Repository of Complex Information Objects as a Source for Development of Semantic Applications

Julia Rogushina[1,2], Anatoly Gladun[3]

[1] Institute of Software Systems of the National Academy of Sciences of Ukraine, 40, Ave Glushkov, Kyiv, 03181, Ukraine
[2]Institute for Digitalization of Education, National Academy of Educational Sciences of Ukraine, 9 M. Berlynskoho Str., Kyiv, 04060, Ukraine
[3] International Research and Training Center for Information Technologies and Systems under NAS and MES of Ukraine, 40, Ave Glushkov, Kyiv, 03680 GSP, Ukraine

Abstract

This paper addresses the challenge of reusing ontological knowledge in semantic web-oriented systems and analyzes the issues encountered in retrieval of some structured objects in external ontologies. We propose the model of repository oriented on representation of complex information objects that enhances the functionality of ontology repository search services at the content level, taking into account the structure elements of these objects and the semantic relations between them. The paper outlines the fundamental requirements for repository and provides an example of its practical application: the development of a semantic web-oriented system called ActiveBook, designed to retrieve educational materials integrated from external sources. The prototype of this system facilitates the selection of relevant textbooks for lecturers and students in educational institutions, aligning with academic programs. It is implemented on base of semantic extension of Wiki technology, with the structural elements of complex information objects sourced from relevant external ontologies.

Keywords

Ontology, complex information object, semantic Wiki, learning object

1. Introduction

The development of various *information analytical systems* (IASs) increasingly relies on obtaining information about subject domains from external *information resources* (IRs) with dynamic structures and content. To automate the acquisition and structuring of information from these IRs, we propose an ontological approach that facilitates the extraction of domain knowledge from external *knowledge bases* (KBs) for web-oriented applications of varying complexity. Formal models of *information objects* (IOs) processed by IASs help standardize and provide a clear interpretation of their semantics and content.

Ontologies serve as a theoretical foundation for defining the domain-specific structure of IOs and the relations between them that are critical for retrieval and comparison. They establish a unified terminology for content processing and define semantic connections with other information sources, such as encyclopedias, regulatory documents, and classifications [1].

2. Complex Information Objects

IO, in its most general sense, represents a formalized abstraction of data that describes various types of material and virtual entities characterized by different properties. The choice of characteristics and methods of representation of IOs depend on purposes of their use and capabilities for processing them. From the perspective of ontological analysis [2], IOs are represented by ontology classes and class instances. Ontology classes are defined by their structure represented as a set of properties and their characteristics, as well as possible relations with other classes. Instances of ontology classes can

ITS-2023: Information Technologies and Security, November 30, 2023, Kyiv, Ukraine
ladamandraka2010@gmail.com (J.Rogushina); glanat@yahoo.com (A.Gladun)
0000-0001-7958-2557(J.Rogushina); 0000-0002-4133-8169(A.Gladun)

also have values of properties.

Many practical tasks require analyzing complex sets of information where IOs are interconnected through specific relations and constraints. Some of these tasks align with the concept of semantic search [3], where the goal is to generate a non-empty set of IOs belonging to the same class that meet specific conditions, possibly ranked by relevance. In more generalized cases, the result is a set of complex information objects (CIOs) — collections of IOs of different types that adhere to certain conditions regarding the relationships between the IOs within these collections and their properties.

CIO models can be categorized into four groups based on their sources:

- manually created by IAS developers based on their understanding of the problem, without external knowledge;
- manually created by IAS developers on base of some external information sources such as relevant KBs and IRs (ontologies and semantically annotated IRs).
- generated automatically by processing of external IRs (for instance, such as ontologies generated on base of the semantically marked Wiki pages);
- obtained from external repositories with expressive metadata that allows automated matching of problem constraints with contextual elements.

Each CIO instance consists of two or more IO instances linked by domain ontology-defined relations, satisfying requirements for the structure and property values of the IOs within the CIO. Examples of CIOs are:

- An organization, its employees, and the projects it undertakes, where IO types are “organization,” “project,” and “employee”;
- Hierarchically related organizational units performing common tasks using shared technical resources, with IO types like “organization,” “task,” and “equipment”;
- The infrastructure of a settlement, including its support systems and personnel;
- An educational institution, its staff, equipment, offered specialties, disciplines, and competencies, with primary IO types such as “competence,” “discipline,” “person,” and “specialty” [4].

We use formal model of domain ontology O_{domain} to define CIO formal model:

$$O_{domain} = < T, R, F >, \tag{1}$$

where

- T contains a finite non-empty set of domain concepts, divided into classes T_{cl} and class instances T_{ind};
- R defines a finite set of relations between class instances from T_{ind};
- F is a finite set of interpretation functions defined on the terms and relations of ontology from T and R.

This CIO model uses non-empty subset of elements of the ontology O_{domain} (1) that separates a set of instances T_{ind} from selected subset of ontology classes from T_{cl} on base of subset semantic relations from R between these instances.

We propose to use CIO model with the following structure:

$$C = < T_C \subseteq T, N_C, R_C \subseteq R, \{(t_j, t_k, r_m), t_j \in T_{ind}, t_k \in T_{ind}, r_m \in R\} > \tag{2}$$

where $T_C = \{t_i, i = \overline{1,p}\} \subseteq T$; $N_C = \{n_i, i = \overline{1,p}\}$.

Main difference of CIO model (2) from model of domain ontology (1) is fixation of positions of classes and class instances into CIO structure. These positions indicate all CIO structural elements by unique names from $N_C = \{n_i, i = \overline{1,p}\}$. Each CIO element has some fixed set of characteristics that define mandatory and multi–values properties, restrictions on intersection with values of some other CIO elements, etc.. If separation of IO positions is not important for domain, several formally different CIOs are joined in single object.

For instance, if structure of CIO "Discipline textbook" contains mandatory IOs of class "Person" denoted by the names "Author of the textbook" that can have more than one instance, their order is not significant for the search, i.e. textbooks that differ only by order of information about the authors are not different at the content level. Conversely, in CIO "Researcher Publications," where order of authors is significant (e.g., first or second author), such distinctions are crucial because they differ papers where some person is a first author, a second one, etc.

