ISSN 1995-0802, Lobachevskii Journal of Mathematics, 2019, Vol. 40, No. 7, pp. 861–875. © Pleiades Publishing, Ltd., 2019.

Information Model of LibMeta Digital Library

O. M. Ataeva[1*] **and V. A. Serebryakov**[1**]

(Submitted by E. K. Lipachev)

[1] *Dorodnicyn Computing Center FRC "Computer Sciences and Control", Russian Academy of Sciences, Moscow, 119333 Russia*

Received March 15, 2019; revised March 20, 2019; accepted March 24, 2019

Abstract—When developing a digital library, heavy emphasis is laid onto the library information model. The content of digital libraries can be featured in different formats and presented in different ways. The library, as defined by the LibMeta system, is a storage of structured and diverse data with the possibility of integrating it with other data sources and specifying its content by analysing the subject area. The semantic content ontology of the library provides basis for further content formalization. The article introduces the basic concepts needed to describe the problem of data integration, as well as concepts required for the definition of an unspecified thesaurus. The ontology's structure enables the possibility to determine the semantic library within any subject area.

DOI: 10.1134/S1995080219070035

Keywords and phrases: *semantic library, data model, ontologies, data sources, data integration.*

1. INTRODUCTION

In different subject areas, the data model of the content of digital semantic libraries can vary significantly both in types of resources and their structure respectively. While developing the aforementioned type of libraries, the data model of the library content is given emphasis.

While speaking about libraries, the authors primarily mean elaborated information system aimed at generating LibMeta semantic libraries [1–4], that creates the semantic library of a subject area and gives relevant description of it.

LibMeta is an information system that implements a set of features that are necessary to work with the content of a prospective semantic library. LibMeta is a special electronic library management system (ELMS). Modern technologies call for redefining both the concept and the content of the library, at the same time content may include traditional descriptions of printed works and any other types of digital content. The content of digital libraries can be featured in different formats and presented in different ways. The LibMeta-implemented library is considered as a storage of structured and diverse data with the ability to integrate it with other data sources with the possibility of specifying its content by defining the subject area.

A prospective subject area is defined by the thesaurus [5], which contains the basic concepts of a subject area, that are hierarchically and horizontally linked. The content of the library is prescribed by the types of resources involved, so that the description of which defines a set of valid objects, that are possibly united in a variety of collections that make up its content associated with a variety of relations with the concepts of the thesaurus.

The article features the study of the set of tools related to content of a semantic library. These tools are required to automate the descriptions of library resources for a specific subject area and the possibility of their chained integration with an external open data-source. The structuring and formalization of knowledge in the field of semantic library content description is a necessary condition.

[*] E-mail: oli@ultimeta.ru

[**] E-mail: serebr@ultimeta.ru

2. THE LIBMETA INFORMATION MODEL

The DELOS project (Digital Library Reference Model, DLRM) [6] is the most complete reference model of the electronic library. The basic concepts for the electronic library (specific DL, DL system, DL management system) were defined, at the same time the categories of users behind these concepts (developer, user, administrator) were identified. The following six main high-level concepts/areas were identified as well: (1) content, (2) user, (3) functionality, (4) quality, (5) policies, (6) architecture. The LibMeta personal open semantic library was developed with a user support system, that was aimed at helping users to work with digital resources of libraries and their collections for the subject area. The development of the library was based on the DELOS conceptual model and its definitions, as well as the ideas of Semantic Web and Linked Open Data. When implementing LibMeta, the developers were guided by a set of basic tasks that the developed system should solve accordingly as follows:

1) the library is required to be able to use media objects or refer to them when describing its objects, including text, audio, video, or any combination of aforementioned digital formats. This requirement is reflected in the title by the word ’digital’;

2) the types of resources used and the relationships between them should be described by means of the system within the framework of the concepts defined in the previous work that make up the semantic description of the library content resources. Moreover, according to the LOD principles, the use of classes and properties of previously used ontologies in the community that favors LOD is supported in the description of resources. This support is expressed either in the direct use of finalized ontologies in describing the resources and the relations between them, or in the possibility of referring to their elements using links at the resource description level. This requirement is reflected in the title with ’semantic’ adjective;

3) the library should serve as an integration node, linking its own data to the data from different sources that are included in the LOD cloud. The service should also be able to provide the option to extract data from this library in machine-readable format. This requirement is mirrored in the title by the adjective ’open’ respectively;

4) the library users should be able to organize their collections in the scientific-preference order, that they are most interested in, adding new terms to the subject thesaurus, thus specifying the scope of their interests. Besides searching among objects within the system, users should also be able to search by data sources, without having to use a specialized language for search queries. This requirement is reflected in the title by the adjective ’personal’.

Ontology of semantic library content serves as a mean of formalization [1, 7, 8]. Based on this specification, we can define the basic concepts of describing the task of data integration from open sources. The article features the basic requirements for such type of data source.

We consider the definition of the thesaurus and the basic standards as the main section of the present thesis. We aim to identify the concepts necessary to describe the content of the semantic library in any subject area as well as to define the basic concepts necessary to describe the problem of data integration from open sources, and to identify the main types of relationships between these concepts [5, 9].

LibMeta is characterized by the configurable metadata storage for its resources and the types of described information resources. The basic requirements for describing resources are as follows: versatility, structure, adaptability. Versatility is the independence of the description of its types of resources and objects from the subject area and the users’ area of interest. The structure of the description, on the basis of the LOD principles, provides support for a link between external and internal resources. Adaptability of resource descriptions allows for adding new properties and links in the process of system development and customizing of user interface to reflect perspective changes. Below are the basic concepts of LibMeta subsystems, which ensure compliance with the aforementioned requirements, derived from the formal model of the semantic library.

2.1. Content Description

The aforementioned definition of versatility of the system's content is based on the set of concepts which represents the LibMeta informational content model: *information resource* and *information object* that define a resource instance. The *information resource* is the basic descriptive unit of library content, and the *information object* represents instance of information resources. Each of them has its own unique identifier. In fact, the semantic meaning of the information resource is equivalent to the concept of *ontology class* with discrepancies in description. The structure of the description of information objects is determined by the concepts of an *attribute* and a textitset of attributes that are defined in the description of the corresponding resource. The attribute is an element of a resource property description, and the *set of attributes* is defined as a collection of different attributes.

