ROH: Towards a highly usable and flexible knowledge model for the academic and research domains

Mikel Emaldi [a,*], Maite Puerta [a], David Buján [a], Diego López-de-Ipiña [a], Emilio Rubiera Azcona [b], José Emilio Labra Gayo [b], Esteban Sota [c] and Ricardo Alonso Maturana [c]

[a] *DeustoTech Institute of Technology, University of Deusto, Av. Universidades 24, 48007 Bilbao, Spain*
E-mails: m.emaldi@deusto.es, mpuerta004@deusto.es, david.bujan@deusto.es, dipina@deusto.es
[b] *Dept. of Computer Science, University of Oviedo, C/Calvo Sotelo S/N, 33007 Oviedo, Spain*
E-mails: UO38239@uniovi.es, labra@uniovi.es
[c] *gnoss.com, Piqueras 31, 26006 Logroño, Spain*
E-mails: estebansota@gnoss.com, riam@gnoss.com

Abstract. This paper presents the work developed by the Hercules-ASIO project, putting special emphasis on the design and development of the ROH network of ontologies. ROH (*Red de Ontologías Hércules*, by its Spanish naming) aims to model thoroughly the main entities and relationships of the academic and research domain, e.g., projects, researchers, academic articles, universities, courses, organizations or research results. In this paper, the methodology followed for the development of ROH is detailed, paying special attention to the implementation and validation phases. Consequently, the most relevant entities are described, as well as their relationships, followed by a wide range of methods applied to continuously evaluate and enhance the ontology's correctness and exhaustiveness.

Keywords: Semantic Web, Ontology, Academic domain, Research domain, CRIS

1. Introduction

This work presents the Hercules Network of Ontologies (ROH, *Red de Ontologías Hércules* by its Spanish naming), a set of ontologies that models the research and academic domain. Specifically, ROH models the research performed in research institutions, administratively and financially, and the academic activities performed by researchers. ROH is able to represent the scientific results, such as, academic articles, journals and their impact; research projects and their funding; events; and research work that has been and is being conducted in different areas of knowledge.

ROH is the result of Hercules project [1], which aims to create a new information management system for Spanish universities, under the supervision of the CRUE (*Conferencia de Rectores de las Universidades Españolas*, Commission of Rectors of Spanish Universities), based on the technologies of Semantic Web and Knowledge Graphs. For this purpose, this project has been divided into several subprojects:

[*] Corresponding author. E-mail: m.emaldi@deusto.es.

1570-0844/$35.00 © 0 – IOS Press and the authors. All rights reserved

- **SGI**: *Sistema de Gestión de la investigación* (Research Management System, RMS). This project aims to create a Current Research Information System (CRIS), i.e. a database or information system to store, manage and exchange contextual metadata for the research activity funded by a research funder or conducted at a research-performing organization. CRIS systems are also known as Research Information Management or RIM Systems (RIMS). The Hercules CRIS should offer a global vision of the research data of the Spanish University System in order to improve the management, analysis and possible synergies between universities and the general public through the development and incorporation of solutions that go beyond those currently available on the market.
- **ASIO**: *Arquitectura Semántica de Datos e Infraestructura Ontológica* (Semantic Data Architecture and Ontological Infrastructure). ASIO is a pre-commercial public procurement project, whose objective is to develop an innovative solution grounded on a Semantic Architecture and Ontological Infrastructure, to be used in the future on a regular basis by both the University of Murcia and the rest of Spanish Universities that belong to the CRUE with similar needs and responsibilities. The core innovative features of the solution are 1) *Semantic Data Architecture*, which consists of developing an efficient platform for storing, managing and publishing research data from the Spanish University System, based on ontologies, with the ability to synchronise and aggregate data coming from different Universities, and 2) *Ontological Infrastructure*, which consists on developing an ontology infrastructure, known as the "Hercules Network of Ontologies" (ROH) modelling with fidelity and fine granularity the research domain.
- **EDMA**: *Enriquecimiento de Datos y Métodos de Análisis* (Data Enrichment and Analysis Methods). It aims to curate, enrich and exploit the data produced by ASIO. ED is the part of this sub-project intended to facilitate data enrichment through the use of information sources available on the Internet and commonly used by the research community. MA (*Métodos de Análisis*) part of this sub-project aims to enable the exploitation and analysis of data for the purpose of inclusion and participation of different levels of stakeholders with distinct data interpretation capabilities.

In this paper we will focus on the description of the contributions achieved in the development of ASIO, specifically on the development of the ROH network of ontologies.

The rest of the paper is structured as follows. Section 2 introduces different ontologies related to the academic and research domain. Section 3 displays the notation used to formally describe the classes and relationships developed at ROH. In Section 4, the methodology applied for the development of ROH is presented. Section 5 includes the specification of the most relevant entities of ROH. In Section 6, the work carried out in order to develop a suitable evaluation of ROH is illustrated. Lastly, Section 7 presents the conclusions and further work plans of this work.

2. Related work

Since the rise of the Semantic Web, many ontologies for describing different aspects of the academic and research domain have been developed. Although at [2], a wide survey about those works is presented, in this Section we introduce those which have been the most relevant in the specification and development of ROH in conjunction with those works which are the most relevant for the Semantic Web community.

Developed within the VIVO project[1], the VIVO ontology [3] aims to represent the academic domain. Specifically, it represents the relationships of people to different academic artifacts such as research projects, publications, degrees, and so on. It allows modelling the resources used by academics, the institutions they work for, their expertise and knowledge, and so on. VIVO allows creating academic web portals aligned with the Semantic Web standards. Because of its completeness, it is the base ontology on top of which ROH has been developed.

The Bibliographic Ontology [4] (BIBO) aims to describe citations and bibliographic references. BIBO is widely used by other ontologies from the academic domain. For example, the mentioned VIVO ontology leverages on a set of terms from BIBO ontology for describing the different types of documents found at the academic domain, among others.

[1] https://duraspace.org/vivo/

The Semantic Web for Research Communities (SWRC) ontology [5] models research communities and related concepts such as projects, organizations, events and publications, among others. Nowadays, this ontology is not available on the web, so it has not been considered to be used at ROH. On the other hand, the SWRC Funding Extension ontology (SWRC-FE) [6] adds capabilities for describing funding sources to SWRC ontology, which have been reintroduced in ROH by extending VIVO.

The Common European Research Information Format (CERIF) ontology [7] was developed within the CRIS (Current Research Information Systems) community [8]. It provides basic concepts and properties for describing research information as semantic data, such as equipment, facilities, curriculum vitae or metrics. CERIF classifies its described entities as base entities, infrastructure entities, second-level entities and result entities, depending on their role within the CRIS data model. ROH's design has taken into account CERIF data model for CRIS to guarantee that all entities and relationships conventionally modeled in a CRIS are included.

The SPAR ontologies[2] [9] are a family of vocabularies undoubtedly related to our own endeavor, although in a more fragmented ("orthogonal and complementary") fashion and specifically addressing the whole aspects of semantic publishing and referencing. Two of its vocabularies relevant for ROH ontology are FRAPO[3] (Funding, Research Administration and Projects Ontology), tackling administrative information of research projects (grant applications, funding bodies, project partners, etc.) and also FaBiO[4] (FRBR-aligned Bibliographic Ontology), an ontology for recording and publishing bibliographic records, based on the Functional Requirements for Bibliographic Records (FRBR) model [10]. On the other hand, CiTO (Citation Typing Ontology) [11] is an ontology for the characterization of bibliographic citations, both factually and rhetorically.

Based particularly on some of those ontologies (e.g., VIVO and BIBO), ROH has developed a wide model which allows to represent the academic domain. In Section 4.2 more details about the usage of state-of-art ontologies is provided.

3. First-order logic notation

The language selected to develop ROH is OWL DL [12]. First-order logic allows to describe the OWL axioms and relationships from an ontology. We have considered both logical and non-logical symbols. The logical symbols are:

- The quantifier symbols: universal $\forall$ and existential $\exists$.
- The connectives symbols: $\wedge$ for "and", $\vee$ for "or", $\rightarrow$ for "implies" and $\leftrightarrow$ for biconditional statements.
- Variables: $x, y, \ldots$, ranging over particulars (individual objects).

As non-logical symbols, we consider unary and binary predicates:

- Unary predicates define the class that a variable has. They are denominated with the name of the OWL class, without the prefix of the ontology in order to simplify the notation. E.g., the unary predicate Document(x) means that the variable x is an instance of `bibo:Document` class.
- Binary predicates define the relationship between two variables. They are denominated with the property name without the prefix. E.g., `documentStatus(x, y)` means that the variable y is the status of the document x.

These two types of symbols make it possible to define the two main OWL restrictions that we used in this ontology: `owl:someValuesFrom`, namely `some` restrictions, and `owl:allValuesFrom`, namely `any` restrictions.

With the `some` restriction, we can ensure that if there exists an instance of the entity that has this restriction, then there exists at least one instance that is related to the first one through the object property that has this restriction. For example, the sentence: "All organizations have to have an name or title" is a `some` restriction that means that

[2] http://www.sparontologies.net/ontologies
[3] http://www.sparontologies.net/ontologies/frapo
[4] http://www.sparontologies.net/ontologies/fabio

for all organization instances, there must be at least one entity, a literal in ROH, that identifies this organization. The mathematical expression of this sentence is described at Eq. (1) and an example of this restriction in an OWL code is described at Listing 1 in lines 12-14.

With the `any` restriction, if there exists an instance of the entity that has this restriction and it is related to another instance through the object property that has the restriction, then it allows us to define the class of this second instance. For example, "if there's an organization that's participated in something, this has to be an activity". So this restriction allows us to define the domain of the instance to which the first one is related. The mathematical expression of this sentence is described at Eq. (2) and an example of this restriction in an OWL code is described at Listing 1 in lines 9-11.

$$\forall x(Organization(x) \rightarrow \exists\, y\ (Literal(y) \wedge title(x,y))). \tag{1}$$

$$\forall x \forall y\ (Organization(x) \wedge participates(x,y)\ \rightarrow Activity(y)). \tag{2}$$

Listing 1: An example of the restrictions some and any in the organization instance.

```
1
2  @prefix : <http://w3id.org/roh#> .
3  @prefix foaf: <http://w3id.org/roh/mirror/foaf#> .
4  @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
5  @prefix vivo: <http://w3id.org/roh/mirror/vivo#> .
6
7  foaf:Organization rdf:type owl:Class ;
8                    rdfs:subClassOf foaf:Agent ,
9                            [ rdf:type owl:Restriction ;
10                                 owl:onProperty vivo:identifier ;
11                                 owl:someValuesFrom rdfs:Literal ],
12                           [ rdf:type owl:Restriction ;
13                                     owl:onProperty :participates ;
14                                     owl:allValuesFrom :Activity ].
```

4. Methodology

Different methodologies for developing ontologies can be found in the literature. At [13], its author reviews different methodologies which can be used for ontology development, such as [14], [15], [16], [17] and [18], as well as a list of nine criteria for analysing each methodology. As evolution of those methodologies, the NeOn Methodology for Ontology Engineering [19] was developed. NeOn identifies the most common nine flexible scenarios for building ontology networks that may unfold during the ontology network development. Most recent ontology development methodologies, based on modern software development methodologies such as Test Driven Development (TDD) [20] or eXtreme Programming (XP) [21] have been developed by [22], [23] and [24]. As stated by [22], *"such kind of methodologies would be preferred when the ontology to develop should be compose by a limited amount of ontological entities – while the use of highly-structure ad strongly-founded methodologies remain valid and, maybe, mandatory to solve and model incredible complex enterprise projects"*. Considering the complexity of the Hercules project the use of an agile methodology have been discarded.

Although it has been developed to fit the ASIO-Hercules project particularities, the methodology carried out when developing ROH fits with four scenarios considered by NeOn, i.e., *Scenario 1: From specification to implementation*, *Scenario 3: Reusing ontological resources*, *Scenario 8: Restructuring ontological resources* and *Scenario 9: Localizing ontological resources*.

For example, in the first scenario, the *ontology requirements specification activity* is performed, in which the ontology requirements specification document (ORSD) is produced. This activity has been performed in ROH as it is described in 4.1, producing the deliverable “EF2.1-1. Hercules Ontology Specification”[5]. At this document, a set of competency questions (CQs, section 6.1) have been proposed, as NeOn suggests.

The third scenario, *Reusing Ontological Resources*, defines the *ontological resource reuse process* which is composed of four activities, i.e.: 1. ontology search (to search for candidate ontological resources that satisfy the requirements), 2. ontology assessment (to inspect the content and granularity of the ontological resources obtained in Activity 1), 3. ontology comparison (to compare the ontological resources assessed in Activity 2) and 4. ontology selection (to select the set of ontological resources that are the most appropriate for their ontology network requirements). In the development of ROH, those activities has been tackled by the *Selection of Ontologies* phase, described at 4.2.

