AN ANALYSIS OF SOME RELATIONS AMONG DOMAIN ONTOLOGIES[1]

Alexander Kleshchev, Irene Artemjeva

Abstract: *Building domain ontologies and applying them to different objectives, researchers faced the fact that many ontologies are associated with one another by one or another relations. Therefore, the problem arose to study relations among different ontologies of the same domains as well as of different ones. A formalization of a relation among domain ontologies is the analogous mathematical relation among mathematical models of these ontologies. The article considers the case when domain ontology model is represented by logical relationship system. Relations among domain ontologies give a possibility to reuse one ontology model when another ontology models are worked out and when new intellectual computer system for same or different domain is worked out.*

Keywords: Mathematical model of domain ontology, ontologies representing the same conceptualisation, resemblance between ontologies, simplification of ontologies, composition of ontologies, intellectual task solver.

ACM Classification Keywords: I.2.4 Knowledge Representation Formalisms and Methods, F4.1. Mathematical Logic

Introduction

Building domain ontologies and applying them to different objectives, researchers faced the fact that many ontologies are associated with one another by one or another relations. Therefore, the problem arose to study relations among different ontologies of the same domains as well as of different ones. Although, as noted in [van Heijst et al, 1996], the field is still in its infancy and many questions are unsolved or even unaddressed (for example, how can ontologies be compared and integrated?), by now there has been some information in professional literature related to this problem. Many works studying this problem considered relations among ontologies within the context of ontology integration.

In [Gangemi et al, 1999] ontology integration is defined as the construction of an ontology C that formally specifies the union of the vocabularies of two other ontologies A and B. Three aspects of an ontology are taken into account: (a) the intended models of the conceptualisations of its vocabulary, (b) the domain of interest of such models, i.e. the topic of the ontology, and (c) the namespace of the ontology. The most interesting case is when A and B are supposed to commit to the conceptualization of the same domain of interest or of two overlapping domains. In particular, A and B may be:

Alternative ontologies: `The intended models of the conceptualizations of A and B are different (they partially overlap or are completely disjoint) while the domain of interest is (mostly) the same. This is a typical case that requires integration: different descriptions of the same topic are to be integrated.

Truly overlapping ontologies: Both the intended models of the conceptualisations of A and B and their domains of interest have a substantial overlap. This is another frequent case of required integration: descriptions of strongly related topics are to be integrated.

Equivalent ontologies with vocabulary mismatches: The intended models of the conceptualisations of A and B are the same, as well as the domain of interest, but the namespaces of A and B are overlapping or disjoint. This is the case of equivalent theories with alternative vocabularies.

Overlapping ontologies with disjoint domains: The intended models of the conceptualizations of A and B overlap while the domain of interest are disjoint. This concerns overlapping theories with different extensions. Actually,

1 This paper was made according to the program of fundamental scientific research of the Presidium of the Russian Academy of Sciences «Mathematical simulation and intellectual systems», the project "Theoretical foundation of the intellectual systems based on ontologies for intellectual support of scientific researches".

it is often the case that some fragments from an ontology A can be reused as components in another ontology B that models a different topic.

Homonymically overlapping ontologies: The intended models of the conceptualizations of A and B do not overlap, but A and B overlap. This is the case of two unrelated ontologies with a vocabulary intersection that – if presented – generates polysemy: this is one of the reasons to maintain ontology modules.

To be sure that A and B can be integrated at some level, C has to commit to both A's and B's conceptualizations. In other words, the intention of the concepts in A and B should be mapped to the intention of C's concepts. The authors call this approach principled conceptual integration.

As noted in [Gangemi et al, 1996], the ontological integration envisaged is at a deeper level than representational integration. In fact, the representational integration concerns heterogeneity of formal languages, or heterogeneity of data base schemata. Ontological integration concerns the heterogeneity among conceptualizations.