Attribute types are as follows: *file, object, numeric, text, string*. In order to manage taxonomies, a new type of attribute is introduced—*taxonomic*, which will be described later in the corresponding section of this work. In addition to defining the range of valid attribute values, there is another important property: a type and a number of limitations of possible values.

These concepts provide a structured description of the content and support its adaptability. This approach also provides a description of specific resources and their objects in the form of RDF[1] triples and provides a SPARQL[2] access point for publishing data.

A specific implementation of the library content model can be based on a particular imported ontology, classes of which are transformed into resources and properties of which are described in terms of LibMeta attributes and sets of attributes, in fact, define ontology property domains. While constructing a library resource model based on such ontology, all the URIs of the properties, relations, and classes of the selected ontology are preserved. If necessary, when importing the selected ontology into the system, you can change the set of concepts, expanding or reducing it by means of the system accordingly. It is given that this method of mirroring ontology on to the concepts of the LibMeta system does not preserve the entire possible list of restrictions, but at the same time its structural part is preserved, which is sufficient to solve the problems defined within the system framework.

Figure 1 shows the basic concepts used to generate the description of the subject area. Some of these concepts will be explained below.

In order to generate a description of any subject area, there is always a certain set of terms, each designating or describing a concept from given domain. The thesaurus is a set of terms describing the subject area with the indication of semantic relations (connections) between the terms. Such relations in the thesaurus always indicate the presence of a semantic connection between such terms.

At the same time, the thesaurus model should not be focused on any of the specific subject areas and should be flexible enough to always keep the dictionary updated and easy to use to define any subject area. A thesaurus with different types of relations allows you to implement a flexible, customizable search that results in a list of domain objects corresponding to the selected terms. The model of a thesaurus at issue is compliant with ISO 25964. This standard defines a thesaurus as a set of terms that are related by their respective relations (links). Terms are expected to maintain the following attributes:

- SN—Scope Note. Note for a term. For example, it represents a verbal explanation of the term or its intended use.

- TT—Top Term. An attribute that identifies the terms at the highest level of the hierarchy (the terms of the most common concepts in the hierarchy of concepts). The relations between terms can be as follows:

- USE—associates the term with the most preferred term for the concept. While A USE B means that the term B is the most preferred term for the concept denoted by the term A.

- UF—Used for. An opposite relation from the USE. Links the most appropriate term with synonyms and quasi-synonyms (less appropriate terms).

[1] https://www.w3.org/RDF/.
[2] https://www.w3.org/TR/sparql11-overview/.

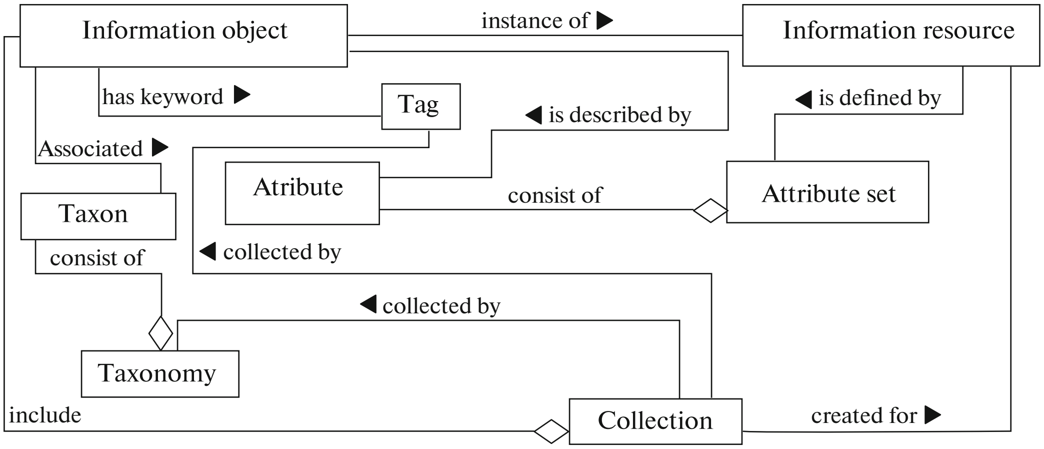


Fig. 1. Basic concepts of the LibMeta system.

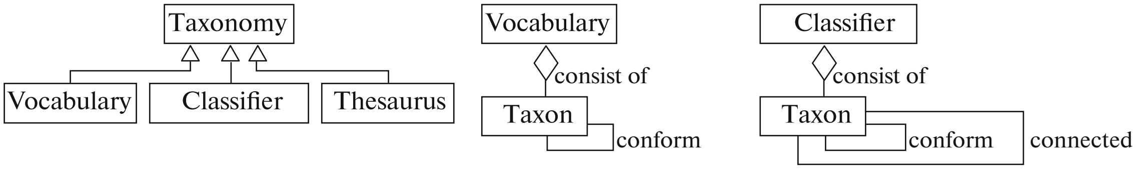


Fig. 2. Taxonomy subconcepts and default connections in taxonomies.

- BT—Broader Term. The connection of the term with the term of a broader concept. A BT B means that the term B denotes a broader concept than the one denoted by the term A.

- BTG—Broader Term Generic. A variant of the BT connection in the case where the term characterizes a variety of concepts defined by a broader term. For example, ’parrots’ and ’birds’. The BTG implies the BT connection in the first place. o BTP—Broader Term Partitive. A variant of the BT connection in the case where the term characterizes a part of the concept defined by a broader term. For example, ’mathematics’ and ’number theory’. The BTP implies the BT connection.

○ NT, NTG, NTP—Narrower Term, Narrower Term Generic, Narrower Term Partitive the opposite of BT, BTG and BTP, respectively.

○ RT—Related Term. Associative connection. Used for semantically related terms that are not in the same hierarchy and are not synonyms or quasi-synonyms. This connection is set when it may be relevant for the user of the thesaurus to search for or index not only a given term, but also a term associated with it.

In order to describe the thesaurus, additional concepts are to be introduced: taxon and taxonomy. A taxon is a taxonomy element with a specific set of properties required for its basic representation, and a taxonomy defines a set of available relationships between taxonomy components and system resources. Given the need to describe additional links between taxon, taxon links are introduced that allow the identification and description of new relations within the information system. By default, only two types of relationships between taxon are available in the system: hierarchical and untyped horizontal.

Figure 2 shows the taxonomy subconcepts based on the definition of a semantic library: *dictionary, classifier*, and thesaurus, and the default connections in taxonomies defined by taxon. For the thesaurus, it is possible to set more attributes as in figures: *T_Text Attribute, T_Object Attribute, T_Taxonomic Attribute*, these attributes allow to extend the definition of a taxon and to include information objects in its definition as well as to specify horizontal connections between taxon.