3. External Sources of Knowledge About CIO Structure

A wide range of ontologies, each characterized by varying levels of expressiveness and detail based on their development goals, is currently available. These ontologies often use open formats and provide open access, with numerous tools supporting the extraction of information about the structure and instances of complex information objects (CIOs) to facilitate knowledge reuse.

The primary challenge lies in search expressiveness: ontology metadata typically describes the ontology as a whole, limiting searches for CIOs within ontology to name-based queries. However, many tasks require more advanced search capabilities to find not only individual ontology classes or their instances but also sets of instances linked by specific relationships. This need highlights the importance of creating CIO repositories that support detailed search functionalities, similar to those found in ontology or document repositories [5].

These repositories can be enriched with information from external ontologies and semantically annotated documents, provided there is a formal mapping between semantic markup elements and ontology elements. Additionally, domain experts can manually enter or edit information about CIOs.

From an ontological analysis perspective, the following functions are essential for CIO repositories:

- search for classes with some defined set of object properties;
- search for class instances with specific property values;
- search for instances of selected classes that have some defined relations with instances of other selected classes;
- check for the existence of class instances that meet specific conditions;
- search for semantically similar classes and compare their instances using various measures of semantic similarity.

An essential condition for the effective operation of such repositories is their openness and the ability to be populated by various external users and information resources (IRs). The technological foundation for these repositories should adhere to FAIR principles [6], which emphasize data that is Findable, Accessible, Interoperable, and Reusable. Unlike Open Data, that is available to everyone without restrictions, FAIR allows for controlled access under specific conditions, offering greater flexibility in managing data throughout its lifecycle. Although FAIR is primarily aimed at scientific research data, it can also be applied to other IRs, such as educational materials (textbooks, manuals, and reference books) and online encyclopedias and vocabularies.

Therefore, the technological basis for CIO repository can use semantically extended wiki technology because this software provides development of IRs that satisfy FAIR requirements [7] and support semantic search by the categories and values of properties of IOs represented in these IRs.

4. Scope of CIO Repositories

The development of a CIO repository should provide the infrastructure and automate the semantic analysis of the ontologies it contains. The main objectives of the CIO repository are:

- to reduce the time required to retrieve information about the structure and properties of IOs that can be utilized by applied IASs;
- to harmonize the terminological basis and enhance the interoperability of knowledge processed and created within such systems;

- to improve the reuse of previously acquired knowledge and increase the relevance of search results.

The functionality of CIO repository differs from ontology repository by providing:

- search capabilities for ontological classes that are similar to selected ones (by structure, name, instances, superclasses, subclasses, etc.);
- search for all classes across different ontologies that contain one or more selected instances (search by sample);
- search for instances of similar classes from different ontologies;
- finding classes and instances that can define values for instance properties of a selected class;
- finding ontologies that contain instances of selected classes connected by certain relations (CIO prototypes).

The functions of the CIO repository are not limited to this list, but most of them can be considered as different subtypes of semantic search, where constraints and results are specified in terms of classes and class instances of domain ontologies and relations between them.

5. Functions of ActiveBook System

The web-oriented information system ActiveBook is designed to find and provide access to educational materials – such as textbooks, reference books, and methodological instructions – to students and lecturers at Ukrainian universities and other persons that take part in learning process. The objects of its operations can be considered as CIOs that contain information about IOs of various types like individuals, disciplines, competencies, and educational organizations. This system addresses a practical task in IAS development, requiring external knowledge that cannot be fully satisfied by the functionalities provided by ontology repositories.

The users of this system are divided into several groups:

- students searching for educational literature that meets their current information needs (pertinent to specific educational module, discipline, or thesis in a particular specialty, etc.);
- lecturers selecting the most relevant and up-to-date textbooks for their courses or analyzing the need for creating their own educational materials;
- other university department employees selecting appropriate textbooks for each module, determining the need for textbooks in the university library, or planning the development of original materials;
- book publishers (both private and state publishing houses) who can provide information about textbooks they have published or analyze the need for new editions.

The content for the "e-Textbook" system can be provided by:

- book publishers and authors, who can register textbooks they publish and set access terms;
- lecturers and other university department employees, who can register textbooks in their specialty (whether created by them personally, in co-authorship, or found in open access).

Main CIOs processed by the ActiveBook system are: "textbook", "author", "publisher", "specialty", "competency", "learning course (module)".

Currently, various educational portals and electronic libraries of educational institutions, information about LO ("Textbook", "Monograph" CIOs, etc.) and the field of their use ("Discipline", "Competence", "Specialty" CIOs, etc.) are represented by different sets of properties, and their similar properties are represented by similar but different names. The use of external knowledge bases is caused by the need to unambiguous and interoperable definition of the structure of the main objects and subjects of this IAS in order to enable the exchange of information with other applications and its automated export from various IRs (such as libraries of other universities). External ontologies can help to unify the semantics of metadata about the LOs submitted in the system.

ActiveBook should support semantic-level retrieval and comparison of CIOs that include attributes of IOs, such as "Textbook," "Specialty," "Competence," and more. This allows searches not only by textbook titles or specialty codes but also by additional parameters such as publication year, language, difficulty level, and integration with other disciplines.

We propose implementing ActiveBook as an analytical web portal, where all content additions and updates are automatically reflected across related pages. Semantic markup ensures content findability and integration, facilitating meaningful navigation. Access to personal information is restricted to specific user groups based on their access levels. For example, a general textbook rating is visible to all users, but only portal administrators can see which users submitted specific ratings.