In [Guarino, 1998] it is noted that information integration is a major application area for ontologies. As well known, even if two systems adopt the same vocabulary, there is no guarantee that they can agree on a certain piece of information unless they commit to the same conceptualization. Assuming that each system has its own conceptualization, a necessary condition to make an agreement possible is that the intended models of the original conceptualizations overlap. Supposing now that these two sets of intended models are approximated by two different ontologies, it may be the case that the two ontologies overlap while the intended models do not. Hence, it seems more convenient to agree on a single top-level ontology rather than relying on agreements based on the intersection of different ontologies.

In [Sowa] ontology integration is defined as the process of finding commonalities between two different ontologies A and B and deriving a new ontology C that facilitates interoperability between computer systems that are based on the A and B ontologies. The new ontology C may replace A or B, or it may be used only as an intermediary between a system based on A and a system based on B. Depending on the amount of change necessary to derive C from A and B, different levels of integration can be distinguished: alignment, partial compatibility, and unification.

Alignment is a mapping of concepts and relations between two ontologies A and B that preserves the partial ordering by subtypes in both A and B. If an alignment maps a concept or relation x in ontology A to a concept or relation y in ontology B, then x and y are said to be equivalent. The mapping may be partial: there could be many concepts in A or B that have no equivalents in the other ontology. Before two ontologies A and B can be aligned, it may be necessary to introduce new subtypes or supertypes of concepts or relations in either A or B in order to provide suitable targets for alignment. No other changes to the axioms, definitions, proofs, or computations in either A or B are made during the process of alignment. Alignment does not depend on the choice of names in either ontology. For example, an alignment of a Japanese ontology to an English ontology might map the Japanese concept Go to the English concept Five. Meanwhile, the English concept for the verb go would not have any association with the Japanese concept Go.

Partial compatibility is an alignment of two ontologies A and B that supports equivalent inferences and computations on all equivalent concepts and relations. If A and B are partially compatible, then any inference or computation that can be expressed in one ontology using only the aligned concepts and relations can be translated to an equivalent inference or computation in the other ontology.

Refinement is an alignment of every category of an ontology A to some category of another ontology B, which is called a refinement of A. Every category in A must correspond to an equivalent category in B, but some primitives of A might be equivalent to non-primitives in B. Refinement defines a partial ordering of ontologies: if B is a refinement of A, and C is a refinement of B, then C is a refinement of A; if two ontologies are refinements of each other, then they must be isomorphic.

Unification is a one-to-one alignment of all concepts and relations in two ontologies that allows any inference or computation expressed in the one to be mapped to an equivalent inference or computation in the other. The usual way of unifying two ontologies is to refine each of them to more detailed ontologies whose categories are one-to-one equivalent.

Alignment is the weakest form of integration: it requires minimal change, but it can only support limited kinds of interoperability. It is useful for classification and information retrieval, but it does not support deep inferences and computations. Partial compatibility requires more changes in order to support more extensive interoperability, even though there may be some concepts or relations in one system or the other that could create obstacles to full interoperability. Unification or total compatibility may require extensive changes or major reorganizations of A and B, but it can result in the most complete interoperability: everything that can be done with one can be done in an exactly equivalent way with the other.