In order to determine the link of taxonomy with information objects, the concept of taxonomic attribute is introduced. This connection provides the ability to connect any of the taxonomies to any type of resources in the system. This approach, on the one hand, allows to avoid redundancy at the

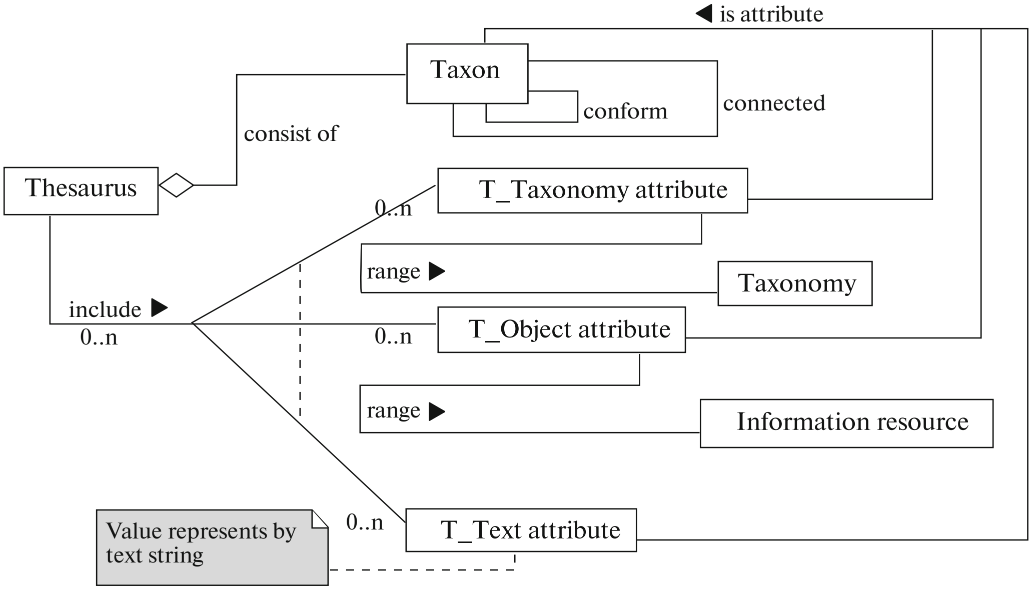


Fig. 3. Attributes of the thesaurus.

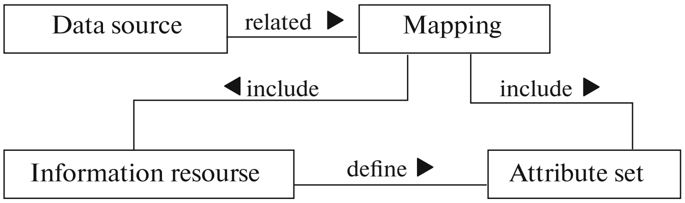


Fig. 4. Connection between the concept.

initial stage of system design, on the other hand, allows for representation of almost any connection type. Taxonomic attributes are specified in the description of a value domain of the information resources attributes.

Given the need to maintain a variety of collections of objects, the concept of a collection of information objects is used, which is determined on the basis of a taxonomy that indicates the types of resources to be collected. A collection can combine information objects of different information resources. You can define multiple collections based on the same taxonomy. This approach will be extremely relevant for creating separate user collections.

At the same time to solve the problem of data integration from external sources, we introduce the concept of a data source. Information resources of the system are aligned to this concept, meanwhile the ratio of the resource attribute set and the properties of the resource from the data source is established. This provides ground for us to generate SPARQL queries to data sources to extract specific information. In this case, the user operates with the typical kind of search options, avoiding the need to write the queries themselves.

In case a particular implementation of the library content model is based on an imported ontology and the ontology is used in the data source, then there is a mechanism for displaying the properties and ontology classes of the connected data set in LibMeta terms in a semi-automatic way. Thus, we form an integration node that allows for establishing a link with data sources. Figure 4 shows the logic behind the connection between the concept of data source and the basic concepts that define the content of the library. Figure 6 illustrates how the user interacts with the subsystem in order to obtain the results of their query.

3. LIBMETA LIBRARY ONTOLOGY

Ontology describes the resources of the subject area and their interconnections. For each subject area, the set of resources may differ both in format and in the set of resources themselves. Therefore,

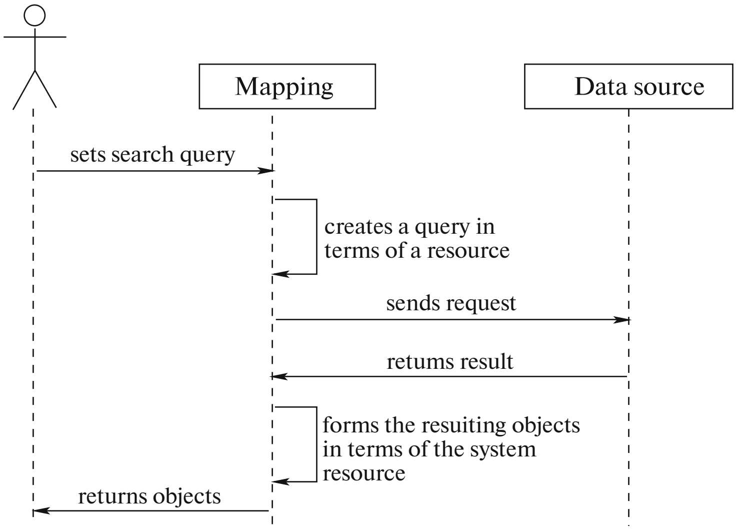

Mapping
Data source
sets search query
creates a query in terms of a resource
sends request
retums result
forms the resuiting objects in terms of the system resource
returns objects

Fig. 5. Interacts with the subsystem.

while setting the definition of the library, it is advised to use resources that make up the content of a particular subject area. Thus, the set of concepts that form the description of the library content should be so versatile that it can adapt to the needs of a particular area. Since one of the main tasks solved within the library, as mentioned above, is the integration of data from various sources, this approach allows for implementation of data integration tools within the library, while these tools are adaptable to the conditions of any subject area regardless of its specifics.