6. Libraries of Educational Institutions and Learning Content

Most higher education institutions currently maintain their own electronic libraries, which contain various educational materials corresponding to the specialties offered by this institution. We analyzed several university libraries and found that they have different structure of knowledge base, use different technological frameworks, and offer different navigation approaches. These differences complicate integration efforts and global searches across multiple systems

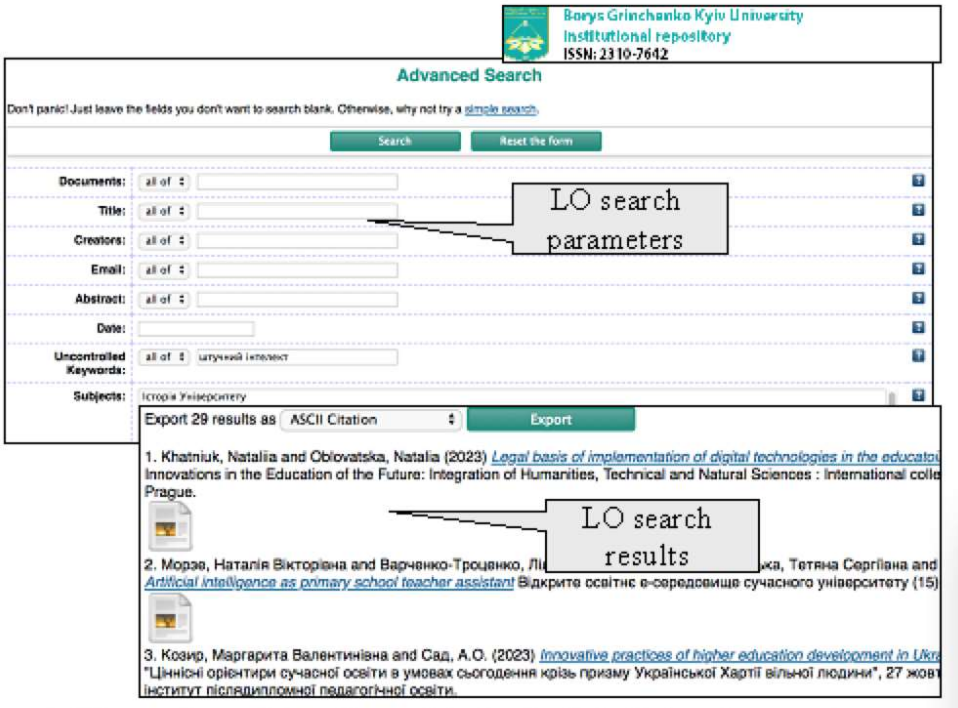

Figure 1: Advanced search into repository of Borys Grinchenko Kyiv University

For instance, the institutional repository of *Borys Grinchenko Kyiv University* (https://elibrary.kubg.edu.ua/) hosts electronic training courses, personal profiles of students and lecturers, evaluation journals, and a catalog of proposed disciplines. The available mobile application allows task processing offline and is designed to accumulate, systematize, and electronically store the intellectual products of the university's scientific community, making them accessible via Internet technologies. This open-access resource is hosted on the university's server and is available globally, at any time. It is registered in the Registry of Open Access Archives (ROAR) and the Directory of Open Access Repositories (OpenDOAR), and is indexed by the European search service BASE (Bielefeld Academic Search Engine). The repository provides a complex search function (elibrary.kubg.edu.ua/cgi/search/advanced) that allows searching by names, titles, subjects, and keywords (Fig. 1). However, this search is not integrated with an ontological representation, and it does not allow the selection of search results based on their form or composition. Additionally, the system only offers full-text documents, not metadata, and lacks the ability to export results into Semantic Web formats.

National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute" (NTU-KPI) also offers several scientific and educational information resources. The electronic catalog (https://kpi.ua/1338-1) provides an advanced search system for accessing electronic textbooks. NTU-KPI's institutional repository (https://ela.kpi.ua/) serves as an electronic archive of scientific and educational materials.

Search capabilities include filtering by authors, departments, directions, years of publication, and keywords, but it does not support complex searches based on multiple parameters. Furthermore, the software implementation lacks a clear visualization of the textbook information structure or the relationships between them.

Another example is *Dmytro Motornyi Tavria State Agrotechnological University* (http://www.tsatu.edu.ua/biblioteka/), which integrates scientific libraries of its territorial units. The library (http://elar.tsatu.edu.ua/) contains about 12000 documents, including educational and methodological materials, monographs, conference materials, and lecture notes, indexed by external search engines. However, it does not support searching by arbitrary parameters and only offers full-text access in PDF format to registered users.

Most Ukrainian higher education libraries are not integrated with each other, are implemented on different platforms, and differ significantly in interface and structure. These issues hinder semantic search, making it difficult for students or lecturers to find and reuse materials across institutions.

It should be noted that some existing systems oriented on support for the learning process partially or fully solve these problems. Currently, a number of such integrated applicable systems are developed for LO search. They differ significantly with functionality, knowledge representation models, and the scope and focus of content. For instance, [8] describes the online service Evdoxus designed to provide university textbooks to students. It provides: a) exact search of information about textbooks for each course/module; b) simple access to textbooks for students; c) relevant mechanisms for needs of publishers; d) dynamic and actual distribution of free e-books and applications; e) prevention of abuse of state resources; and f) more transparency and less bureaucracy. Service can be used by book publishers who register their textbooks, by lecturers who search for appropriate textbooks for their courses, by department staff who register textbooks selected by lecturers for each module of the curriculum or course, and finally, by students after registering in the university information system and choice of appropriate learning modules. But they do not have a Ukrainian-language interface, they do not support the existing Ukrainian means of classification of learning disciplines and specialties, and these features complicate the possibility of their integration with LOs in Ukrainian.

7. Problem Definition

The analysis of university libraries highlights significant variability in metadata structures and descriptive parameters for learning objects. Libraries use different parameters to describe the same of similar their content and such differences cause a need to build a more universal structure of main CIOs to integrate all information available for search without loss of important parameters.

Therefore, we consider the existing approaches to CIO structuring in various ontologies to choose such representation form that integrates main significant properties of conventional structures. Search expressiveness proposed by repositories of ontologies is not sufficient to find structural elements that correspond to some selected CIOs (or this task is not trivial). Key Issues:

- Inconsistent Metadata: Different libraries use varied parameters to describe similar content, complicating integration.
- Search Limitations: Ontology repositories often lack the expressiveness needed to locate CIOs based on complex structural relations. Manual analysis of classes and subclasses is frequently required to map relevant CIOs.

- Interoperability Gaps: Current systems do not fully support semantic integration or automated data exchange across repositories.

Proposed solution is based on developing a universal CIO structure that incorporates essential properties from existing metadata models and supports semantic search to enhance integration and retrieval capabilities.

