

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

185,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Examples of Ontology Model Usage in Engineering Fields

Larysa Globa, Rina Novograduska,
Alexander Koval and Vyacheslav Senchenko

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/intechopen.74369>

Abstract

The proposed research deals with the improvement of engineering knowledge classification and recognition by means of ontology usage. Ontology model allows structure information as well as to raises the effectiveness of search. Research describes the development of ontology models for engineering knowledge in Internet portal and modeling system for the classification and recognition of marine objects. The ontology model usage for the engineering knowledge portal development allows to systematize data and knowledge, to organize search and navigation, to describe informational and computational recourses according to the meta-notion standards. The description of modeling system subject domain is based on ontology that allows to realize the recognition of marine objects based on their parameters.

Keywords: ontology, knowledge portals, marine object, the marine objects classification and recognition

1. Ontology model of engineering knowledge portals

There exists a great number of engineering equipment and software to solve specialized tasks of different types. However, they are either very expensive or much closed. The important theoretical and practical results obtained by the researches are mostly limited to specific scientific institutions. Thus, the process of knowledge concentration is going on and knowledge becomes being accessed only by limited groups of people. That is why the most urgent task is to give the possibility of using this knowledge by wider groups of investigators for whom this knowledge is intended to and is in need of. Besides access to knowledge, such shortcomings are characterized as poorly structured and insufficiently systematized information on the Internet. In addition, the

information is widely distributed through various Internet sites, numerous electronic archives and libraries. From the preceding, it is evident that the necessity of new method is urgent to develop specialized Internet knowledge portal using vast amounts of various informational and computational resources in definite sphere. Ontology model takes into the account specifics of Internet portals. Suggested model gives the possibility to systematize and structure Internet portals information as well as to organize informational search.

The aim of this research is to improve access to engineering knowledge by means of designing specialized Internet portal of engineering knowledge (e.g., the portal in the field of strength of materials). The following tasks are to be solved:

- qualitative knowledge presentation on the portal;
- to systematize and structure information;
- to formalize the engineering knowledge and
- to organize an effective and purposeful search.

It is necessary to design the model of knowledge representation at the portal of engineering knowledge. It is clear that the model of high-quality designing to represent knowledge at the portal may allow realizing all the abovementioned requirements. Ontology was used as a model knowledge representation at the portal [1].

Formally, the ontology may be specified as

$$O = \{C, A, R, T, F, D\}$$

where

- C is the set of classes that describes the notions of a subject domain;
- A is the set of attributes that describes the features of notions and relations;
- R is the set of relations specified for classes:

$$R = \{R_{AS}, R_{IA}, R_n, R_{CD}\},$$

where

R_{AS} is the associative relation $R_{AS}(O_2) = \{C_i(O_2) \times C_j(O_2), M(R_{AS}) = \{str\}$, where M is the type of relation meaning,

R_{IA} is the relation "is-are" $R_{IA}(O) = C_k(O) \subset C_m(O)$,

R_n is the relation of "heredity" $R_n(O) = a_i, r_i | A_{C_m}(O) \rightarrow a_i, r_i | A_{C_k}(O)$,

R_{CD} is the relation "class-data" $R_{CD}(O) = C_j(O) \subseteq D_i(O)$;

- T is the set of standard types of attribute values;
- F is the set of limits for values of attribute notions and relations;
- D is the set of class exemplar.

Ontology described here may serve to present notions that are necessary for describing knowledge in the field of strength of materials as well as for engineering activities performed in this context.

1.1. Elements of portal ontology

The ontology of the portal in the field of strength of materials includes four ontologies such as engineering activity ontology, engineering knowledge ontology, engineering computations ontology and subject domain ontology [2]: $O_{portal} = \{O_1(O_2, O_3, O_4), O_5\}$ (**Figure 1**).

Engineering ontology consists of engineering activity ontology, engineering knowledge ontology and engineering computations ontology:

$$O_2 \supset O_2, O_3, O_4$$

Engineering activity ontology includes general classes of notions related to the organization of engineering activities such as *Person, Organization, Activity, Event, Literature, Documents, Teaching materials, Publication* and *Location*:

$$O_2 = \{C_{O_2}, A_{O_2}, R_{O_2}, T_{O_2}, F_{O_2}, D_{O_2}\}$$

Engineering knowledge ontology includes the meta-notions that specify structures to describe the problem. The classes of this ontology correspond to *Research method, Research object, Research result, Research purpose* and *Research equipment*:

$$O_3 = \{C_{O_3}, A_{O_3}, R_{O_3}, T_{O_3}, F_{O_3}, D_{O_3}\}$$

Engineering computations ontology unites classes that describe calculation abilities realized at the portal. Engineering computations ontology includes classes such as *Calculation, Service, Service Parameters, Results* and *Interface*:

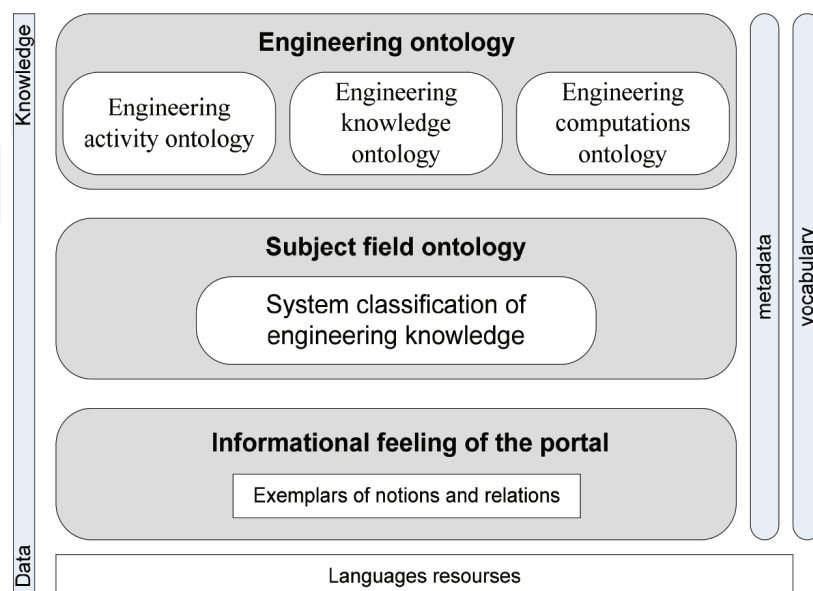


Figure 1. Ontologies of the portal.

$$O_4 = \{C_{O_4}, A_{O_4}, R_{O_4}, T_{O_4}, F_{O_4}, D_{O_4}\}$$

The classes enumerated are related to each other with classes of one ontology and to the classes of other ontologies by means of associative relations. For example, classes of engineering activity ontology “*Person*” and “*Organization*” are related through associative relations “*Be a member of*”. It means that in real life, a person may be a member of some organization. Associative relations may correlate not only with classes of one ontology, but also with classes that belong to different ontologies. For example, class “*Literature*” being a class of engineering activity ontology is associatively correlated by relation “*Describe*” with the engineering computations ontology class “*Research result*”. In addition to associative relations in working up the portal ontologies, the relations of the type “is-are” to relations of subclasses with their parent classes are used. For example, class “*Literature*” is related by means of “is-are” with classes “*Documents*”, “*Training materials*” and “*Published materials*”. It means that class “*Literature*” is parent class for its subclasses “*Documents*”, “*Training materials*” and “*Published materials*”.

Subject domain ontology represents general knowledge of subject domain such as hierarchy of notion classes and their semantic relations. Ontology of subject domain describes the strength of materials as a whole as science and its parts, notions and their connections. These notions are realizations of meta-notions of engineering knowledge ontology and may be put in the order into hierarchy “is-are”. For example, “*Research methods*” (class of engineering knowledge ontology) correspond to such methods as methods of strain, deflection, stress distribution, and so on in the field of strength of materials [3]. “*Research objects*” are materials, material groups or specific material properties. Main class of engineering computations ontology “*Calculation*” corresponds to such notions from the field of strength of materials as limit state design, deformation analysis, stress calculation, and so on.

1.2. Searching process based on ontology

The search for information is also based on ontology model. Due to this fact, user can set the search request not only with the help of keywords, but also with the help of terms of subject domain, which are well-known to the user. The main elements of such search request are basic notions of ontology: its classes, attributes and relations of various kinds. Search request that is formed by means of ontology is simple for user and is full from a perspective of information found. For example, the search request: “*To find the results of research, that was held with steel, received by Gain V.A. in 2008 and that are described in the book “Steel behavior,” published by the German Institute of Material Science*” in a formal way can be presented as:

Class: “Research result”

Relation: “Was held with”

Class: “Research object”

Attribute: “Name”

Class exemplar: “Steel”

Relation: “Received”

Class: "Person"

Attribute: "First name, Last name, Surname"

Class exemplar: "Gain V.A."

&

Class: "Research result"

Attribute: "Year"

Class exemplar: "2008"

Relation: "Described"

Class: "Literature"

Relation: "is-are"

Class: "Teaching materials"

Attribute: "Name"

Class exemplar: "Steel behavior"

Relation: "Published"

Class: "Organization"

Attribute: "Name"

Class exemplar: "German Institute of Material Science"

After presentation of search request in the terms of ontology model, let us make the formalized presentation of them:

$C_1 = \{\text{research result}\}, R_{AS_1} = \{\text{was held with}\},$

$C_2 = \{\text{research object}\}, A_{C_2} = \{\text{name}\},$

$D_{C_2} = \{\text{steel}\}, R_{AS_2} = \{\text{received}\},$

$C_3 = \{\text{person}\}, A_{C_3} = \{\text{First name, Last name, Surname}\},$

$D_{C_3} = \{\text{Gain V.A.}\}, C_1 = \{\text{research result}\},$

$A_{C_2} = \{\text{year}\}, D_{C_2} = \{\text{2008}\},$

$R_{AS_3} = \{\text{Described}\}, C_4 = \{\text{literature}\},$

$R_{IA_4} = \{\text{is-are}\}, C_5 = \{\text{teaching materials}\},$

$A_{C_5} = \{\text{name}\}, D_{C_5} = \{\text{steel behavior}\},$

$R_{AS_4} = \{\text{published}\}, C_6 = \{\text{organization}\},$

$A_{C_6} = \{\text{name}\}, D_{C_6} = \{\text{German Institute of Material Science}\}.$

For search request realization such description of classes and relations are actual:

- classes: $C_2(A_{C_2}, D_{C_2}, R_{C_2}), \quad C_3(A_{C_3}, D_{C_3}, R_{C_3}), \quad C_4(A_{C_4}, D_{C_4}, R_{C_4}), \quad C_5(A_{C_5}, D_{C_5}, R_{C_5}),$
 $C_6(A_{C_6}, D_{C_6}, R_{C_6}),$
- relation of “is-are” type: $R_{IA_{45}} \leftarrow C_4 \subset C_5,$
- associative relations: $R_{AS_1} = \{ C_1 \times C_2 \}, R_{AS_2} = \{ C_1 \times C_3 \},$
- $R_{AS_3} = \{ C_1 \times C_4 \}, R_{AS_4} = \{ C_5 \times C_6 \}.$

With the help of formal description given earlier, search request could be given as shown in **Table 1**. **Figure 2** shows the process of search through the elements of portal ontology. Class and relations exemplars, set by user in the terms of search request, are marked with bold text and bold line. Class “*Research result*” is connected by appropriate associative relations with classes “*Research object*,” “*Literature*,” “*Person*” and “*Organization*.” In these classes, the exemplars set by the user are found: Steel, Gain, Steel behavior, German Institute of Material Science. In such a way, exemplar of class “*Research result*” that corresponds to these exemplars could be found. Therefore, this exemplar is the theory of strength.