However, the main scenario defined by Neon which has been tackled at the development of ROH is the *Scenario 8: Restructuring Ontological Resources*. This scenario refers to *“those cases where the knowledge contained in the conceptual model of the ontology network should be corrected and reorganized to obtain the network that covers the ontology requirements”*. At this scenario the *ontology restructuring activity* is carried out, which is divided in different sub-activities. Among those sub-activities, the *ontology pruning activity* and the *ontology enrichment activity* are carried out. The first one, refers to *“prune those branches of the taxonomies included in the ontology network that are considered not necessary to cover the ontology requirements”*. As described in sections 4.2 and 4.3, during the development of ROH different ontologies have been reused, but including only the branches significant for the project instead including the entire ontology. Regarding to the second sub-activity, it is composed by two sub-activities, i.e., *ontology extension activity* and *ontology specialization activity*. In this sense, ROH has specialized different branches of reused ontologies, e.g., `vivo:AcademicDegree` has been specialized through the creation of the `roh:BachelorsDegree`, `roh:DoctoralDegree` and `roh:MastersDegree`.

A last scenario defined by NeOn has been implemented at ROH, i.e., *Scenario 9: Localizing Ontological Resources*. This scenario refers to the localization and translation of different labels of the ontology. As described in section 4.1.4, different labels from ROH concepts are translated into English and all the official languages from Spain.

The implementation of the ROH network of ontologies was carried out following an iterative and incremental methodology divided in 5 different phases, see Figure 1, namely, requirements analysis, selection of ontologies, implementation and evaluation. All of these phases have been defined according to the following design principles:

- **Reusability**: re-modelling any concept that could be represented by any other ontology has been avoided. For example, for modelling the concept of the position a person occupies in an academic organization, ROH leverages the VIVO Ontology for Research Discovery [3], in which this concept is widely documented.
- **Extensibility**: Although academic information modelling shares many aspects universally, there are aspects that are country-specific, e.g. the *“sexenios”* (six-year periods) or diverse positions that exist only at the Spanish university system. This has led to the development of a *core* ontology which can be extended by country-specific sub-modules.
- **Maintainability**: the modular design applied to ROH seeks an easier maintainability of the ontology.
- **Integrity**: restrictions and validation scripts in languages like SHACL [25] have been applied, to preserve the integrity of the ontology.
- **Usability**: ROH has been designed with the aim of being comprehensive and exhaustive, i.e. covering the maximum number of academic world concepts and their properties, but also, and very importantly, to make it easily usable. In ontological design, often entities and properties are very superficially described, following the open world principle. However, ROH has been developed to be usable by those that need to instantiate it, independently on whether they are ontology engineers or just developers. Developers working in a CRIS need to understand which properties are compulsory, which are optional, and what data types they need to use to generate semantic data through ROH. This explains why in ROH a big effort has been paid to generate a proper documentation and to introduce ontological restrictions which validate the correct instantiation of classes and properties of the ontology.

[5] https://github.com/HerculesCRUE/ROH/blob/main/docs/6%20-%20OntologySpecification.pdf

The design of the ontology has followed a five-step process:

1. **Requirement analysis**: during the first stage of the development, an analysis of the requirements for modelling academic information was delivered, describing all the concepts to be modelled within ROH. This process was validated by the University of Murcia.
2. **Selection and analysis of ontologies describing the academic domain**: at this phase, taking the state of the art on academic domain ontologies delivered during the previous phase as starting point, the set of ontologies to be reused during the development of ROH were selected.
3. **Implementation of the main concepts and relationships related to the modelling of the academic domain**: from the requirements detected at the first step, and the ontologies selected at the second step, the main concepts required for representing the academic domain were implemented, as well as the relationships among them. At this step, a widely used ontology modelling tool, i.e. Protégé[6] [26], which uses the OWL language for the ontology modelling task, was used.
4. **Evaluation of the flexibility, completeness and integrity of ROH**: for that, three different evaluation processes were carried out:
   - **Competency Questions** set up by University of Murcia after a thorough survey issued to domain experts in order to check if the developed network of ontologies fits to the requirements identified during the first phase. Those competency Questions were translated into SPARQL [27] queries and executed against synthetic data modelled using ROH. In addition, a dataset based on real data has been produced.
   - **Use of SHACL** (Shapes Constraint Language) [25] for validating the data modelled according to ROH, particularly during instantiation, against a set of conditions, creating a set of SHACL shapes derived from the restrictions defined by the ontology.
   - **Mapping of FECYT's CVN to ROH**. FECYT (*Fundación Española para la Ciencia y la Tecnología*, Spanish Foundation for Science and Technology), provides the CVN (*Currículum Vítae Normalizado*, Standardised Curriculum Vitae) model which is required for applying to different research funding grants. Within the ASIO project, an API for translating a CVN into a RDF dataset modelled according to ROH was developed, which allowed us to validate if all the required concepts and relationships defined in CVN were modelled.
5. **Continuous refinement validated by automated regression tests**: a test suite based on SPARQL competency Questions created in the previous phase was integrated in a CI/CD (Continuous Integration and Continuous Delivery) workflow, in order to check that every modification applied to the ontology could be integrated properly into the existing work. As a matter of fact, every change committed to the ontology is automatically validated, through this automatic process, before such changes are integrated in a new ontology release.

Figure 1 summarizes the methodology applied by GNOSS-Deusto, the temporal organization that was created to define ROH. In this work, we cover all these steps. The first step is described in Section 4.1, while the second step is described in Section 4.2. The implementation details of the main concepts, i.e., the third step, is described in Section 4.3. Lastly, Section 6 describes the evaluation and the continuous refinement of the ontology.

4.1. Requirement analysis

The requirement analysis was split in four different steps: 1) analysis of use cases in research management; 2) analysis of the main functionalities of a CRIS; 3) identification of entities and relationships; and 4) analysis of non-functional requirements of a CRIS.

4.1.1. Analysis of use cases in research management.

Table 1 shows the set of usage scenarios identified within the Hercules project for advanced exploitation of data related to research management. Each scenario is accompanied by a description and a preliminary identification of entities and possible queries that could be made. The analysis performed at this step allowed us to identify a preliminary set of entities and relationships to be modelled at the ontology.

[6] https://protege.stanford.edu/

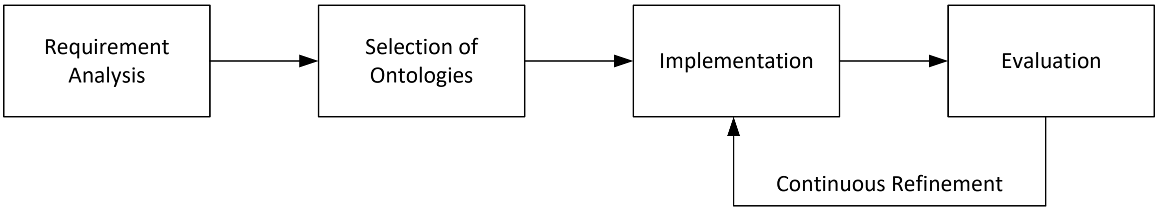

Fig. 1. Methodology followed during the development of ROH.

Table 1

Scenarios identified within the Hercules project.

#	Description	Involved entities	Example queries
1	Analysis, at different levels, of the different types of expenditure made by national groups on funded research projects.	Project, Funding Organization, Expense, Expense Classification, Project Classification, Organization, Consortium, Role.	Total income by group, total income by type of funder, total income by organization, breakdown of income by project, expenditure breakdown by project.
2	Creation of a national knowledge map, enabling the objective identification of knowledge hubs in thematic areas.	Organization, Knowledge Area, Document.	Collaborations between universities, research groups and people; JCRs per year per group, per researcher, per department, per faculty and per university; projects per year by research group, department, faculty, university and, researcher.
3	Implementation of flexible dashboards, where the hierarchical structure of ontologies will allow to establish the granularity of the analysis.	Knowledge Area, Project, Organization, Researcher Role, Research Group, Metric.	Funds associated to a research topic by university, by department, by group, by researcher; publications associated with a topic by university, by department, by group, by researcher; history of income and publications, by group, by researcher; metrics about researchers, groups, departments, faculties and universities.
4	Implementation of search engines for groups with specific profiles and available for participation in specific research calls.	Knowledge Area, Project, Organization, People, Researcher Role.	List of participants for a project belonging to a specific research topic; list of authors of publications at specific topic.
5	Finding of those persons whose academic/researcher profile best match a public or private offer, and even to implement methods for automatic designing optimal consortia from existing data.	Research Group, Curriculum Vitae, Metric.	List of organizations participating in projects belonging to a specific research topic; list of organizations whose staff are authors of publications at specific topic.
6	Allowing the regional, national and International (e.g., European) project offices to have a better knowledge of the profile of a given research groups and to be able to carry out a more precise active task of recommending calls for proposals.	Funding Program, Project, Research Group, Curriculum Vitae.	List of research groups participating in projects belonging to a specific H2020 call; list of research whose members have been granted by the European Commission in the past.
7	Automatic and up-to-date generation of reports in the form of HMTL pages or PDF documents from the entities and their relationships modelled in the graph.	Curriculum Vitae, Webpage, Report.	List of publications, projects, research groups, and positions belonging to a researcher.

4.1.2. *Analysis of the main functionalities of research management systems.*

As a result of this analysis, the main concepts to be held by the CRIS/RMS were defined:

- **Projects**: management of research projects, their definition, origin, purpose, economic management, annual payments and other basic associated information.
- **Research groups**: management of the creation, maintenance, and deletion of research groups as well the additions and dropouts of members of the research groups.
- **Calls for applications and grants**: management of different calls for applications for the distribution of funds, grants and scholarships. Financial management and procedures, concession, refusal and scale or evaluation system.
- **Project personnel**: management of staff associated with projects and payments to projects' staff.
- **Scientific production**: management of the scientific production of researchers (articles, theses, conferences, various publications and so on).
- **Curriculum Vitae**: management of the curriculum vitae of researchers in CVN ("Currículum Vitae Normalizado", Standardised Curriculum Vitae) format.
- **Contracts and patents**: management of contracts and patents, and research work in which universities and private companies collaborate.
- **Research group web page manager**: management of the information associated with research groups to increase their visibility through their own web pages.
- **Research bulletin**: Management of a news system for the dissemination of relevant information for the research community.
- **Consortia and partners**: information on the consortia and a valuation by the research groups of the partners (classified by type, university, SME, company, tech center) with whom they have collaborated, allowing the university to know the most valued partners. In addition, it should allow the university to select the partners with whom to form a consortium on the basis of previous experience.

4.1.3. *Identification of entities and relationships*

Once the requirements associated with the scenarios were defined and the expected functionality of the CRIS/RMS was explored, the entities identified in these scenarios were analysed. For each entity identified, the following was specified: a) a taxonomy or hierarchy of entity classes associated to such higher order entity; b) essential attributes that such entities must have in order to satisfy the modelling and querying of the knowledge graph; and c) the fundamental relationships with other key entities. Table 2 shows the entities identified within ROH. In this table, cardinality is identified by the symbols $+$ (one or more occurrences) and $*$ (zero or more occurrences). This table shows the preliminary concepts and relationships identified within the project, and refined during the development of the ontology. Thus, the concepts shown in Table 2 could not concur directly with the ones modeled at ROH.

4.1.4. *Analysis of non-functional requirements of the RMS*

During this step, different non-functional requirements were identified. Among those requirements, the following ones are worth mentioning:

- **Follow Linked Open Data principles** [28]: 1) use URIs as names for things, 2) use HTTP URIs so that people can look up those names, 3) when someone looks up a URI, provide useful information, and 4) include links to other URIs, so that they can discover more things.
- **Follow FAIR principles** [29]: data must be *Findable* through a persistent identifier and including metadata, *Accessible* through the universal HTTP protocol, *Interoperable* using widely adopted vocabularies and *Reusable*, published using user licenses that promote reusability.
- **Use of persistent identifiers**: use of IDs that are permanently assigned to a resource even if the location of the resource changes over time, such as purl.org or w3id.org.
- **Multilingualism**: labels and descriptions of both classes and properties of the ontology should be expressed in English, Spanish and the rest of the official languages from Spain's autonomous communities, through the usage of the `rdfs:label` and `rdfs:comment` properties. In addition, several vertical modules, modelled in SKOS, also exploit the properties `skos:prefLabel` and `skos:altLabel` for multilingual purpose. On the other hand, the related notion of multi-scriptalism [30], concomitant to multilingualism, is also fully deployed in the vertical modules for those languages using different writing systems, such as Cyrillic, Greek

Table 2
Entities and relationships identified within the Hercules project.