8. Search for CIO Structure Knowledge in Ontology Repositories and Knowledge Graphs

The development of CIO repository relies on key theoretical and technological advancements in intelligent information processing. Key technologies supporting this development are:

- standards and toolkit of the Semantic Web project that provide interoperable means of representation and processing of distributed knowledge on base of the ontological approach (OWL, RDF, SPARQL);
- Linked Data concept [9];
- FAIR concept for providing access, search and reuse of data and documents;
- existing ontological models of CIOs that formalize LO structure;
- knowledge graphs;
- semantic extensions of wiki technology that provide the possibility of semantic markup of natural language texts and multimedia with tags of an arbitrary ontological structure and creating templates for representing instances of such CIOs that can be used as a source of information about the structure of CIOs.

All these elements can be used by CIO repositories for representation of content elements and for unification of its structure. RDF (Resource Description Framework) is a standard that provides the ability to formulate statements as suitable for computer processing based on the "object-attribute-value" data model for metadata [10]. Widely used in a variety of fields, RDF Schema enables developers to define a specific vocabulary for RDF data and the kinds of objects that can use these attributes. In other words, the RDF Schema engine provides an underlying type system for RDF models. On its basis, such large knowledge bases as Dbpedia and Wikidata become available. In addition, search engines such as Google and Bing also support RDF. OWL (Web Ontology Language) ontology is defined as an ordered set of axioms, facts and links to other ontologies that can be defined by their URIs. OWL extends the capabilities of such formal languages as XML, RDF, RDF Schema and DAML+OIL [11].

Linked Data is a set of advanced methods for publishing and combining structured data on the Web. These methods that integrate various best practices selected by data providers are aimed on development of global information space named the Web of Data. Linked Data is based on two fundamental web technologies: Uniform Resource Identifiers (URIs) and Hypertext Transfer Protocol (HTTP). As opposed to URLs, use of URI provides a more general means of entity identifying.

URIs and HTTP are complemented by RDF technology critical to the Web of Data. While HTML provides means of structuring and linking documents on the Web, RDF provides a general graph-based data model that can be used to structure and link data describing a variety of entities. Data representation of RDF model is based on triples "subject-predicate-object" where subject and object use URIs for resource identification or contain constants. Triple predicate defines relation between subject and object, and it also uses URI for identification of predicate resource.

Knowledge graphs (KGs) provide a powerful way of representation structured knowledge and integrating information from different sources [12]. Every KG consists of non–empty sets of nodes and edges: every KG node represents some concept, and each KG edge represents some connection between two concepts.

KGs are used in several of tasks that require semantic processing of information. Examples of these tasks are semantic retrieval, data analysis, generation of recommendations, natural language processing and pattern recognition. One of the important key aspects of KG approach is the use of ontologies to provide a formal representation of entities and their relations. Ontologies provide a logical derivation of KGs, as well as a check of their consistency. In addition, integration of KGs with different sources on base of ontological analysis provide interaction schemas and common term systems of domain.

The use of KG supports the Linked Data concept by defining relations linking IO instances of the system with corresponding DBpedia records, using the Wikipedia and DBpedia SPARQL search engine. Such KGs correspond to lightweight ontologies that can be processed by many open tools. Therefore KGs can be used for development of various repositories for integration of information from various external sources. In order to automatically integrate data from such different sources such as libraries of educational institutions and catalogs of publishing houses, KGs and RDF format can be used as technological means of converting them into open formats of knowledge representation.

Linked open data (LOD) [13] is an expansion of Linked data concept based on KGs. Main aim of LOD is to transform the Web data into more interoperable, accessible and reusable representation according to Linked Data principles: concepts are identified by URIs, HTTP provides their search, and RDF is used for data structuring with controlled vocabularies [14]. LOD supports data combining from heterogeneous sources and domains, performs semantic queries. LOD datasets examples: DBpedia (https://www.dbpedia.org/), Wikidata (https://www.wikidata.org/), and GeoNames (https://www.geonames.org/).

9. Semantic Information Resources for CIO Structure

The development of the knowledge base structure for any IAS can benefit from the use of external ontologies to define the structure of CIOs in accordance with established standards. For instance, LO search can leverage external ontologies that represent various aspects of knowledge about the educational process, subject domains and learning specialties. Unfortunately, direct search for such information in relevant ontologies and in ontology repositories is not a simple task that can be performed automatically.

Currently, a lot of ontologies with various characteristics describes research activities, education and scientific publications [15], but most of such academic ontologies are related only to scientific research and publications (for instance, Microsoft Academic Knowledge Graph [16]) and Open Research Knowledge Graph [17]), and only a few of them include aspects related to educational process and specifics of LO search for learning courses with selected properties, educational management, and learning technologies [18].

For instance, the *Linked Open Vocabularies* (LOV) repository (https://lov.linkeddata.es/dataset/lov) that ensures the reuse of information resources in the field of scientific research and education [19] currently contains 782 ontologies, and only 6 of these ontologies pertain to educational activities, while the rest focus on research and publishing activities. Furthermore, not all these educational ontologies include information in English. For instance, the *Education Ontology* (https://schema.edu.ee) may not have comprehensive English-language support, and some ontologies reflect education systems specific to certain countries, such as the *EduProgression Ontology* (http://ns.inria.fr/semed/eduprogression/).

VIVO ontology (http://vivoweb.org/ontology/core) focuses on academic research, publication activities and relations between researchers. While it partially covers the structure of universities and study modules, it does not link these modules to curricula or hierarchically connect academic units to each other [20]. Thus, VIVO is not fully suited to describe educational structures in a way that supports detailed course content or materials.

AIISO ontology (https://vocab.org/aiiso/schema-20080925.html) characterizes the internal organizational structure of academic institutions. It provides classes and properties for describing universities, departments, curricula, and courses. However, AIISO does not cover textbooks and their relationships with courses and modules, limiting its utility in contexts requiring educational content details.

Teaching Core Vocabulary *TEACH* (https://lov.linkeddata.es/dataset/lov/vocabs/teach) is a lightweight ontology that enables educators to link various elements within their courses. This ontology facilitates the description of courses/modules and learning materials (e.g., textbooks, books), but it does not address the relations between educational institutions. TEACH is based on practice requirements established by providing workshop and course descriptions as linked data (therefore users can see all relations between main terms – Fig.2). This ontology allows the description of courses/modules and LOs (e.g. textbooks, books), but it does not extend to university-wide structures.

