to ensure the inclusion of subsystems for automatic building of solvers for classes of tasks into program components. Thus, the program components of the maintenance system are the following: subsystem to maintain subprogram library for nonstandard quantities, subsystem to maintain knowledge editors library (Fig. 4), subsystem to maintain solvers library, subsystem to maintain library of systems for automatic building of solvers for tasks specification (Fig. 4). Rule-based language processor is used while developing the problem-solving methods for the tasks presented as a set of rules.

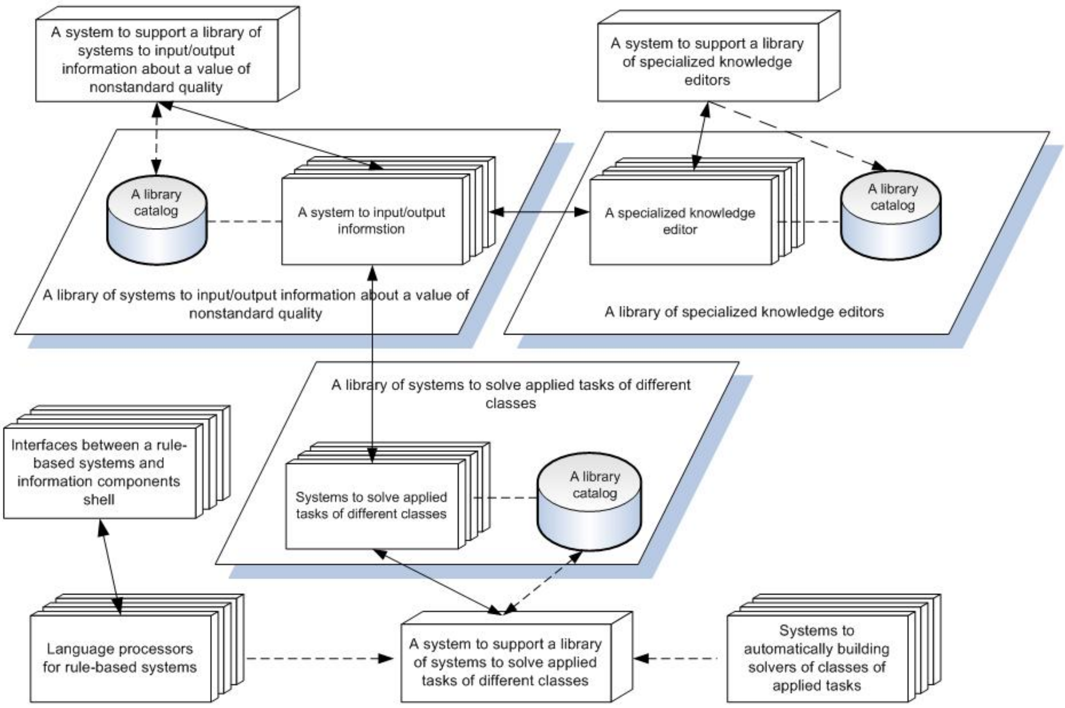

Fig. 4. Components of the maintenance subsystem.

The maintenance subsystem for a specialized shell is to make it possible to add new subsystems to input/output values of nonstandard quantities, in particular for those quantities supporting the method of graphic representation of these values, new classes of tasks and subsystems for their solution. It is also to ensure the inclusion of subsystems for automatic building of solvers for classes of tasks into program components. Thus, the program components of the maintenance system are the following: subsystem to maintain subprogram library for nonstandard quantities, subsystem to maintain knowledge editors library (Fig. 4), subsystem to maintain solvers library, subsystem to maintain library of systems for automatic building of solvers for tasks specification (Fig. 3). Rule-based language processor is used while developing the problem-solving methods for the tasks presented as a set of rules.

The maintenance subsystem for a specialized shell is to make it possible to add new subsystems to input/output values of nonstandard quantities, in particular for those quantities supporting the method of graphic representation of these values, new classes of tasks and subsystems for their solution. It is also to ensure the inclusion of subsystems for automatic building of solvers for classes of tasks into program components. Thus, the program components of the maintenance system are the following: subsystem to maintain subprogram library for nonstandard quantities, subsystem to maintain knowledge editors library (Fig. 4), subsystem to maintain solvers library, subsystem to maintain library of systems for automatic building of solvers for tasks specification (Fig. 4). Rule-based language processor is used while developing the problem-solving methods for the tasks presented as a set of rules.

Acknowledgements

This paper was made according to the program № 2 of fundamental scientific research of the Presidium of the Russian Academy of Sciences, the project 09-I-П2-04.

Bibliography

[Artemieva et al, 1997] Artemieva I.L., Gavrilova T.L., Kleshchev A.S., Logical models of the second order for domains // Scientific and technical Information, Ser. 2. 1997. № 6. P.14-30.

[Artemieva, 2007] Artemieva I.L. Domains with complicated structures and their ontologies // Inf. Theories and Appl. – 2008. - Vol. 15.- № 4. – PP. 330-337. – ISSN 1310-0513.

[Artemieva et al, 2007] Artemieva I.L., Reshtanenko N.V., Tsventikov V.A. Multilevel ontology of chemistry // All-Russian Conference with International Participants «Knowledge-Ontologies-Theories». Novosibirsk, 14-16 September 2007. Novosibirsk: Institute of Mathematics. Vol.1. P. 138-146.

[Artemieva et al, 2008] Artemieva I.L., Reshtanenko N.V. Intelligent system based on multilevel ontology of chemistry // Program products and systems. 2008. № 1. P. 84-87.

[Kleshchev and Artemjeva, 2007] Kleshchev A.S., Artemjeva I.L. Mathematical models of domain ontologies // Inf. Theories and Appl. – 2007. - Vol. 14, № 1. - PP. 35-43. – ISSN 1310-0513.

[Artemieva and Tyutyunnik, 2008] Artemieva I.L., Tyutyunnik M.B. Parallelization of Logical inference for confluent rule-based system // International Book Series "Information Science and Computing". Book 1 "Algorithmic and Mathematical Foundations of the Artificial Intelligence ". 2008. PP.81-87

[Artemieva et al, 2008] Artemieva I.L., Reshtanenko N.V., Tsvenikov V.A. Upgradable tree levels editor of metaontologies, ontologies and knowledge for a chemistry // International Book Series "Information Science and Computing". – Number 5. – Supplement to the International Journal "Information Technologies & Knowledge". – Vol. 2. – 2008. – PP. 119-124. – Sofia, Bulgaria. – ISSN 1313-0455.

[Commercial shells] http://inf.susu.ac.ru/~polak/expert/commercial/faq=doc-7.htm

Authors' Information

Irene L. Artemieva – Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of Sciences; 5 Radio Street, Vladivostok, Russia; e-mail: artemeva@iacp.dvo.ru