

The Use of Ontologies in the Integrated Graphical Environment

Valery A. Stennikov¹, Evgeny A. Barakhtenko², Dmitry V. Sokolov³

*Pipeline Energy Systems Department,
Melentiev Energy Systems Institute of Siberian Branch of the Russian Academy of Sciences,
Lermontov str., 130
Irkutsk, Russia*

¹*E-mail: sva@isem.irk.ru*

²*E-mail: barakhtenko@isem.irk.ru*

³*E-mail: sokolov_dv@isem.irk.ru*

Abstract

The paper presents a methodological approach to the construction of a system of ontologies to store the knowledge about energy pipeline systems, their properties, modeling problems related to these systems and the software employed. This system of ontologies consists of a metaontology and applied ontologies. The ontologies applied in the integrated graphical environment make it possible to automate the stages of the software development, fill the user interface with data and ensure effective operation with the model of pipeline system.

Keywords: Ontologies, applied ontologies, metaontology, methodological approach, automation of programming, integrated graphical environment.

1. Introduction

Nowadays, modern pipeline systems (i.e. heat, water, oil, gas and other systems) represent engineering structures that are unique in scale and complexity. They are becoming increasingly more important in the energy, economy, industry, residential-municipal sector and other spheres of the country and society. It is impossible to effectively solve the problems of their design, reconstruction, operation and dispatching control without calculations and information support of the decision making processes.

Researchers of Melentiev Energy Systems Institute of Siberian Branch of the Russian Academy of Sciences develop methodological approaches to the construction of integrated environments for computer modeling of pipeline systems. The integrated graphical environment represents a software package that can be used to solve applied information (work with the computer model of a pipeline system and its elements on the location plan in an interactive mode), calculation (solving the engineering problems) and analytical (analysis of graphs, tables and calculation results) problems within a single user interface. Such an environment should support the multi-level modeling, when the pipeline system model can be represented as a set of models of

its different subsystems (network, pumping stations, sources, etc.).

Pipeline systems of a certain type have their characteristic properties, a set of applied problems and specialized software designed to solve them. The knowledge about them should be stored in a form suitable for processing by software systems, and used to solve various problems. To this end, it is necessary to arrange the storage of this information in the form of ontologies [1-6]. In the knowledge engineering, the ontology is taken to mean a description of some subject domain which is used for formal and declarative definition of its conceptualization [7, 8]. During the automatic software design, we use the ontologies to formalize the description of objects of the subject domain, their properties and interactions between these objects. Nowadays, ontologies are widely used to solve the engineering problems. General issues in the field of development and use of ontologies are discussed in [9-12]. In [13-17], the authors present the approaches to the software development based on ontologies.

This paper is focused on the construction and practical application of the ontologies in the framework of the proposed approaches to the automated design of integrated graphical environments.

2. Specific features of ontology application

One of the specific features of the methodology for solving the problems of energy pipeline system modeling is that the implementation of the software intended for solving these problems is the final stage of the development of methods, mathematical models, techniques and algorithms. The application of this software to solve scientific and practical engineering problems leads to the accumulation of experience enabling us to develop more accurate mathematical models, specify reference data, increase the speed of algorithms, improve the convergence of methods, and obtain an original solution to some practical problems. Normally, the gained experience is fixed by introducing some changes in the software, which improves its quality and correspondence to real engineering systems. The described approach to the development leads to a situation where the software becomes the only tool for formalization and storage of all the accumulated experience. As a result, this experience is unavailable for a wide range of specialists to study and use it.

There are different types of pipeline systems. Each of the systems has their characteristic features, and their components are represented by different graphical and mathematical models. The new algorithms are developed and implemented within the applied software. This results in a great number of methodological and software developments. Their description should be made in a form suitable for multiple application.

To overcome the above difficulties it is necessary to switch to the software development paradigms within which the once- formalized knowledge will be used many times to develop and use the software. The Model Driven Engineering paradigm [18-21] makes it possible to devise methodological approaches to solve this problem. Within this approach, we propose making a formal description of knowledge about a subject domain in the form of ontologies, which will allow their repeated use in automation of the applied software construction.