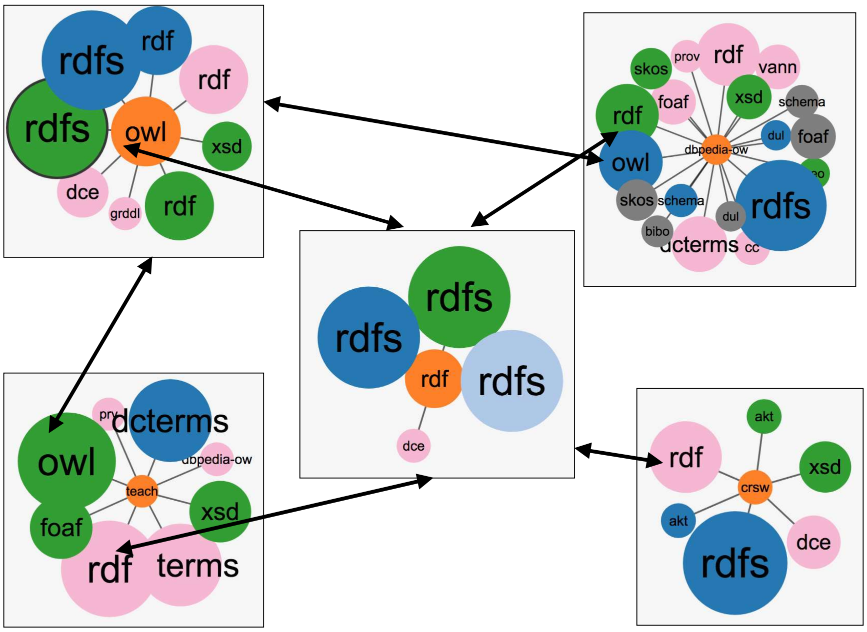

Figure 2: Links between ontology elements represented by Linked data

Some ontologies specialize in modeling of learning infrastructure. For instance, the *ReSIST* ontology (https://lov.linkeddata.es/dataset/lov/vocabs/crsw) represents educational courses and learning resources with focus on the internal structure of learning modules and the software that supports their use. The *Bologna Educational Ontology* (https://gist.github.com/lsarni) models the academic environment proposed by the Bologna reform principles [21]. It characterizes administrative procedures in European universities and concepts for describing learning programs. Such ontologies can be used as a source of information about the structure and properties of learning programs and specialties.

It is advisable to use ontologies that formalize learning outcomes for describing specialties and related educational modules. ESCO ontology (European Skills, Competences, Qualifications and Occupations, https://ec.europa.eu/esco/portal/home) 【22】, which classifies professions, skills, and qualifications related to the European labor and education market.

It is also advisable to use ontologies that formalize learning outcomes for description of specialties and related to them learning modules. An example is the *ESCO ontology* (European Skills, Competences, Qualifications and Occupations) (https:/ /ec.europa.eu/esco/portal/home) [22] that describes, defines and classifies profession, skills and qualifications related to the European Union (EU) labor and education market. ESCO can be used as a dictionary of terms applied for describing and matching of competencies, skills qualifications and vacancies. Currently, ESCO contains about 3000 descriptions of professions and about 15 000 descriptions of skills related to these professions. This information is translated in 27 languages (all official languages of EU and some additional ones):

each concept is associated with at least one term in every language but some different several terms can be used for the same concept. Within the ESCO data model, terms are linked with other ones by pertinent relations. ESCO uses Linked Open Data formats (SKOS-RDF, CSV) that developers of various applications can use to provide services such as job search, career guidance and self-assessment. Users can integrate the ESCO classifier into their applications and services. In addition, support of local API and APIs for web services provides applications information from ESCO classifier in real time.

These examples highlight that it is not enough to simply locate ontologies related to the subject area of the designed IAS (in this case, education or scientific research). It is essential to manually analyze the content of each ontology to identify the classes that can serve as a basis for the CIO structure and their relationships with other CIOs in the system.

In addition to specialized academic and educational ontologies, information about the structure and relations between the IOs of the system at the top level can be acquired from general-purpose ontologies such as Schema.org, DBpedia, and Wikidata.

Schema.org Dictionary (https://schema.org/docs/schemas.html) supports schemas for structured data. It currently contains more over 2000 types and properties that cover concepts, actions and relations between them and, and this set of structural elements can be expanded according to task needs. Many sites and applications (such as Google and Microsoft) use this dictionary to mark up web pages.

DBpedia ontology (http://mappings.dbpedia.org/server/ontology/ classes/) is a cross-domain ontology manually constructed from the most frequently used information blocks of Wikipedia. It is currently one of the largest general-purpose ontologies: it contains hierarchically organized set classes described by almost 3 thousand properties. DBpedia is the central hub of a cloud of Linked open data, and it can be used as a source of entity relations if information about them is not found in more specific ontologies.

Wikidata [23] is an open knowledge base used as a central data management platform for Wikipedia and other Wikimedia projects. The Wikidata repository consists mostly of elements and assertions about those elements. Elements that have unique identifiers, labels and descriptions are used to represent a variety of entities, including concepts and objects. Instructions used to write data about elements consist of at least one “property-value” pair to connect elements to each other’s and result in a Linked data structure. Elements are divided into classes and class instances that are connected by a relation of belonging. Classes can be hierarchically related to each other by the taxonomic relation "subclass".

WikiProject ontology (https://www.wikidata.org/wiki/ Wikidata:WikiProject_ Ontology) consists of several top-level classes and properties that aim to support broad semantics of interaction between other well-known ontologies such as DOLCE, BFO, SUMO, Lemon, RDA, etc., and to integrate the main branches of the wiki data core concept tree. Such ontology can be used as a knowledge source to reconcile different term systems of related domains for search.

10.Implementation of CIOs for ActiveBook

The development of the ActiveBook system, considered as an example of an IAS requiring an external CIO repository, is currently in the prototype phase. The following steps have been completed:

- the main subjects and objects of the system are defined and classified, their structure is formalized and model of interaction between them is built;
- the authorities of subject groups (users) regarding the ability to edit and view information are defined;
- the initial set of basic properties for CIOs is determined, and ontologies that allow for their clarification and improvement are selected;

- the software environment based on MediaWiki and Semantic MediaWiki is analyzed for implementation;
- examples of pages corresponded to main IOs of IAS are created, and their content is semantically marked according to selected ontology;
- examples of requests for content integration that allow automated generation of information on pages are created.