In [Wielinga et at, 1994] more general and more special ontologies are considered. Ontologies can have a recursive structure, meaning that ontology expresses a viewpoint on another ontology. Such a viewpoint entails a reformulation and/or reinterpretation on other ontology. This multi-level organization raises research questions such as the required expressiveness of the mapping formalisms for expressing viewpoints between ontologies. At least two different mapping operations can be identified. The first one is the mapping of terminology in one formalism onto the terminology of another formalism. The other one is the adding of supplementary commitments to one ontology by the mapping of the terms of the ontology onto the terms of the other ontology that takes additional commitments. The first terminology mapping will occur frequently. Since the ontology describes the meaning of the domain theory, for which it is a meta-model, without commitment to the language, in which this meaning is expressed, it will be confronted with meta-models, which partially convey the same meaning, but with different terminology. In this case merging of the two ontologies, or translation of the one ontology into the other is simply a mapping of terminology (e.g. boat in one ontology can be mapped on ship in another ontology if they refer to the same type of object in the universe of discourse (note that the knowledge bases described by these ontologies, even when they describe the same object in the real world, may be totally different!)). The second type of mapping occurs when it is necessary to provide an interpretation of underlying ontology or to provide a more specific interpretation that takes additional commitments. If the more restrictive ontology is already available (such as, sometimes, the ontology of a task or of a method) than it is necessary to map this ontology on the more general one. An example of this type of mapping occurs when there exists a model of the problem-solving task, that should be accomplished, and an existing ontology of the domain of the application. In this case, it is necessary to map terminology from the task (e.g. hypothesis) on terminology of the domain ontology. A simple mapping will not always be possible. Sometimes the ontology - introducing the additional commitments - needs to be constructed. This will often be the case with domain-model oriented ontologies.

In [Laresgoiti et al] and [Schreiber et al] a combination of ontologies is introduced. An example of some artifact such as a ship is considered. One can define multiple viewpoints on a ship. Well-known examples of such viewpoints are the physical structures (what are the parts of a ship?) and the functional structure (how can a ship be decomposed in terms of functional properties?). Although these two viewpoints often partially overlap, they constitute two distinct ways of “looking” at a ship. The purpose of ontology is to make those viewpoints explicit. For a design application such as CAD application, one would typically need a combined physical/functional viewpoint: a combination of two ontologies. For a simulation application (e.g. modeling the behavior of a ship), one would need an additional behavioral viewpoint. Many other viewpoints exist such as the process type in the artifact (heat, flow, energy, ...). Each ontology introduces a number of specific conceptualizations, that allow an application developer to describe, for example, a heat exchange process.

In [Studer et al, 1998] constructing ontologies from reusable ontologies is considered. Assuming that the world is full of well-designed modular ontologies, constructing a new ontology is a matter of assembling existing ones. There are several ways to combine ontologies. In [Studer et al, 1998] the most frequently occurring ones are only given. The simplest way to combine ontologies is through inclusion. Inclusion of one ontology into another has the effect that the composed ontology consists of the union of the two ontologies (their classes, relations, axioms). In other words, the starting ontology is extended with the included ontology. Conflicts between names have to be resolved. Another way to combine ontologies is by restriction. This means that the added ontology only is applied on a restricted subset of what it was originally designed for. The last way to assemble ontologies that is discussed in [Studer et al, 1998] is polymorphic refinement, known from object-oriented approaches.

It is possible to make some conclusions from this overview.

Many authors consider supporting interoperability as a main objective of ontology integration. But if this objective is reached, then it is not clear, what properties integrated ontologies and the result of their integration will have. Before studying these relations and building their formal models, it seems necessary to declare the fundamental properties, that all the relations among ontologies will have.

Consideration of overlapping but different conceptualizations as a necessary condition for possibility of ontology integration seems slightly speculative. If a conceptualization is adequate [Kleshchev et al, 2000a], then it must include the domain reality. In this case, the reality must be a subset of the intersection of these conceptualizations. But the conceptualization that is their intersection is adequate, too. And any top-level conceptualization is worse (wider) than initial ones and especially than their intersection.

Vocabularies (concept systems) are only external structures, by which sets of intended situations, sets of intended knowledge systems and correspondences between them are expressed. Thus, it is unlikely that the union of the vocabularies can be considered as a principal property of ontology integration.

In the same way a mapping of concepts between two ontologies can be but one of ways to determine relations between ontologies. This way cannot be always applied to do this. If there is a mapping between concepts of two ontologies, then this fact alone does not allow us yet to call corresponding concepts as equivalent. The notion of equivalence is defined in mathematics as reflexive, symmetric and transitive relation.

When defining relations among ontologies, any references to properties of inferences or computations cannot be considered as admissible because they darken rather than clarify the meaning of introduced relations. The condition that all the inferences or computations are equivalent cannot be verified.