Concepts forming the LibMeta ontology library are divided as follows:

—content descriptions of the subject area;

—formation of the thesaurus of any subject area;

—descriptions of thematic collections;

—description of the task of integrating library content with data from external sources.

Semantically significant connections between these groups are defined.

Let us consider the basic formal definitions necessary to describe the ontology.

Definition 1. The *content* of the library $C = \langle IR, A, IO\rangle$ is determined by the types of its information resources described by the *associated attribute sets* A, and the set of input data that defines IO *information objects*, which are objects directly stored in the library.

Definition 2. The *library thesaurus* $TH = \langle T, R\rangle$ is defined by the *terms* T and the *relations* R between them. The set of T terms that make up the subject area description is strictly defined.

Definition 3. *Semantic marks* $M = \{mi\}$ of an information object are terms that are not included in the thesaurus but are necessary to specify the subject of the information object. Semantic marks are not related, opposite to the terms of the thesaurus, to each other or to the terms of the thesaurus, but provide an additional thematic division of information objects within the subject area.

Definition 4. The *objective* of the *data integration library* $IT = \langle DS, R, A, M, D, DS\rangle$ with external sources DS is determined by the types of library resources and a set of attributes A, the mapping M resources R on a data source scheme S and a set of relations of the DS with the data from the source.

Definition 5. A *collection of information objects* $C = \langle IO, T, M, DS\rangle$ is a set of objects unified on the basis of a set of the following features:

1. by their term of the subject area thesaurus;

2. by semantic marks;

3. by the data source from which the objects were received.

The collection can include objects of different resource types specified when describing the library content. At the same time, collections for each feature can be formed automatically and we call it automatic collections. In the case when the attributes are defined by the user, we will call such collections simply collections. The main classes of ontology that are based on the definitions were introduced. According to the 1st definition, the following classes are established:

1. ***IResource*** (library information resource), which contains general information about the resource type, name, URI,[3] and information about the attribute set used to describe the structure of the resource.

2. ***IObject*** (library information object), which is an instance of a resource with the composition of the attributes corresponding to a set of attributes of the associated resource. In order to describe the corresponding values for an information object, there is a multivalued value property, values of which are instances of the AttributeValue helper class that contains information about the specific value of the object as well as the corresponding attribute.

3. ***Attribute*** (attribute, element of information resource description), which has the following properties:

   a) *name*—self-explanatory;

   b) *type*—contains information about the value type of this attribute and can include values such as string, number, date, resource type (the values are objects of a selected resource type);

   c) *view*—indicates the scope of the attribute within the system. This property can have the following values: searching (participates in formation of search forms), identifying (obligatory value) and descriptive (contains additional information on the described object).

4. ***AttributeSet*** (a set of attributes that groups attributes that correspond to a single resource).

   Based on Definition 2, according to the previously described ISO 2788-1986 standard for thesaurus, the following classes are introduced.

5. ***Thesaurus*** (subject area thesaurus), contains general information about the thesaurus: title and authors (organizations and individuals). The presence of this entity allows you to download ready-made thesauruses without mixing them with those that may already be in the system.

6. ***Concept***—an entity containing information about the concepts of the thesaurus. Contains the following attributes:

   a) *Name*—the name of the concept. In the case that the concept does not have names presented in text form, any ID is used;

   b) *RepresentationType*—the type of representation of the concept. The concept cannot always be verbalized, sometimes it is much more suitable to use formula or image, so you need to be able to add concepts in any form;

   c) *Image*—self—explanatory;

   d) *Note* —self—explanatory.

7. ***ConceptGroup***—themes concepts of the thesaurus.

8. ***HierarchicalRel***—hierarchical relations that define the tree structure of the dictionary. Contains the attributes that control communication in accordance with the standard (BT, BTG, BTP).

9. ***FamilyRel***—horizontal connections. They set the typological connection between concepts and allow you to find publications on similar topics. It also contains attributes that define connections according to the standard (NT, NTG, NTP).

[3] https://tools.ietf.org/html/rfc3986.

10. ***PrefferedTerm***—descriptors of the concept. Each concept corresponds to a single descriptor in each language.

11. ***NonPrefferedTerm***—this includes synonyms. A single descriptor can have many synonyms. The visibility attribute is a property that is responsible for the visibility of the term. It has two values—global and private, global and private scope, respectively. This attribute is introduced to solve the problem of multiple terminologies—different people can name the same objects differently (even if these names are similar). In order to make it comfortable for users to work in the system, they are given the opportunity to create their own terms, in case there are none in the global part of the thesaurus. They can associate these terms with other terms from the global part and mark their publications with them. Thus, if two users have created keywords convenient for them in their local repositories, marked their publications with them and linked these keywords with the same term from the global thesaurus, they will be able to find and receive each other's publications using their own respective terminology.

12. ***Term***—is a generic class that combines descriptors and synonyms. It contains a set of properties that, if necessary, allows you to arbitrarily expand the text descriptions of terms and determine the relationship with the information objects of the system.

    Based on Definitions 3 and 5, the following classes are introduced:

13. ***SemanticTag***—class of semantic labels, which has the following properties:

    a) *title*—a brief title for the semantic label;

    b) *description*—extended description of the semantic label.

14. ***ICollection***—the collection class that possesses the following properties and is defined by a person:

    a) *name* —the name of the collection;

    b) *definition*—description of the collection;

    c) *resources*—types of resources included in this collection.

    Based on the 4th definition the following classes are introduced:

15. ***DataSource*** (external data sources)—a class that has the following properties:

    a) *name*—name of the source;

    b) *description*—description of the source;

    c) *url*—entry point for data extraction;

    d) *resourceMapping*—contains information about the types of resources displayed for this source and the corresponding source classes. The values are instances of the ResourceMapping class.

