Representation of formal ontologies of basic entity classes in intelligent computer systems

Stanislau Butrin
Belarusian State University of Informatics and Radioelectronics
Minsk, Belarus
Email: stas.butrin1331@gmail.com

***Abstract*—In the article, the ontological approach to the design of knowledge bases of next-generation intelligent computer systems is considered. The main subject domains and ontologies containing the description of basic entity classes are allocated. The connections and relations between the basic concept classes are described. The results obtained will improve the efficiency of designing knowledge bases of intelligent computer systems.**

***Keywords*—knowledge base, ontology, top-level ontology, ontological approach to designing knowledge bases, basic entity class, subject domain.**

I. Introduction

The number of application fields for various computer systems increases every year together with the number of intelligent problems that require automation. This leads to the need to improve intelligent computer systems and expand their functionality.

A key element of such systems is knowledge bases, which largely determine the level of their intelligence [1].

A knowledge base is a systematized totality of knowledge stored in the memory of an intelligent computer system and sufficient to ensure purposeful (appropriate, adequate) functioning (behavior) of this system both in its external and internal environment (in its own knowledge base) [2].

To ensure the joint usage of different types of knowledge included in the knowledge base, it is necessary to ensure their compatibility with the specified knowledge base, which includes semantic compatibility that implies an unambiguous and unified interpretation of the used concepts for all fragments of the knowledge base.

II. Analysis of existing approaches to solving the problem

Among the variety of means for knowledge representation, the most effective are ontologies [3]. The essence of this approach when designing the knowledge base is to consider the knowledge base as a hierarchical system of selected subject domains and their corresponding ontologies. However, ontologically, it is possible to specify knowledge in different ways. To solve this problem, top-level ontologies are designed.

Let us consider the currently available top-level ontologies [4], [5]:

- **The Standard Upper Ontology** (SUMO) [6]
  - The key concept in the SUMO ontology is "Entity", and this concept includes the "Physical" and the “Abstract” concepts. The first category includes everything that has a position in space and time, and the second category includes all the rest.
  - The ontology covers the following areas of knowledge: general kinds of processes and objects, abstractions (set theory, attributes, relations), numbers and units of measurement, temporal concepts, parts and a whole, agents and intentions.
- **Descriptive Ontology for Linguistic and Cognitive Engineering** (DOLCE) [7]
  - A DOLCE ontology is focused on capturing the ontological categories underlying natural language and human common sense.
  - The top-level concept in the ontology is “Concrete”, indicating that all instances of this and its subtypes are private.
- **Cyc’s upper ontology** (OpenCyc) [8], [9]
  - At the top of the collection hierarchy is a universal collection named “Something”, which, by definition, contains everything that exists within the domain being described.
  - An OpenCyc knowledge base contains information from various subject domains: Philosophy, Mathematics, Chemistry, Biology, Psychology, Linguistics.

Usage of modern top-level ontologies in the development of knowledge bases of intelligent computer systems involves the problems of ensuring their compatibility. Since the original purpose of creating top-level ontologies was to ensure the compatibility of subject domain ontologies and applied ontologies but not the intelligent systems themselves.

Such problems are:

- unconstrained interpretation of concepts, caused by the lack of their clear definition;

- the lack of a unified technology of designing knowledge bases on the basis of top-level ontologies;
- the lack of association of top-level ontologies with any technology, which does not allow using them as reusable components.

Therefore, there is a need to develop such a top-level ontology system, which could provide semantic compatibility between a large number of ontologies of different subject domains.

III. Proposed approach

In this article, to solve the problems mentioned earlier, we propose to use an OSTIS Technology. This technology is a set of models, tools, and methods for the development of next-generation intelligent computer systems.

The proposed models are based on the following basic principles of the OSTIS Technology:

- using unified semantic networks with a basic set-theoretic interpretation of their elements as a method of knowledge representation;
- orientation on the semantic representation of knowledge;
- unification of knowledge base models of intelligent systems;
- orientation not only on the specification of knowledge but also on the specification of the model of how this knowledge will be processed.

The *OSTIS* Technology is based on the usage of unified semantic networks with a basic set-theoretic interpretation of their elements as a method of knowledge representation. This way of knowledge representation is called an *SC-code*, and the semantic networks, represented in the *SC-code*, are called *sc-graphs* (*sc-texts*, or *texts of the SC-code*). The elements of such semantic networks are called *sc-elements* (*sc-nodes* and *sc-connectors*, which, in turn, can be *sc-arcs* or *sc-edges* depending on their orientation). The *Alphabet of the SC-code* consists of five basic elements, on the basis of which SC-code constructions of any complexity are built, including the introduction of more particular kinds of sc-elements (e.g., new concepts). The memory storing SC-code constructions is called semantic memory, or *sc-memory*.