Properties of Relations among Domain Ontologies

Any domain is characterized by its reality, i.e. by the set of all the possible situations that have ever taken place in the past, are taking place now and will take place in the future [Kleshchev et al, 2000a]. Since the reality is known only partially, the domain knowledge system gives a more comprehensive idea of it. The knowledge system determines the set of situations admitted by the system, i.e. of such situations that are considered as possible in the reality by this knowledge system. So an observer comes across only situations of the reality, but a person possessing a knowledge system is able to imagine situations admitted by the knowledge system. Where does he or she take these imaginary situations from? They are determined by a conceptualization, that can be imagined as the implicitly given set of all the intended situations, i.e. all the situations which can be imagined within the framework of this conceptualization. In this case, the set of the situations admitted by a knowledge system is a subset of the set of all the intended situations.

An investigation of a domain, i.e. of its reality, is aimed at obtaining such a knowledge system that admits the set of situations being as near to the reality as possible. So the set of the situations admitted by a knowledge system is considered as an approximation of the reality, and the investigation of the domain is aimed at obtaining the best (the most adequate) approximation of its reality. This investigation perpetually gives birth to new knowledge systems instead of outdated ones. Where does these knowledge systems come from? They are determined by a conceptualization, too. So a conceptualization can be imagined also as the implicitly given set of all the intended knowledge systems, i.e. of such knowledge systems that can be formed within the framework of the concept system introduced by the conceptualization.

Ontology of a domain is an explicit representation of a conceptualization of the domain. Since the ontology can represent the conceptualization imprecisely, it determines two external approximations both for the set of all the intended situations and for the set of all the intended knowledge systems.

A relation among knowledge systems of the same or different domains is a relation defined on the sets of the situations admitted by these knowledge systems. If this relation takes place among these knowledge systems, and another, more adequate, knowledge system is found instead of one of them, then, in the general case, this relation does not have to take place among the renewed collection of knowledge systems. But from practical needs, it is quite desirable to have a possibility to determine with what other knowledge systems the new knowledge system is in the same relation.

A relation among ontologies of the same or different domains is a relation defined on the sets of all the intended knowledge systems of these ontologies (i.e. a subset of the Cartesian product of these sets) possessing the property that only the tuples consisting of knowledge systems belong to the relation that are in the analogous relation. Thus, if relations among ontologies are determined, then it determines the analogous relation among all the intended knowledge systems of these ontologies. In this article the relations possessing this property are considered only.

A formalization of a relation among domain ontologies is the analogous mathematical relation among mathematical models of these ontologies. The article considers the case when domain ontology model is represented by logical relationship system [Kleshchev, 2000a, 200b].

Ontologies Representing the Same Conceptualization

Domain ontology is a collection of agreements. It defines domain terms, determines their interpretations, contains statements that restrict the meaning of these terms and also gives interpretations for these statements. These agreements are the result of understanding among some members of the community working in this domain [Kleshchev et al, 2000a]. Different members of this community can advance different ontologies of this domain. The question arises: do these ontologies represent the same conceptualization or different ones? Let us discuss this question on the assumption that the models of these ontologies have the form of unenriched logical relationship systems [Kleshchev et al, 2000b].

If a conceptualization is considered as a set of all the intended situations, then two ontologies can represent the same conceptualization only when the sets of terms for situation description in these ontologies are the same. If a conceptualization is considered as a set of all the intended knowledge systems, then two ontologies can represent the same conceptualization only when the sets of terms for knowledge description in these ontologies are the same (they can be empty sets).

Let us consider the case when two different ontologies have the same sets of terms for situation description as well as the same sets of terms for knowledge description. In this case, to be different, these ontologies must have different sets of ontological agreements. Two points of view are possible on the condition under that these ontologies represent the same conceptualization: (1) when both the sets of intended situations and the sets of intended knowledge systems determined by these ontologies are the same; (2) when, following the definitions of the previous section, the sets of intended knowledge systems determined by these ontologies are the same, and for any knowledge system the sets of situations admitted by this knowledge system in these two ontologies are also the same.