16. ***ResourceMapping***—a class that contains information about the information resources of the library displayed for the data source:

    a) *resource*—resource type displayed for the source;

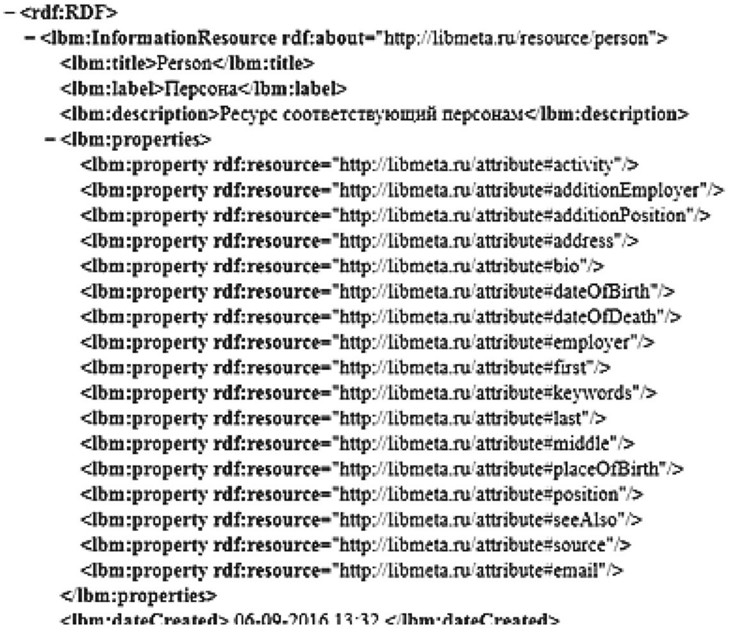
    b) *class*—reference to the corresponding data source class;

    c) *attributeMappings*—contains instances of the AttributeMapping class that has information about displaying attributes that correspond to the resource.

17. ***AttributeMapping***—a class that contains information about the attributes displayed for the data source from the set of attributes corresponding to the library information resource:

    a) *attribute*—the attribute that is displayed for the source;

    b) *property*—reference to the corresponding property of the data source class.



```
– <rdf:RDF>
  – <lbm:InformationResource rdf:about="http://libmeta.ru/resource/person">
      <lbm:title>Person</lbm:title>
      <lbm:label>Персона</lbm:label>
      <lbm:description>Ресурс соответствующий персонам</lbm:description>
    – <lbm:properties>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#activity"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#additionEmployer"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#additionPosition"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#address"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#bio"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#dateOfBirth"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#dateOfDeath"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#employer"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#first"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#keywords"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#last"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#middle"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#placeOfBirth"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#position"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#seeAlso"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#source"/>
        <lbm:property rdf:resource="http://libmeta.ru/attribute#email"/>
      </lbm:properties>
      <lbm:dateCreated> 06-09-2016 13:32 </lbm:dateCreated>
```

Fig. 6. Example of description of information resource in terms of the LibMeta ontology.

Figures 6 and 7 set examples of the description of the concrete information resource and information object in terms of this ontology according to Definition 1. Figure 6 sets as an example the description of the Person resource model. And Fig. 7 is an example of description of a specific person, according to its model.

4. EXTERNAL DATA SOURCE INTEGRATION

Based on the basic concepts, the content model of the library G is a set of resources $R = r_j$, a set of attributes $a = a_i$ and for each resource defined a set of attributes $N(r) \subset A$, that is $r_j(a_1, .., a_n)$, $a_n \in N(r)$. Each set of attributes includes so-called identifying attributes, for unambiguous identification of information objects of this resource let us denote them as $I(r) \subset N(r) \subset A$. We introduce the $simp(a)$ function, which assigns each attribute a type of its value and returns 0 for simple types, and 1 for object types.

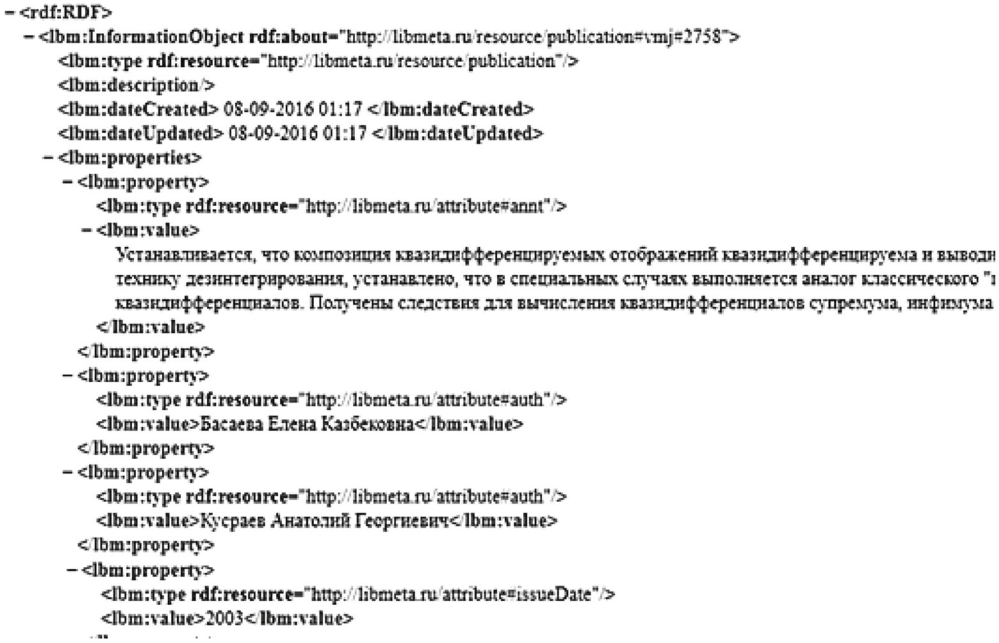
Formally, the I_T sub-system integration is represented by a triple of $\langle G, S_i, M_i \rangle$, where G is a predefined content model, consisting of a set of resources R and their descriptions in the form of a set of attributes $N(r)$, S_i schema of i-source connected to the system, M_i is the mapping of the i-source, $1 \le i \le n$, where n is the number of data sources.

The resource schema of both the S data source and the G content library can be represented as a graph that includes objects and connections. Each object can be related to another object values of which are represented by simple data types (strings, numbers, dates) or relations with other objects values of which correspond to some resources. At the same time, to display the resource, we can use its Z_s representation, which means to choose not a complete set of its attributes and relations with other objects to display on the G scheme, but which necessarily includes a set of attributes values of which allow to uniquely identify the object in the system.

The semantic relations between the Z_s representation of some resource corresponding to S and the resource corresponding to G define the elements of the data source mapping to the content library model. The semantic link m_i, that defines the mapping of the data source element S to the content element of the library $g \in G$, is denoted as $(m_i(Z_s) \Rightarrow s \sim g)$.