Entity	Taxonomy (subclasses)	Main Attributes	Related entities
Researcher Role	PhD candidate after dissertation, Numerary staff, Contracted staff, Fellowships, Staff in special services, Research fellowship, Honorary collaborating professor.	ID, name, surname, contract.	Project*, ResearchResult*, Internship*, ProjectExpense*, KnowledgeArea+, CV+, ResearchMetric*
Project	Private project, Agreement, Tender, International project, State project, European project.	ID, title, description, abstract, type, duration, status, supporting documents.	Funding+, Organization+, KnowledgeArea+, FundingAmount*, ProjectExpense*, ResearcherRole+, ResearchResult*, Metric*
Funder	Private, Public.	ID, name, URL, description, address, contact email.	Organization*, FundingProgram*
Funding-Program	Grant, Loan, Subcontracting.	ID, name, URL, description.	FundingOrganization+, Project*
Funding	-	ID, name, description, resolution.	FundingProgram, Project*, Funding-Amount+.
Funding-Amount	Personnel cost, Subcontracting, Travel, Equipment, Research Infrastructure, Other goods and services.	ID, Income modality, Amount, Year.	Funding+, Project+, Organization*
Project-Expense	Personnel cost, Subcontracting, Travel, Equipment, Research Infrastructure, Other goods and services.	Expense Classification, monetary amount, date.	Project+, Researcher*, ExpenseClassification*
Research-Result	Publication, Software, Dataset, Patent, Dissemination article.	ID, Result type, Repository, Date, Keywords, License, Version.	Project+, ResearcherRole+, Funding+, Organization+
Publication	Book, Book section, Conference paper, Journal article, Magazine article.	ID, Type of publication, Publisher	ResearchResult.
Subject	Bachelor's degree Subject, Master's degree Subject, PhD Subject, Continuous training Subject.	ID, name, description, programme, student guide, contents.	Organization, Teacher Role+, AcademicDegree
Degree	Bachelor's degree, Master's degree, PhD degree, Continuous training.	ID, name, description, title.	Organization, TeacherRole+, AcademicSubject+
Academic-Activity	Lecture, PhD thesis (defence), Graduation event, Conference, Stay.	ID, title, type, description, place, period.	Organization, KnowledgeArea+, ResearcherRole+
Placement	Predoc, Postdoc, Research, Education.	ID, title, type, description, place, period.	Organization, KnowledgeArea+, ResearcherRole+, FundingProgram*
Organization	University, Faculty, Department, Undergraduate degrees, Master degrees, PhD degrees, Research groups.	ID, name, description, type, place, date of foundation.	Researcher+, KnowledgeArea+, AcademicDegree*
Infrastructure	Facility, Equipment.	ID, name, description, type, place.	Funding*, Organization
Geographical-Scope	Administrative hierarchy in line with Geonames (https://www.geonames.org/export/codes.html).	Geoname ID, longitude, latitude, feature code, name, country code.	Organization, FundingOrganization
Knowledge-Area	Popular taxonomies such as UNESCO and FECYT.	Code, name of the concept in different languages, hierarchical relationship of the concept.	ResearcherRole*, Project*, Organization*
Contract	Employment contract, Project contract.	ID, document.	ResearcherRole*, Project*, Organization+

or Arabic scripts. Finally, also *multilocalism* is consistently exploited whenever diverse locales for a given language are used.

- **Interoperability with existing ontologies**: sometimes ontologies are no longer available on the Internet, usually because of the lack of maintenance. Considering that one of the principles adopted for the development of ROH is the ontology reutilization, ontologies used by ROH will be hosted by the Hercules project, to the extent allowed by their licenses, allowing their reuse by third parties in the future.

- **Integration with existing information sources**, both from the university itself and from third party organizations.
- **Integration of the CRIS/RMS with external knowledge networks**, such as DBPedia [31] or Wikidata [32].
- **Release of ontologies and source code**, using the Creative Commons 4.0 BY-SA[7] or equivalent license.

4.2. Ontology reutilization

In order to maximize the reuse of popular ontologies and the compatibility of new developments within the framework of ASIO, priority has been given to the reuse of those entities (both classes and properties) that already fulfilled the objective of modelling the different aspects required. As described at the beginning of Section 4, the ontology reutilization has been performed manually including those concepts useful for the development of ROH. As one of the requirements of the Hercules project was that all the reused ontologies should persist over time, those ontologies have been backed up at the project's source code repository, and persistent URIs powered by w3id.org have been assigned to them. This allows reaching ontologies which are not currently available on the web, e.g. CERIF ontology (http://eurocris.org/ontology/cerif).

These reused entities have been combined among them and with entities explicitly created in ROH in order to model the data properly. Table 3 illustrates the most relevant entities reused in ROH[8]. Table 4 depicts the usage of the different reused ontologies in ROH.

4.3. Implementation

When designing and developing the ontology, priority has been given to its flexibility in order to ensure easy extensibility. This has been achieved thanks to two factors: the categorization of concepts instead of the use of hierarchies and the modularity of the ontology. By avoiding hierarchies, the ontology can be much more flexible. For instance, different institutions can use different hierarchies to classify their projects (e.g., universities that classify their projects according to the geographical scope of the call, as opposed to other universities which could classify them according to the public or private nature of the call). To tackle this, the use of categories has been prioritized, properties that allow the categorization of entities according to different criteria. For example, instead of creating a complex hierarchy under the `vivo:Project` class to represent the different types of projects, a Project could be categorized through the `roh:hasProjectCategorization` property. The range of this relation is `roh:ProjectClassification` which is a subclass of `skos:ConceptScheme`. Under `roh:ProjectClassification` each organization which wants to use ROH to model its knowledge, could develop its own vocabulary based on SKOS ontology to classify its projects. Within ROH the *project-classification.owl*[9] module has been developed describing the different European and Spanish project modalities. But, as said before, each organization could develop its own project classification, according to its specific needs. The same concept applies to the different specializations developed at ROH under `skos:ConceptScheme`, e.g., `roh:CompanyClassification`, `roh:ExpenseClassification`, `roh:FundingProgramClassification` or `roh:HRClassification`.

However, thanks to its modular design (*core* and *vertical* modules, see 4.3.1), our ontology allows any European country, territorial administrative entity or university to develop its own sub-ontology (refinements and extensions of ROH) adapted to its reality.

In the same way, and to avoid the explicit declaration of hierarchies, a series of *defined classes* have been implemented. A defined class is a class that should not be instantiated directly, but rather, an instance will belong to it only if it complies with a series of restrictions. These classes have been used to define, for example, when an or-

[7] https://creativecommons.org/licenses/by-sa/4.0/

[8] To make reading easier, only classes have been described. Reused object properties and data properties could be checked at the ontology published at https://w3id.org/roh.

[9] http://w3id.org/roh/project-classification

Table 3

Overview of entities reused in ROH. Prefixes are shown at Table 4.

Entity	Comments
vivo:AcademicDegree	Describes the degrees offered by a vivo:University and obtained by different people (foaf:Person). Specializations of this class created at ROH: roh:BachelorsDegree, roh:DoctoralDegree and roh:MastersDegree.
vivo:Certificate	Describes a document confirming certain characteristics of a person or organization, usually provided by some form of external review, education, or assessment. Specializations of this class created at ROH: roh:Award, roh:CourseCertificate and roh:LanguageCertificate.
vivo:License	Licenses are usually issued in order to regulate some activity that is deemed to be dangerous or a threat to the person or the public or which involves a high level of specialized skill.
foaf:Organization	In ROH, different specializations of this entity has been created in order to describe different actors participating in the RMS, specifically: roh:AccreditationIssuer, roh:EthicsCommittee, ManagementUnit, roh:ResearchGroup and roh:UniversityDivision. Other specializations of this entity defined at VIVO, such as vivo:Department, vivo:AcademicDepartment, vivo:Foundation, vivo:GovernmentAgency, vivo:University have been reused too.
vivo:Company	This entity, defined as a specialization of foaf:Organization has been specialized by the implementation of the company types defined by the European Commission: roh:LargeEnterprise, roh:MediumEnterprise, roh:SmallEnterprise and roh:MicroEnterprise.
vivo:Institute	This entity, defined as a specialization of foaf:Organization has been specialized by the implementation of roh:ResearchInstitute.
foaf:Person	This entity, which describes an instance of a human being, has been reused in ROH to model all the human participants in the RMS.
skos:ConceptScheme	This entity represents an aggregation of different entities belonging to skos:Concept entity. In ROH, the following specializations of this entity have been created in order to classify collections of instances which categorizes different entities, such as roh:AcademicSubject (including its specializations roh:BachelorsDegreeSubject and roh:MastersDegreeSubject), roh:AdministrativeEntity, roh:CompanyClassification, roh:Country, roh:ExpenseClassification, roh:FundingProgramClassification, roh:HumanResourceClassification, roh:KnowledgeArea, roh:ProjectClassification, roh:PropertyClassification and roh:ResearchProblem.
geonames:Feature	This entity represents any feature (location) form the Geonames dataset; it has been reused in ROH to describe the locations in different contexts.
bibo:Document	This entity and its subclasses have been widely used at ROH. Entities such as roh:PeerReviewedArticle, roh:BlogPost, roh:WorkshopPaper, roh:PressArticle, roh:Catalog, or roh:CurriculumVitae, among others.
bibo:Report	Subclass of bibo:Document, specializations of this entity have been created, such as roh:EthicalReport (and its specializations roh:EthicalAudit and roh:EthicalValidation), roh:EvaluationSummary, roh:Justification and roh:TechnicalReport.
bibo:Thesis	Specializations of this entity has been created in ROH, i.e., roh:DegreeThesis, roh:MastersThesis and roh:PhDThesis.
vivo:Contract	In ROH the following entities have been created in order to represent different types of contracts: roh:PatentContract, roh:PersonContract, roh:ProjectContract, roh:ServiceContract.
vivo:Position	This entity is crucial in ROH as it allows modelling the position a foaf:Person holds in a foaf:Organization. In ROH the following additional specializations have been developed: roh:FacultyPositionEmeritus, roh:LibrarianPositionEmeritus and roh:ResearcherPosition.
vivo:Relationship	In addition to those entities modelled under vivo:Position, different classes have been modelled as subclasses of vivo:Relationship, such as SupervisingRelationship (and its specializations Bachelors/Masters/PhDSupervisingRelationship).
obo-bfo:BFO_0000023 (*Role*)	In addition to the vivo:Position class, *obo-bfo:Role* class is one of the most important, since it allows defining the role of different actors in organizations, projects, activities, and so on. Different specializations have been created in addition to existing ones, e.g.: roh:AuditeeRole, roh:AuditorRole, roh:ExternalMemberRole, roh:SuperviseeRole, roh:SupervisorRole or roh:ThirdPartyContractorRole.

Table 4
Ontologies reused in ROH

Prefix	Ontology Name	Classes	object properties	data properties
bibo	Bibliographic Ontology http://purl.org/ontology/bibo	26	6	13
foaf	FOAF (Friend of a Friend) http://xmlns.com/foaf/0.1	5	6	6
gn	Geonames ontology http://www.geonames.org/ontology#	1	1	0
obo-bfo	OBO Foundry, Basic Formal Ontology http://www.obofoundry.org/ontology/bfo.html	5	2	0
obo-iao	OBO Foundry, Information Artifact Ontology https://github.com/information-artifact-ontology/IAO/	5	0	0
obo-ero	OBO Foundry, eagle-i Research Resource Ontology (ERO) https://open.catalyst.harvard.edu/wiki/display/eaglei/Ontology	29	0	0
obo-ro	OBO Foundry, Relations Ontology http://www.obofoundry.org/ontology/ro.html	0	5	0
roh	Red de Ontologías Hercules http://w3id.org/roh	144	109	48
skos	SKOS Simple Knowledge Organization System RDF Schema http://www.w3.org/2004/02/skos/core#	2	1	0
terms	DCMI Metadata Terms https://www.dublincore.org/specifications/dublin-core/dcmi-terms/			
vcard	vCard Ontology - for describing People and Organizations https://www.w3.org/2006/vcard/ns#	8	6	1
vivo	VIVO Ontology for Researcher Discovery http://vivoweb.org/ontology/core#	46	22	14
cito	The Citation Typing Ontology (CiTO) http://purl.org/spar/cito	0	3	0
oa	The Web Annotation Data Model http://www.w3.org/ns/oa#	2	1	0

ganization is a Funding Organization. A Funding Organization is defined as an Organization or any of its subclasses (University, Research Organization, Government Agency, etc.), which provides funds to some Funding and promotes some Funding Program or Funding Source. So, in the case that a instance meets the following restrictions, the OWL reasoner will automatically classify it as a Funding Organization, through the rule expressed in Eq. (3).

$$\begin{aligned}\forall x \forall y \forall z (Organization(x) \land (FundingProgram(z) \lor FundingSource(z)) \land \; promotes(x,z) \; \land \\ Funding(y) \land funds(x,y) \rightarrow FundingOrganization(x)).\end{aligned} \tag{3}$$

Last, different ontology design patterns have been used in order to implement ROH, i.e. *partOf*[10], *Participation*[11] and *AgentRole*[12] patterns. The objective of the *partOf* pattern is to represent entities and their parts. This pattern is used in multiple relations in ROH, e.g., a `foaf:Organization` `obo-ro:hasPart` `foaf:Organization`, that is, an organization can be composed of sub-organizations, and this can be `obo-ro:partOf`, that is, be part

[10] http://ontologydesignpatterns.org/wiki/Submissions:PartOf
[11] http://ontologydesignpatterns.org/wiki/Submissions:Participation
[12] http://ontologydesignpatterns.org/wiki/Submissions:AgentRole

of a parent organization. On the other hand, the *Participation* pattern allows the representation of the participation of an object in an activity or event. In ROH, we have used this design pattern to model, for example, the roles that a `foaf:Agent` through an `obo-bfo:Role` can play (`obo-bfo:realizedIn`) in a `roh:Activity`. Last, the *AgentRole* pattern, represents the agents and the roles played by those agents. This pattern has been used when representing the roles held by organizations and individuals at projects or research publications. See Section 5 for examples of the usage of this role.