The models of these ontologies have the same sets of unknowns and the same sets of parameters but different sets of logical relationships. Formalization of the conditions above means that:

1. the sets of logical relationships for the models of these ontologies are equivalent as applied logical theories (two applied logical theories are equivalent, if they have the same set of models [Kleshchev et al, 2000b]);
2. the models of this domain determined by the models of these ontologies for the same knowledge model have the same models of the reality, i.e. the models of these ontologies are equivalent as unenriched logical relationship systems [Kleshchev et al, 2000b].

It is easily seen that both these conditions are equivalent. Thus, equivalent transformations of the logical relationship set for a domain ontology model (as an applied logical theory) lead to a model of another ontology representing the same conceptualization. These transformations can be, for example, transformation of an applied logical theory to a disjunctive normal form, a conjunctive normal form and so on.

Now let us consider the case when two ontologies of the same domain have the same sets of terms for situation description but different sets of terms for knowledge description. In this case, following the previous section, we can consider these ontologies as representing the same conceptualization, if there is a one-to-one correspondence between their knowledge system sets, and for any corresponding knowledge systems the sets of the situations admitted by these knowledge systems are the same. When passing to models, it means that the models of these ontologies are equivalent [Kleshchev et al, 2000b].

Now let us consider the case when two ontologies of the same domain have different sets of terms for situation description but the same sets of terms for knowledge description. In this case, following the previous section, we can consider these ontologies as representing the same conceptualization if for any knowledge system there is a one-to-one correspondence between the sets of the situations admitted by this knowledge system in both these ontologies. When passing to models, it means that the models of these ontologies have the same sets of all possible enrichments and are isomorphic [Kleshchev et al, 2000b].

Resemblance between Ontologies

In the case when both terms for situation description and terms for knowledge description are different in two ontologies, it is possible to speak of resemblance between these ontologies only (of the same or different domains).

Two knowledge systems related to different ontologies (of the same or different domains) can be considered as resembled if there is a one-to-one correspondence between the sets of situations admitted by these knowledge systems. So two ontologies of the same or different domains can be considered as resembled if there is such a one-to-one correspondence between their sets of intended knowledge systems that any corresponding knowledge systems are resembled. It means that the models of these ontologies are isomorphic [Kleshchev et al, 2000b].

If terms of an ontology are substituted by different terms (by abstract designations), then, as a result, a resembled ontology will be obtained. The resemblance between ontologies is a relation of equivalence. It is reflexive, symmetric and transitive.

Simplification (Coarsening) of Ontologies

Comparing different ontologies of the same domain, one can sometimes say that one of these ontologies is a simplification (coarsening) of another. In the same way considering ontologies of different domains, one can sometimes say that an ontology of one of these domains resembles a simplified ontology of another domain. The availability of more simple and more complex ontologies of the same domain can be important to develop knowledge based systems for specialists of different qualifications (for example, medical systems for physicians of high qualification and for doctor’s assistants).

One can say that a knowledge system related to an ontology is more simple than a knowledge system related to another ontology (of the same or different domains) if for every situation admitted by the second knowledge system (of the more complex ontology) the only situation admitted by the first knowledge system (of the more simple ontology) can be set as corresponding. Then one can consider an ontology as more simple than another ontology (of the same or different domains) if for every knowledge system of the second ontology the only more simple knowledge system of the first ontology can be set as corresponding. It means that a model of the first ontology is a homomorphic image of the second ontology [Kleshchev et al, 2000b].

A domain model <O1, k2> is a simplification (coarsening) of a domain model <O1, k1> if the enriched logical relationship system <O1, k2> is a homomorphic image of the system <O1, k1>. A coarsened model of medical diagnostics can be obtained, for example, by elimination of a few signs.