If $s \sim g$, where $g = r$, then to construct a working map of M resource r to source S it is enough to construct the following map:

$$M(S, r) = (m_r(Z_s) \Rightarrow \sim g = r) \vee (m_i(Z_s) \Rightarrow s \sim g = a_i) | r(a_1, .., a_n) \in R, a_i \in I(r).$$



```
- <rdf:RDF>
  - <lbm:InformationObject rdf:about="http://libmeta.ru/resource/publication#vmj#2758">
      <lbm:type rdf:resource="http://libmeta.ru/resource/publication"/>
      <lbm:description/>
      <lbm:dateCreated> 08-09-2016 01:17 </lbm:dateCreated>
      <lbm:dateUpdated> 08-09-2016 01:17 </lbm:dateUpdated>
    - <lbm:properties>
      - <lbm:property>
          <lbm:type rdf:resource="http://libmeta.ru/attribute#annt"/>
        - <lbm:value>
            Устанавливается, что композиция квазидифференцируемых отображений квазидифференцируема и выводи
            технику дезинтегрирования, установлено, что в специальных случаях выполняется аналог классического "
            квазидифференциалов. Получены следствия для вычисления квазидифференциалов супремума, инфимума
          </lbm:value>
        </lbm:property>
      - <lbm:property>
          <lbm:type rdf:resource="http://libmeta.ru/attribute#auth"/>
          <lbm:value>Басаева Елена Казбековна</lbm:value>
        </lbm:property>
      - <lbm:property>
          <lbm:type rdf:resource="http://libmeta.ru/attribute#auth"/>
          <lbm:value>Кусраев Анатолий Георгиевич</lbm:value>
        </lbm:property>
      - <lbm:property>
          <lbm:type rdf:resource="http://libmeta.ru/attribute#issueDate"/>
          <lbm:value>2003</lbm:value>
```

Fig. 7. Example of the description of the information object in terms of the LibMeta ontology.

In the case when $g = a$, the line $s \sim *g$ will be called precise, if the linking elements are identical in meaning, $s \sim +g$ is redundant if s is wider in the sense of g, $s \sim \hat{}g$—*inaccurate*, if s is already in the sense of g.

4.1. Set of Standard Operations for Rendering

Operation eq. If $simp(a) = 0$, $g = a$ and $s \sim *g$, then we will use the $eq(s, g)$ operation to estimate the coincidence of the values of the corresponding attributes of the object from the source with the value of the attribute of the information object from LibMeta.

Operation ext. In the case where $simp(a) = 0$, $g = a$ and $s \sim +g$ we will use the operation $ext(s, g)$ to extract the value of the object attribute from the source to search it for the corresponding occurrence of the information object attribute value from LibMeta.

Operation inc. In the case where $simp(a) = 0$, $g = a$ and $s\hat{}g$ we will use the $inc(s, g)$ operation to extract the values of the corresponding object attributes from LibMeta to find the corresponding occurrence of the object attribute value from the source.

Operation split. If $simp(a) = 0$, $g_i = a_i, g_j = a_j$ and for different g_i and g_i is performed $s_k \sim +g_i$, $sk + gj$, then the data from the source must be dissected to match the data of the final content schema, for this is determined by a pair of operations, $split(s_k, g_i), split(s_k, g_i)$.

Operation app. If $simp(a) = 0$, $g_k = a$, for different s_i and s_j and $s_i + g_k$, $s_j \sim +g_k$ is performed, then the data from the source must be combined to match the data of the final content schema, for this is determined by a pair of operations, $app(s_k, g_i), app(s_k, g_i)$.

Operation norm. If necessary, with $simp(a) = 0$, $g = a$, the operations $eq, ext, inc, split, app$ can use auxiliary operations to normalize the data according to the descriptions of the relevant attributes in the system, for example, $s' = norm(s)$, $g' = norm(g)$, $inc(s', g')$.

The set of operations $op = eq, ext, inc, split, app, norm$ perform the tasks of converting data from source system terms to source terms.

Operation see. In the case where $g = r$ and $s \sim g$, we will determine the connection of the resource from the source with the information resource of the object from LibMeta. We do not try to build a complete mapping of resources or their hierarchy, as in other approaches to the integration of schemes [10, 11]. We only need a connection, which means that instances of resources within the library and the data source can belong to the same object. This approach allows us to build partial resource mappings that are sufficient to uniquely identify objects and determine the relationship between them.

Operation res_eq. In the case of $simp(a) = 1$, $g = a$ and $s \sim *g$, we will use the operation $res_eq(s, g)$ to evaluate the coincidence of the objects—the values of the corresponding attributes.

Operation add_att. If for s there is no relevant g such that $g = a$ and $s \sim g$, we will use the $add_att(s)$ operation to add such $g = a$ to the resource schema for which $simp(a) = 0$ or $simp(a) = 1$.

Operation add_res. If there is no g to perform the add_att operation such that $g = r$ and $s \sim g$, for $simp(a) = 1$, we will use the add_res(s) operation to add a new resource to the scheme such that $g = r$ and $s \sim g$.

Operation add = add_att, add_res makes the integration subsystem customizable for any data source with the possibility of adding the data that is already available in the system. The dynamic definition of the G model actually includes the stage of analysis of the resources of the integrated data source and the expansion or refinement of the original G model by expanding the set of R or the set of A. The possibility of performing these operations is provided by the adopted adaptive data model of the system.

4.2. Map Building

The process of building the map can be divided into several main stages that are presented below:

—Data source connection. Each data source is characterized by a corresponding unique *URL* address and a set of parameters required to access the data. A preliminary analysis of the information available from the source is carried out and the types of its resources and their properties involved in the integration are determined. The result of the first step is to determine the part of the Si source schema by means of which the data will be extracted.

—Determining the types of library resources corresponding to the types of source resources. For each source resource defined by its schema that is extracted at this stage, the library resource is mapped. The result of this step is establishing a link between the library resource and the source resource by using the see operation, which declares that there are instances of these resources that correspond to the same real-world object. Based on certain $see(r, r_s)$ relationships, the next step is to map the attributes.

—For each resource, the mapping of attributes to the corresponding properties of the data source resource is defined. First, identifying attributes, which are mandatory, are mapped, then the others. For each pair (r, r_s), the operation see (r, r_s) is defined, and the type of communication is determined according to $s \sim g$, where $g = a_i$, and a set of $op(s, g)$ operations is defined.

This mapping let us get a set of rules relying on which we can represent each found object in the source within the concepts of our library and, accordingly, allow it to be stored in its entirety in local storage at the user's request, or simply keep the link between the found object in the source and the object in the library.

4.3. Queries to Integrated Data Sources

After building the map, it is possible to perform search queries on data sources. Any instance of the data source resource for which the mapping is built can serve as a response to a user request.