4.3.1. Modularity

In this section we are going to provide more details about the modularity applied to ROH which is split into a central and also a number of peripheral components. The inspiration of this architecture comes from a loose reading of [33], obviously adapting Fodor's cognitivist approach to computer science and information architecture. It distinguishes between two fundamentally different types of information processing, relying upon information architecture and datasets: a central type, or core, and vertical types.

On the basis of this distinction, we develop an architecture of the ontological organization as involving both very specialized modules (*vertical modules*) and what we call domain-general, non-modular knowledge (*core ontology*). Two properties of modularity in particular, informational encapsulation and domain specificity, make it possible to tie together questions of functional architecture with those of knowledge content.

ROH network of ontologies is thus divided into 2 main parts:

- The generic ontology, *core* module, contains the most important entities and properties to model information in the academic domain. It contains the central part of the network of ontologies. It covers the academic domain, being agnostic to the country or the research organization whose information wants to be modelled with it.
- A set of vertical modules which include, on one hand, specializations of some academic concepts for a given country domain. For instance, the figure Associate Professor in the Spanish academic domain would be defined in the vertical module `university-HR-es` and is identified by the URI http://w3id.org/roh/university-HR/es#ProfesorTitularDeUniversidad. To incorporate specific modules to the ontology, it is enough to create a new ontology, import the required higher level ontology entities and create the new classes or properties needed. For example, if a new Spanish university wants to make use of ROH to add a series of positions of its own, it could import the `university-HR-es` ontology, and under `vivo:Position`, where the hierarchies for the typical university positions appear, create its own specific positions as subclasses. On the other hand, within ROH a set of vertical modules including different Knowledge Area classifications have been included, e.g., the academic subject areas described by the UNESCO. Similar to the previous case, custom classifications could be added under the `skos:ConceptScheme` instance, e.g., the knowledge areas provided by the Spanish FECYT.

Figure 2 depicts the implemented modules, which are described next:

- `roh-core`: this module implements the core concepts and relationships, those which can be considered as common and universal to all the university systems worldwide.
- `geopolitical`: it is a module focusing on administrative subdivisions of countries relying upon standard codes. It includes as departing samples the whole subdivisions, up to three levels, of Andorra, Spain and Portugal:
  - ∗ Andorra: implementing just the first-level subdivisions (*parròquies*)
  - ∗ Portugal: implementing both the first (*distritos e regiões*) and the second-level (*municípios*) subdivisions.
  - ∗ Spain: covering the first (*comunidades autónomas*), the second (*provincias*) and the third-level subdivisions (*municipios*).

  SKOS-Core was also chosen to model a clearly hierarchical domain and the dataset is massively enriched multilingually and ponderously linked to relevant national and international vocabularies, such as the EU's Country Named Authority List[13].

[13] https://data.europa.eu/data/datasets/country?locale=en

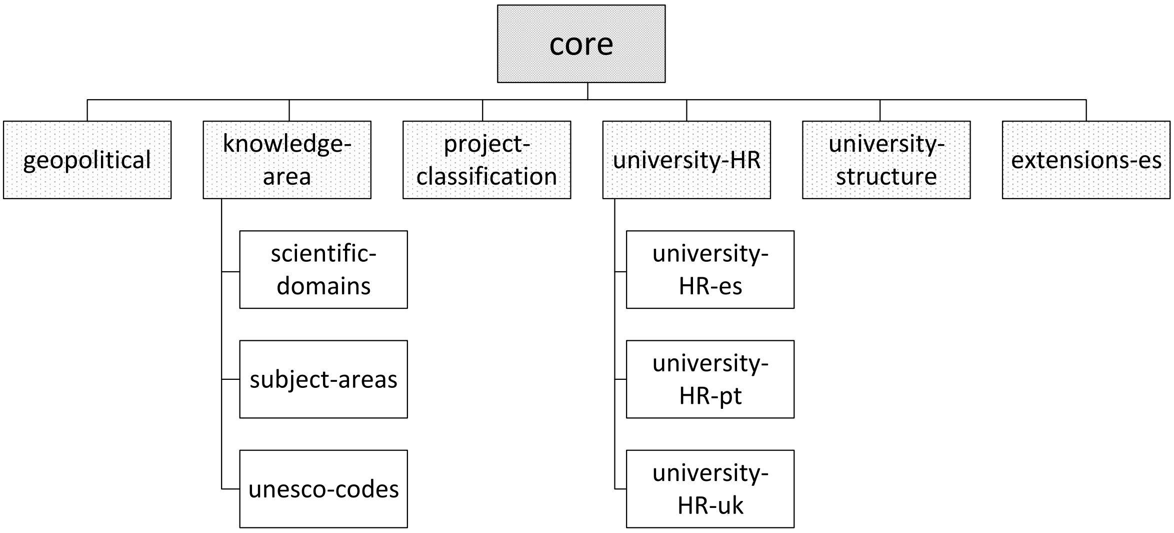

Fig. 2. Modules implemented at ROH.

The main goal of this vertical module, which is in a way also transversal, is to geopolitically *locate* agents, organizations and other resources included in ROH ontology with an encompassing granularity.

- `knowledge-area`: This module implements concepts related to the knowledge area of an instance. For this project, we implemented three different schemes of knowledge areas, each of them in one module:
  * `es-scientific-domain`: Spain's Ministry of Science, Innovation and Universities[14], through its State Research Agency[15], published a document featuring a number of relevant agency-related Scientific domains[16] which are the basis for several ones among the competency questions provided by the University of Murcia in order to model the ontology. The document is sourced in PDF and thus not computationally processed. A conversion into SKOS was carried out to feed this module, hence reaching a high-quality, or 5-star quality[17], linked data format ("non-proprietary format (e.g. CSV instead of excel), open standards from W3C (RDF and SPARQL) to identify things and link your data to other people's data to provide context").
  * `es-subject-area`: The same approach has been used to create the related vertical module Subject areas, from the same State Research Agency[18], which is used for slightly different cases with the core ontology, but was modelled equally following the schema provided by SKOS.
  * `unesco-codes`: this module implements the UNESCO nomenclature for fields of science and technology[19]. This module was originally developed by the University of Murcia [34].

At ROH an alignment among those three different scheme has not been done. However, they can be used together to define the knowledge area a research object belongs. Listing 2 shows an example of how to define the knowledge area of a project through ROH.

[14] http://www.ciencia.gob.es/

[15] http://www.ciencia.gob.es/portal/site/MICINN/menuitem.8d78849a34f1cd28d0c9d910026041a0/?vgnextoid=664cfb7e04195510VgnVCM1000001d04140aRCRD

[16] http://www.ciencia.gob.es/stfls/MICINN/Ayudas/PE_2013_2016/PE_Promocion_e_Incorporacion_Talento_y_su_Empleabilidad/FICHEROS/SE_Incorporacion/Ayudas_contratos_RYC_2016/Clasificacion_areas_cientificas_2016_AEI.pdf

[17] https://www.w3.org/community/webize/2014/01/17/what-is-5-star-linked-data/

[18] http://www.ciencia.gob.es/portal/site/MICINN/menuitem.8d78849a34f1cd28d0c9d910026041a0/?vgnextoid=664cfb7e04195510VgnVCM1000001d04140aRCRD

[19] https://skos.um.es/unesco6/

Listing 2: An example of how to use the different knowledge area modules to define a project related to Mathematics.

```

@prefix rohsubj:     <http://w3id.org/roh/subject-areas#> .
@prefix rohsci:  <http://w3id.org/roh/scientific-domain#> .
@prefix uneskos-individuals:  <http://w3id.org/roh/unesco-individuals#> .
@prefix roh: <http://w3id.org/roh#> .

[...]

<http://w3id.org/roh/data#my_project>
        a                           vivo:Project ;
        roh:hasKnowledgeArea        uneskos-individuals:12 ,
                                    rohsci:ES_SCIENTIFIC_DOMAIN_LEVEL_2_MTM ,
                                    rohsubj:ES_SUBJECT_AREA_LEVEL_2_MTM .
```

- `project-classification`: this module implements the classification of different calls granted by the Spanish Government and the European Commission, such as Horizon2020 or ITN-ETN, among others. This module incorporates mainly a scheme of the classification of the projects, which had to be integrated under `roh:ProjectClassification`. The object property `roh:hasProjectClassification` relates the resource to the concept (instance of `skos:Concept`) which has to be in the custom schema (this relationship is modelled through `skos:inScheme`) that is integrated under `roh:ProjectClassification` entity Eq. (4).
- `university-HR-es`, `university-HR-pt` and `university-HR-uk`: those modules implement the different human resource classifications followed by universities in Spain, Portugal and UK. They incorporate mainly a scheme of the human resource classifications, which had to be integrated under `roh:HRClassification`. The object property `roh:hasHRClassificacion` relates the resource with the concept (instance of `skos:Concept`) which has to be in the customized schema (this relationship is modelled through the `skos:inScheme` object property) that is integrated under `roh:HRClassification` entity Eq. (5).
- `university-structure`: another vertical module includes the entire list of the universities of Spain, for which some rich data was retrieved from the RUCT[20] portal. Also modelled using SKOS, it includes encompassing metadata about each institution, such as specific codes for each centre, multilingual labels when applicable and other information. It also includes a limited sample of subdivisions (schools, faculties, centres) from the universities of Murcia, Oviedo, Santiago de Compostela and the Basque Country, and it receives as well special care regarding multilingualism, official codes from the Ministry, etc.
- `extensions-es`: in this module concepts related to Spanish taxes and accounting are implemented.

$$\forall x \forall y (Project(x) \wedge hasProjectClassification(x,y) \rightarrow Concept(y) \wedge \exists z (inScheme(y,z) \wedge ProjectClassification(z))). \tag{4}$$

$$\forall x \forall y (hasHRClassification(x,y) \rightarrow Concept(y) \wedge \exists z (inScheme(y,z) \wedge HRClassification(z))). \tag{5}$$

In `roh-core`, there are other entities which allow to incorporate a new customized schema under them, e.g., `roh:FundingProgramClassification`. In this case the object property which relates a `roh:FundingProgram` and its `roh:FundingProgramClassification` is `roh:hasFundingProgramClassifi-`

[20] https://www.educacion.gob.es/ruct/consultacentros.action?actual=centros

cation and their rules are similar to those described at Eq. (6). The same applies to roh:ExpenseClassification and roh:hasExpenseClassification.

$$\forall x \forall y (FundingProgram(x) \wedge hasFundingProgramClassification(x, y) \rightarrow Concept(y) \wedge \exists z (inScheme(y, z) \wedge FundingProgramClassification(z))) \tag{6}$$

In this paper, we focus on the main concepts and relationships implemented by the roh-core module, which will be described in the following section.

5. ROH concepts

In this section, we pay special attention to the foaf:Agent, vivo:Project, roh:Funding, Information Content Entity[21], roh:Activity, roh:ResearchObject and roh: Metric classes and their relationship with other classes. These classes are the most important ones in ROH and represent the main concepts of roh-core module. Figure 3 shows the main relationships among these classes.

The source code of ROH is hosted in a GitHub repository[22], in which more information about other classes and relationships defined in roh-core module can be found.

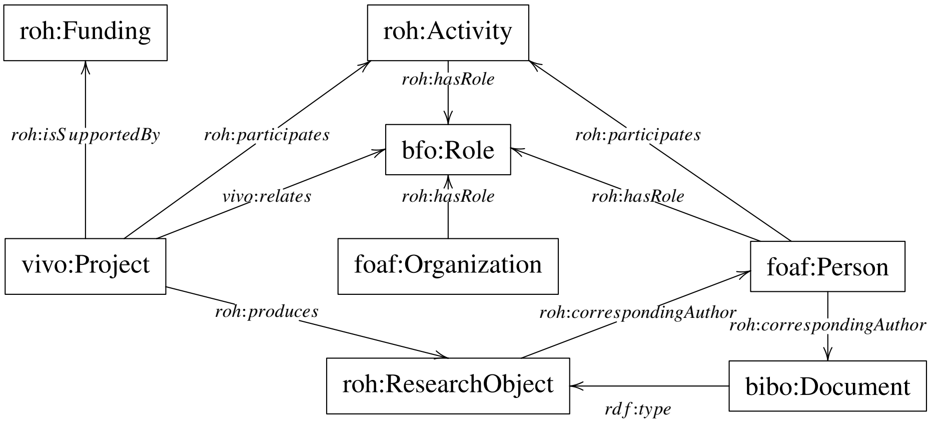

Fig. 3. Main concepts described at roh-core

5.1. Agent

The foaf:Agent class, has been imported from FOAF ontology [35], being foaf:Person, foaf:Organization and foaf:Group its subclasses. In ROH, we make extensive use of two of these subclasses, i.e., foaf:Person and foaf:Organization.