From the practical point of view, implementation of the method will make it possible to give the most optimal result for search request. As the search process is held according to the tree of terms determined in the ontology model, the search query implementation makes all of its terms associated with the relationship of the ontological model. Thus, the result of a search query will be the most appropriate term assignments to the terms in the search query.

1.3. Ontology model integration into knowledge portals

Most of the portals use relationship databases to organize and to process data. To provide information on the existing knowledge portal in the form of ontological models, it is possible to use a relational database to get data for ontological model designing.

This module should give API to represent ontological model for knowledge portal. This model is based on data from the database. Ontological model that is accessible via modules API is in RDF format. **Figure 3** presents the portal with ontology integration schema of used relational database. The schema includes:

- Relational database;
- Knowledge portal component;

Search request	Description
$C_1 \vdash$	Research result
$\xrightarrow{R_{AS_1}} D_{C_2}$	Held with steel
$\xrightarrow{R_{AS_2}} D_{C_{3_1}} \ \& \ D_{C_{3_2}}$	Received by Gain V.A. in 2008
$\xrightarrow{R_{AS_3}} D_{C_5} \xrightarrow{R_{AS_4}} D_{C_6}$	Described in the book “Steel behavior”, published by the German Institute of Material Science”

Table 1. Search request formalization.

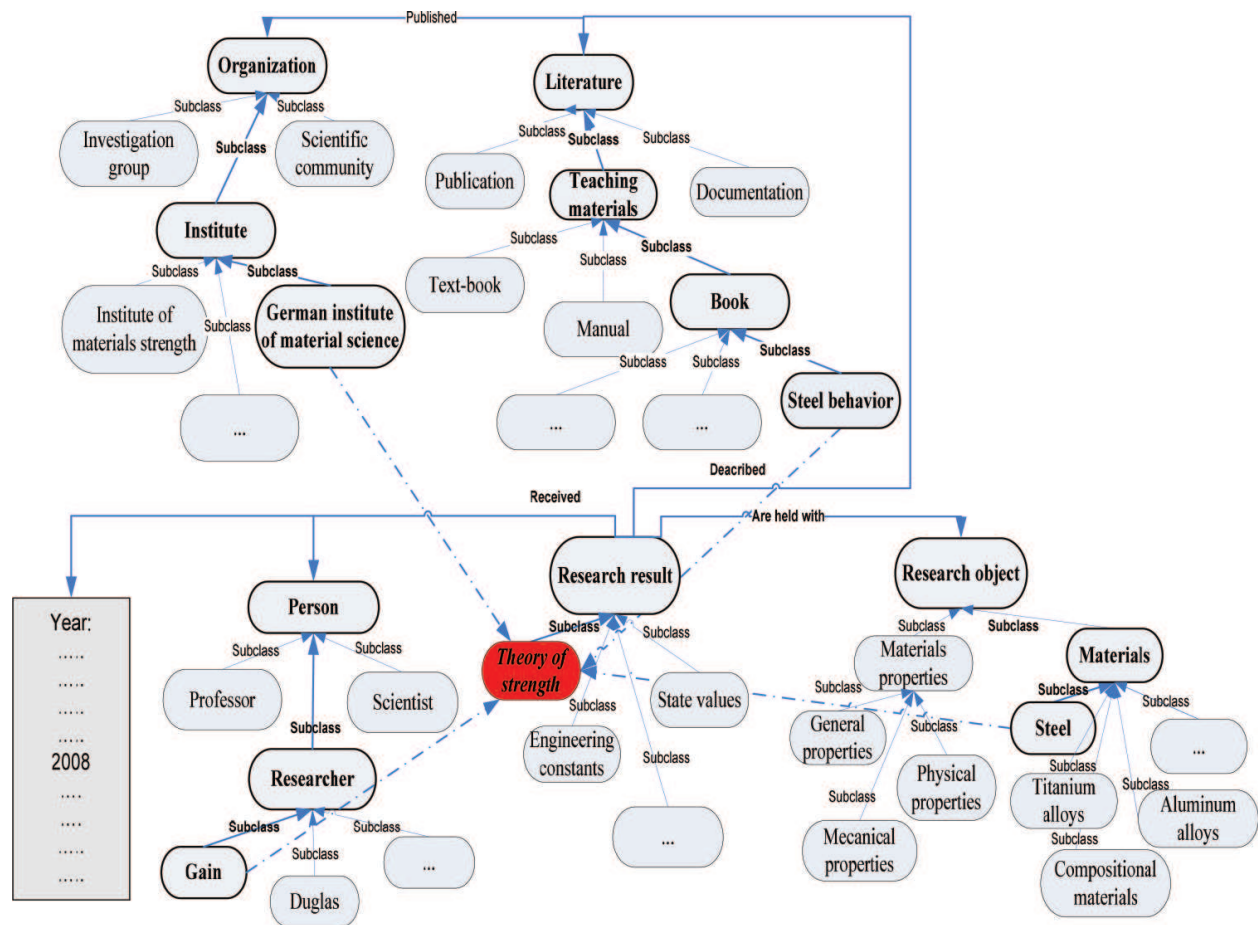


Figure 2. Search request realization process.

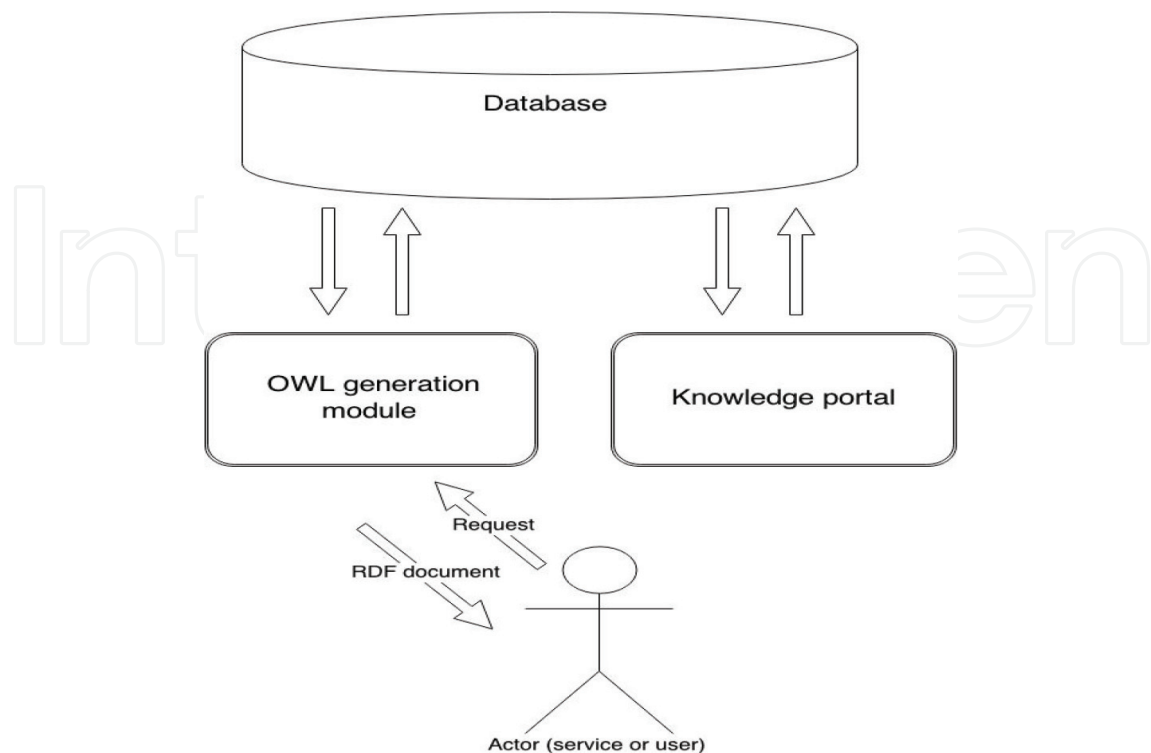


Figure 3. Schema of the portal with ontology integration.

- Generator of Web Ontology Language (OWL) representation and
- Actor (service or user).

To get ontological representation, actor makes requests to OWL generator, OWL generator uses portal database to generate RDF document that corresponds to the ontological model. The main advantage of this approach is that it is not necessary to modify existing knowledge portal.

Let us consider the process of generating RDF document (Figure 4). To generate RDF document using specific program module it is in need to link ontology classes with SQL query. Module gets query using the class name and execute SQL request to database of portal. When database returns appropriate data, the module generates this data in RDF view. After all processing, the module returns RDF document to the client.

Knowledge portals have complex structure. It should be possible to visualize the ontological model on the portal. The end user should have tools for the knowledge portal structure modification using its functionality. Basic requirements for visualization module are as follows:

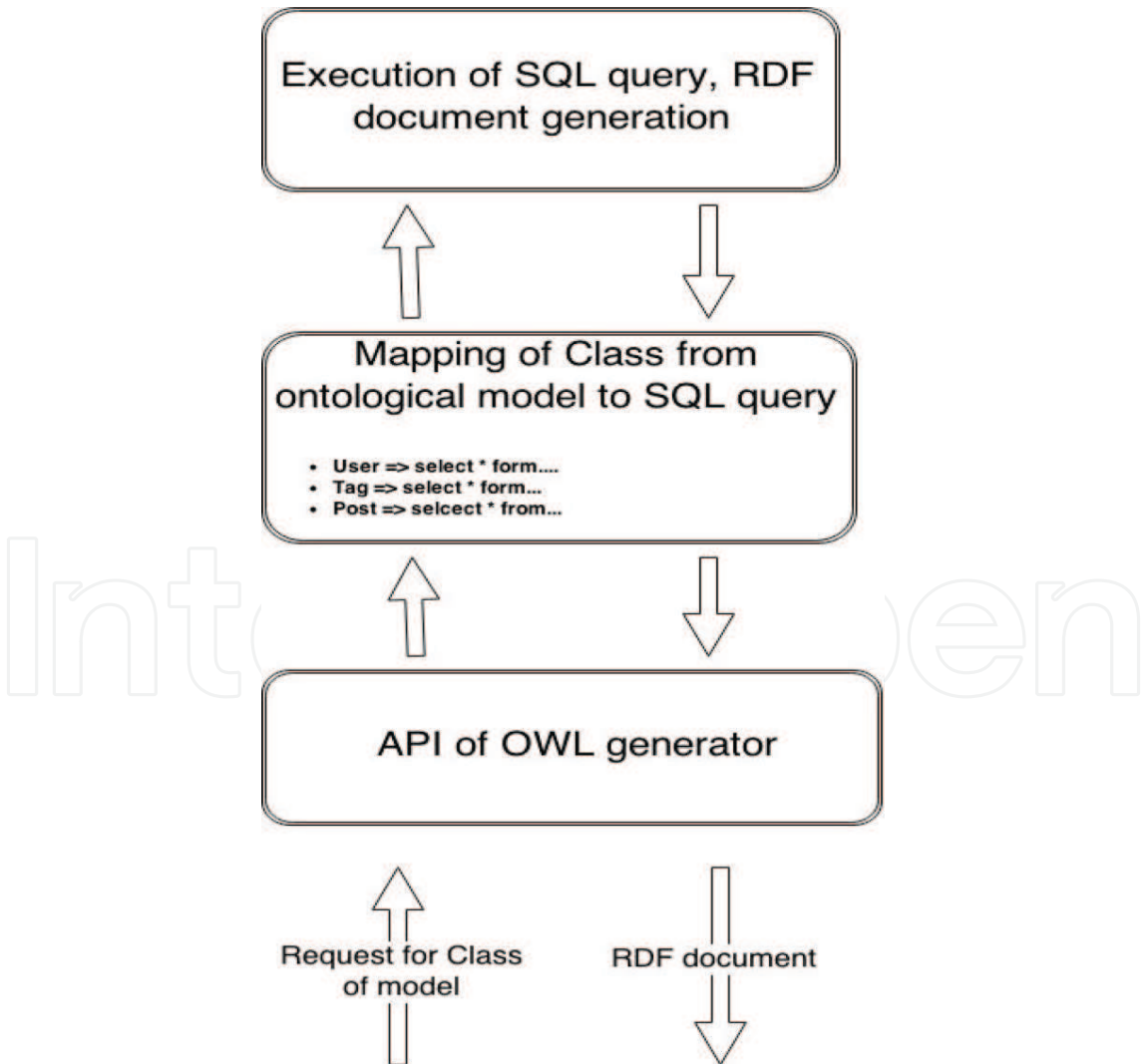


Figure 4. The process of RDF document generation.

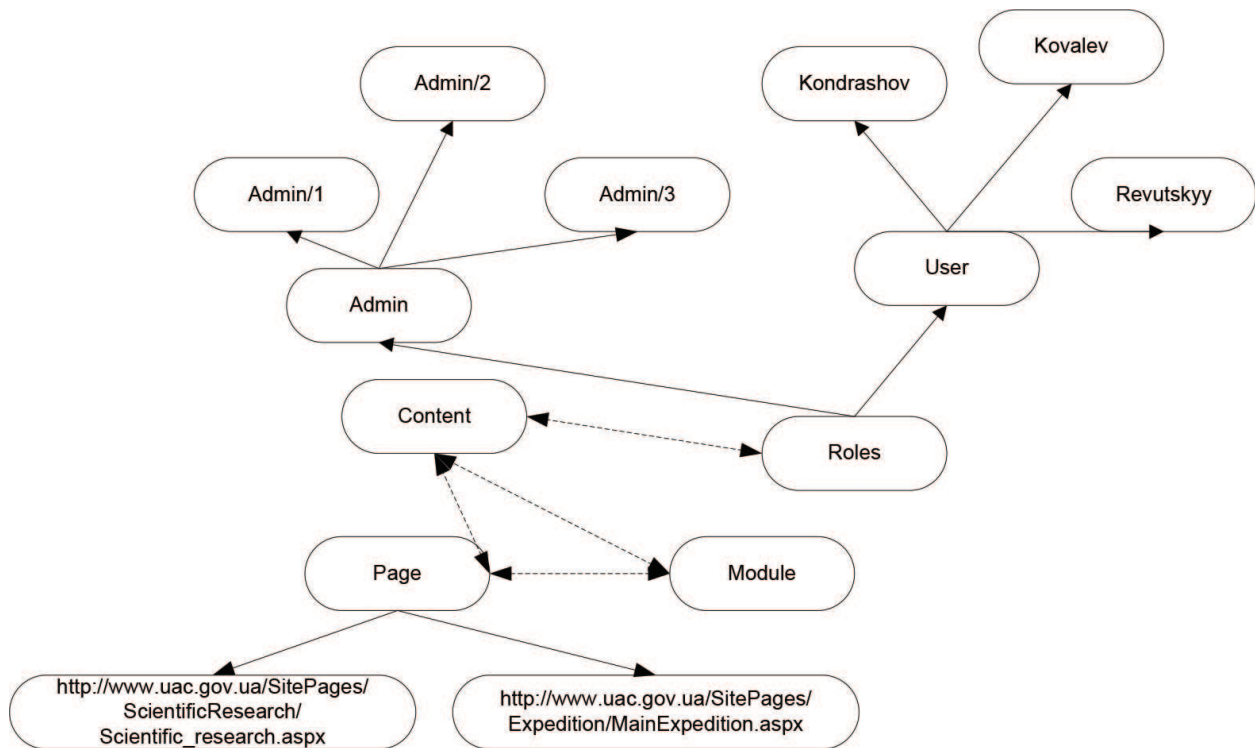


Figure 5. Visualization of classes.

- ability to integrate with portal interface,
- visualize ontological model as a graph,
- each class of models should have unique color and
- visualization should be able to adapt to complex models

Proposed solution developed as a JavaScript module that can be integrated easily with portal interface. Examples of visualization are shown in **Figure 5**.

The classes have different colors. Class “*User*” (user of a portal) has five fields. Class *Post*, which represent note on portal, has nine fields. Also, there are six instances of class *Post*.

2. The ontology model for classification and recognition of marine objects

2.1. Features of semantic modeling technology

The main feature of semantic technology is to store and maintain the integrity of semantics (meaning of knowledge) separately from the contents of data files and from the code of the programs that implement them [4]. Semantic simulation technology differs from traditional methods that combine the meaning of data and the processing procedures directly in the program code. This often leads to the need for a radical manual redevelopment of data structures and total revision of programs during their development or migrate to another platform.

Semantic technologies allow obtaining logical conclusions based on the rules of conceptual models and perform automatic redesign of data structures. Experience the semantic modeling of intelligent systems using ontologies indicates that any subject domain (SD) can be described considerable number of ontologies. From the methodological point of view, it is quite understandable—each ontology reflects a person's perception of the developer of the functioning model of ontology (the main entities, classes, subclasses and their relationships within the general idea about the subject domain). Therefore, it is advisable for the developer to apply such methodology and tools, which allow not only to develop an ontology model, but also to correct it in the process of mastering it, and understanding the features of its functioning, been aimed at the most correct model development.

Under the term ontology, we understood a system of concepts domain, which is represented by a set of entities and their properties, interconnected relations, in order to develop knowledge bases on their basis. Consequently, the main purpose of ontological modeling is to develop a formalized knowledge model of the domain, which is stored electronically and may further improved through a more in-depth understanding of the features of the subject domain.

For the implementation of knowledge-based systems, it is expedient to use common language for describing ontologies such as OWL Lite, OWL DL and OWL 2 [5] that use discriminatory logic for knowledge work. In this case, the semantic modeling process has been performed using the open knowledge base connectivity (OKBC) [6]. This model is based on the theory of frames and uses concepts such as “conceptualization of classes,” “objects,” “slots,” “facets” and “inheritance” for representing knowledge about the subject domain, which allows to develop various knowledge-based applications with a high level of interoperability.

The formal semantics of the subject domain on the OWL describes how to obtain logical investigations, having the ontology of SD, which is to obtain facts that are not represented literally in the ontology, but logically follow from its semantics.

2.2. Methodological aspects of semantic modeling using web ontology language

An applied ontology should describe concepts that depend both on the ontology of tasks and on the ontology of the subject domain. The purpose of the applied ontology is to develop an electronic model of knowledge that allows:

- creation of general terminology of a subject domain, for common use and understanding by all users—system developers;
- give an exact and consistent definition of the meaning of each term and
- provide semantic tasks using axioms that automatically allow you to answer the main questions about the subject domain.

One of the most common languages for representing ontology is the Web Ontology Language (OWL). The OWL language contains elements such as classes, properties and individuals [7]. All concepts of the subject domain are divided into classes, subclasses and instances (copies). The tag describes classes as:

`<owl:Class rdf:about = "http://untitled-ontology-9#APM_MK">`

Thus, the process of semantic modeling using the unified OKBC model consists of the following steps:

- Definition of the concepts of the subject domain, that is, the basic concepts such as classes, entities, categories (Active Ontology, Entities and Classes) that describe the SD;
- Determination of the set of properties, which describes the properties of the concepts of SD and allow in the final sense to develop a knowledge base of the domain. Formation of concept properties in AKVS is performed using the mechanisms of definition, attributes and roles (Data Properties);
- Establishing relations between concepts of the subject domain and their properties using the mechanisms of forming predicates (Object Properties), which should be taken into account the functional orientation of relations, for example, "solves problems", or "is part of", and so on;
- Setting numerical or logical constraints, which are used to describe the properties of instances (Individuals) of the knowledge base. This is realized through the mechanisms of axiom definition (Axiom) or facet (Value). For example, the maximum speed for terrestrial objects is limited to the value;
- Formation of knowledge base using the mechanism of description of instances of the knowledge base and its filling (Individual by class);
- Development of typical query patterns for the knowledge base using the query language, DL Query and the output machine, Reasoner;
- Checking the correctness of the functioning of the ontological model of SD from the point of view of its correspondence to the initial goals and the task and finding gaps in the ontology using the OntoGraf research mechanism. The evaluation is based on the analysis of the results of testing by various output machines (Reasoner) and the compilation of various types of requests;
- Development of a strategy for improving the ontological model of SD and carrying out the relevant work.

Taking into account the complexity and ambiguity that arises in the process of describing the subject domain, modern science offers several approaches to the creation of ontology:

- Top-down. The use of this method requires the definition of the most general concepts of the domain, with further detail of objects in the class hierarchy and the concepts of the subject domain.
- Down to the top. This approach begins with the definition of detailed and specific classes (the end of the tree of the hierarchy), followed by grouping into more general concepts.
- Combination of the first two methods. First, a description of the concepts that are fully understood, then associate them into groups and develop more complex concepts of subject domain.

2.3. Definition of the depth and scope of the subject domain

It is advisable to start developing an ontology with the purpose of determining its scope and scale. That is, the answer to a few basic questions:

- Which SD will cover the ontology?
- What types of questions should answer information in ontology be?
- Who will use and maintain an ontology?

Of course, the answers to these questions change during the process of designing ontology software, but at any time, they allow you to limit the scale of the ontology if it becomes too complicated.

Consider the methodology for creating an ontology in an example of An Intelligent System For Studying Hydroacoustical Processes.

The main quality of such systems is the accumulation of two types of information:

- real data—hydroacoustic as a result the marine area scanning,
- data modeling—obtained as a result of mathematical modeling, the behavior of the object.

Modeling of hydroacoustical processes requires the development of component models, including the information model of the water area, the database of the parameters of the marine environment (sea noise, ground, coastline, water temperature, deep, salinity) and the parameters of hydroacoustic devices and their interaction with the modeling medium.