We define $I_T = (G', S_i, M_i)$ as a data integration system, and $L = L|S_i \in I_T$ as an interpretation of sources in I_T. If the data source S_i has a set of objects obj, which can be instances of the resource $r \in G'$, then at the request of the user q in terms of G' there can be built query q_L on the interpretation L, such that the answer to it will be the objects of the set obj, presented in terms of G'.

A separate problem of presenting a consolidated response to the user from different data sources is the identification of duplicate objects. For this purpose, identification attribute values are used. Of course, there is no optimal set of such attributes for any type of resource, but at least the expert can choose the optimal set of resources within a given subject area, on the basis of which it is decided whether the objects from different sources are identical or not.

The described integration model may not be optimal for any case and has its drawbacks typical for the general approaches described above. But within the framework of the LibMeta system boundaries and external data sources outlined by the statement, it was possible to achieve acceptable results in procedures to simplify the connection of search and navigation through such sources.

5. DESIGNING SEMANTIC LIBMETA LIBRARY

In order to be able to use the thesaurus of a specific subject area and the ontology of the library content, the following sequence of their use should be followed when constructing a semantic library within LibMeta system.

- Based on the introduced model, a set of information resources used in the library is given. It is necessary to provide descriptions of the content of the future library in terms of the proposed model. On the basis of the classes defined for the content-description of the library, the module is implemented in which you set the basic properties and attributes for resources and specify relations between them.
- The structure of the thesaurus is finally set up. On the basis of certain classes, according to the definition of the thesaurus, a module for building a thesaurus is implemented, in which the used links between terms are set, the term description is expanded if necessary, the links with the system resources are determined as well.
- To select semantic labels, you can use additional dictionaries on the subject area or leave their prospective definition (extension of definition) for later.
- On the basis of the specified classes, according to the definition of the integration task, a module is implemented, within which external data sources are connected. This action can be performed at any stage of the post-development phase of the system.
- On the basis of the specified classes, according to the definition of collections, a module is implemented, within which collections are created and filled. This action can also be performed at any stage. On the basis of the performed actions, the user interfaces of the system are automatically adapted to the specified descriptions of the resources that make up the content of the library. The user interface is divided into the following categories:
- search interfaces;
- viewing interfaces;
- editing interfaces;
- data loading interfaces.

6. EXAMPLE OF DEVELOPING

Let us consider a simple example of LibMeta library implementation based on publications from the electronic library 'Scientific Heritage of Russia' [4]. There are only two main types of resources that are defined for this data: person and publication. For the thematic classification of these publications, the SRSTI classifier is used and each publication has a UDC number.

The authors did not aim to create a reduced copy of the 'Scientific Heritage' library. The main objective pursued in the context of the proposed system is to link this data to external data and to publish it so that other systems can access it. This example uses DBpedia[5] as the data source for binding.

Thus, the main purpose in the developing of the content description is to present the description simpler in order to facilitate the implementation of the data search procedure in external sources.

[4] http://e-heritage.ru.

[5] http://dbpedia.org.

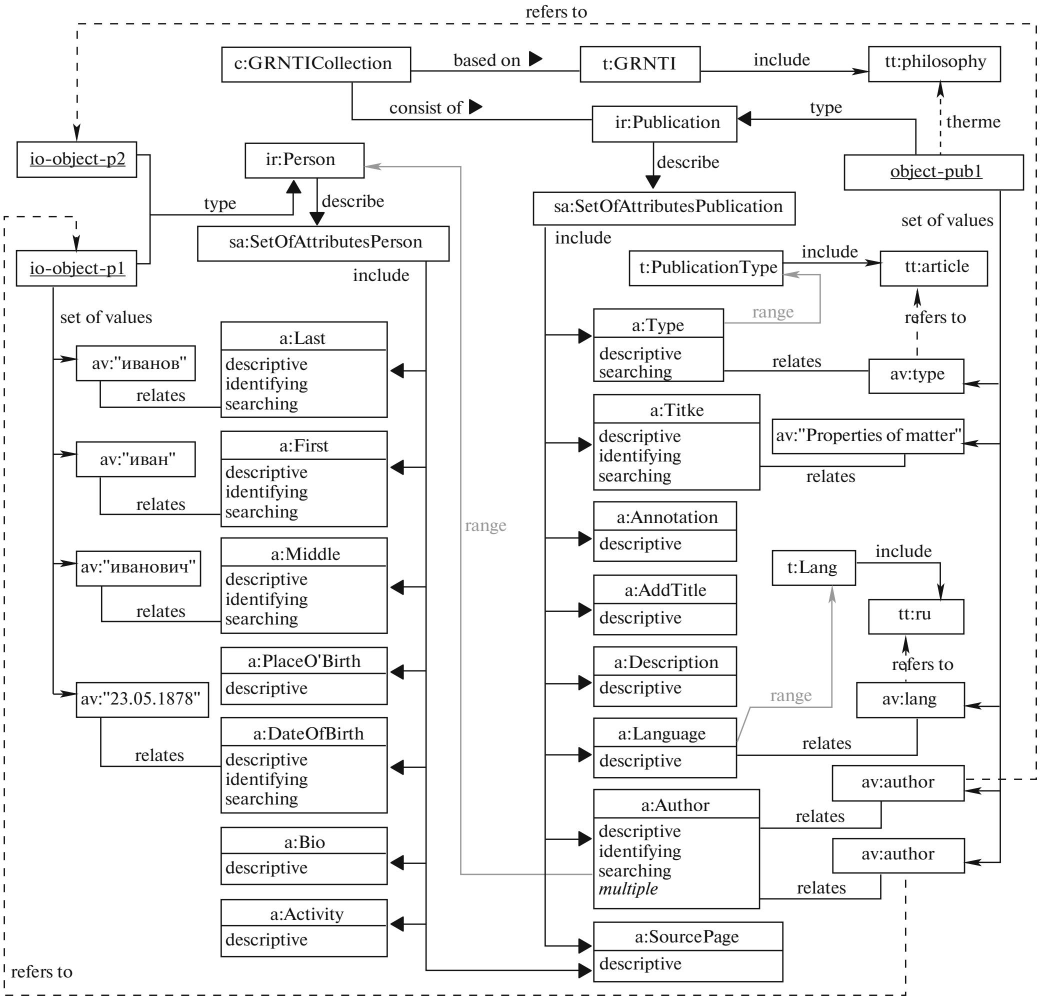


Fig. 8. Designing of Information Resources.