The ontologies in the information graphical environment are used to solve the following problems:

- Automated construction of a program system that implements the integrated graphical environment;
- Input of information in the integrated graphical environment;
- Application of the integrated graphical environment to solve the applied problems.

3. A methodological approach to structuring the knowledge about a subject domain

We propose an original methodological approach to construction of a system of ontologies to store the knowledge about energy pipeline systems, their properties, modeling problems related to these systems, and employed software. The methodological approach includes the following components:

- A three-level representation of the subject domain content;
- The principles of structuring and formalizing the basic notions of the subject domain in the form of metaontology;
- The principles of structuring and formalizing the knowledge about energy pipeline systems in the form of pipeline system ontology;
- The principles of structuring and formalizing the knowledge about the modeling problems in the form of problem ontology;
- The principles of structuring and formalizing the knowledge about the used software in the form of software ontology;
- The principles of ontology development.

3.1. A three-level representation of the subject domain content

We have developed a three-level representation to demonstrate the structure and specific features of the application of ontologies, relations between them and objects defined by them (Fig. 1).

The upper level of the metaontology contains basic notions of the subject domain that are used to describe the knowledge about the energy facilities, applied problems and software. Then, at the next level, there are applied ontologies that contain a description of certain types of pipeline systems, their typical elements, used standard equipment, applied problems and used software. At the lowest level there are specific objects: computer models for specific pipeline systems, conceptual and mathematical statements of the applied problems, software with software support of methods,

Basic concepts	Metaontology		
	Applied ontologies	Pipeline systems ontologies (heat supply system, water supply system, gas supply system etc.)	Applied problem ontology
Objects	Computer model of pipeline system	Statement of applied problem	Software

Fig. 1. A three-level representation.

algorithms and models of pipeline system components (graphical and mathematical).

3.2. Structuring the basic notions of the subject domain

Metaontology contains the basic notions that are true for all the applied ontologies. The Fig. 2 presents the main part of this ontology. Any pipeline system is a system, and its modeling is related to one of the problems of the theory of hydraulic circuits. Therefore, corresponding software is used to solve this problem.

Fig. 3 demonstrates a fragment of metaontology from the “System” concept. The pipeline system consists of the components that are divided into nodes and branches and can be in one of the following physical states: operable, emergency, disconnected or under reconstruction. The nodes in turn are subdivided into source, connection and consumer nodes. The branches are subdivided into active and passive. The active branches contain equipment that actively affects the flow distribution in the network. The branches of this type include pumping stations and branches with

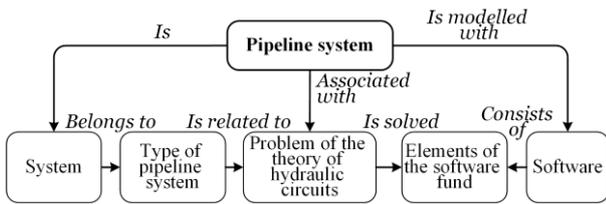


Fig. 2. A fragment of metaontology.

flow rate or pressure controllers. The passive branches contain pipeline sections of the pipeline systems. Each component has a mathematical model defining the laws of transported medium flow, and a graphical model that determines how it will be displayed.

Further, we will consider a structure of metaontology from the concept “Problem of the Theory of Hydraulic Circuits”. The problems of modeling the pipeline systems are subdivided into the classes including the problems of analysis, synthesis, identification and control. Each specific applied problem belongs to one of these classes. It has a name, a statement and a method of solving. The statement of the problem includes its formulation, description of input and output data. Solving a specific applied problem involves the methods, each of which has their algorithm and conditions for its application.