Hydroacoustic processes allow to take into account both problems of direct modeling (for the purpose of obtaining objective data-knowledge about the marine facilities under study), and combine the obtained data with expert knowledge (represented as a characteristic set of parameters of real objects and their assessments based on the expert's experience).

Generalized structure of the knowledge-based modeling system for the identification, classification and definition of parameters of movement of marine facilities is shown in **Figure 6**. The conceptual model consists of such structural components [8]:

- Simulation of marine environment performs the functions of creating a simulation scene (parameters of depths, temperatures and salinity, type of bottom, coastline, etc.) and location and specification of the parameters of marine objects (type, size, direction, speed, etc.);
- Modeling of hydroacoustic device sets the scene of the location of fixing devices and their parameters (type, dimensions, sensitivity);
- Hydroacoustic signal analysis is a set of tools for creating models for generating sonar signals and working out methods for their analysis (fast Fourier transforms, digital filtering, spectral, frequency, correlation analysis, etc.);
- Knowledge base maintenance contains a set of tools for testing models of object recognition, their identification, classification and definition of the parameters of the movement of objects, including methods of fuzzy logic;

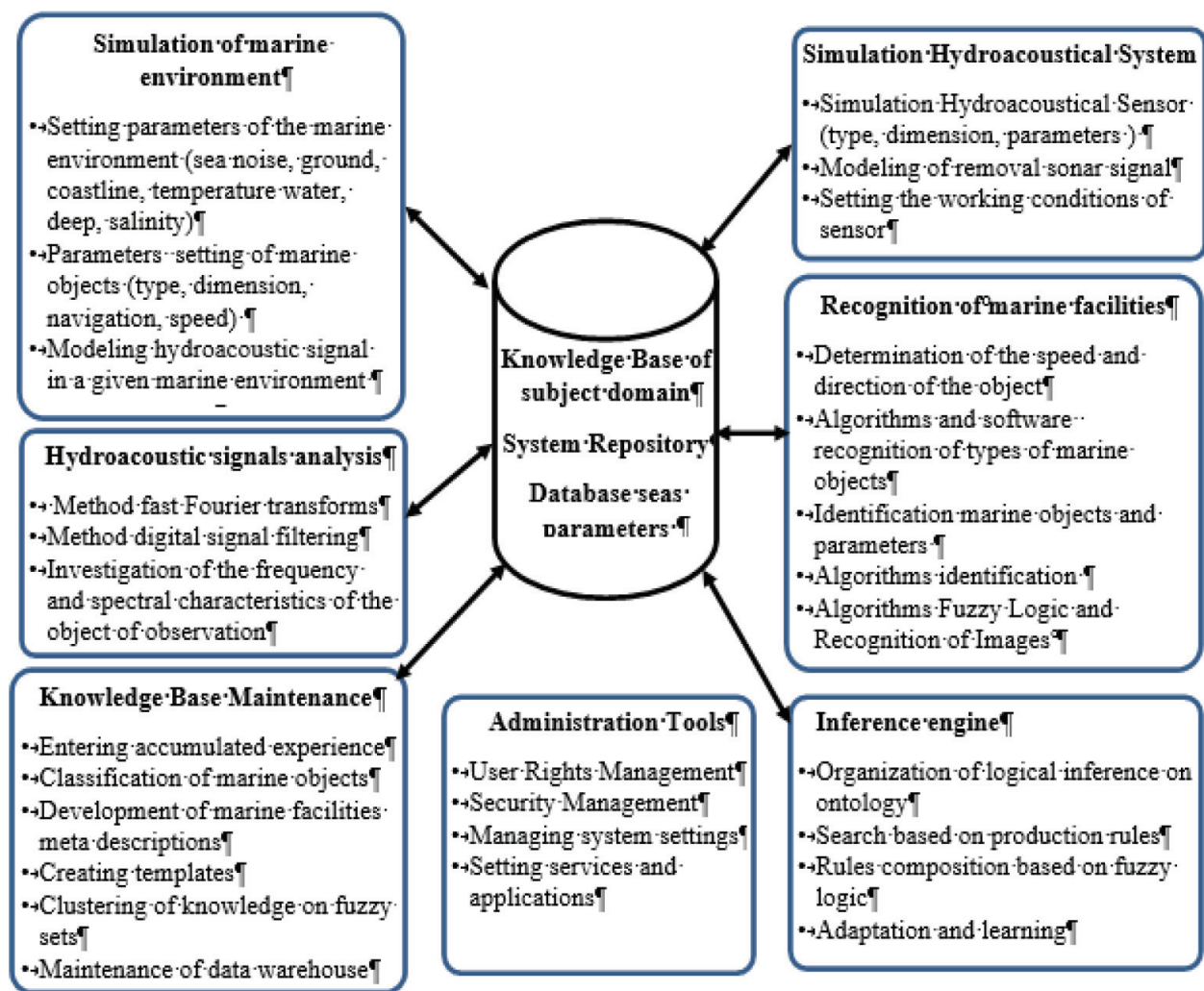


Figure 6. Modeling system structure.

- Knowledge base maintenance is intended for the organization of tools for forming a knowledge base for solving problems of identification and classification of marine objects;
- Inference engine designed to organize logical inference based on accumulated knowledge, including means of composition of product rules, self-learning and adaptation;
- Information storage is the core of the system and provides information to all the structural components of the modeling system, and also contains information with expert assessments of the experiments conducted;
- Administration and management tools provide settings for services and applications to manage user access rights to information resources, manage security and performance of the modeling system.

The proposed system provides opportunities for end-to-end documentation of the processes of hydroacoustic experiments, which gives additional advantages in the formation and accumulation of knowledge about the studied processes, including the formation of scenes and modeling scenarios. Simulation involves the following steps:

- 1 Creation and maintenance of library of hydroacoustic models, including signal generation models, signal extraction models, signal analysis models, implementation of algorithms for classification and identification of objects.
- 2 Creation and maintenance tools for entering both types of data such as model and real data (scanning water areas) into the knowledge base;
- 3 Development of subject domain ontology;
- 4 Development of algorithms and software for recognition and identification types of marine objects
- 5 Organization of logical inference on ontology
- 6 Creation of learning system (scripts and learning algorithms considering SD).
- 7 Maintenance tools for rules setting based on fuzzy inference;
- 8 Development of algorithm identification based on logical inference on the ontology and production rules.
- 9 Development of algorithms for knowledge classification and clustering;
- 10 Formation of classifiers of the system (noise-emitting objects, hydroacoustic systems, water areas).
- 11 Scene formation and creating an experiment scenario
- 12 Fixing the experiment results in DB
- 13 Evaluation of simulation results

2.4. Conceptual model of the subject domain ontology

Ontology model $Ont(SD)$ of the intellectual system contains the basic concepts (entities—basic concepts of the subject domain), their attributes and describes the relations between them and can be represented as:

$$Ont(SD)_{IS} = \langle C^{(Ax)}, Ex^{(C)}, Rel^{(H)}, T^{(Q)}, Ax^{(s)}, Rul^{(S)} \rangle$$

where $C^{(Ax)}$ -classes (Classes) are a finite set of basic concepts of hydroacoustic processes;

$Ex^{(C)}$ -set exemplars classes of ontology;

$Rel^{(H)}$ -relations between classes and their types:

$T^{(Q)}$ -attributes of each class, their data types and value fields;

$Ax^{(s)}$ -axioms that define the basic concepts of SD, which are always true for it and

$Rul^{(S)}$ -rules of logical conclusion.

Detailed consideration of the concepts of the subject domain and functional problems leads to the allocation of the following main classes:

- 1 Experiment class $C^{(Exp)}$: defining characteristics of the conducted experiment—identification number, date of conduct, researcher identifier, service data.
- 2 Models_Hydroacoustic_processes class $C^{(Mod)}$: describing various models, methods and algorithms for generation, selection and analysis of signals as well as methods and algorithms for classification and identification of marine objects.
- 3 Hydroacoustic_objects class $C^{(Obj)}$: consisting of some subclasses (marine objects, underwater objects and air objects) and describing properties objects/
- 4 Aquatorium class $C^{(Sea)}$: describing the maritime area chosen for the experiment: the type of aquatorium, the name and refinement parameters for the type modeling algorithms (coordinates, depths, temperatures, salinity).
- 5 Experiment_scene class $C^{(Sn)}$: describing the simulation scene, taking into account the characteristics of the marine region as well as the characteristics of the objects that participate in the experiment.
- 6 Hydroacoustic_System class $C^{(Div)}$: describing the coordinates, composition and types of hydroacoustic devices.
- 7 Modeling_Scenario class $C^{(Sig)}$: specifying the sequence of individual stages of the simulation. Each step is a set of procedures “start-run-fix the result.”
- 8 Model_estimation class $C^{(Val)}$: for fixing the results of modeling and estimating the correctness of models.
- 9 Acoustic signal class $C^{(Sig)}$: defining the signal characteristics, including the date of the signal detection and the parameters of the locking device.
- 10 Waveguide class $C^{(Noise)}$: describing the hydroacoustic interference affecting the propagation and distortion of the hydroacoustic signal as well as the means of neutralizing interference.

In the ontology model, in addition to the main classes, subclasses representing instances of the corresponding classes are also included.

In accordance with the basic rules of the OWL language, a triplet called the RDF graph describes the class. In this graph, vertices are objects and objects, and as arcs are predicates [9]. From a mathematical point of view, the triplet is an instance of an element of a certain binary relation. The expression of the triplet asserts that certain relations indicated by the predicate connect objects marked as the subject and object in particular in the triplet [10].

One of the tools for semantic modeling is the well-known ontology visual editor Protégé 5—Designed by the University of Stanford [11]. Visual methods of designing ontologies help quickly and fully understand the structure of knowledge of the subject domain, which is

especially valuable for researchers working in the new subject domain. The Protégé 5 supports all phases of the ontology life cycle in accordance with ISO/IEC 15288: 2002 [7] requirements—from the development of a semantic network and the creation of a knowledge base on its basis, to the formation of user requests to these bases in order to obtain knowledge.

The main window of the Protégé 5 editor consists of tabs that represent the various tools to develop model of knowledge such as <Active Ontology>, <Entities>, <Classes>, <Data properties>, <Object properties>, <Individuals by class>, <DL Query>, <SPARQL Query>, <OntoGraf> and so on (Figure 7).

2.5. Formation of the hierarchy of the main classes: Taxonomy of the subject domain

Defining classes and creating their hierarchy (taxonomy) are keys in the development of ontology SD. The taxonomy of classes is a tree of descriptive terms that have a hierarchical structure.

In Protégé 5 editor, creating classes $C^{(Ax)}$ occurs in the bookmark <Classes>. In the OWL, classes are interpreted as a subset of individuals that are part of a defined class.

The peculiarity of designing in the environment Protégé is that classes are considered as subclasses of the general ontology **THING**. According to the CamelCase notation for OWL [12], all class names must begin with a capitalization and should not contain spaces. For securing the classifications, simply press the <Add subclass> button, in the window that opens, you must enter the name of the class. (Figure 8).