The simplification determines a partial order of ontologies. If B is more simple than A, and C is more simple than B, then C is more simple than A. If one ontology is simpler than another, and the second ontology is simpler than the first ontology, then they resemble one another.

Composition of Ontologies

When we speak about complex domains, we must usually bear in mind that these domains include knowledge from other different domains. Thus, when knowledge and reality of complex domains are described, concepts related to other domains are used. These other domains are components of the complex domain. Ontologies of complex domains are built from components, which are ontologies of other domains.

We can consider that a (starting) knowledge system related to a complex domain consists of knowledge systems (components) related to other domains if every component is more simple than the starting knowledge system, and the transfer from any situation admitted by the starting knowledge system to corresponding situations admitted by components takes place without the loss of information. The latter statement means that for any two different situations admitted by the starting knowledge system the two sets consisting of the situations corresponding to these two situations and admitted by all the components are different. In this case a starting ontology of a complex domain can be considered as consisting of components which are ontologies of other domains if every component is more simple than the starting ontology, every knowledge system of the starting ontology consists of knowledge systems of components, and the transfer from any knowledge system of the starting ontology to corresponding knowledge systems of components takes place without loss of information. The latter statement means that for any two different knowledge systems of the starting ontology the two sets consisting of the knowledge systems of components corresponding to these two knowledge systems are different. It follows from these definitions that every model of a starting ontology for a complex domain is the product of ontology models that are components [Kleshchev et al, 2000b].

Using Relations among Domain Ontologies for Working out Intellectual Solvers for Applied tasks

At present time, a demand arises to develop program systems for different domains having means for adaptation of problem solving methods to alteration of knowledge in these domains. Such program systems are called the intellectual solvers for applied problems. The base of developing an intellectual solver is domain ontology. Intellectual problem solvers on domain should permit experts and specialists to form and edit ontology and knowledge on domain and to get the programs for solving applied problems in this domain.

If there are alternative points of view on the same domain then we can speak about equivalence or resemblance between different ontologies of the domain. Recognition of the equivalence between alternative points of view on the same domain can give a possibility to solve tasks arising within the framework of a point of view using methods worked out within the framework of another point of view. Recognition of a resemblance between ontologies of the same domain can give a possibility to solve the tasks described within the framework of one concept system by methods developed within the framework of the other concept system. Recognition of a resemblance between ontologies of different domains can give a possibility to solve the tasks of one domain by reasoning using analogy in the case when methods for solving analogous tasks of the other domain have been developed.

A mathematical specification of an applied task can contain a domain model, input and output data of the task, task conditions (a set of formulas), and also criterion of selecting solutions. All the components of the applied task specification are represented in terms of the domain model. If every value of input data is replaced by a variable (different variables correspond to different values) in the task specification then the mathematical specification of the task will be transformed into a mathematical specification of a class of applied tasks. These variables will be called variables of the class of applied tasks. There is a one-to-one correspondence between the set of tasks belonging to the class and the set of all the admissible substitutions of values instead of these variables. To get the mathematical specification of an applied task belonging to a class it is necessary to replace all the variables of the class by values of input data.

If the domain model is replaced by the domain ontology model and knowledge base of the domain are considered as another set of input data of all the tasks of the class then the mathematical specification of the class of applied tasks will be transformed into the mathematical specification of the class of applied tasks corresponding to the domain ontology. There is a one-to-one correspondence between the set of tasks belonging to the class and the Cartesian product of the set of all the admissible substitutions of values instead of variables of the class of the tasks by the set of all the possible knowledge bases for the domain ontology model. To get the mathematical specification of an applied task belonging to a class of tasks corresponding the domain ontology it is necessary to replace all the variables of the class by values of input data and to enrich the domain ontology model by an appropriate knowledge base.

Finally, if domain terms in the mathematical specification of the class of applied tasks corresponding to a domain ontology are replaced by abstract designations then this mathematical specification of the class will be transformed into a mathematical task. The transformation of a mathematical specification of a class of applied tasks corresponding to a domain ontology into a mathematical task is important because different classes of applied tasks corresponding to ontologies of different domains, generally speaking, can be reduced to the same mathematical task.