The drawback of this idea is, perhaps, some simplification of the content structure, in contrast to the expressiveness represented by means of the OWL[6] language, as will be shown below, but at the same time we get flexibility in building an integration node for different types of resources, the description of which can be expanded in the process of system's life.

In fact, the concepts of person and publication are instances of the information resource class defined as the basic unit of semantic library content. As each resource has a set of attributes, each of these instances has its own set of attributes that are pre-described in the system. The set of attributes for information resources consists of the following elements: name in the original language, name in Russian, surname, name, patronymic, email address, date of birth, abstract, ID, author, occupation, publication type, place of birth, biography, description, additional title, language.

Specific persons are objects that represent instances of the information object class, they are defined by the person information resource and are represented by the attribute values of the corresponding resource. In addition to the properties specified by the attributes represented in the attribute set of its information resource, each object also has properties that are common to all information objects, such as tags, description, creation date, modification date, owner, unique identifier.

Figure 8 shows simplified plan that is provided for these types of resources. The diagram illustrates the relationship between the *person* and *publication* instances of information resources and specific instances of the information object classes (object names object—p1, object—p2, object—pub1 are underlined). Prefixes *'io', 'ir', 'c', 't', 'tt', 'sa', 'a'*, separated by a colon, indicate that the instance

[6] https://www.w3.org/2001/sw/wiki/OWL.

belongs to the classes of *information object, information resource, collection, taxonomy, taxon, set of attributes, attribute*. For the thematic classification of the objects of the publication, a collection based on the classification of SCSTI was used.

Gray arrows coming from the instances of the attributes indicate the range of their possible values. The scope of the other attribute values are simple data types. In the plan, the attribute values are represented by the objects of the auxiliary class with the *attribute value* prefix *'av'*. Objects of this class contain their values for simple attribute types (for example, the values of the text attributes *last name, first name, title* are represented in quotation marks on the plan). For the *author* object attribute, its value contains a reference to the corresponding instance of the information object with the *person* type, which is shown in the plan with a dotted arrow. Taxonomic attributes *type, language* as a value scope indicate the corresponding taxonomy publication type and language, which are linear dictionaries elements of which (taxon) are used as attribute values.

For each attribute every type is specified: *descriptive, identifying*, or *searching*. An attribute can belong to multiple types at the same time. Searching attributes are used to dynamically generate a search form for objects of a certain resource type. Descriptive attributes are used to generate a form to represent information about an object to the user. A set of identifying attribute values is required, as the name implies, to identify an object. In the attribute set for *publications*, the author attribute is marked as *multiple*. This attribute can have several values when describing information objects that correspond to *publications* by resource type, which is reflected as an example in the plan.

Designing of the structure can be done using the user interfaces of the system or by loading RDF/XML with a description of the content structure in the appropriate section of the system by the user with the appropriate permissions.

In fact, LibMeta's original content ontology contains the necessary concepts, relationships, and axioms. When describing a specific subject area, separate instances of the concepts defined in the ontology are added to the ontology, which make up the content of the library that is being generated.

7. PROSPECTS

Currently there is undergoing work with the full texts of the submitted articles. One of the objectives is to generate the information pattern of articles to highlight microthesaurus based on the semantic labels and terminology of the domain for each article, with further definition of the extensibility of the used thesauri or key concept cloud of different areas of knowledge. Another objective is to semantically process the formulas from the full texts and define their keywords with the prospect of further search by formulas, highlight different approaches and mathematical schools, bearing in mind that the formulas are considered as a separate type of system resources.

FUNDING

The research is partially supported by the Russian Foundation for Basic Research, projects nos. 17-07-00214 and 18-00-00297 komfi.

REFERENCES

1. O. M. Ataeva and V. A. Serebryakov, "Basic concepts for building a formal model of semantic libraries and describing integration processes in it," Program. Produkty Sist., No. 4, 193–200 (2015).
2. V. A. Serebryakov and O. M. Ataeva, "Libmeta personal digital library as an related open data integration environment," in *Proceedings of the 16th All-Russia Conference on Digital Libraries: Promising Methods and Technologies, Digital Collections RCDL'2014* (2014), pp. 66–71.
3. V. A. Serebryakov and O. M. Ataeva, "Information model of the open personal semantic library Libmeta," in *Proceedings of the 18th All-Russia Conference on Scientific Service on the Internet, Novorossijsk, Sept. 19–24, 2016* (IPM im. M. V. Keldysha RAN, Moscow, 2016), pp. 304–313.
4. O. M. Ataeva and V. A. Serebryakov, "Ontology of the digital semantic libraty LibMeta," Inform. Primen. **12**, 2–10 (2018).
5. M. H. Nguen and A. S. Adzhiev, "Description and use of thesauri in information systems, approaches and implementation," Elektron. Bibliot. **7** (1), 16–45 (2004).

6. L. Candela, D. Castelli, M. Dobreva, N. Ferro, Y. Ioannidis, H. Katifori, G. Koutrika, C. Meghini, P. Pagano, S. Ross, M. Agosti, H. Schuldt, and D. Soergel, The DELOS Digital Library Reference Model Foundations for Digital Libraries, IST-2002 2.3. 1. 12. Technology-enhanced Learning and Access to Cultural Heritage, Version 0.98, December 2007. `http://www.delos.info/files/pdf/ReferenceModel/ReferenceModel/DELOS_DLReferenceModel_0.98.pdf`. Accessed 2019.
7. T. R. Gruber, "A translation approach to portable ontologies," Knowledge Acquis. **5**, 199–220 (1993).
8. V. A. Serebryakov, "What is a semantic digital library," in *Proceedings of the 16th All-Russia Conference on Digital Libraries: Promising Methods and Technologies, Digital Collections RCDL* (OIYaI, Dubna, 2014), pp. 21–25.
9. E. I. Moiseev, A. A. Muromskii, and N. P. Tuchkova, *Information Search with Thesaurus in Application Area of Ordinary Differential Equations* (MAKS Press, Moscow, 2005) [in Russian].
10. L. Zhao and R. Ichise, "Ontology integration for linked data," J. Data Semantics **3**, 237–254 (2014).
11. L. Zhao and R. Ichise, "Integrating heterogeneous ontology schema from LOD," in *Proceedings of the 26th Annual Conference of the Japanese Society for Artificial Intelligence, 2012*.