The next step involves supplementing of real-world LOs with their metadata, and their processing can caused refinement and expansion of CIO models. Information about additional properties of CIO elements can be acquired from LO descriptions, from metadata structure of external libraries and from external ontologies that describe similar IOs or LOs. Other aspect of further system development is a creation of user-friendly interface for performing individual semantic queries by parameters defines by CIO properties. From the point of view of this study, the creation of ActiveBook prototype demonstrates the need for CIO repository that facilitates the search for such objects in relevant ontologies, allows importing information about the structure of these objects, and supports informing about changes in external ontologies used to build the knowledge base of the system.

11. Representation and search of CIO in the Semantic MediaWiki Environment

The formalization of the CIO structure in ActiveBook IAS is based on semantic properties of the Semantic MediaWiki (SMW) framework. SMW enables the creation of templates, with parameters representing semantic properties of wiki pages where the templates are used. It names links between pages or page elements by terms that connect them with certain domain concepts by semantic properties of the page. Semantic properties and their types are the main tools for entering semantics into Wiki resources. Such semantic marking of the text provides a much more expressive representation of information in comparison to the traditional categories used by Wikipedia. Information becomes available not only for reading, but also for automated machine processing.

SMW allows the automatic integration of information from multiple pages, the generation of complex semantic queries, and the visualization of results. This technology supports the construction of ontological knowledge bases that formally represent the semantics of CIOs and perform logical inference. Therefore, the development of analytical portal based on this technological platform meets the requirements for creating an open directory of learning resources.

SMW plug-in supports the SMW-QL query language that provides capabilities for semantic search into content of wiki resources. This query language allows filtering pages by specified criteria and displaying only the selected parts of information instead of the entire content of a wiki page. This possibility can be used to integrate dynamically relevant information and to represent it in user-friendly and more understandable forms: as diagrams, geographical maps, tables, etc. If the pages with information that satisfy query requirements are changed then the query results are updated automatically, ensuring data consistency.

Semantic queries can be embedded into the text of Wiki templates represented by Wiki pages with unique names situated in the special namespace. Templates can use parser functions, special names and a simple scripting language. Templates provide representation of information blocks replicated on many Wiki pages, often with customizable elements. Parameters of Wiki templates can be incorporated dynamically into the page.

SMW can use in templates the values of the Wiki page semantic properties as parameters: if some template used by Wiki page receives some values of parameters then these values can be retrieved as values of the page semantic properties. If Wiki pages represent some IO (or CIO) it is ad visible to develop the template that formalize structure of these objects, unified it and help in its dissemination. Fig. 3 shows a fragment of the "e-Textbook" prototype knowledge base, demonstrating the representation of basic CIOs.

The "Textbook" template (it can also describe some other types of LOs) defines the connections of class individuals with other IOs defined by semantic relations with use of properties “Author”, “Publisher”, etc. Information about other CIO instances is entered directly into the system using this template, unlike other CIOs, where some data may be generated automatically via semantic queries.

Although this CIO structure may seem redundant because some parameters duplicate each other in different forms, it enhances automated information integration from various sources. For instance, some sources have information about the code of the specialty for which the textbook is used, and other ones – about the name of this specialty, and the transformation of one information into another is quite simple, but in practice it is more appropriate to enter all available information automatically, and to perform further transformations under the control of a human expert.

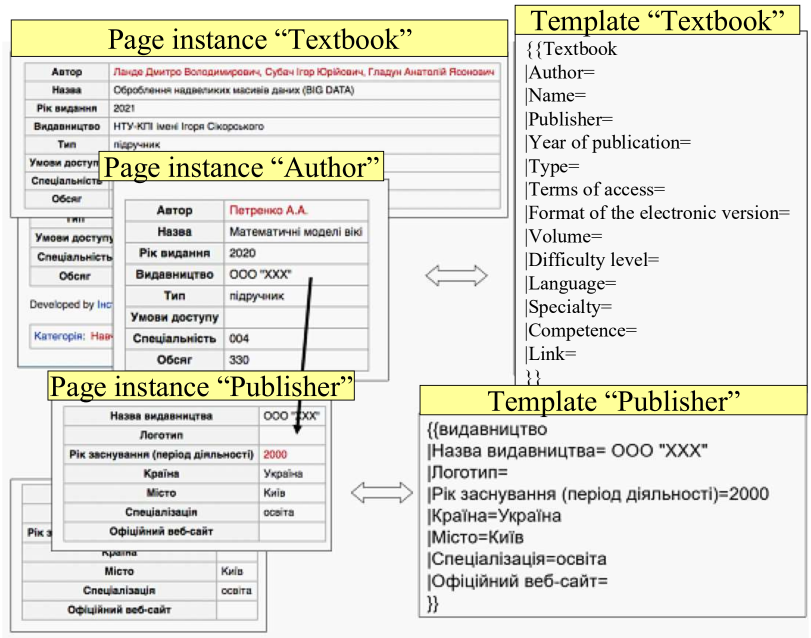

Figure 3: The structure and relations of CIOs in the ActiveBook (fragment)

Some parameters in this CIO template can be considered as object properties from an ontological perspective (such as "Author", "Publisher", "Specialty"), which reference other IOs, and templates of these IOs make it possible to define their structure. In the terms of wiki technology, such properties are of the "Reference" type, and markup tags (or appropriate template parameters) define the semantics of these links. Other template parameters from the point of view of ontological analysis can be considered as data properties (such as "Year of publication" of “Integer” type and "Language" of “Text” type) (Fig.4).