Now we will consider the part of metaontology that concerns software. The software consists of the software components of the integrated graphical environment and

software packages used to solve the applied problems. The software component has a name, a programming language, an access technique and interface. The software components of the integrated graphical environment are organized in the form of libraries that are subdivided into basic software components, models of pipeline system components, components of

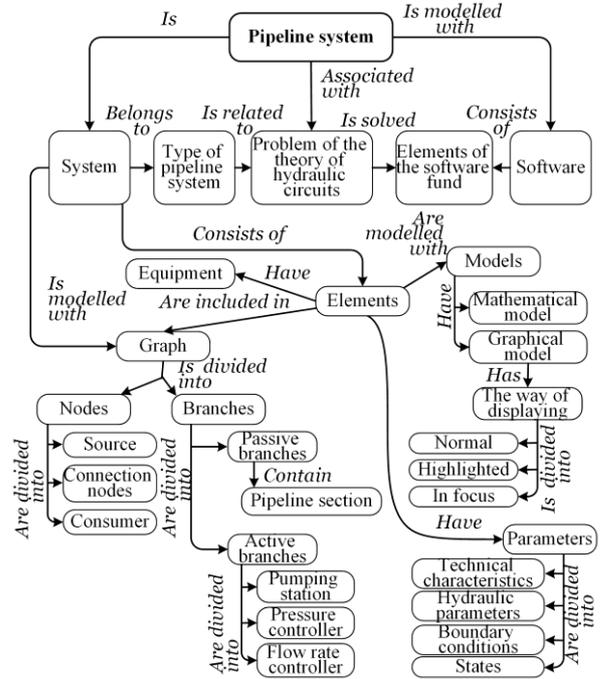


Fig. 3. A fragment of metaontology from the “System” concept.

graphical interface construction, adapter components, database access components, and layers of a geographic information system. The library of the basic software components includes data structures, validator components, graphical primitives and basic mathematical algorithms. The software packages are designed to solve a concrete applied problem, have an access method and work with a certain type of database management system.

3.3. Structuring the knowledge about energy pipeline systems

An integrated graphical environment for operation with a specific pipeline system is constructed in an automated mode based on the knowledge about the pipeline system of this type from a corresponding pipeline system ontology. The ontology of the pipeline system includes the description of the elements of this pipeline system and their properties. A fragment of the pipeline system ontology is presented in Fig. 4.

3.4. Structuring the knowledge about the modeling problems

Knowledge about on the problem is used to adjust the computer model of pipeline system to a concrete applied problem and automate the process of data preparation for the integrated graphical environment to call for the applied software. Storage of this knowledge is organized in the problem ontology that contains the description of the applied problems and methods for solving them, a description of algorithms, enumeration

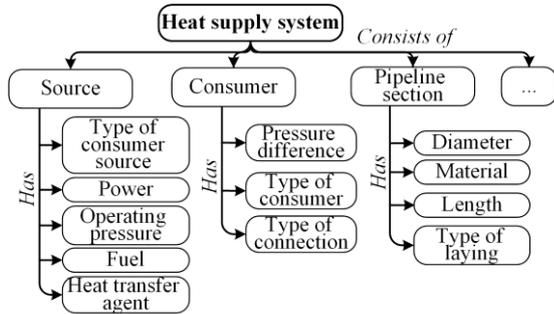


Fig. 4. A fragment of the pipeline system ontology.

of parameters that are input data and parameters to be obtained from the problem solving. Fig. 5 presents a fragment of this ontology that contains a description of the optimization problem of the pipeline system parameters.

3.5. Structuring the knowledge about the used software

The software ontology is intended for the storage of knowledge necessary to automate the construction and use of the software. This ontology contains the description of:

- The software intended for solving applied problems;
- The software used to construct the integrated graphical environment;

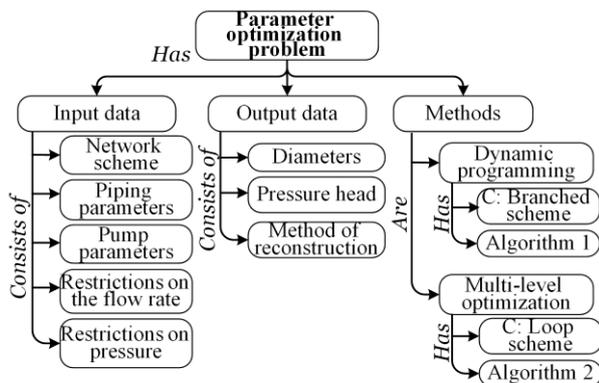


Fig. 5. A fragment of the problem ontology.

- Metadata (input and output parameters, description of data formats);
- Technologies and interfaces of access to the software components.