By default, classes in OWL can intersect. In order to divide the classes, they need to be disjointed. This ensures that an individual cannot be an instance of more than one class. To do this, in the <Classes> tab you need to define a class that should not intersect, then in the <Description> field, you need to click on the + side of the Disjoint With function and in the <Class hierarchy> window, open the class that should not overlap with the specified class (Figure 9).

OWL allows you to develop annotations with various information (comments, creation date, author, links to resources, etc.) and metadata classes, properties, individuals and ontologies.

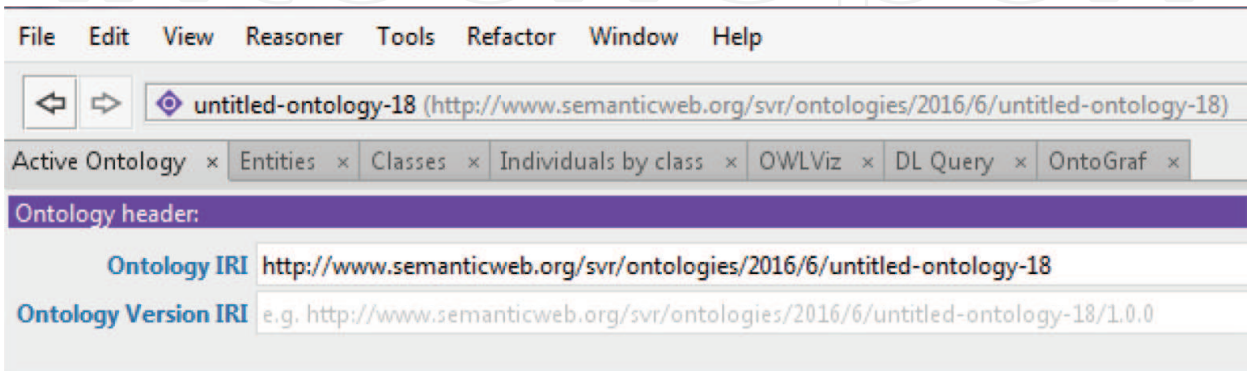


Figure 7. Main functional toolbar of the Protégé 5 editor.

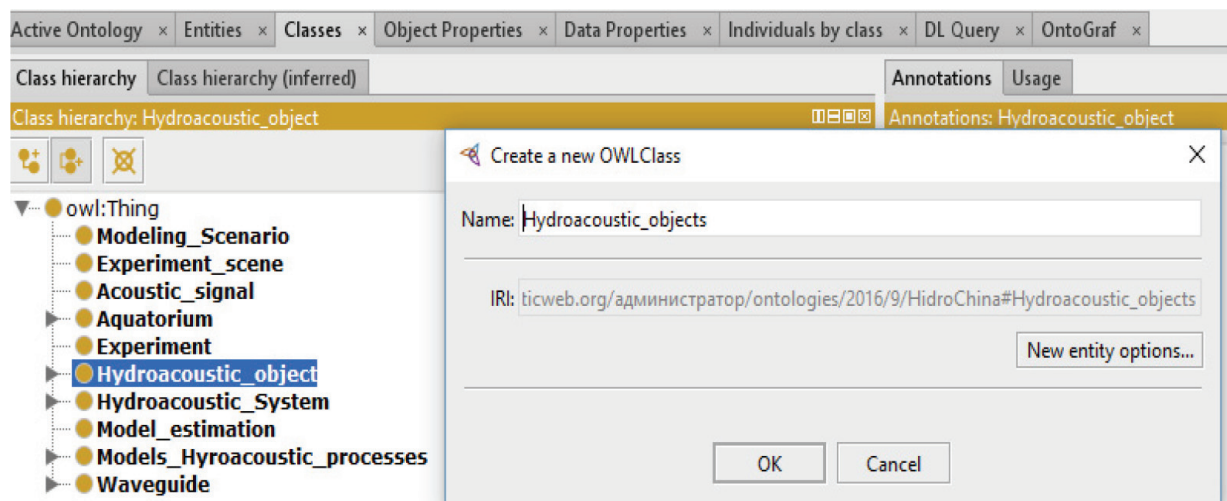


Figure 8. An example of creating classes of the modeling system.

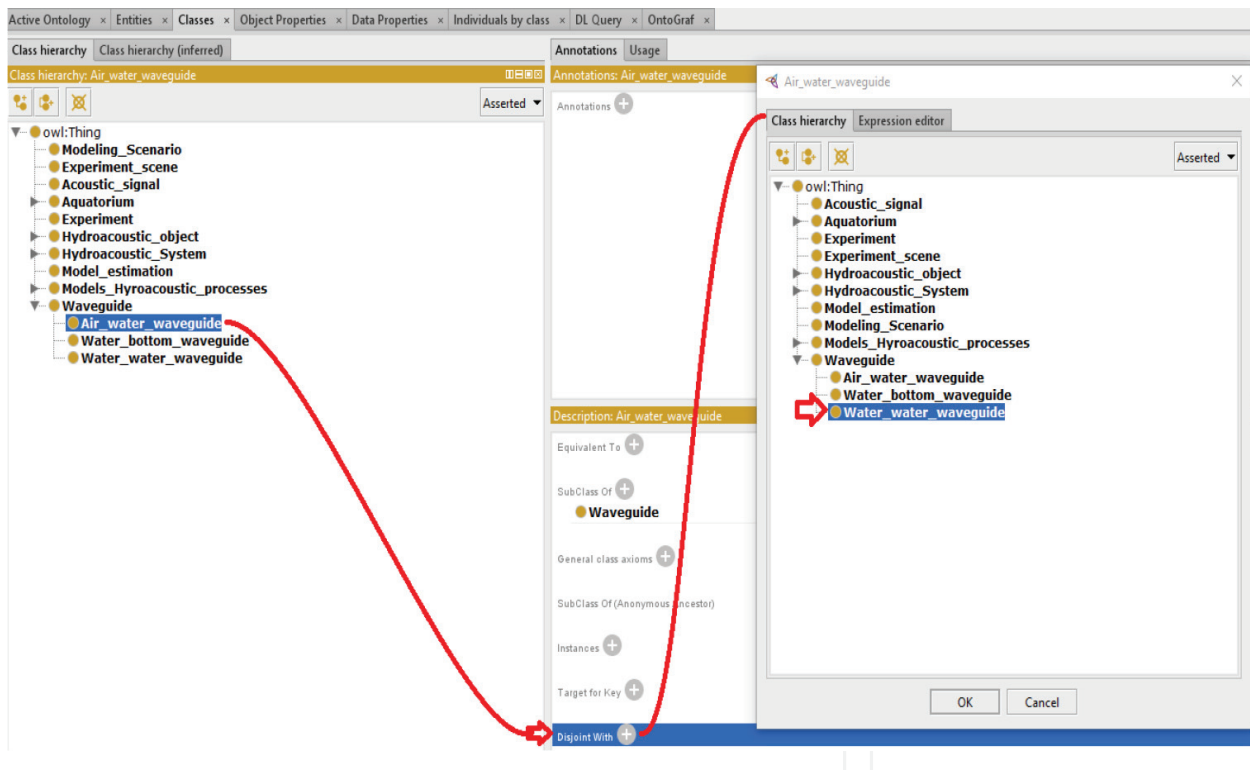


Figure 9. Separation of classes into disjoint ones.

Tabs <Classes>, <Object Properties> and <Data Properties> are used to develop annotations. Next, in the <Annotations> tab, click on the "+" and in the opened window, enter the desired comment (Figure 10).

In the Protégé editor, each class defined by properties that describe the relations between classes are divided into two types [13]:

- <ObjectProperty> —describes relations $Rel^{(H)}$ (types of relations) that are established between particular classes of ontology.

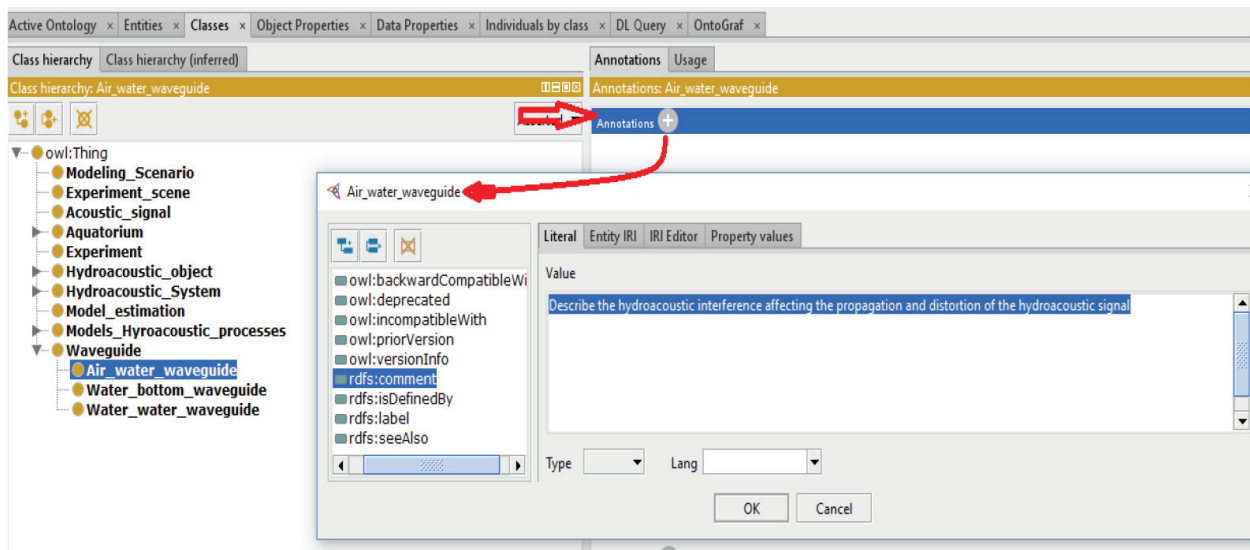



Figure 10. Adding comments to the selected class.

- *<DatatypeProperty>* —describes the specific attributes $T^{(Q)}$ (characteristics) that define the class. For example, the speed of the marine objects, their dimensions, technical parameters, and so on

2.6. Formation of a subset of the domain attributes

A subset of attributes $T^{(A)}$ describes the properties of classes $C^{(Ax)}$ and is used to enter specific values of instances in classes - $Ex^{(C)}$.

The creation of attributes subset by the Protégé editor has been performed by using the *<Datatype Properties>* tab on toolbar of the Protégé editor 5. Next, the window containing the *<OWL: DataProperty>* tab is displayed. When you click a button  in toolbar, opens the window *<Create a new OWLDataProperty>*, in which you can enter a name property such as *<Cruising Speed>* and click OK (Figure 11). After performing the corresponding actions provided by the process, a new type of property appears in the left frame.

Repeating this procedure can form a whole set of attributes for a SD. For a complex domain, *<OWL: DataProperty>* can be represent as a hierarchy structure. A set of properties for different instances of classes can be completely individual.

The predefined attributes has been specified in the XML schema dictionary and can be represented in a variety of data formats such as integers, floating point, lines, logical values, and so on.