5.1.1. Person

This class, imported from FOAF ontology [35], represents a human participant in the academic and research process. A Person could be defined in ROH including some of the basic FOAF properties such as foaf:name, foaf:surname, foaf:nickname, foaf:title, foaf:mbox, or vivo:description among others. In

[21] obo-iao:IAO_0000030

[22] https://github.com/HerculesCRUE/ROH. The OWL file of roh-core module can be found at https://herculescrue.github.io/ROH/roh/core/roh-core.owl

ROH, a Person is characterised through its Role (obo-bfo:BFO_0000023) within an Organization. Since ROH describes the research and academic domain, a Person holding a Researcher Role could be identified through his/her vivo:researcherId, vivo:scopusId, or roh:ORCID among others. Figure 4 shows the main relationships that a Person may exhibit within roh-core:

- roh:AuthorMetric: represents the value of the research metrics of some Person such as the h-index or the i10-index.
- roh:CurriculumVitae: represents the CV of a Person.
- vivo:Position: represents the Position a Person has in an Organization, e.g., vivo:FacultyAdministrativePosition or vivo:FacultyPosition.
- roh:ResearchObject: represents the different research resources authored by a Person, either if he/she is the main author or a contributor.
- roh:Activity: represents an Activity in which the Person participates, such as a bibo:Conference, a vivo:Internship or a vivo:Meeting among others.
- bfo:BFO_0000023 (Role): represents the Role a Person has in an Activity, Project or Relationship, among others.

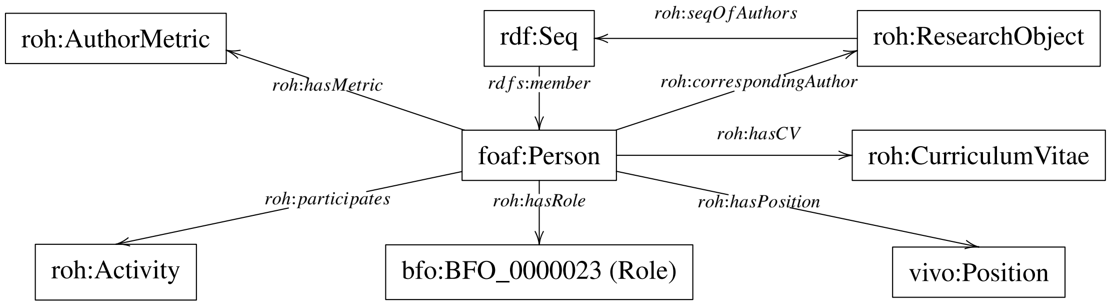

Fig. 4. Main relationships of foaf:Person class.

5.1.2. *Organization*

An Organization in ROH (foaf:Organization) encompasses the different types of organizations that may exist in the research domain. This class has a deep hierarchy of subclasses mostly imported from VIVO, such as vivo:Center, vivo:Company, vivo:Department, vivo:Institute, vivo:University or vivo:Foundation, among others. Also, some classes such as roh:ResearchGroup, roh:UniversityDivision, roh:EthicsCommittee have been defined in ROH in order to include the different organizations involved in a research process. Figure 5 shows the main relationships that an Organization may have in roh-core:

- roh:Accreditation: represents accreditations, e.g. roh:ResearchAccreditation or roh:AcademicAccreditation that an Organization may have. These accreditations are issued by an Organization of type roh:AccrediationIssuer.
- vivo:Company: this class models the spin-offs an Organization may have.
- gn:Feature: represents the geographical location an Organization may have.
- vivo:DateTimeInterval: represents the time interval associated to the existance of an Organization.
- roh:FundingAmount: represents the funding amounts part of a roh:Funding that an Organization may receive.
- bfo:BFO_0000023 (Role):represents the Role an Organization has in an Activity, Project or Relationship, among others.
- foaf:Organization: an Organization could be related to another one if it is the successor or predecessor of the first one, or if it belongs to a bigger Organization.

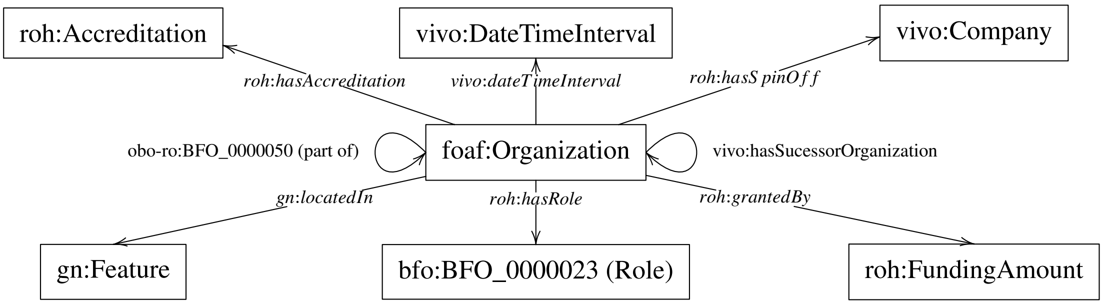

Fig. 5. Main relationships of `foaf:Organization` class.

5.2. Project

A Project is a collaborative process in business and science that often involves research or design and is carefully planned to achieve a particular goal. In ROH, the `vivo:Project` class has been reused to represent a Project. In ROH, a Project must be related to its starting date and finishing date (if any) and to the People and Organizations participating on it, as it is described by Eq. (7).

$$\forall x(Project(x) \rightarrow \exists y(relates(x,y) \land Role(y)) \land \exists t(dateTimeInterval(x,t) \land DateTimeInterval(t))). \tag{7}$$

$$\forall y(Role(y) \rightarrow \exists z(hasRole(z,y) \land Agent(z))). \tag{8}$$

The main classes related to `vivo:Project` are shown in figure 6:

- `roh:Activity`: represents an Activity where a Project participates, e.g., `vivo:InvitedTalk` or `bibo:Conference`.
- `roh:ProjectExpense`: an Expense produced by the execution of a Project.
- `roh:ResearchObject`: represents the research results produced by a Project, for example a `roh:PhD-Thesis`.
- `roh:Dossier`: represents a collection of documents, which could include different documents related to a Project, such as the `vivo:ResearchProposal`, a `roh:EvaluationSummary` or a `bibo:Report`, among others.
- `roh:Funding`: represents the Funding supporting the expenses of a Project.
- `roh:ProjectClassification`: is a subclass of `skos:ConceptScheme` which describes the taxonomy of the projects promoted by the European Commission. Following the modular approach of ROH, explained at Section 4.3.1, each organization can create its own taxonomies.

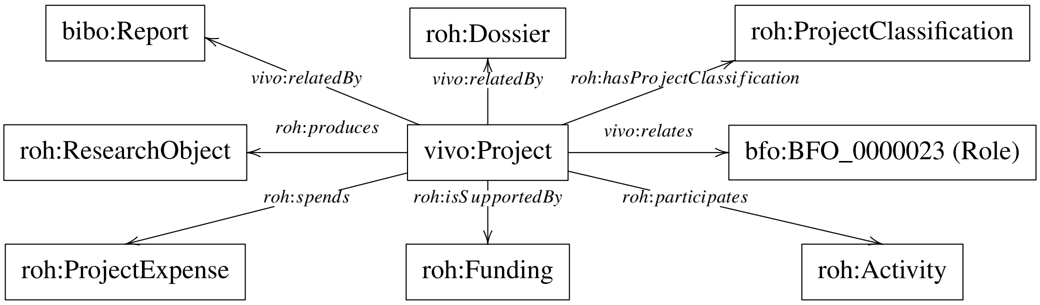

Fig. 6. Main relationships of `vivo:Project`.

5.3. Funding

The roh:Funding class represents the specific Funding action which funds a Project. For example, the specific Funding action of the Hercules-ASIO project is the one identified by the ID E-CON-2018/88/OT-AM. The Funding concept and its related concepts have been modelled to allow the description of complex funding schemes, e.g., projects funded by different public and private initiatives. Figure 7 depicts the main classes related to roh:Funding:

- roh:FundingAmount: a Funding is divided into several Funding Amounts, which grant different Organizations. A FundingAmount represents the monetary amount received by an Organization for a time period. A FundingAmount is intended to represent the different reporting periods a Project could have. A FundingAmount is described by the vivo:dateTimeInterval it covers and the received roh:monetaryAmount for that time period, among others.
- roh:FundingProgram: represents the Funding Program or initiative which provides funds to a specific Funding action.
- roh:FundingSource: represents the source from which the Funding for a specific Funding Program comes, e.g., different regional Funding Programs could be funded by the European Regional Development Fund.
- roh:FundingOrganization: represents the Organization which promotes different Funding Programs and Funding Sources and funds a Funding. As seen at Section 4.3, roh:FundingOrganization is a defined class. To belong to this class, the classes must fulfill the rules described at Eq. (3).

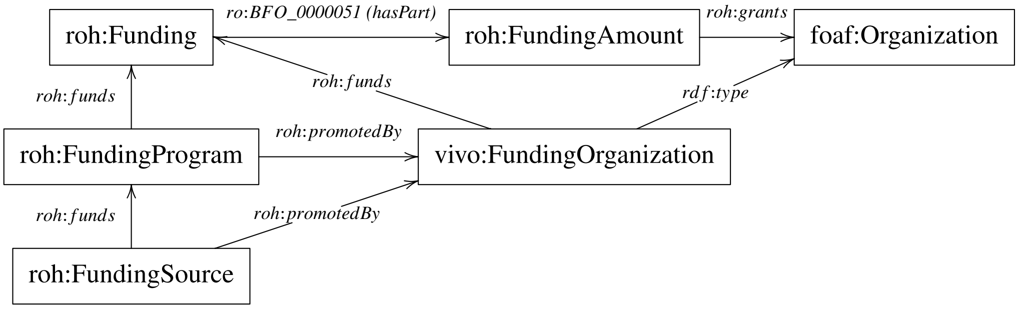

Fig. 7. Main relationships of roh:Funding class.

5.4. Information Content Entity

The obo-iao:IAO_0000030 class (Information Content Entity) has been imported from the Information Artifact Ontology (obo-iao)[23]. It represents a wide collection of publications, patents, documents, repositories, or web pages, so it has a deep taxonomy under it. The most relevant classes of this taxonomy can be seen in Figure 9. In this paper, we focus on two subclasses widely used in the research domain: Article and Journal Article.

The bibo:Article class represents a written composition on a specific topic. In ROH, an Article must necessarily be related to its issue date, its authors and its corresponding organization, as described at Eq. (9). The main classes related to an Information Content Entity are shown in Figure 8:

- vivo:DateTimeValue: represents the creation or publication date of an Article.
- rdf:Seq: represents the list of Persons that contributed to an Article or Document.
- foaf:Person: represents the corresponding author or a contributor of an article.

[23] http://www.obofoundry.org/ontology/iao.html

- bibo:Book or bibo:Collection: represent the Book or bibo:Collection where an Article is published. Examples of subclasses of bibo:Collection are roh:Dossier or bibo:Journal.
- roh:PublicationMetric represents the metrics of those Articles that are published in a Journal.
- vivo:Project: represents the Project which produces a Document or Article.

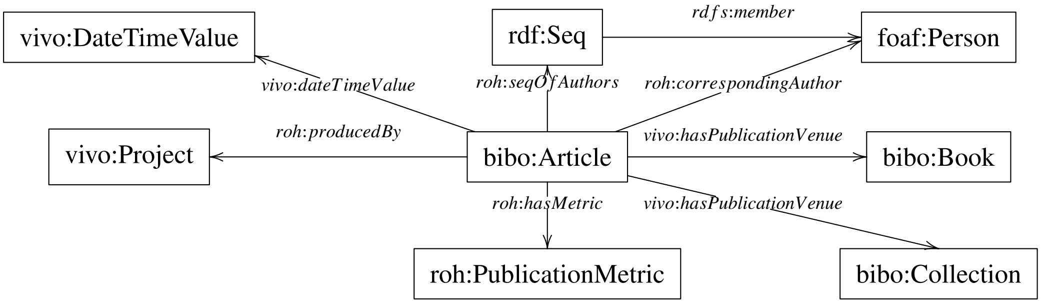

Fig. 8. Main relationships of obo-iao:IAO_0000030 class.