The technology also offers several universal options for visualizing *SC-code* constructions, such as *SCg-code* (graphical variant), *SCn-code* (nonlinear hypertext variant), *SCs-code* (linear string variant).

Within this work, fragments of structured texts in the SCn-code [10], which are simultaneously fragments of the original texts of the knowledge base, understandable both to a human and to a machine, will often be used. This allows making the text more structured and formalized, while maintaining its readability. The symbol ":=" in such texts indicates alternative (synonymous) names of the described entity, revealing in more detail certain of its features.

The OSTIS Technology uses subject domains to formalize knowledge, allowing allocating from the diversity of the World only a certain class of entities under study, focusing attention only on something specific.

The proposed approach implies the development of families of Subject domains and ontologies, which would contain descriptions of all the necessary basic classes of entities for the building of the knowledge base of an intelligent computer system.

Such Subject domains and ontologies include:

- Subject domain and ontology of sets;
- Subject domain and ontology of connectives and relations;
- Subject domain and ontology of parameters, quantities, and scales;
- Subject domain and ontology of numbers and number structures;
- Subject domain and ontology of structures;
- Subject domain and ontology of temporal entities;
- Subject domain and ontology of temporal entities of ostis-systems knowledge bases;
- Subject domain and ontology of semantic neighborhoods;
- Subject domain and ontology of subject domains;
- Subject domain and ontology of ontologies;
- Subject domain and ontology of logical formulas, propositions, and formal theories;
- Subject domain and ontology of external information constructions and files of ostis-systems;
- Global subject domain of actions and problems and its corresponding ontology of methods and technologies.

These subject domains are part of the Knowledge base Kernel, which should be part of every intelligent system. This Kernel guarantees the compatibility of intelligent computer systems due to the common conceptual apparatus. Depending on the specifics of particular systems different Knowledge base Kernels can be allocated, but the presence of the basic part, which includes the subject domains and ontologies mentioned above, should remain unchanged.

The following Subject domains and ontologies are considered as part of this article:

- Subject domain and ontology of sets;
- Subject domain and ontology of connectives and relations;
- Subject domain and ontology of numbers and number structures;
- Subject domain and ontology of parameters, quantities, and scales;
- Subject domain and ontology of structures;
- Subject domain and ontology of temporal entities;
- Subject domain and ontology of situations and events, describing the dynamics of ostis-systems knowledge bases.

To each *subject domain*, it is possible to assign:

- a family of ontologies of different kinds corresponding to it;
- a set of semantic neighborhoods describing the research objects of this subject domain.

A ***Subject domain*** is a ***structure***, which includes:

- the main research (description) objects – primary and secondary ones;
- various classes of research objects;
- various connectives whose components are the research objects (both primary and secondary ones), and possibly other such connectives, that is, the connectives (as well as the research objects) may have different structural levels;
- different classes of the above connectives (i.e., relations);
- different classes of objects that are neither research objects nor the above-mentioned connectives but are components of these connectives.

For the formal specification of the relevant subject domain, a type of knowledge such as an *ontology* is used.

ontology
:= [sc-ontology]
:= [semantic specification of any knowledge that has a fairly complex structure, any coherent fragment of a knowledge base: a subject domain, a method for solving complex problems of some class, a description of a certain type of activity, a description of the area of performing a certain set of actions, language of problem-solving methods, etc.]

The concepts and relations considered in the subject domains are discussed in more detail in the Standard of the OSTIS Technology [2].

IV. Subject domain and ontology of sets

Subject domain of sets
:= [Set theory subject domain]
:= [Subject domain of the set theory]
:= [Subject domain whose research objects are sets]
∈ *subject domain*
∋ *maximum class of research objects'*:
set

This subject domain describes:

- concepts: finite set, infinite set, countable set, uncountable set, set without multiples, multiset, multiplicity of belonging, class, class of primary sc-elements, class of sets, class of structures, class of classes, fuzzy set, clear set, set of primary entities, family of sets, non-reflexive set, reflexive set, set of primary entities and sets, formed set, unformed set, empty set, singleton, pair, pair of different elements, triple;
- relations: belonging, inclusion, Cartesian product, Boolean, example, strict inclusion, combination, subdivision, intersection, pair of intersecting sets, pairwise intersecting sets, intersecting sets, pair of non-intersecting sets, pairwise non-intersecting sets, non-intersecting sets, difference of sets, symmetric difference of sets, family of subsets, equality of sets.