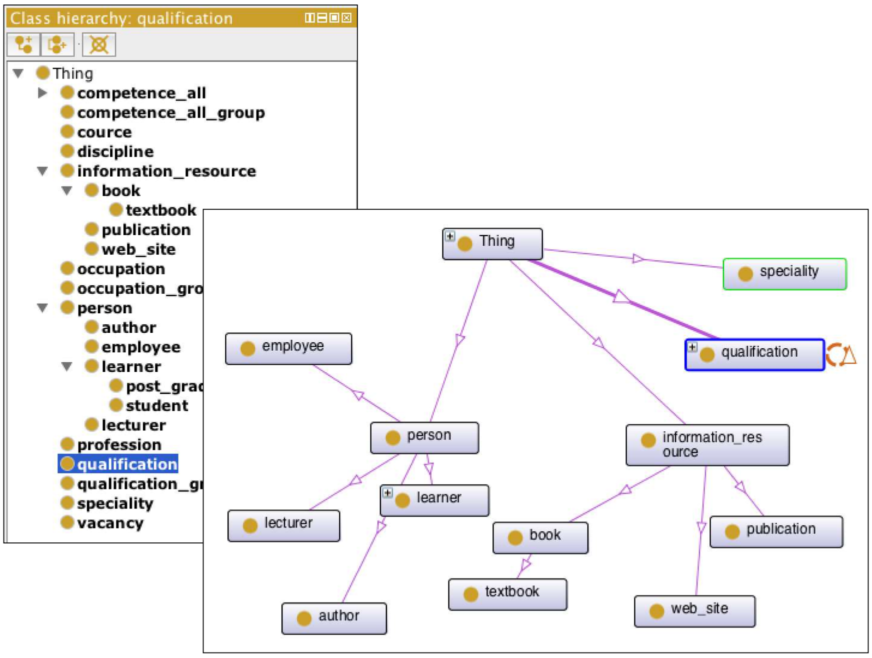

Figure 4: Wiki ontology of ActiveBook (fragment)

Search tools in semantic wiki resources can take into account the application specifics and use CIO structure and characteristics. These search tools can be categorized into the following groups:

- *Search by Keywords:* This type of search focuses on specific terms found in the titles of the wiki pages that correspond to CIO instances such as "Textbook", "Author", "Specialty", "Competence". For example, a user might search for textbooks by the initial letters of the title or the author's name;
- *Search by Domain Topics and IO Type:* In this search, users can filter by categories and subcategories of CIOs (e.g., "Learning Courses", "Monographs"). This type of search (Fig.5) simplifies navigation within the resource by grouping similar CIOs under specific domains and topics, making it easier to explore related educational materials;

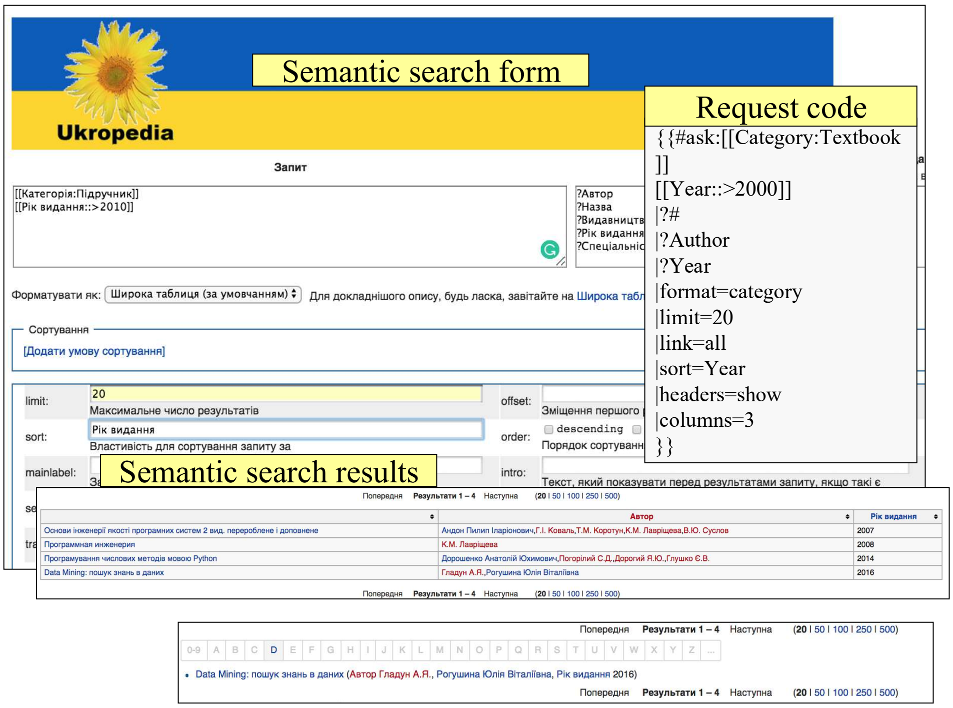

Figure 5: Semantic search in ActiveBook

- *Search According to the IO Semantics:* This search method combines the requirements for category and semantic property values (such as the range or value of certain properties). For example, users can search for textbooks that are categorized under a specific domain and also match certain property values, such as particular language or year of publishing.

In the ActiveBook system, semantic search goes beyond simple keyword searches. One advanced search method involves retrieving semantically similar objects, where the matching is done based on a set of properties, taking into account domain knowledge about the proximity of these properties' values. For instance, a search could identify educational materials for a particular competency, even if that competency is not explicitly specified in the materials. This could be possible because the competency is a subclass or superclass of the given one, and can be identified using taxonomy of competencies.

However, SMW does not inherently support this kind of advanced semantic search. This type of functionality could be integrated into the system through the development of separate API modules and specialized services. These services would enable more sophisticated queries that are capable of understanding and matching semantically similar concepts within the system. But practice shows a lot of various relevance problems of domain and top-level ontologies [26] ontological models: these models do not reflect automatically all recent changes in the structure of IAS knowledge base.

As a result, it would be more effective to provide users with links to the relevant elements of the CIO repository that accurately correspond to the current state of the IAS knowledge base. This approach ensures that the search results are consistently aligned with the most up-to-date data and knowledge.

12.Conclusions

The development process of the ActiveBook IAS highlights the need for a repository of complex information objects that facilitates the search for instances of various ontological classes linked by specified types of semantic relations. An analysis of existing ontology repositories reveals that this functionality is lacking, and this fact significantly complicates the development of intelligent applications that rely on external knowledge sources and require the ability to track changes in those sources.

This paper outlines the essential requirements for such repository of complex information objects, examines the technologies that can be used to populate it, and provides examples of its practical applications. The proposed method and services can be utilized in artificial intelligence applications for various domains.

References

[1] J. Rogushina, A. Gladun, Use of Ontological Knowledge for Multi-Criteria Comparison of Complex Information Objects, UkrProg-2022: Proc. of the 13th International Conference of Programming, CEUR Vol-3501, (2022): 222-231. URL: https://ceur-ws.org/Vol-3501/s21.pdf.