The obo_iao:0000013 (Journal Article) is a subclass of bibo:Article and represents those Articles which have been published in a Journal. A Journal Article needs to be related to several classes to ensure it contains the minimum information required by the ASIO project. In this sense, a Journal Article class must be related to its corresponding author and the Journal in which has been published. Those restrictions are described at Eq. (10).

$$\begin{aligned}\forall x(Article(x) \rightarrow & \exists y(dateIssued(x,y) \wedge dateTimeValue(y)) \wedge \\ & \exists t(seqOfAuthors(x,t) \wedge Seq(t)) \wedge \\ & \exists z(correspondingOrganization(x,z) \wedge Organization(z))).\end{aligned} \tag{9}$$

$$\begin{aligned}\forall x(JournalArticle(x) \rightarrow & \exists y(hasPublicationVenue(x,y) \wedge Journal(y)) \wedge \\ & \exists t(hasMetric(x,t) \wedge PublicationMetric(t)) \wedge \\ & \exists p(correspondingAuthor(x,p) \wedge Person(p))).\end{aligned} \tag{10}$$

5.5. Activity

The roh:Activity class represents the activities in which Agents and Projects take part. The Activity must necessarily be related to the interval of time when it happens, as described at Eq. (11). The main classes related to roh:Activity are shown in Figure 10:

- vivo:DateTimeInterval: represents the time interval when an Activity occurs.
- vivo:Project or foaf:Agent: represent the Project or Agent participating in the Activity.
- bfo:BFO_0000023 (Role): represents the Role of an Agent in the Activity. The relationships among these three classes can be seen in Figure 10.
- bibo:Document: represents a Document involved in the Activity.
- gn:Feature: describes the place where an Activity occurs.

$$\begin{aligned}\forall x(Activity(x) \rightarrow & \exists y(DateTimeInterval(y) \wedge dateTimeInterval(x,y)) \wedge \\ & \exists z((Agent(z) \vee Project(z)) \wedge participatedBy(x,z))).\end{aligned} \tag{11}$$

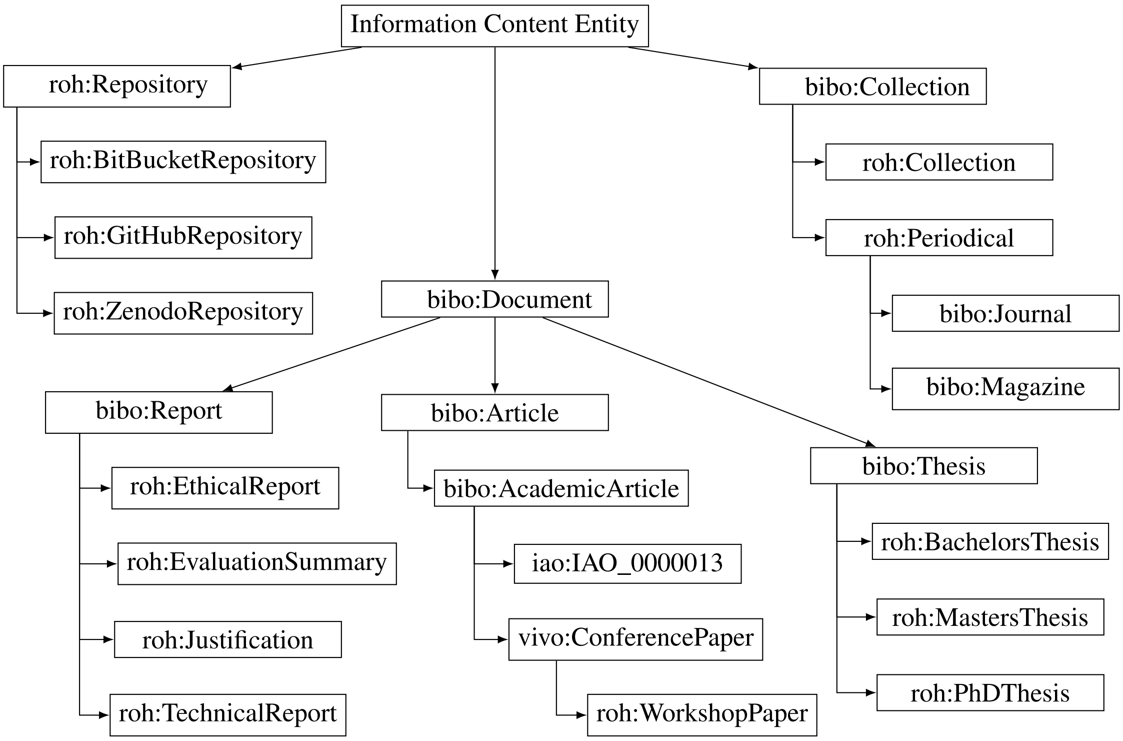

Fig. 9. The most relevant subclass of `obo-iao:IAO_0000030` (Information Content Entity) class.

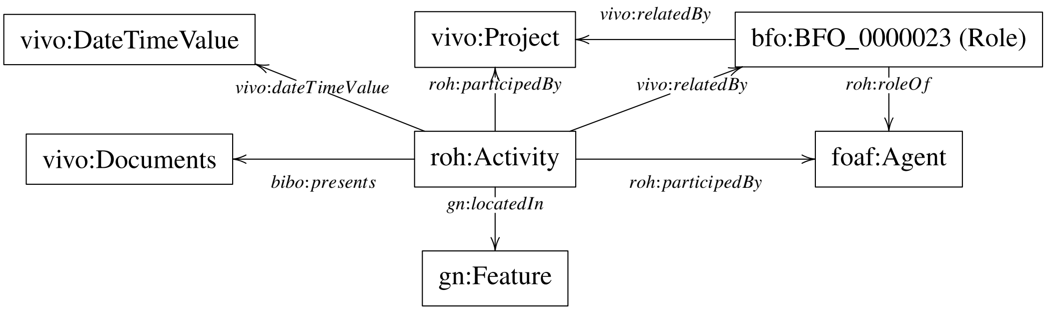

Fig. 10. Main relationships of `roh:Activity` class.

5.6. *Research Result and Research Object*

A Research Result is composed of the different research objects generated by a researcher through work on a project. Each author can create her own research results, including the research objects that she considers.

The `roh:ResearchObject` class represents a particular research result generated by a researcher. Usually a `roh:ResearchObject` results from working on a `vivo:Project`. It is a defined class that follows the Eq. (12, 13), and a subclass of the Research Result. The main relationships associated with `roh:ResearchObject` are shown in Figure 11:

- `rdf:Seq`: is the seq of persons that contributed in the creation of a Research Object.
- `foaf:Person`: represents the corresponding author of the research object.
- `foaf:Organization`: represents the corresponding organization of the research object.
- `roh:ResearchResult:` is the research result that contains the research object.

- `roh:KnowledgeArea`: is the set of knowledge areas that are related to the research object.
- `vivo:Project`: is the project within which researchers generate research objects as result of their work on it.

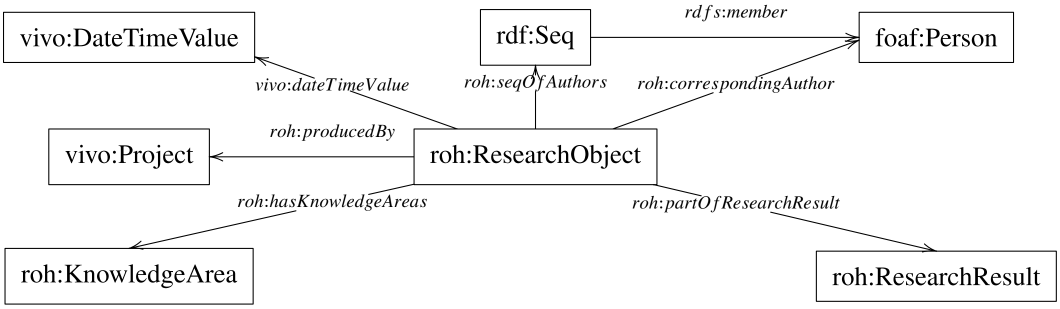

Fig. 11. Main relationships of `roh:ResearchObject`.

$$\forall x \forall y (Project(x) \wedge producedBy(y,x) \rightarrow ResearchObject(y)). \tag{12}$$

$$\forall x \forall y (ResearchResult(x) \wedge partOfResearchResult(y,x) \rightarrow ResearchObject(y)). \tag{13}$$

5.7. Metric

We define three different metrics in `roh-core` module for three different elements: authors, publications, and journal articles; and these metrics are represented by three subclasses of `roh:Metric` class: `roh:AuthorMetric`, `roh:PublicationMetric` and `roh:JournalMetric`. The instance and its metric have to be associated through `roh:hasMetric` object property.

The `roh:AuthorMetric` class has three metrics (h-index, i10-index, and citation count), related to the research quality of the author and available at `roh:citationCount`, `roh:h-index` and `roh:h10-index` data properties.

The other two metrics are more closely related to each other. But the main difference between these metrics is that `roh:JournalMetric` describes information that can't be changed about the impact of the journal at the time an article was published, while `roh:PublicationMetric` describes information that can be changed about how often the article was cited.

The `roh:PublicationMetric` class describes the intrinsic information of a publication: its citations. As in the field of research there are multiple citation networks, each with its own collection of publications, each network has a different number of citations for the same publication. Therefore, a journal article can have as many publication metrics as there are networks where this article is available. A publication metric has two features:

- `roh:metricName`: The citation network within which the number of citations was determined.
- `roh:citationCount`: The number of citations in this network. As we explained before, if this information is updated, this number has to be updated too.

The journal in which an article is published has to be related to its impact at the time of this publication, namely its impact factor. There are several metrics to define the impact of a journal, and for each of them it's possible to define one journalMetric from the same journal at the same time. A journal metric has 3 features:

- The impact factor of the journal is the name of the metric to measure the impact of the journal at the moment of its publication. The most common are Journal Impact Factor (JIF) and SCImago Journal & Country Rank (SJR). The name has to be registered by the data property `roh:impactFactorName` and the number of this impact through the data property `roh:impactFactor`. These two properties are necessary to define this metric.

- The quartile and the ranking of the journal based on its impact factor. This metric is represented with the data property `roh:quartile` and `roh:ranking` respectively.
- The issue date of this impact factor. This information is registered by `vivo:dateIssued` object property .

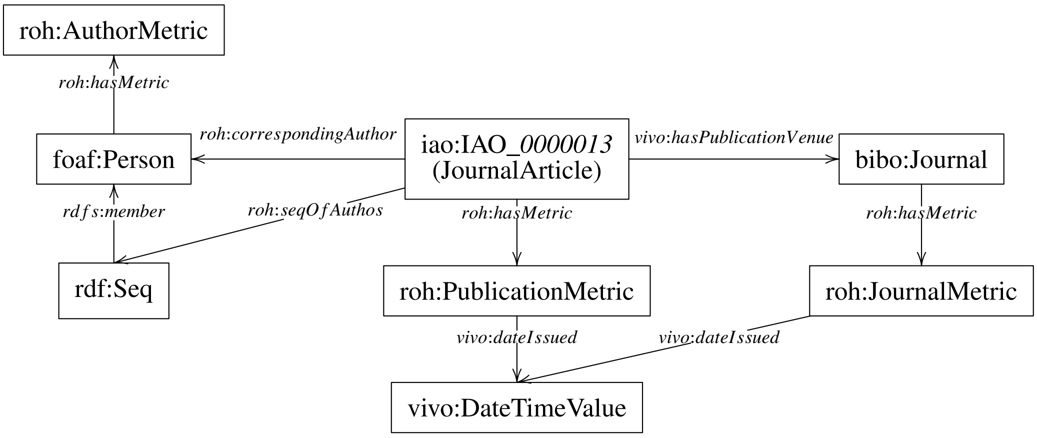

Fig. 12. Main relationships of `roh:PublicationMetric`, `JournalMetric` and `roh:AuthorMetric`.

6. Evaluation and continuous refinement

Within Hercules-ASIO project, different actions have been carried out in order to evaluate the ROH ontology. In this section, three mechanisms used to validate the ontology are explained, namely, the competency query, the mapping of the CVN from FECYT to ROH and the SHACL validations. Lastly, the continuous refinement process devised for continuous development and refinement of ROH is explained.

6.1. Competency questions

Competency questions are a set of questions set up by the University of Murcia in order to check if the ontology fits the requirements defined. For that, two datasets[24], modelled through ROH was prepared: a synthetic dataset and a dataset based on the database from the web page of the MORElab[25] research group. Those datasets contain the description of different concepts from the academic domain, and their relations as well. Afterwards, competency questions were translated to SPARQL queries, and executed against a SPARQL endpoint in which both the sample dataset with instance data and ROH ontology were loaded. The results obtained were analyzed in order to check if they were the expected ones. If not, ROH underwent a refinement process. For executing the SPARQL queries Apache Jena Fuseki[26] has been used as SPARQL endpoint, and Openllet[27] as a reasoner.

Below, some examples of these SPARQL queries are presented. Notice that 68 queries were developed in total. Section 6.4 provides more details about the usage of competency questions.

- QA: this query retrieves Research Groups and Research Institutes working on the Knowledge Area of Artificial Intelligence (`uneskos-individuals:120304`), and the name of the University they belong.
- QB: this query retrieves Researchers working on the Knowledge Area of Artificial Intelligence (`uneskos-individuals:120304`), and the Position they have within their Research Group.

[24] https://github.com/HerculesCRUE/ROH/blob/main/validation-data/rdf/roh_data_edma.ttl
[25] https://morelab.deusto.es/
[26] https://jena.apache.org/documentation/fuseki2/
[27] https://github.com/Galigator/openllet

– QC: lists the scientific production (Research Objects) from a Research Center in the Knowledge Area of Artificial Intelligence (`uneskos-individuals:120304`) in a given date range. In addition, the type of research object returned and the corresponding author are provided.

Listings 3, 4 and 5 depict the SPARQL queries mentioned above, executed against the synthetic dataset, whilst Tables 5, 6 and 7 show the results obtained.