Fig. 6 presents a fragment of this ontology. It describes software and a software component for determining optimal pipeline system parameters. The software component is implemented in Java and has Command interface. The SOSNA software calls optimizer that solves the problem of determining optimal parameters. The software is accessed through the corresponding executable file. This software works with the data through the database-management system (DBMS) Firebird.

3.6. Ontology design principles

XML language is used as a tool for formal representation of ontologies in the framework of the proposed methodological approach. A specific feature of the proposed formal representation is the use of own language instead of standard ontology tools and languages. The reason for the development of own language is the fact that in this case the ontology has an applied character and is intended for automated design

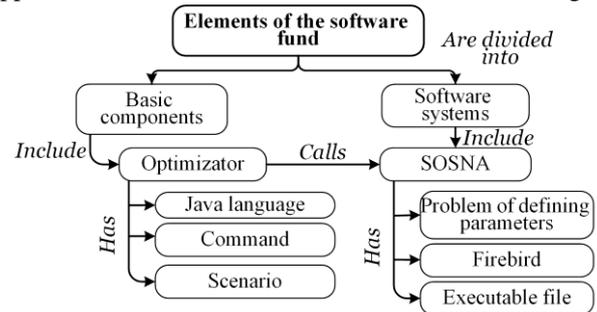


Fig. 6. A fragment of the software ontology.

of an integrated graphical environment and its information content. In this context, it is sensible to use own subject-oriented language. The SVG and MathML languages, being XML subsets, are used for representation of graphical and mathematical models, respectively.

4. Practical application

The proposed methodological approach is used to implement a software prototype of an integrated graphical environment for computer modeling of pipeline systems. The application of the ontologies ensured automated design of the integrated graphical environment and its information content. Fig. 7 presents

a scheme of the integrated graphical environment design and operation.

The developed prototype was successfully tested by solving the problems of determining optimal parameters of a heating system. The proposed approach ensured flexible adjustment of the software to a wide range of equipment in the modelled system. The integrated graphical environment prototype was employed in the calculations that made it possible to determine optimal parameters and make recommendations on rational

The developed methodological approach was applied to automate the design of the software system that represents a software prototype of the integrated graphical environment for computer modeling of energy pipeline systems of different types and purposes.

Acknowledgements

This work is supported by the Russian Foundation for Basic Research (grant No.16-07-00948).

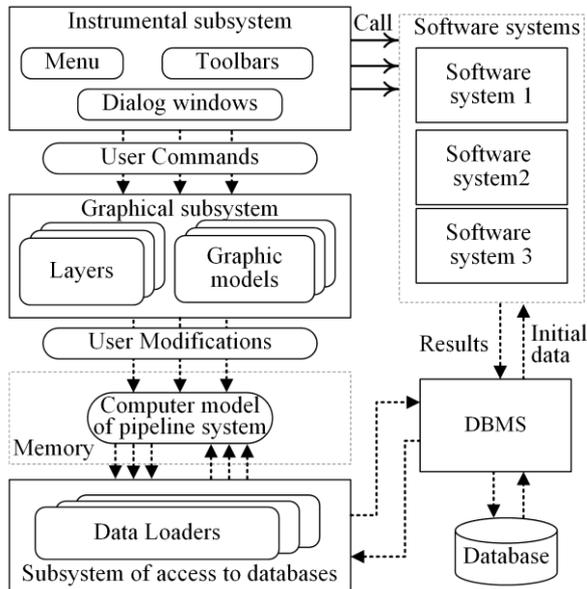


Fig. 7. A scheme of the integrated graphical environment design and operation.

reconstruction and expansion of heating systems in real cities.

5. Conclusion

Pipeline systems of a certain type (heat, water, oil, gas, etc.) have characteristics, a set of applied problems and specialized software used to solve these problems. Knowledge about them should be stored in a form suitable for their processing by software systems, and using to solve various problems. To this end, it is necessary to organize the storage of this knowledge in the form of ontologies, which can be used to automate software development stages, create the user interface content, and ensure effective work with the computer model of the pipeline system.

The paper proposes an original methodological approach to build a system of ontologies to store the knowledge about energy pipeline systems and their properties, applied problems related to these systems, and the software used.

