Control graph based solver development for knowledge based systems

V.V. Gribova[1], Ph.M. Moskalenko[1], V.A. Timchenko[1], E.A. Shalfeeva[1]

[1]*Institute of Automation and Control Processes – FEB RAS, Radio 5, Vladivostok, Russia, 690041*

Abstract
A methodology for using declaratively defined control graph structures for developing problem solvers for systems with knowledge bases is proposed. Special attention is paid to the processing of knowledge base fragments and the possibility of placing architectural components of the solver on different computing nodes. The structure of such an information-and-control graph includes related constructs (operators), one of which is marked as "beginning". These structures can be divided into: computing control constructs and information processing constructs. The developer can group individual operators into blocks – for their repeated usage (including organization of recursive calculations). The structure of the graph also includes a description of the data.

Keywords
knowledge base, problem solver, control graph, software engineering, program development methodology, information processing

1. Introduction

Reducing the complexity of developing software for solving complex problems of science and practice, as well as ensuring their viability are urgent tasks of software engineering. For systems with knowledge bases (KB), as noted in literature, this problem is most acute, despite the fact that significant results have been achieved in its solving. This problem is especially critical for those systems in which one of the requirements is a time limit for decision-making. Given that the used solution methods belong to the class of exhaustive ones that have a large computational complexity, parallelization is required to reduce the execution time. To ensure the understanding, verifiability and maintainability of ontology-based systems with knowledge bases (KBS), solutions from software engineering (in the field of development of complex software systems and composite applications [1]) can be used with their adaptation to the features of KBS.

2. Results

A methodology for the development of KBS problem solvers is proposed, which increases their transparency and is based on the declarative formation of a structure called an information-control graph (ICG). Special attention is paid to the processing of KB fragments and the possibility of placing architectural components of the solver on different computing nodes.

An important aspect of the development of intelligent systems is to separate the database with domain knowledge into a separate component and present it in a declarative form (in the form of digraph). This ensures their understandable formation and modification. Pursuing the same goal, it is also necessary to form solvers (procedural knowledge) in the most declarative form possible. It is important to present the entire solver algorithm visually, to develop it from transparent components that implement the required processing of related domain concepts. Reusable components that implement solving of subproblems (like diagnostics, forecasting, etc.) are preferred.

To ensure transparency of the solver (procedural) components, their description is performed in terms of information necessary and sufficient for their operation, including input and output parameters, and the method of transferring control to them. For transparency of integration of procedural elements within a single solver, it is necessary to define sets of parameters and processed data, data and control dependencies, and, if necessary, the computing nodes on which they are placed and run.

The ICG structure includes related constructs (operators), one of which is marked as "beginning". These constructs can be divided into two main types: computational control constructs and information-processing constructs. As in all modern programming languages, the ICG can contain not only individual operators, but also so-called blocks that group operators for multiple calls, including recursion organization. In addition, the structure of the ICG includes a description of the data. The main class of constructs, the presence of which contributes to the creation of transparent intelligent problem solvers, is operations on an (ontological) information resource (resources). By the nature of processing, they are divided into those that: control the existence of an information resource (its creation and deletion) and manipulate the content of an information resource (in accordance with the generally accepted types of CRUD operations).

A special subtype of the operation for creation of a fragment of an information resource is the transformation of a fragment of an information resource (graph).

Due to the fact that KBS process related information resources that have a complex graph structure, the data type "fragment of an information resource" ("subgraph", "reference to a vertex") is added to the generally accepted processed data types (strings, numbers, boolean values, etc.). Accordingly, operations "value calculator" and "predicate value calculator" must support data processing of this type (reading vertex labels, navigating the vertex hierarchy, matching fragments of an information resource with a pattern, etc.).

The problem solver execution is provided by an ICG interpreter that works within a certain runtime environment (platform).

3. Conclusion

The authors have developed the IACPaaS cloud platform, which supports the creation of viable KBS as services [2]. According to the development technology [3, 4], the solver is created from declarative-procedural modules (agents) that perform information processing in accordance with specific types of causal and other relationships characteristic of the problem being solved. The specification (declarative description) of the problem solver sets the agent that starts the work. In the case of applying the described approach for the development of KBS based on ICG in its solver, the *ICG interpreter* agent must be specified as the "root" agent of the solver, and the ICG itself must be specified as a system parameter.

4. Acknowledgements

The work is partially financially supported by RFBR (grants nos. 19-07-00244 and 20-07-00670).

5. References

[1] Smirnov, P.A. Knowledge-based support for complex systems exploration in distributed problem solving environments / P.A. Smirnov, S.V. Kovalchuk, A.V. Boukhanovsky // Communications in Computer and Information Science. – 2013. – Vol. 394. – P. 147-161.

[2] Gribova, V.V. The development of viable intelligent systems with controlled declarative components / V.V. Gribova, Ph.M. Moskalenko, V.A. Timchenko, E.A. Shelfeeva // Information and mathematical technologies in science and control. – 2018. – Vol. 3(11). – P. 6-17.

[3] Gribova, V.V. Basic technology for intelligent service development on the IACPaaS cloud platform. Part 1. Development of a knowledge base and a problem solver // Software engineering. – 2015. – Vol. 12. – P. 3-11.

[4] Gribova, V.V. Basic technology for intelligent service development on the IACPaaS cloud platform. Part 2. Development of agents and of message templates // Software engineering. – 2016. – Vol. 1. – P. 14-20.