Listing 3: SPARQL query for competency question QA.

```
SELECT ?center ?centerName ?universityName
WHERE {
  ?center     a                     ?centerClass ;
              roh:hasKnowledgeArea  uneskos-individuals:120304 .
  ?university a                     vivo:University ;
              ro:BFO_0000051+       ?center ;
              roh:title             ?universityName .
  ?center     roh:title             ?centerName .
  FILTER(?centerClass = roh:ResearchGroup || ?centerClass = roh:ResearchInstitute).
}
```

Table 5

Results for the SPARQL query QA.

center	centerName	universityName
data:research-group-1	Research Group 1	University 1
data:research-group-3	Research Group 3	University 2
data:research-group-2	Research Group 2	University 1

Listing 4: SPARQL query for competency question QB.

```
SELECT ?researcher ?center ?positionClass
WHERE {
  ?researcher roh:hasKnowledgeArea  uneskos-individuals:120304;
              roh:hasPosition       ?position.
  ?position   a                     roh:ResearcherPosition.
  ?position   vivo:relates          ?center ;
              a                     ?positionClass.
  ?center     a                     ?centerClass.
  FILTER (?centerClass = roh:ResearchGroup || ?centerClass=roh:ResearchInstitute)

  FILTER NOT EXISTS {
      ?position   a ?otherClass.
      ?otherClass rdfs:subClassOf ?positionClass .
      FILTER (?otherClass != ?positionClass)
  }
}
```

Listing 5: SPARQL query for competency question QC.

```
SELECT DISTINCT ?researchObject ?researchObjectClass ?organization ?author ?dateTime ?title
WHERE {
  ?researchObject a                         roh:ResearchObject;
                  a                         ?researchObjectClass;
```

Table 6

Results for the SPARQL query QB.

researcher	center	positionClass
data:researcher-3	data:research-group-1	roh:ResearcherPosition
data:researcher-2	data:research-group-1	roh:ResearcherPosition
data:researcher-1	data:research-group-1	roh:ResearcherPosition

```
                    roh:correspondingOrganization  ?organization;
                    vivo:dateIssued                ?dateIssued ;
                    roh:hasKnowledgeArea           uneskos-individuals:120304 .
  ?organization     a                              ?classOrganization .
  ?dateIssued       vivo:dateTime                  ?dateTime .
  ?researchObject   roh:title                      ?title .
  ?researchObject   roh:correspondingAuthor        ?author .
  FILTER (YEAR(?dateTime) >= "1980"^^xsd:integer && YEAR(?dateTime) <= "2022"^^xsd:integer)
  FILTER (?classOrganization = roh:ResearchInstitute || ?classOrganization = roh:ResearchGroup)
  FILTER NOT EXISTS {
            ?researchObject a                        ?otherClass .
            ?otherClass         rdfs:subClassOf ?researchObjectClass .
            FILTER (?otherClass != ?researchObjectClass)
  }
  FILTER (str(?researchObjectClass) != "http://w3id.org/roh#ResearchObject")
  FILTER (str(?researchObjectClass) != "http://www.w3.org/2002/07/owl#NamedIndividual")
}
```

Table 7

Results for the SPARQL query QC.

researchObject	researchObjectClass	organization	author	dateTime	title
data:software-1	obo-ero:ERO_0000071	data:research-group-1	data:researcher-1	2020-04-27T00:00:00	A great software
data:experimental-protocol-1	roh:ExperimentalProtocol	data:research-group-1	data:researcher-1	2020-04-27T00:00:00	A great experimental protocol.
data:journal-article-2	obo-iao:IAO_0000013	data:research-group-1	data:researcher-1	2017-04-27T00:00:00	My great journal article
data:journal-article-1	obo-iao:IAO_0000013	data:research-group-1	data:researcher-1	2016-04-27T00:00:00	My great journal article
data:researcher-3-phd-thesis	roh:PhDThesis	data:research-group-1	data:researcher-3	2010-04-27T00:00:00	My fabulous PhD Thesis

On the other hand, the dataset created from the database of the MORElab research group[28] represents mainly research projects and publications. This dataset has been created using the Morph-KGC tool [36]. The resultant dataset[29] has more than 29,000 triples in which more than 600 researchers, 150 projects and more than 400 research articles are described. Thanks to the application of the ROH at a real research database, some minor issues have been detected an fixed into the ontology. Unfortunately, the database from the MORElab research groups lacks of some features covered by ROH, such as the Knowledge Areas. However, Listings 6 and 7 show some examples of queries could be executed against the MORElab dataset, while Tables 8 and 9 show a sample of the result of those queries.

Listing 6: SPARQL query for retrieving information about journal articles at MORElab database

```
SELECT ?title ?correspondingAuthorName ?correspondingAuthorSurname ?dateIssued ?journal
```

[28] https://morelab.deusto.es/

[29] Available at https://raw.githubusercontent.com/HerculesCRUE/ROH/main/validation-data/rdf/Dataset%20MORElab.ttl

```
?impactFactor ?quartile
WHERE {
  ?journalArticle   a iao:IAO_0000013 ;
                    roh:title ?title ;
                    roh:correspondingAuthor ?correspondingAuthor ;
                    vivo:hasPublicationVenue ?publicationVenue .
  ?correspondingAuthor foaf:name ?correspondingAuthorName ;
                       foaf:surname     ?correspondingAuthorSurname .
  ?publicationVenue roh:title ?journal ;
                    roh:hasMetric ?metric ;
                    vivo:dateIssued ?publicationDateIssued .
  ?metric roh:impactFactor ?impactFactor ;
          roh:quartile ?quartile .
  ?publicationDateIssued vivo:dateTime ?dateIssued .
}
```

Table 8

Results for the SPARQL query at Listing 6

title	correspondingAuthorName	correspondingAuthorSurname	dateIssued	journal	impactFactor	quartile
Exploring LOD through metadata extraction and data-driven visualizations	Oscar	Peña	"2016-06-08T00:00:00+00:00"^^xsd:dateTime	Program: electronic library and information systems	"0.556"^^xsd:float	Q4
An Ambient Assisted Living Platform Integrating RFID Data-on-Tag Care Annotations and Twitter	Diego	López-de-Ipiña	"2010-06-28T00:00:00+00:00"^^xsd:dateTime	Journal of Universal Computer Science	"0.578"^^xsd:float	Q4
Towards Citizen Co-Created Public Service Apps (sensors)	Mikel	Emaldi	"2017-06-02T00:00:00+00:00"^^xsd:dateTime	Sensors	"2.475"^^xsd:float	Q2
Context Management Platform for Tourism Applications	David	Buján	"2013-06-24T00:00:00+00:00"^^xsd:dateTime	Sensors	"1.953"^^xsd:float	Q1

Listing 7: SPARQL query for retrieving the list of projects and their respective leader organization, funding programs and funding organizations at MORElab database

```
SELECT ?title ?leaderOrganization ?fundingProgram ?fundingOrganization
WHERE {
  ?project  a vivo:Project ;
            roh:title ?title ;
            roh:isSupportedBy ?funding .
  ?leaderRole   a vivo:LeaderRole ;
                roh:roleOf ?leader ;
                vivo:relatedBy  ?project .
  ?leader       foaf:name ?leaderOrganization .
  ?funding      roh:fundedBy    ?fundingProgramObject .
  ?fundingProgramObject roh:title ?fundingProgram ;
                        roh:promotedBy ?fundingOrganizationObject .
  ?fundingOrganizationObject foaf:name ?fundingOrganization
}
```

Table 9

Results for the SPARQL query at Listing 7

title	leaderOrganization	fundingProgram	fundingOrganization
Broad Data Stack	Eurohelp Consulting SL	Hazitek	Gobierno Vasco
ASIO-HERCULES	GNOSS	FEDER	European Commission
EDI	DeustoTech	H2020	European Union
REACH	Commissariat Al Energie Atomique Et Aux Energies Alternatives	H2020	European Union

6.2. FECYT CVN mapping

As introduced in Section 4, one of the evaluations carried out to test ROH was to model the CVN (“Currículum Vítae Normalizado”, Standardised Curriculum Vitae) provided by the FECYT (“Fundación Española para la Ciencia y la Tecnología”, Spanish Foundation for Science and Technology). The CVN defines a standard format to present researcher’s CV which allows the interoperability among different databases of the Spanish public administration. CVN allows researcher presenting their CV in a unified way in different funding calls from Spanish and regional governments.

At ROH, a tool which takes the XML version of the CVN as an input and generates an RDF file mapping of the CVN to ROH has been developed[30]. The objective of this tool, is to evaluate if ROH is complete enough to model a researcher’s CVN.

Listing 8 depicts the generated RDF document modelled using ROH ontology from the XML representation of the CVN. The application of this mapping process demonstrates that ROH can be used to model the main aspects of the CVN. This validation demonstrates how ROH is comprehensive and exhaustive enough to incorporate academic knowledge modeled according to external non-semantic data models to seamlessly integrate with them.

Listing 8: Representation of a publication from the CVN in Turtle format.

```
data-um:ca69bb23-9650-4000-94b8-b59b4b104de a    vivo:ConferencePaper ;
    roh:seqOfAuthors data-um:9c227529-fdcc-4724-a9ec-18dfdd9e4be1 ;
    roh:title         "INSUFICIENCIA CARDIACA: MANEJO CLINICO"^^xsd:string .

data-um:9c227529-fdcc-4724-a9ec-18dfdd9e4be1 a rdf:Seq ;
    rdf:_1 data-um:79dd74b4-730e-4ae6-a04e-30feea4efc08 .

data:79dd74b4-730e-4ae6-a04e-30feea4efc08 a foaf:Person ;
    roh:correspondingAuthorOf     data-um:ca69bb23-9650-4000-94b8-b59b4b104de ;
    foaf:name                     "DOMINGO ANDRES PASCUAL FIGAL"^^xsd:string ;
    foaf:primaryTopic             true .
```

6.3. SHACL Validation

The ROH ontology defines a set of restrictions, as data types or multiplicity, that must be used to have a CRIS graph aligned and coherent with the semantic model once an organization decides to adopt the semantic model to build its ROH graph.

We have carried out some ROH graph materializations to test the scalability of our system loading a CRIS dataset from different sources (e.g. CERIF or Hercules SGI Data Model) in order to evaluate ROH ontology against them.

To do so, a set of SHACL Shapes have been defined in a way that are aligned with the ontology definitions and have been applied to the datasets before loading them into a graph modelled following ROH. For example, Listing 9 of a domain shape that declares that all the subjects of the property `roh:grants` must be instances of the `roh:FundingAmount` class. Listing 10 defines a shape that declares that the targets of property

[30] https://github.com/deustohercules/CVN

`roh:foundationDate` must have `xsd:dateTime` datatype and Listing 11 shows another shape that declares that the targets of property `roh:hasAccreditation` must be instances of the `roh:Accreditation` class.

Listing 9: Example of a SHACL domain shape.

```
roh:domainroh__grantsShape
      a sh:NodeShape ;
      sh:targetSubjectsOf roh:grants ;
      sh:class roh:FundingAmount.
```

Listing 10: Example of a SHACL data type shape.

```
roh:rangeDatatyperoh__foundationDateShape
      a sh:NodeShape ;
      sh:targetObjectsOf  roh:foundationDate ;
      sh:datatype xsd:dateTime.
```

Listing 11: Example of a SHACL object shape.

```
roh:rangeClassroh__hasAccreditationShape
      a sh:NodeShape ;
      sh:targetObjectsOf  roh:hasAccreditation ;
      sh:class roh:Accreditation.
```

This task has helped us to detect possible issues related to ontological constraints defined in the model when loading large datasets from external sources. Thanks to this method, some problems caused by binding and multiplicity constraints present or not sufficiently defined have been identified. These problems could block the loading of datasets due to some data that are not always present in the sources; or due to absent data, which could allow the materialization of a graph that is not consistent with ROH model.

In practice, different restrictions should be added to manage some other validations, not included in ROH as ontological restrictions, before loading a RIS dataset and could be tuned to be more or less restrictive, depending on the data source. Typically, these are controls related with the quality of the data source (e.g. the starting date of a research project cannot be bigger than the current date plus 6 months).

6.4. Continuous refinement

As stated in Section 4, a continuous refinement process has been carried out during the development of ROH. MLOps or ML Ops is a set of practices that aims to deploy and maintain machine learning models in production reliably and efficiently. The word is a compound of "machine learning" and the continuous development practice of DevOps in the software field. In this work DevOps practices have been applied to ontology development, which we can name *Ontology* Ops or *Onto Ops*. This process is based on a CI/CD workflow implemented through the GitHub Actions[31] tool. Whenever a pull request is issued to integrate new changes into the main branch of the source code, the workflow is executed in order to check the integrity of those changes. Figure 13 depicts the main steps of the workflow described below, which can be shown at Listing 12:

1. **Environment setup (lines 12-35)**: at this phase the runtime environment is prepared, i.e., Java, Python and its dependencies, and Pellet reasoner are installed. At this step the source code of the Pellet reasoner [37][32]

[31] https://github.com/features/actions

[32] https://github.com/stardog-union/pellet

is downloaded and compiled, as the developers do not distribute the compiled library. Pellet reasoner is used for inference and reasoning tasks needed for the execution of the competency questions. Next the *javadoc* documentation is created, which generates an HTML page with the description of each test.

2. **Checkout ROH source code (lines 37-40)**: at this phase the source code is downloaded from the git repository to the workflow execution environment.
3. **Execute competency questions (lines 42-51)**: at this phase, a set of tests have been implemented using the JUnit[33] library. For each competency question, a unit test has been implemented. For each test, the corresponding SPARQL query and the expected result have been defined. Before launching the tests, an RDF model is created through the Jena[34] library, in which the synthetic instance data and ROH are loaded. If the returned result does not match the expected result, an error is raised and the pull request is labeled as not ready to be integrated into the main branch of the repository. Tests are launched using the Maven Surefire plugin[35]. This plugin allows generating different reports from the results of the tests.
4. **Generate documentation (lines 53-88)**: in case the previous step did not arise any errors, the documentation of the ontology is generated and the pull request is labeled as valid to be integrated into the main branch. This phase is formed by different steps. First, the Maven Site plugin[36] is used to generate the project site, which includes the report generated by the Surefire plugin. Next, a custom Python script is executed in order to generate a more friendly HTML page to show the result of the execution of the validation questions. Last, WIDOCO [38] tool is used in order to generate the documentation of ROH.