[2] N. Guarino, Formal Ontology and Information Systems. Formal Ontology in Information Systems, in: Proc. of FOIS'98, by N. Guarino (ed.). Trento. Italy, Amsterdam, IOS-Press. 1998. pp. 3–15. URL: https://www.researchgate.net/publication/272169039_Formal_Ontologies_and_Information_Systems

[3] J. Rogushina, I. Grishanova, Semantic Information Resources with a Complex Structure: Knowledge Representation, Scaling and Search Problems, UkrProg-2022: Proc. of the 13th International Conference of Programming, CEUR Vol-3501, (2022): 158-171. URL: ceur-ws.org/Vol-3501/s15.pdf

[4] J. Rogushina, S. Priyma, Use of competence ontological model for matching of qualifications, Chemistry: Bulgarian Journal of Science Education, Vol 26, Number 2 (2017): 216–228. URL: http://elar.tsatu.edu.ua/bitstream/123456789/3181/1/2.pdf

[5] J. Rogushina, A. Gladun, R. Valencia-Garcia, Reuse of Ontological Knowledge in Open Science: Models, Sources, Repositories, Communications, Communications in Computer and Information Science (CCIS), V.1873, (2023): 157–172. Cham: Springer Nature Switzerland.
[6] FAIR data. URL: https://en.wikipedia.org/wiki/FAIR_data.
[7] J. Rogushina, Ontological Approach in the Smart Data Paradigm as a Basis for Open Data Semantic Markup, COLINS, CEUR Vol-3403, (2023): 12-27. URL: ceur-ws.org/Vol-3403/paper2.pdf.
[8] N. Bassiliades, EvdoGraph: A Knowledge Graph for the EVDOXUS Textbook Management Service for Greek Universities, in: Proc. Of 15th International Conference on Knowledge Engineering and Ontology Development KEOD, 2023, Rome, Italy. URL: https://intelligence.csd.auth.gr/wp-content/uploads/2023/08/EvdoGraph-CR.pdf
[9] C. Bizer, T. Heath, T. Berners-Lee, Linked data: The story so far, in: Semantic services, interoperability and web applications: emerging concepts, IGI global, (2011): 205-227.
[10] M. Wylot, M. Hauswirth, P. Cudré-Mauroux, S. Sakr, RDF data storage and query processing schemes: A survey. ACM Computing Surveys (CSUR), 2018, 51(4): 1–36.
[11] G. Antoniou, Web ontology language: Owl / G. Antoniou, Van Harmelen F. In: Handbook on ontologies. Springer Berlin Heidelberg, 2004, pp. 67–92.
[12] A. Hogan, E. Blomqvist, M. Cochez, C. D'amato, G. Melo, C. Gutierrez, A. Zimmermann, Knowledge graphs. ACM Computing Surveys, 2022, 54(4): 1–37. doi:10.1145/3447772.
[13] L. Yu, L. Yu, Linked open data. A Developer's Guide to the Semantic Web, 2011, pp. 409–466.
[14] C. Binding, D. Tudhope, D. Improving interoperability using vocabulary linked data. In: International Journal on Digital Libraries, 17, 2016, pp.5-21.
[15] K. Stancin, P. Poscic, D. Jaksic, Ontologies in education - state of the art. Education and Information Technologies, 2020, 25(6): 5301–5320. doi:10.1007/s10639-020-10226-z.
[16] M. Färber, The Microsoft Academic Knowledge Graph: A Linked Data Source with 8 Billion Triples of Scholarly Data. ISWC 2019, LNCS. 11779, 2019, pp. 113–129. Springer. doi:10.1007/978-3-030-30796-7_8.
[17] M. Y. Jaradeh, A. Oelen, K. E. Farfar, M. Prinz, J. D'Souza, G. Kismihók, S. Auer, Open research knowledge graph: Next generation infrastructure for semantic scholarly knowledge. KCAP 2019, ACM, (2019): 243–246. doi:10.1145/3360901.3364435.
[18] B. Abu-Salih, Domain-specific knowledge graphs: A survey. Journal of Network and Computer Applications, 185, 2021. doi:10.1016/j.jnca.2021.103076.
[19] P.-Y. Vandenbussche, G. A. Atemezing, M. Poveda-Villalón, B. Vatant, Linked Open Vocabularies (LOV): A gateway to reusable semantic vocabularies on the Web. Semantic Web, 8(3), (2017): 437–452. doi:10.3233/SW-160213.
[20] J. Corson-Rikert, S. Mitchell, B. Lowe, N. Rejack, Y. Ding, C.Guo The VIVO Ontology. In: VIVO, Synthesis Lectures on Data, Semantics, and Knowledge, 2012, pp. 15–33. doi:10.1007/978-3-031- 79435-3_2.
[21] G. Demartini, I. Enchev, J. Gapany, P. Cudré-Mauroux, The Bowlogna ontology: Fostering open curricula and agile knowledge bases for Europe's higher education landscape. Semantic Web, 4(1), (2013): 53–63. doi:10.3233/SW-2012-0064.
[22] ESCO (the European Multilingual Classifier of Skills, Competences, Qualifications and Occupations. URL: https://ec.europa.eu/esco/portal/home.
[23] D. Vrandečić, K. Krötzsch, Wikidata: a free collaborative knowledgebase. Communications ACM, 10, (2014) 78–85. doi:10.1145/2629489.
[24] J. H. Gennari, M. A. Musen, R. W. Fergerson, W. E. Grosso, M. Crubézy, H. Eriksson, S. W. Tu, The evolution of Protégé: an environment for knowledge-based systems development. International Journal of Human-computer studies, 58(1), (2003): 89-123.
[25] J. Rogushina, Semantic Wiki resources and their use for the construction of personalized ontologies. UkrProg-2016: Proc. of the 10th International Conference of Programming, CEUR Vol-1631, (2016): pp.188-195.
[26] C. Partridge, A. Mitchell, A. Cook, J. Sullivan, M. West, A Survey of Top-Level Ontologies-to inform the ontological choices for a Foundation Data Model. 2020. URL: https://api.repository.cam.ac.uk/server/api/core/bitstreams/829fe47b-d4fa-4430-a639-3f0904e1684b/content.