If intellectual solver can solve mathematical tasks then it can be used for any domain which ontology model is isomorphic or equivalent to ontology model from mathematical task specification.

Let's consider a set of mathematical specifications of applied tasks such that every specification contains the same domain model. Such a set will be called an applied multitask. Just as an applied task was transformed into a class of applied tasks, the latter was transformed into a class of applied tasks corresponding to a domain ontology, and the latter was transformed into a mathematical task, so an applied multitask can be transformed into a class of applied multitasks, the latter can be transformed into a class of applied multitasks corresponding to a domain ontology, and the latter can be transformed into a mathematical multitask. An intellectual solver is intended for solving applied multitasks of a class of applied multitasks or for solving applied multitasks of a class of applied multitasks corresponding to a domain ontology.

The availability of simpler and more complex ontologies of the same domain can be important to develop intellectual solvers for specialists of different qualifications. As this takes place, working out methods for solving tasks based on a more simple ontology can be a simplification of methods for solving the corresponding tasks based on a more complex ontology. The same can also take place for ontologies of different domains.

The same methods often can be used for solving a few tasks and subtasks. Abstraction of applied tasks to mathematical ones gives a possibility of reusing methods for their solving. If different applied tasks can be reduced to the same mathematical task then a method for solving the mathematical task can be used for solving these applied tasks too. A decomposition of a mathematical task into mathematical subtasks in working out a method for solving the mathematical task gives an additional possibility for reusing methods. In this case, the same mathematical subtasks can be components of decompositions of different mathematical tasks and methods for solving these subtasks can be components of methods for solving different mathematical tasks.

Ontologies of complex domains are built from components, which are ontologies of other domains. The fact that an ontology of a complex domain is a composition of other domain ontologies can be used to work out methods for solving tasks in the complex domain. These tasks can be divided into subtasks corresponding to tasks for components of the ontology. If methods for solving these tasks have been already known, working out a method for solving the whole task may be considerably simplified.

Conclusions

In this article, general properties of relations among domain ontologies have been considered. Examples of these relations can be the relation between ontologies representing the same domain conceptualization, the relation of resemblance between ontologies, the relation “to be more simple or more complex” and the relation among an ontology consisting of components, which are other ontologies, and these components. A formalization of these relations has been suggested. This formalization preserves the properties above. These results show that the definitions of an ontology and its model given in [Kleshchev et al, 2000a] allow us to recognize these relations among ontologies. Relations among domain ontologies give a possibility to reuse one ontology model when another ontology models are worked out and when new intellectual computer system for same or different domain is worked out.

References

[van Heijst et al, 1996] van Heijst G., Schreiber A.T., and Wielinga B.J. Using Explicit Ontologies in KBS Development. In International Journal of Human and Computer Studies, 1996, 46 (2-3): 183-292.

[Gangemi et al, 1999] Gangemi A., Pisanelli D.M. and Steve G. An Overview of the ONIONS Project: Applying Ontologies to the Integration of Medical Terminologies // In Data & knowledge Engineering, Vol. 31, N 2, 1999, pp. 183–220

[Gangemi et al, 1996] A. Gangemi, G.Steve, F. Giacomelli. ONIONS: An Ontological Methodology for Taxonomic Knowledge Integration. In P. van der Vet (ed.) Proceedings of the Workshop on Ontological Engineering, ECAI96, 1996.

[Guarino, 1998] Guarino N. Formal Ontology and Information systems. In Proceeding of International Conference on Formal Ontology in Information Systems (FOIS’98), N. Guarino (ed.), Trento, Italy, June 6-8, 1998. Amsterdam, IOS Press, pp. 3-15.