Listing 12: GitHub Action workflow for continuous refinement of the ontology. Some steps have been omitted to facilitate reading.

```
1  name: Validation questions and Widoco documentation
2  on:
3    push:
4      branches: [ 'main' ]
5    pull_request:
6      branches: [ '*' ]
7
8  jobs:
9    build:
10     runs-on: ubuntu-latest
11     steps:
12     - name: Deploy Java
13       uses: actions/setup-java@v1.4.3
14       with:
15         java-version: 1.8
16
17     - name: Set up Python 3.7
18       uses: actions/setup-python@v2
19       with:
20         python-version: 3.7
21
22     - name: Install dependencies
23       run: |
24           python -m pip install --upgrade pip
25           pip install json2html
26
27     - name: Checkout pellet
28       uses: actions/checkout@v2
```

[33] https://junit.org/
[34] https://jena.apache.org/
[35] https://maven.apache.org/surefire/maven-surefire-plugin/
[36] https://maven.apache.org/plugins/maven-site-plugin/

```
        with:
          repository: stardog-union/pellet
          path: pellet

      - name: Install Pellet reasoner
        working-directory: pellet
        run: mvn install

      - name: Checkout repo
        uses: actions/checkout@v2
        with:
          submodules: true

      - name: Launch tests
        working-directory: /home/runner/work/ROH/ROH/validation-questions
        run: |
          mvn surefire-report:report -Dmodel=https://raw.githubusercontent.com/HerculesCRUE/ROH
              /main/roh/modules/core/roh-core.ttl,
          https://raw.githubusercontent.com/HerculesCRUE/ROH/main/validation-data/rdf
              /roh_data_edma.ttl,
          https://raw.githubusercontent.com/HerculesCRUE/ROH/main/roh/modules/knowledge-area/
              unesco-knowledge-area.rdf
          -DqueryFolder=/home/runner/work/ROH/ROH/validation-questions/sparql-query/

      - name: Generate maven site
        working-directory: /home/runner/work/ROH/ROH/validation-questions
        run:  mvn site

      - name: Create html files
        run: |
          cd validation-questions/src
          python jsonTohtml.py "/home/runner/work/ROH/ROH/validation-questions/sparql-query"
              "/home/runner/work/ROH/ROH/validation-questions/html"

      - name: Compile documentation
        run: |
          cd widoco
          mkdir config
          java -jar ./widoco.jar \
          -ontFile ../roh/modules/core/roh-core.ttl \
          -oops \
          -webVowl \
          -includeAnnotationProperties \
          -outFolder output/roh \
          -rewriteAll \
          -confFile ../roh/modules/core/doc/widoco.config.txt \
          -includeImportedOntologies \
          -uniteSections \
          -excludeIntroduction
          mv output/roh/index-en.html output/roh/index.html
          mv output/roh ../docs/
          cp -r ../mirror ../docs/

      - name: Publish on Github Pages
        uses: crazy-max/ghaction-github-pages@v2.3.0
        with:
          build_dir: docs
          publish_dir: target/site
        env:
          GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
```

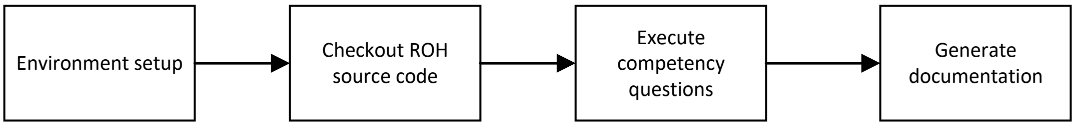

Fig. 13. Main steps of the continuous refinement CI/CD workflow.

This CI/CD workflow allows to ensure the integrity of the ontology when new instances are modelled. On the other hand, the documentation generated[37] allows developers and ontologists to understand the basics of ROH.

7. Conclusions

This paper has presented the work carried out in the Hercules-ASIO project, from which the development of ROH ontology, its main outcome, has been specifically detailed. The paper has showcased as well the methodology applied in the development of the ontology and how its different stages were tackled during the development. The different classes and properties that are part of ROH have been modelled in order to allow describing the complexity of the academic domain in a flexible and exhaustive although still approachable (easily graspable) manner. The described classes allow modelling the main concepts of the research domain such as organizations, metrics, university staff, research projects and so on. The modular approach applied during the design and development of ROH allows third parties to extend the ontology easily, facilitating its adoption in different academic institutions worldwide. Importantly, the paper has also described the process of validation of the flexibility, completeness and usability of the ontology by means of a large range of competency questions, designed to check if the ontology meets the modeling requirements for an exhaustive academic domain knowledge graph. Besides, a mapping of CVs in CVN format to ROH semantic model additionally checks the ontology capability to interoperate with other external systems modeling academic knowledge. Indeed, this has been further validated by importing contents of already existing CRIS systems into knowledge graphs powered by ROH, with the support of a suite of SHACL scripts. These wide range of validations methods have demonstrated the utility of ROH and allowed to detect and fix different issues through a thorough refinement process spanning more than 6 months. Remarkably, ROH has been designed to maximize its usage and adoption. For that, the resulting network of ontologies has been enriched with a wide assortment of ontological restrictions. Furthermore, a very detailed documentation has been provided to support the extensive usage of ROH by third parties.

Future work will continue through the Hercules-ASIO project, i.e., Hercules-EDMA. This follow-up project is addressing the holistic management of Research Objects by providing a more detailed ontology for describing those concepts. Besides, machine learning techniques will be applied to continuously enhance the existing contents of universities' knowledge graphs with contents coming from external non-semantic sources. Some examples of this could be the creation of new knowledge, such as suggesting collaboration of researchers from different institutions based on their publications or projects they have participated in, or the development of Business Intelligence tools to enhance the "Academy Analytics" concept.

Acknowledgements

This work was funded by Hercules-ASIO project (2018/88/OT-AM).

[37] https://herculescrue.github.io/ROH/roh/

References

[1] Hercules - Universidad de Murcia. https://www.um.es/web/hercules/.

[2] V.B. Nguyen, V. Svátek, G. Rabby and O. Corcho, Ontologies Supporting Research-Related Information Foraging Using Knowledge Graphs: Literature Survey and Holistic Model Mapping, in: *International Conference on Knowledge Engineering and Knowledge Management*, Springer, 2020, pp. 88–103.

[3] K. Börner, M. Conlon, J. Corson-Rikert and Y. Ding, VIVO: A semantic approach to scholarly networking and discovery, *Synthesis lectures on the Semantic Web: theory and technology* **7**(1) (2012), 1–178.

[4] B. D'Arcus and F. Giasson, Bibliographic ontology specification, *Madrid: Biblioteca Nacional Española* (2009).

[5] Y. Sure, S. Bloehdorn, P. Haase, J. Hartmann and D. Oberle, The SWRC ontology–semantic web for research communities, in: *Portuguese Conference on Artificial Intelligence*, Springer, 2005, pp. 218–231.

[6] O. Pena, J. Lázaro, A. Almeida, P. Orduna, U. Aguilera and D. López-de-Ipina, Visual Analysis of a Research Group's Performance thanks to Linked Open Data, *Linked Data for Knowledge Discovery* (2014), 59.

[7] B. Jörg, CERIF: The common European research information format model, *Data Science Journal* (2010), 1006280236–1006280236.

[8] K.G. Jeffery and A. Asserson, Supporting the Research Process with a CRIS, *Enabling Interaction and Quality: Beyond the Hanseatic League* (2006), 121–130.

[9] S. Peroni and D. Shotton, The SPAR ontologies, in: *International Semantic Web Conference*, Springer, 2018, pp. 119–136.

[10] S. Peroni and D. Shotton, FaBiO and CiTO: ontologies for describing bibliographic resources and citations, *Journal of Web Semantics* **17** (2012), 33–43.

[11] K. Saur, IFLA Study Group on the functional requirements for bibliographic records. Functional requirements for bibliographic records: final report, UBCIM Publications-New Series, 1998.

[12] D.L. McGuinness, F. Van Harmelen et al., OWL web ontology language overview, *W3C recommendation* **10**(10) (2004), 2004.

[13] M. Fernández-López, Overview of methodologies for building ontologies (1999).

[14] D.J. Schultz et al., IEEE standard for developing software life cycle processes, *IEEE Std* (1997), 1074–1997.

[15] M. Uschold and M. King, *Towards a methodology for building ontologies*, Citeseer, 1995.

[16] G. Schreiber, B. Wielinga, W. Jansweijer et al., The KACTUS view on the 'O'word, in: *IJCAI workshop on basic ontological issues in knowledge sharing*, Vol. 20, Citeseer, 1995, p. 21.

[17] M. Fernández-López, A. Gómez-Pérez and N. Juristo, Methontology: from ontological art towards ontological engineering (1997).

[18] K. Knight, I. Chander, M. Haines, V. Hatzivassiloglou, E. Hovy, M. Iida, S.K. Luk, R. Whitney and K. Yamada, Filling knowledge gaps in a broad-coverage machine translation system, in: *IJCAI'95: Proceedings of the 14th international joint conference on Artificial intelligence*, Vol. 2, ACM, 1995, pp. 1390–1396.

[19] M.C. Suárez-Figueroa, A. Gómez-Pérez and M. Fernández-López, The NeOn methodology for ontology engineering, in: *Ontology engineering in a networked world*, Springer, 2012, pp. 9–34.

[20] K. Beck, *Test-driven development: by example*, Addison-Wesley Professional, 2003.

[21] J. Shore and S. Warden, *The art of agile development*, " O'Reilly Media, Inc.", 2021.

[22] S. Peroni, A simplified agile methodology for ontology development, in: *OWL: Experiences and directions–reasoner evaluation*, Springer, 2016, pp. 55–69.

[23] C.M. Keet and A. Ławrynowicz, Test-driven development of ontologies, in: *European Semantic Web Conference*, Springer, 2016, pp. 642–657.

[24] V. Presutti, E. Daga, A. Gangemi and E. Blomqvist, eXtreme design with content ontology design patterns, in: *Proc. Workshop on Ontology Patterns*, 2009, pp. 83–97.

[25] H. Knublauch and D. Kontokostas, Shapes constraint language (SHACL), *W3C recommendation* **20**(07) (2017).

[26] M.A. Musen, The protégé project: a look back and a look forward, *AI matters* **1**(4) (2015), 4–12.

[27] W.W.W. Consortium et al., SPARQL 1.1 overview (2013).

[28] C. Bizer, T. Heath and T. Berners-Lee, Linked data: The story so far, in: *Semantic services, interoperability and web applications: emerging concepts*, IGI global, 2011, pp. 205–227.

[29] M.D. Wilkinson, M. Dumontier, I.J. Aalbersberg, G. Appleton, M. Axton, A. Baak, N. Blomberg, J.-W. Boiten, L.B. da Silva Santos, P.E. Bourne et al., The FAIR Guiding Principles for scientific data management and stewardship, *Scientific data* **3**(1) (2016), 1–9.

[30] F. Coulmas, *The Blackwell encyclopedia of writing systems*, Wiley-Blackwell, 1999.

[31] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak and Z. Ives, Dbpedia: A nucleus for a web of open data, in: *The semantic web*, Springer, 2007, pp. 722–735.

[32] D. Vrandečić and M. Krötzsch, Wikidata: a free collaborative knowledgebase, *Communications of the ACM* **57**(10) (2014), 78–85.

[33] J.A. Fodor, *The modularity of mind*, MIT Press, 1983.

[34] J.A.P. Sánchez, F.J.M. Méndez, R.L. Carreño and J.V.R. Muñoz, UNESKOS. publicación como Linked Open Data de la Nomenclatura Internacional de Ciencia y Tecnología y del Tesauro UNESCO, in: *II Congreso ISKO España y Portugal/XII Congreso ISKO España*, 2013, pp. 1022–1044.

[35] M. Graves, A. Constabaris and D. Brickley, Foaf: Connecting people on the semantic web, *Cataloging & classification quarterly* **43**(3–4) (2007), 191–202.

[36] J. Arenas-Guerrero, D. Chaves-Fraga, J. Toledo, M.S. Pérez and O. Corcho, Morph-KGC: Scalable Knowledge Graph Materialization with Mapping Partitions, *Semantic Web* (2022). http://www.semantic-web-journal.net/system/files/swj3135.pdf.

[37] E. Sirin, B. Parsia, B.C. Grau, A. Kalyanpur and Y. Katz, Pellet: A practical owl-dl reasoner, *Journal of Web Semantics* **5**(2) (2007), 51–53.
[38] D. Garijo, WIDOCO: a wizard for documenting ontologies, in: *International Semantic Web Conference*, Springer, 2017, pp. 94–102.
[39] D.C.M. Initiative et al., Dublin core metadata element set, version 1.1 (2012).