V. Subject domain and ontology of connectives and relations

Subject domain and ontology of connectives and relations
∈ *subject domain*
∋ *maximum class of research objects'*:
relation

This subject domain describes:

- concepts: binary relation, sc-connector, non-atomic binary relation, non-binary relation, non-oriented relation, oriented relation, class of equal-powered connectives, class of connectives of different power, unary relation, binary relation, quasi-binary relation, ternary relation, non-binary relation, reflexive relation, antireflexive relation, partially reflexive relation, symmetric relation, antisymmetric relation, partially symmetric relation, transitive relation, antitransitive relation, partially transitive relation;
- relation: relation attribute, first domain, second domain, relation composition, factor set, correspondence, correspondence relation, departure domain, arrival domain, image, prototype, surjective correspondence, non-surjective correspondence, all-round definite correspondence, partially definite correspondence, unambiguous correspondence, inverse correspondence, reversible correspondence, ambiguous correspondence, injective correspondence, one-to-one correspondence, set of combinations, set of placements, set of permutations.

VI. Subject domain and ontology of numbers and number structures

Subject domain of numbers and number structures
∈ *subject domain*
∋ *maximum class of research objects'*:
number

This subject domain describes:

- concepts: natural number, whole number, rational number, irrational number, real number, complex number, negative number, positive number, arithmetic expression, arithmetic operation, Pi number, zero, unit, minor unit, number structure, number

system, decimal number system, binary number system, hexadecimal number system, fraction, regular fraction, decimal fraction, digit, Arabic digit, Roman digit;
- relations: opposite numbers, modulus, sum, product, exponentiation, greater than, equal to, greater than or equal to.

VII. Subject domain of parameters, values, and scales

This subject domain allows describing the properties and characteristics of objects, both qualitative and quantitative ones. The maximum class of research objects for the subject domain of parameters, values, and scales is the parameter.

Subject domain of parameters, values, and scales
:= [Subject domain of parameters and equivalence classes that are their elements (values, quantities)]
∈ *subject domain*
∋ *maximum class of research objects'*:
parameter

parameter
:= [characteristic]
:= [feature]
:= [property]
⇒ *explanation**:
[Each ***parameter*** is a class that is a family of all possible equivalence or tolerance classes defined by either a *equivalence relation* or a *tolerance relation* (symmetric, reflexive but partially transitive).]

Each particular parameter (characteristic), i.e. each element of a class of all possible parameters (characteristics) is essentially a sign of classifying entities with that characteristic according to the equivalence (similarity of value) of that characteristic. For example, the *color* parameter divides the set of entities with color into classes, each of which includes entities with the same color.

A description of the parameter example is shown in Figure 1.

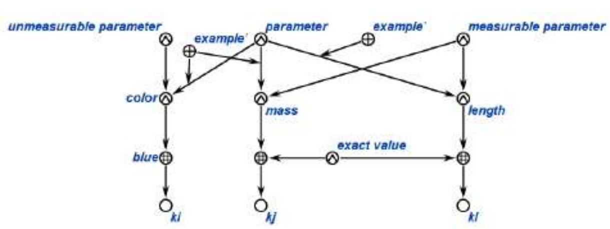

Figure 1. A description of the parameter example

value
:= [value of quantitative parameter]
⇒ *inclusion**:
- *exact value*
- *non-exact value*
- *interval value*

⇒ *explanation**:
[Each ***value*** represents an unambiguous and scale-independent measurement result for some characteristic of some entity]

exact value
:= [exact parameter value]
:= [set of all exact parameter values]
:= [the parameter value that is a family of equivalence classes corresponding to some equivalence relation]
:= [equivalence class]
⇒ *explanation**:
[Each ***exact value*** has one fixed value in some unit of measurement or on some scale. It is assumed that all elements of such a class have the same value of a given parameter, and deviations can be ignored.]

scale measurement
:= [scale]
⇐ *family of subsets**:
measurement
⇒ *explanation**:
[Each ***scale measurement*** is a subset of the *measurement* relation and is characterized not by a unit of measurement but by some reference point for that ***scale***. As the result of a ***scale measurement***, some point on the scale will serve, which is a certain distance away from the reference point in the required direction (smaller or larger).]