References

1. V. A. Stennikov, E. A. Barakhtenko and D. V. Sokolov, Model-Driven Engineering in the Software for Parameter Optimization of Heat Supply Systems, *Software Engineering* **7**(3) (2016) 108–116.
2. V. A. Stennikov, E. A. Barakhtenko and D. V. Sokolov, Metaprogramming in the software for solving the problems of heat supply system schematic and parametric optimization, *Software Engineering* **6** (2011) 31–35.
3. V. A. Stennikov, E. A. Barakhtenko and D. V. Sokolov, Ontologies application for implementation of the software for solving optimal design problems of heat supply systems, *Information technologies* **3** (2013) 2–7.
4. V. A. Stennikov, E. A. Barakhtenko and D. V. Sokolov, A methodological approach on the basis of Model-Driven Engineering and ontologies for the development of a software for heat supply system design, *Information technologies* **21**(3) (2015) 201–209.
5. V. A. Stennikov, E. A. Barakhtenko, D. V. Sokolov and T. B. Oshchepkova, Problems of modeling and optimization of heat supply systems: new methods and software for optimization of heat supply system parameters, in *Sustaining Power Resources through Energy Optimization and Engineering* (IGI Global, Hershey PA, 2016), pp. 76–101.
6. V. A. Stennikov, E. A. Barakhtenko and D. V. Sokolov, Use of Multilevel Modeling for Determining Optimal Parameters of Heat Supply Systems, *Thermal Engineering* **64**(7) (2017) 518–525.
7. T. R. Gruber, A translation approach to portable ontology specifications, *J. Knowledge Acquisition*, **5**(2) (1993) 199–220.
8. S. Staab and R. Studer, *Handbook on Ontologies*, 2nd edn. (Springer-Verlag, Heidelberg, 2009).
9. M. Yudelson, T. Gavrilova and P. Brusilovsky, Towards User Modeling Meta-ontology, *Lecture Notes in Computer Science*, **3538** (2005) 448-452.
10. N. Guarino, Understanding, building and using ontologies, *International Journal of Human-Computer Studies* **46**(2–3) (1997) 293-310.
11. A. Gómez-Pérez, M. Fernández-López and O. Corcho, *Ontological Engineering with examples from the areas of Knowledge Management, e-Commerce and the Semantic Web*, 1st edn. (Springer-Verlag, London, 2004).
12. M. Uschold and M. Grüninger, Ontologies: Principles, methods, and applications, *J. The Knowledge Engineering Review* **11**(2) (1996) 93–113.
13. F. Ruiz and J. R. Hilera, Using ontologies in software engineering and technology, in *Ontologies in software engineering and software technology* (Springer-Verlag, Berlin, 2006), pp. 62–119.
14. S. Martin, *Applied Ontology Engineering in Cloud Services, Networks and Management Systems* (Springer-Verlag, New York, 2012).
15. D. Gaevi, D. Djuri and V. Deved, *Model Driven Architecture and Ontology Development* (Springer-Verlag, Berlin, 2006).
16. M.-C. Valiente, C. Vicente-Chicote and D. Rodríguez, An Ontology-Based and Model-Driven Approach for Designing IT Service Management Systems, *International Journal of Service Science, Management, Engineering, and Technology* **2**(2) (2011) 65-81.
17. S. Staab, T. Walter, G. Gröner, F. S. Parreiras, Model Driven Engineering with Ontology Technologies, *Lecture Notes in Computer Science* **6325** (2010) 62-98.
18. D. C. Schmidt, Guest Editor's Introduction: Model-Driven Engineering, *Computer* **39**(2) (2006) 25-31.
19. M. Volter, T. Stahl, J. Bettin, A. Haase and S. Helsen, *Model-Driven Software Development: Technology, Engineering, Management* (Wiley, New York, 2006).
20. M. Brambilla, J. Cabot and M. Wimmer, *Model Driven Software Engineering in Practice. Synthesis Lectures on Software Engineering* (Morgan & Claypool, San Rafael, 2012).
21. A. R. Silva, Model-driven engineering: A survey supported by the unified conceptual model, *Computer Languages, Systems & Structures* **43** (2015) 139-155.