An example of the owl tag: *DatatypeProperty<URI>* in XML, which describes the class *<Underwater_object>*:

```
<owl: DatatypeProperty rdf: about = "http://www.semanticweb.org/svr/ontologies/ -9# = "#Underwater_
object_dimensions"/>
```

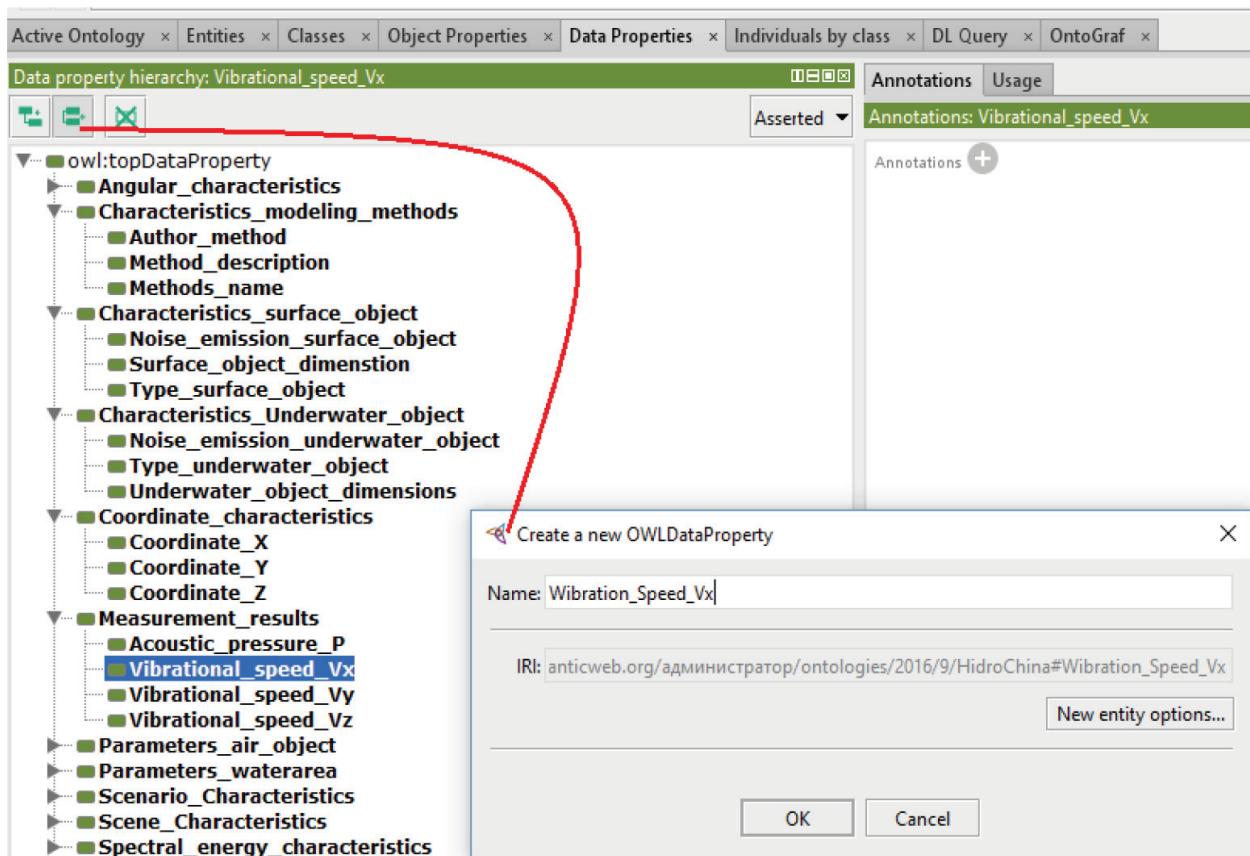



Figure 11. Creation of a new attribute in the Protégé editor.

2.7. Formation of instances of classes for filling the knowledge base

Examples of classes $Ex^{(C)}$ are specific objects of SD (instance) that belongs to a certain class $C_i^{(Ax)}$. Every class $C_i^{(Ax)}$ describes a subset of attributes $T_i^{(A)}$.

Instance creating process is executed through the tab *<Individuals by class>*, which is written by the type tag.

`<owl:NamedIndividual rdf:about = "....">`

Process of creating a new instance consists of the following steps:

- 1 Select a class by using the *<Classes>* or *<Entities>* tab. For example,
- 2 Select tab *<Individuals by class>* in a toolbar of the Protégé editor. New window is activated. In this window, it is possible to develop an ontology instance.

- 3 To create a new instance of the selected class (*Southern seas*), you should click on the button



, as shown in Figure 12. This opens a new window *<Create a new OWLNamedIndividual>*, which is intended to input the name of an instance. After performing the corresponding actions provided by the process, a new instance of the selected class appears in the left frame, for example, *<Taiwan Strait>*.

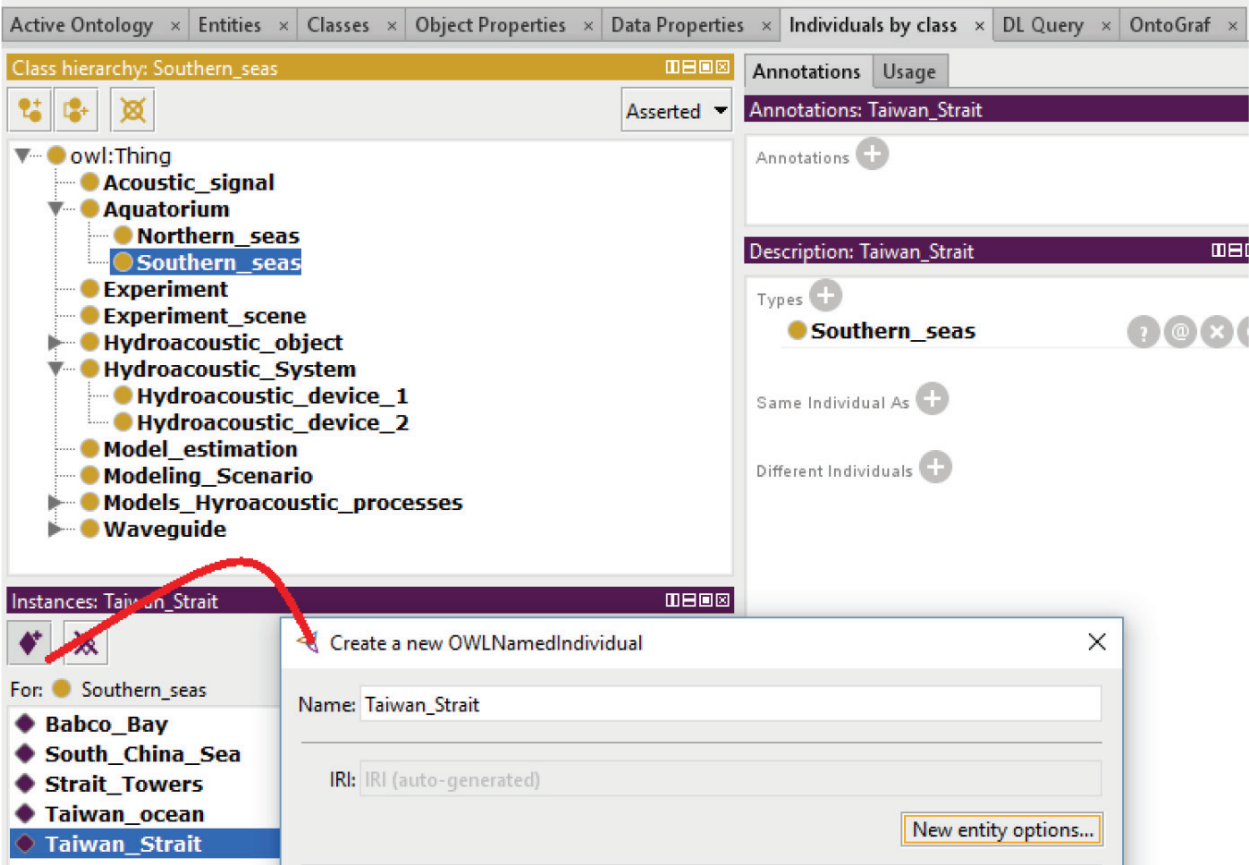


Figure 12. Example of adding a new instance.

The procedure *<Create a new OWL Named Individual>* can also be used to fill the knowledge base with test samples. To do this, select the desired instance of the specified class (*Southern_Seas*) and in the field *<Property assertions>* with the instance name (*Babco Bay*), click on the icon *<+>* next to the function *<Data Property assertions>*. When clicked the icon, a new *<Data Property>* window appears where, required in entity, *<Owl:TopDataProperty>* the user selects the appropriate attribute, for example, *<Maximum_depth>* and enters its value in the field *<Value>* of right window (Figure 13). After clicking the OK button in the ontology database, its value is written.

Since an instance of the class is described by some subset of attributes, this value-adding procedure must be repeated for each attribute.

A set of properties for different instances of classes can be completely individual.

2.8. Defining and forming a relation subset and linking classes

In the Protégé editor, each class is defined by properties that describe the relationship between classes. In the Protégé editor, the object properties describing [14] by tab *<ObjectProperty>* are relations between two classes and individuals.

With the help of the *<ObjectProperty>* tab, the following actions are performed:

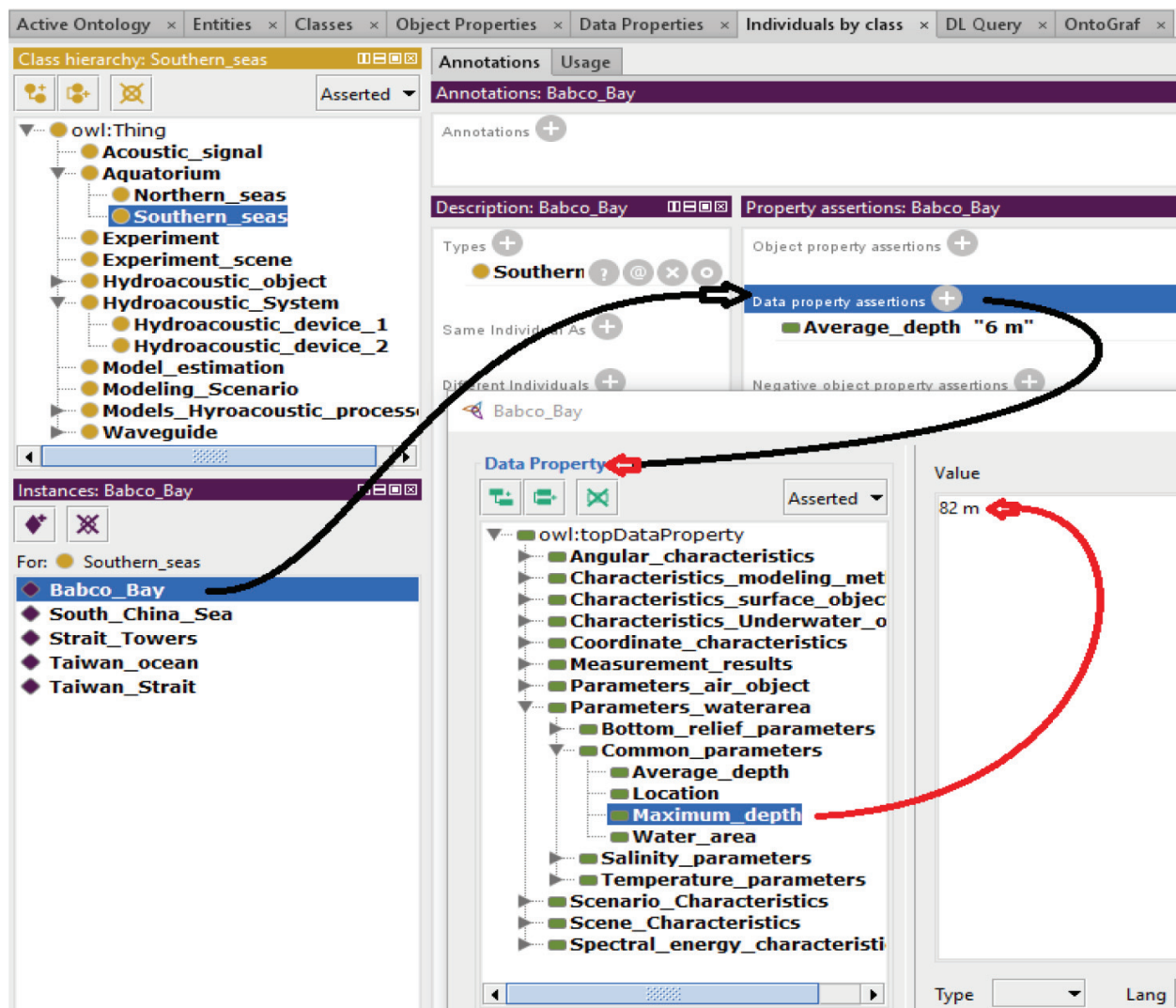


Figure 13. Example of entering the value of a particular attribute.

- 1 Formation of a subset of the relationship $Rel^{(H)}$ between classes $C^{(Ax)}$ in a defined subject domain by adding types of relationships to the list $\langle owl: topObjectProperty \rangle$.
- 2 Formation of a subset of axioms $Ax^{(s)}$ in a defined subject domain by binding certain class ontology relations using the predicates from $\langle owl: topObjectProperty \rangle$.
- 3 Formation of relationships between individuals using the predicates contained in $\langle owl: topObjectProperty \rangle$

Forming a relationship subset in a defined subject domain.

To create a subset of relationships and adding it to $owl: topObjectProperty$, you must select the $\langle ObjectProperty \rangle$ tab in the Protégé 5 and go to the $owl: topObjectProperty$ line. The toolbar opens the tools to describe relationships. To add a new type of relationship, you need to click a button. In the window named $\langle Create a new OWLObjectProperty \rangle$, you must enter a name for the type of relation, for example, $\langle has_communication_ \rangle$ (Figure 14) and press OK.

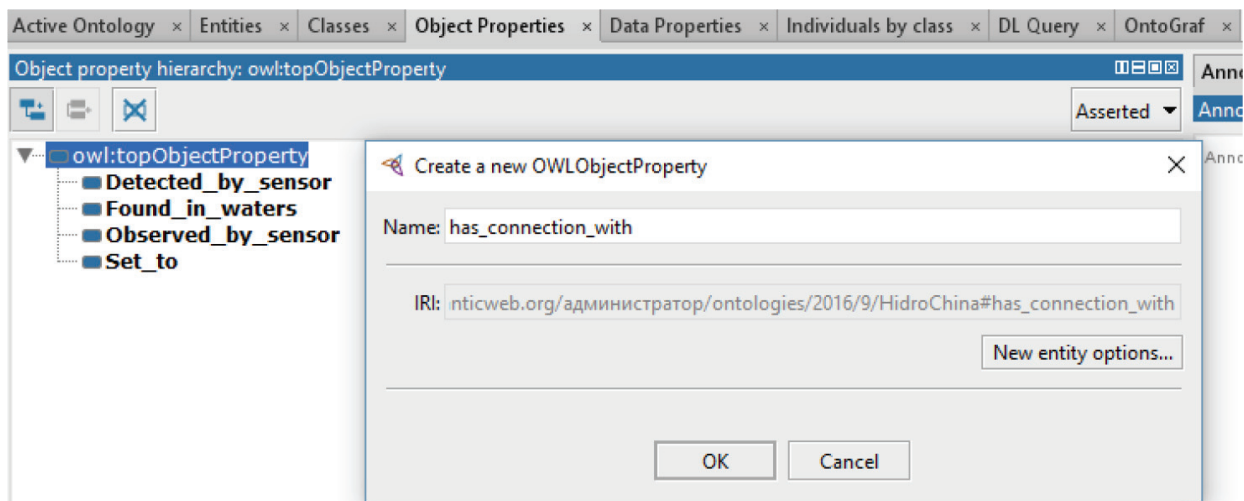


Figure 14. An example of formation of a new type of relationship.

After performing the appropriate actions provided by the process, a new type of connection appears in the `<owl: topObjectProperty>` subset. Repeating this process for SD, it is possible to form the whole subset of the types of zips for the selected SD.

3. Formation of a subset of axioms

Under the axiom are assertions that introduced into the ontology in the finished form, from which other statements can be deduced. Axioms bind two classes (the concept) with certain relations.

The link between classes is described by the RDF graph.

The created relationship between classes (axioms) is described by the RDF graph: `<subject-predicate-object>`:

Subject $C_{si}^{(Ax)}$ - is the class of ontology;

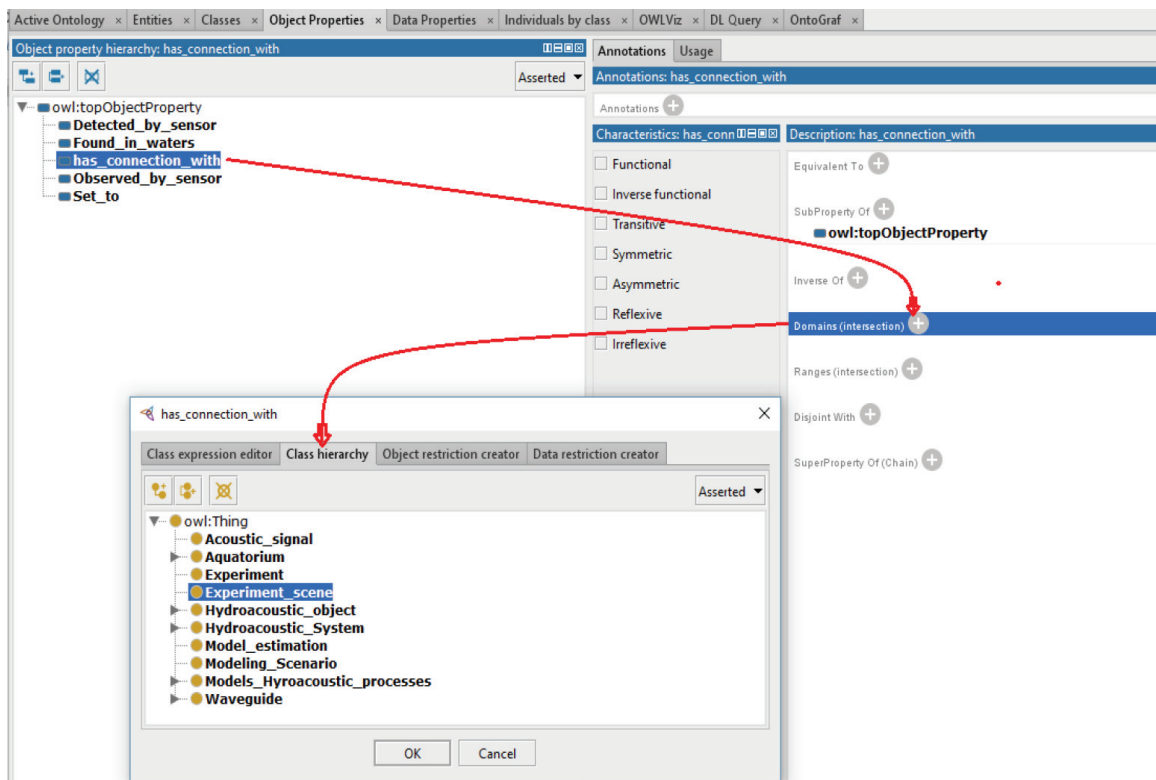
Rel $_{ij}^{(H)}$ -is the predicate binding of two classes;

Object $C_{oj}^{(Ax)}$ - is the class of ontology.

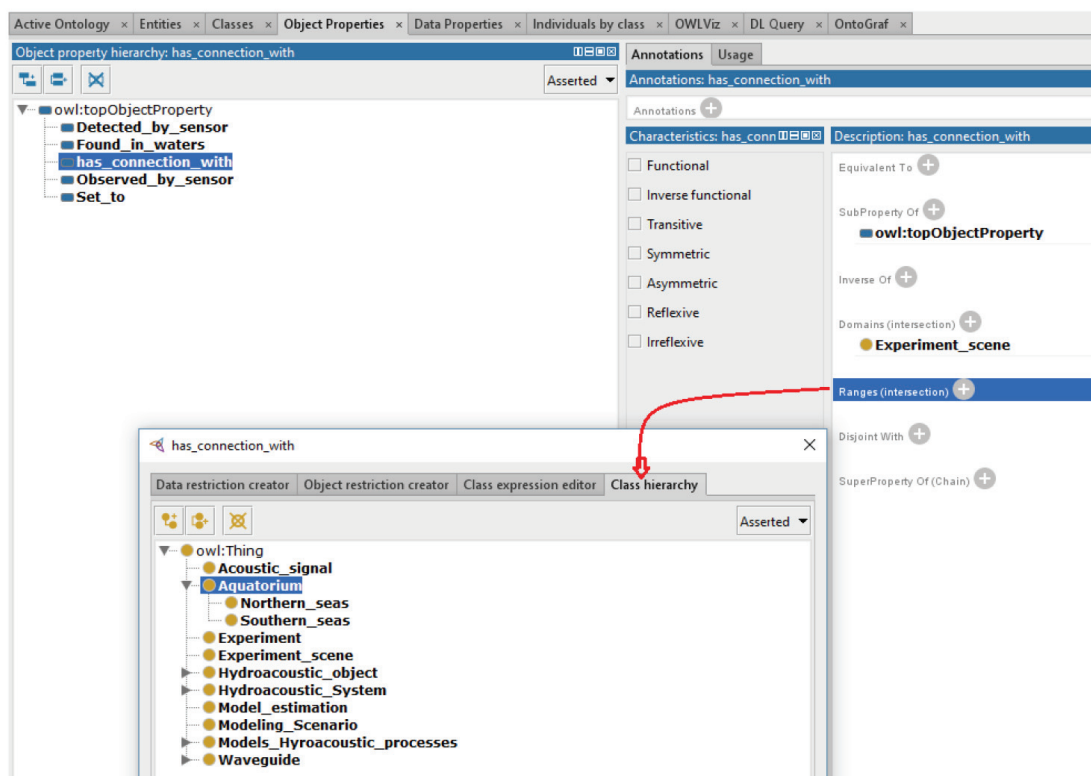
In the editor Protégé 5, two functions used to define the axiom of subject domain:

- Domains (intersection)—asserts that the subjects of such property statements must belong to the class extension of the indicated class description.
- Ranges (intersection)—asserts that the values of this property must belong to the class extension of the class description or to data values in the specified data range.