A description of an example of a scale measurement is shown in Figure 2.

In this example, ***ki*** denotes the class of entities that have a exact temperature of 330 K and ***bi*** is a specific example of such an entity.

This subject domain describes:

- concepts: measurable parameter, unmeasurable parameter, equivalence class level, non-exact value, interval value, parametric model, fixed unit measurement, arithmetic expression on values, arithmetic operation on values, action. measurement, problem. measurement;
- relations: definition area of a parameter, standard, measurement, exactitude, unit of measurement, zero mark, unit mark, sum of quantities, product of quantities, exponentiation of quantities, greater quantity, equality of quantities, greater or equal quantity.

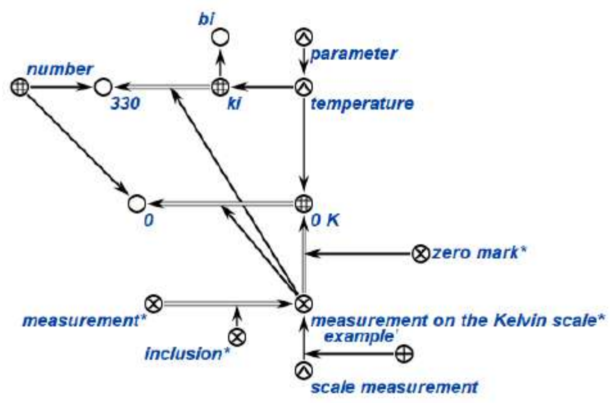

Figure 2. A description for an example of a scale measurement

VIII. Subject domain and ontology of temporal entities

The subject domain is necessary to describe entities and processes that occur in time, since the existing top-level ontologies do not handle this problem well and rarely operate anything other than the present tense.

Subject domain of temporal entities
:= [Subject domain of temporal connections and relations]
:= [Subject domain of temporal entities]
∈ *subject domain*
∋ *maximum class of research objects'*:
temporal entity

temporal entity
:= [temporarily existing entity]
:= [non-stationary entity]
:= [entity that has either a beginning and/or an ending of its existence]
:= [sc-element that is a sign of some temporarily existing entity]
:= [entity with temporal characteristics (duration, starting point, ending point, etc.)]
⇒ *subdividing**:
{• *past entity*
• *present entity*
• *future entity*
}
⇒ *subdividing**:
{• *temporal relation*
• *temporal structure*
:= [structure containing at least one temporal entity]
⇒ *inclusion**:
structure
⇒ *subdividing**:
{• *situation*
• *process*
}
• *material entity*
}
⇒ *subdividing**:
{• *continuous temporal entity*
⇒ *subdividing**:
{• *point temporal entity*
• *longtime continuous temporal entity*
}
• *discrete temporal entity*
:= [temporal entity that can be decomposed into a sequence of point temporal entities]
• *interrupted temporal entity*
:= [temporal entity with interrupts]
}

It should be noted that the above classification of *temporal entities* characterizes not so much the *temporal entities* themselves as our knowledge about them and the degree of granularity of knowledge about these entities with which they are described in the knowledge base. Thus, if it is not important for solving specific problems how a certain *temporal entity* changed within any period of time but only its initial and final state, then it can be considered as a *point temporal entity*.

This subject domain describes:

- concepts: past entity, present entity, future entity, process in sc-memory, process in the external environment of the ostis-system, material entity, influence, class of temporal relations, class of temporal and permanent relations, situational set, non-situational set, partial situational set, temporal connection, temporal relation, beginning, ending, duration, millennium, century, year, month, day, hour, minute, second;
- relations: influencing entity, influencing object, initial situation, causal situation, final situation, event, last added sc-element, temporal inclusion, temporal part, initial stage, final stage, intermediate stage, temporal inclusion without coincidence of initial and final moments.

IX. Subject domain of situations and events that describe the dynamics of ostis-systems knowledge bases

Since it is necessary to distinguish the temporal nature of the sc-element and the temporal nature of the entity denoted by this element, it becomes necessary to use the Subject domain of situations and events that describe the dynamics of ostis-systems knowledge bases to describe the dynamics of the knowledge base itself.