[Kleshchev et al, 2000a] Kleshchev A.S., Artemjeva I.L. Mathematical Models of Domain Ontologies. Technical Report 18-2000. Vladivostok, 2000. 43 p. (available in http://iacp.dvo.ru/es/)

[Kleshchev et al, 2000b] Kleshchev A.S., Artemjeva I.L. Unenriched Logical Relationship Systems. Technical Report 1-2000. Vladivostok, 2000. 43 p. (available in http://iacp.dvo.ru/es/)

[Laresgoiti et al] L. Laresgoiti, A. Anjewierden, A. Bernaras, J. Corera, A.Th.Schreiber, B.J.Wielinga. Ontologies as Vehicles for Reuse: a Mini–expirement. Available from http://ksi.cpsc.ucalgary.ca/KAW/KAW96/laresgoiti/k.html

[Schreiber et al] G.Schreiber, W.Jansweijer, B.Wielinga. Framework & Formalism for expressing Ontologies (version 2). Technical Report, University of Amsterdam, DO1b2. http://www.swi.psy.uva.nl/projects/NewKACTUS/Reports.html

[Sowa] Sowa J., Knowledge Representation: Logical, Philosophical and Computational Foundations. In http://www.bestweb.net/sowa/ontology/gloss.htm

[Studer et al, 1998] R Studer, V.R. Benjamins, D. Fensel. Knowledge Engineering: Principles and methods. In Data & Knowledge Engineering 25, 1998, p 161–197

[Wielinga et at, 1994] Wielinga, B., Schreiber A.T., Jansweijer W., Anjewierden A. and van Harmelen F. Framework and Formalism for Expressing Ontologies (version 1). ESPRIT Project 8145 KACTUS, Free University of Amsterdam deliverable, DO1b.1, 1994. http://www.swi.psy.uva.nl/projects/NewKACTUS/Reports.html

Author's Information

Alexander S. Kleshchev – kleschev@iacp.dvo.ru

Irene L. Artemjeva – artemeva@iacp.dvo.ru

Institute for Automation & Control Processes, Far Eastern Branch of the Russian Academy of Sciences

5 Radio Street, Vladivostok, Russia

METHODS OF ADAPTIVE EXTRACTION AND ANALYSIS OF KNOWLEDGE FOR KNOWLEDGE-BASE CONSTRUCTION AND FAST DECISION MAKING

Alexander Kuzemin, Darya Fastova, Igor Yanchevsky

Abstract: An approach for knowledge extraction from the information arriving to the knowledge base input and also new knowledge distribution over knowledge subsets already present in the knowledge base is developed. It is also necessary to realize the knowledge transform into parameters (data) of the model for the following decision-making on the given subset. It is assumed to realize the decision-making with the fuzzy sets' apparatus.

ACM Classification Keywords: I.2.5 - Expert Systems; I.2.6 - Knowledge acquisition

Introduction

The problem of knowledge representation in the process of the expert system (ES) design is the central one as the knowledge base (KB) main function implementation, i.e. new knowledge gaining, depends on its successful solution. Starting from this the structure and form of models and methods for knowledge representation making the decisive action on the ES efficiency and external information

In the majority experts' opinion the expert system power is defined by the volume of the knowledge the given system offers. Despite the fact that a lot of instrumental means helpful in gaining knowledge has appeared recently this problem still remains poorly defined and laborious one. Knowledge gaining is inseparably connected with the process of their check-out consisting in detection of insufficient knowledge and their introduction to the system, the KB check on non-inconsistency and completeness, check of the managing mechanism, the ES analysis and modification.

The process of compatible ES development implies creation of the specialized instrumental systems. Such systems support execution of the life cycle main stages, they commonly fix presentation of the used information, the knowledge presentation language, the knowledge interpreter (display) and a set of software instruments intended for a number of problems solution. However, these systems are oriented to the support of a user from the knowledge engineer class [2,3,4] and not information carriers (experts). Thus, they do not take into account a modern approach to creation of means for information processing (MIP), which consists in exclusion of a knowledge engineer, from this process as a redundant mediator.