To create a subset of axioms, you must select the `<ObjectProperty>` tab in toolbar of the Protégé 5 and go to the `owl: topObjectProperty` line. In the `<owl: topObjectProperty>` subset, select the



a)



b)

Figure 15. Example of created relationship between classes.

desired relationship type, which binds two classes, for example, class *<Experiment_scene>* connected with class *<Aquatorium>* relationship *<has_connection_with>*.

To do this, on the *<ObjectProperty>* tab in the *<Description>* window, you need to click on the “+” next to the function *<Domains (intersection)>*. When you click “+”, a separate window opens where you want to select the *<Class hierarchy>* tab (**Figure 15a**) and define the class (*<Experiment_scene>*) in which the selected property is given.

When selecting second class, you need to click on the “+” next to the function *<Ranges (intersection)>* in the *<Description>* window. Further, in the opened window, you must select the class (*<Aquatorium>*) and click OK (**Figure 15b**). After completing the corresponding actions, a new relationship between the classes has created and stored in the model ontology of Subject Domain.

4. Formation of relationships between individuals

For formation of relationships between two individuals in the Protégé 5, you need to click in the *<Entities>* tab of toolbar. In the opened window *<Individuals by type>*, there is a class hierarchy that has instances. Next, in the class hierarchy (e.g., *<Helicopter>*), you must select an instance (subject) and open it (*<BoeingCH-47>*). Then, two windows open, that is, *<Description>* and *<Property Assertions>*. For formation of relationships between selected individual, you need to click on the “+” next to the function *<Object Property Assertions>*. Then, a new form to enter two fields *<Object property name>* and *<Individual name>* opens (**Figure 16**).

For example, individual *<BoeingCH-47>* characterized by a high level of acoustic noise, the limiting values of which are recorded in the individual *<Big_acoustic_noce>*. In the field *<Object property name>*, you need to enter the relationship *<Has_acoustic_noce>* and in the field *<Individual name>*, register individual *<Big_acoustic_noce>* and then click OK.

After completing the corresponding actions, a new relationship between the individuals has created and stored in the model ontology of Subject Domain.

Figure 17 presents The hierarchy of classes developed by means of the Protégé.

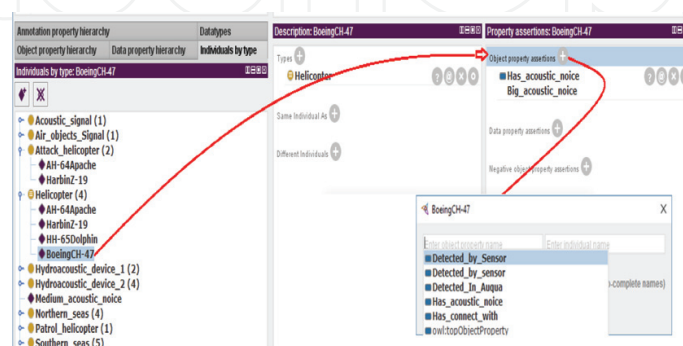


Figure 16. Process of creation of relationship between individuals.

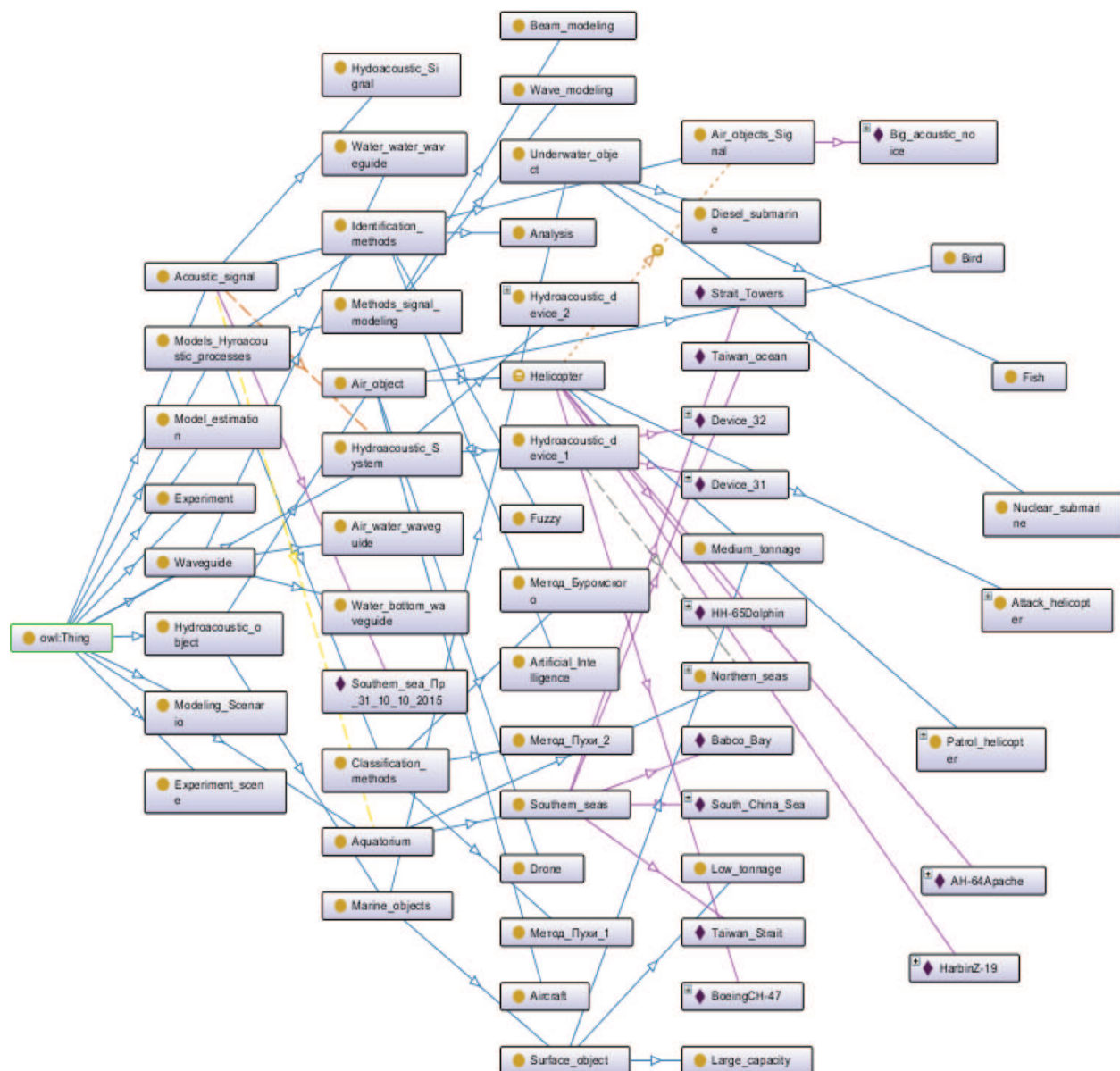


Figure 17. Graph of ontology model using the Protégé.

Subject domain ontology includes 34 classes, each of them has 5–15 attributes, nearly 30 axioms, 23 associative relations, 15 “is-are” relations, nearly 70 heredity relations and 50 “class-data” relations are described; standard types of attributes values and limits for values of attributes are highlighted.

For example, acoustic signal—class $C^{(Sig)}$ has the following attributes:

- Frequency of the signal
- Phase
- Amplitude
- Interference level

- Spectral-energy characteristics
- Vector-phase characteristics to identify the object
- Coefficient of total attenuation (pressure signals, speed along the axes: V_x , V_y and V_z)
- Location coordinates
- Distance to the object,
- Speed of propagation,
- Position of the object relative to the receiver,
- Pressure at the depth of the receiver,
- The results of modulation of the noise spectra of objects
- Coefficient of noise emission.

Thus, ontology represents the description of subject domain notions in terms of knowledge processing theory. It allows to describe the subject domain notions in terms of ontology classes, subclasses and classes exemplars as well as to define relations between them. After all ontology elements are described, it is possible to use algorithms of descriptive logic for ontology information processing.

5. Summary

The approach to the Internet portal in the field of materials' strength is presented. The usage of ontology model for portal knowledge representation allows to structure and systematize portal data and knowledge as well as to organize meaningful search through portal informational space.

The ontology model for the classification of marine objects was proposed, and elements of such ontology was described. The process of ontology development using the Protégé is depicted. The usage of ontology gives the possibility to execute the recognition of marine objects within the developed modeling system.

Author details

Larysa Globa^{1*}, Rina Novogrudska¹, Alexander Koval¹ and Vyacheslav Senchenko²

*Address all correspondence to: lgloba@its.kpi.ua

1 National Technical University of Ukraine "Kyiv Polytechnic Institute", Kyiv, Ukraine

2 Institute for Information Recording, Kyiv, Ukraine

References

- [1] Zagogulko YA, Borovikova OI. The technology of ontologies constructing for the portals of scientific knowledge. Gornal of NGU. 2007;5:12-15
- [2] Globa LS, Novogrudskaya RL. Engineering knowledge portal designing approach. Workshop Intellectual analysis of information 2010. Kyiv: KPI; 2010. pp. 53-62
- [3] Fridman YB. Mechanical properties of metal. M: Mechanical engineering. 1974
- [4] Semantic Technology. Available at: https://en.wikipedia.org/wiki/Semantic_technology
- [5] Knowledge Representation Languages: Available at: <http://www.cs.man.ac.uk/~stevensr/onto/node14.html>
- [6] Open Knowledge Base Connectivity: Available at <http://www.ai.sri.com/~okbc/>
- [7] Web Ontology Language: Available at: <http://www.w3.org/TR/owl-features/>
- [8] Senchenko VR, Koval OV, Globa LS, Novogrudska RL. Intelligent modeling system based on cloud-technology. Proceedings of International Conference Radio Electronics & Info Communications (UkrMiCo). DOI:10.1109/UkrMiC.2016.7739646. IEEE Digital Library, 2016
- [9] RDF Model Theory: Available at: <https://www.w3.org/TR/2002/WD-rdf-mt-20020429/>
- [10] Semantic Web Technologies. 2013: Available at: <https://open.hpi.de/course/semanticweb>
- [11] Protégé - Free, open-source ontology editor and framework for building intelligent systems: Available at <http://protege.stanford.edu>
- [12] Camel Case Notation: Available at https://en.wikipedia.org/wiki/Camel_case
- [13] OWL 2 Web Ontology Language Document Overview (Second Edition) W3C Recommendation 11 December 2012: Available at: <http://www.w3.org/TR/owl2-overview/>
- [14] Jureta I, Mylopoulos J, Faulkner S. Revisiting the Core Ontology and Problem in Requirements Engineering, In: RE; 2008. pp. 71-80