Subject domain of situations and events that describe the dynamics of ostis-systems knowledge bases

:= [Subject domain describing the dynamics of the knowledge base stored in sc-memory]
∈ *subject domain*
∋ *maximum class of research objects'*:
situation

elementary event in sc-memory
⊂ *event in sc-memory*
⇒ *explanation**:
[***elementary event in sc-memory*** is defined as an *event* that changes the state of only one *sc-element*]

.

⇒ *subdividing**:
{
- *event of adding an sc-arc going out of a given sc-element*
- *event of adding an sc-arc coming into a given sc-element*
- *event of adding an sc-edge incident to a given sc-element*
- *event of deleting an sc-arc going out of a given sc-element*
- *event of deleting an sc-arc coming into a given sc-element*
- *event of deleting an sc-edge incident to a given sc-element*
- *event of deleting an sc-element*
- *event of changing the file contents*

}

This subject domain describes:
- concepts: sc-element, present sc-element, logically deleted sc-element, number, uncalculated number, calculated number, concept, main concept, non-main concept, concept going from main to non-main concept, concept going from non-main to main concept, specified entity, not enough specified entity, enough specified entity, medium specified entity, structure, file, event in sc-memory;
- relations: elementary event in sc-memory, event of adding an sc-arc going out of a given sc-element, event of adding an sc-arc coming into a given sc-element, event of adding an sc-edge incident to a given sc-element, event of deleting an sc-arc going out of a given sc-element, event of deleting an sc-arc coming into a given sc-element, event of deleting an sc-edge incident to a given sc-element, event of deleting an sc-element, event of changing the file contents.

X. Conclusion

In the article, an approach to designing top-level ontologies of knowledge bases of next-generation intelligent computer systems is considered. This approach is based on the representation of the knowledge base of intelligent computer systems grounded on the OSTIS Technology as a hierarchical structure of interrelated subject domains and their ontologies.

The main subject domains and ontologies containing the description of basic classes of entities are allocated. Connections and relations between basic concept classes are described.

These ontologies can be used to develop a universal Knowledge base Kernel, which will ensure interoperability of intelligent computer systems.

The results obtained will improve the efficiency of knowledge base development for next-generation intelligent computer systems, at the same time ensuring and preserving their compatibility.

The author would like to thank the research group of the Departments of Intelligent Information Technologies of the Belarusian State University of Informatics and Radioelectronics for its help in the work and valuable comments.

References

[1] C. Gavrilova, *Gavrilova T.A. Knowledge Bases of Intelligent Systems / T.A. Gavrilova, V.F. Khoroshevsky. - SPb: Peter, 2000*, 2000.

[2] V. Golenkov, N. Guliakina, and D. Shunkevich, *Otkrytaja tehnologija ontologicheskogo proektirovanija, proizvodstva i jekspluatacii semanticheski sovmestimyh gibridnyh intellektual'nyh komp'juternyh sistem [Open technology of ontological design, production and operation of semantically compatible hybrid intelligent computer systems]*, V. Golenkov, Ed. Minsk: Bestprint [Bestprint], 2021.

[3] I. Davydenko, "Ontology-based knowledge base design," *Otkrytye semanticheskie tekhnologii proektirovaniya intellektual'nykh system [Open semantic technologies for intelligent systems]*, pp. 42–50, 2018.

[4] C. Partridge, *A survey of Top-Level Ontologies*. The Construction Innovation Hub, 2020.

[5] V. Mascardi, *A Comparison of Upper Ontologies*. DBLP, 2007.

[6] (2022, Nov) Suggested Upper Merged Ontology (SUMO). [Online]. Available: https://github.com/ontologyportal/sumo

[7] (2022, Nov) DOLCE: Descriptive Ontology for Linguistic and Cognitive Engineering. [Online]. Available: http://www.loa.istc.cnr.it/dolce/overview.html

[8] (2022, Nov) Cycorp – Cycorp Making Solutions Better. [Online]. Available: https://cyc.com/

[9] (2022, Nov) Github: Opencyc. [Online]. Available: https://github.com/asanchez75/opencyc

[10] (2021, Jun) IMS.ostis Metasystem. [Online]. Available: https://ims.ostis.net

Представление формальных онтологий базовых классов сущностей в интеллектуальных компьютерных системах

Бутрин С. В.

В работе рассмотрен онтологический подход к проектированию баз знаний интеллектуальных компьютерных систем нового поколения. Выделены основные предметные области и онтологии, содержащие описание базовых классов сущностей. Полученные результаты позволят повысить эффективность разработки баз знаний интеллектуальных компьютерных систем.

Received 01.11.2022