Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 146 (2019) 29–38

Procedia Computer Science

www.elsevier.com/locate/procedia

14th International Conference on Current Research Information Systems, CRIS2018

The Research Core Dataset (KDSF) in the Linked Data context

Tatiana Walther*[a], Christian Hauschke[a], Anna Kasprzik[a]

[a]*German National Library of Science and Technology (TIB), Welfengarten 1B, Hannover 30167, Germany*

Abstract

This paper describes our efforts to implement the Research Core Dataset ("Kerndatensatz Forschung"; KDSF) as an ontology in VIVO. KDSF is used in VIVO to record the required metadata on incoming data and to produce reports as an output. While both processes need an elaborate adaptation of the KDSF specification, this paper focusses on the adaptation of the KDSF basic data model for recording data in VIVO. In this context, the VIVO and KDSF ontologies were compared with respect to domain, syntax, structure, and granularity in order to identify correspondences and mismatches. To produce an alignment, different matching approaches have been applied. Furthermore, we made necessary modifications and extensions on KDSF classes and properties.

Peer-review under responsibility of the scientific committee of the 14th International Conference on Current Research Information Systems, CRIS2018.

Keywords: research information management; Linked Data; VIVO; Research Core Dataset; ontology matching

1. Introduction

In Germany, publicly funded organizations are obliged to provide extensive reports to various German governmental and European funding agencies, federal ministries and offices of statistics [1]. A harmonization of the processes and workflows to record and aggregate information for various reports should save time and work otherwise spent on determining key figures and lists for every report anew.

KDSF [2] is a specification recommended by the German Council of Science and Humanities ("Wissenschaftsrat") for a uniform, standardized collection, and provision of data on research activities. It is meant to support research institutions in their reporting efforts while serving different reporting purposes [1].

* Corresponding author. Tel.:+49 51176217895.
E-mail address: tatiana.walther@tib.eu

1877-0509
Peer-review under responsibility of the scientific committee of the 14th International Conference on Current Research Information Systems, CRIS2018.
10.1016/j.procs.2019.01.074

It is currently being implemented across German academic and research-related institutions. Since the integration of the elements of KDSF into a research information system (CRIS) is regarded as a viable option, several efforts to implement the standard in different types of CRIS such as Pure, Converis or various in-house developments were undertaken.

KDSF is – among other standards – also available in the Web Ontology Language (OWL). Hence, classes and properties that are already defined in this ontology comply with the Linked Data principles at least formally, and the OWL version of KDSF is therefore considered suitable to be used within a Linked Data application. The KDSF specification mentions VIVO as a potential software for its technical implementation [3].

VIVO is a CRIS based on Linked Data technologies. It aggregates research related information, interlinks it semantically and represents it in accordance with web standards. As an open source software, VIVO is flexible and can be easily adapted to different needs. As a Linked Data application, VIVO is built upon on a knowledge base that models information using established Linked Data vocabularies such as the Friend of a Friend Ontology (FOAF) and others [4]. Collecting data in VIVO using Linked Data technologies has several implications:

- In VIVO, information is not merely recorded and used as character strings, but rather as classified, semantically linked, URI-named and reusable items. A one-time collection is sufficient to link one entity with another in a variety of ways and to be able to use it again and again.
- Inference rules are built into the VIVO data model. They are interpreted by the inference engine and allow implicit knowledge to be derived.
- Classes and properties already present in VIVO can be re-used to reflect facts in a variety of contexts.

With the implementation of KDSF in VIVO we expect to be able to use research information in different contexts. On the one hand, information about people, organizations, projects, and publications in form of profiles can be used for representation on the web. On the other hand, the integration of a VIVO compatible ontology developed from the base data model ("Basisdaten") of KDSF would support data acquisition and central reporting within a single application.

Thus, the overall goal of our work is the integration of KDSF into VIVO to enable a KDSF-compliant reporting by means of structured, semantically supported and user-friendly data collection and report generation. Since the integration of external ontologies within VIVO requires an alignment of these ontologies with the VIVO data model, KDSF has consequently to be matched to the VIVO ontology.

To this end, this paper is organized as follows. Section 1 presents the background information and motivation of our work. Section 2 provides a short insight into the basics of ontology matching and introduces the related work. Section 3 is dedicated to the various steps of the process of implementation which begins with the general analysis and comparison of VIVO and KDSF ontologies. This is followed by the matching approaches as well as the transformation that we have applied to KDSF. Finally, section 4 concludes the paper, presenting the results and directions for future work.

2. Background and related work

The following section introduces shortly the theoretical background and related work considering alignment of VIVO and KDSF to other vocabularies.

The alignment between VIVO and KDSF ontologies introduced in this paper is based on pre-existing approaches of ontology matching. The term "ontology" used in the paper, is limited to a formal ontology expressed in OWL and consisting of classes, individuals, relations (subsumption, disjointness, and instantiation), data types, and data values

Ontology matching is subject of various publications throughout the last two decades. In the majority of cases this research has been confined to computational matching [5]. However, various experts point out that computational matching can hardly yield satisfactory results without human assistance, intervention or expertise [6], [7], [8]. Even the matching of two relatively simple and small ontologies can push a matching device to its limits [8]. Accordingly, a test with a computational matching device (LogMap) proved to be insufficient.

Thus, we decided to use an approach of intellectual matchmaking. The generalized definition of "ontology matching" by Euzenat and Shvaiko as "*a process of finding relationships or correspondences between entities of different ontologies*" [9] is most applicable to our work. According to this definition, the process of matching results in an alignment consisting of correspondences between two or more ontologies. Each correspondence is an assertion declaring a relation between two entities (classes, properties, or individuals) [9].

The complexity of a matching process as well as matching scenarios can vary depending on the purpose and the application of ontologies to be matched. If the concepts of an ontology are used to support annotating and cataloguing of information, a simple set of correspondences, would suffice to provide an alignment [10]. If an ontology is applied to conceptualize data in a triple store, more complex matching and mediation patterns are required [10].

With regard to VIVO and KDSF, similar work was done to provide alignment of both of the vocabularies to the Common European Research Information Format (CERIF). To ensure the international interoperability and applicability of KDSF in common CRIS, it has been partially mapped to CERIF [1]. For the sake of interoperability, efforts to map VIVO to CERIF were undertaken by Leczano et. al. In their work the authors compare these two data models and introduce the results of the mapping as well as various methods of mappings such as SPARQL query rewriting, linked data technologies and OWL axioms. The work of Leczano et al. focuses on data integrating and translating data from CERIF and VIVO systems "*to answer particular queries without necessarily having to perform a wholesale conversion of data from one format to the other*" [11], whereas the work presented in this paper aims at providing of ontology alignment for data recording and subsequent report generation within a VIVO application.

A tool for translation of data from VIVO into CERIF and visa-versa, based on the XSLT and XQuery processor Saxon [12] has been developed at the Alcalá University (Spain) [13].

3. Implementation

In this section we describe the steps we have made to adapt KDSF for using it in VIVO. We provide an example of matching for a part of KDSF describing publications as well as some transformation examples.

The adaption process we have applied consisted of the following steps:

- General analysis of the ontologies
- Matching
  - Computational matching
  - Equivalences and subsumptions
- Transformation and extension
  - Classes to instances
  - Datatype Properties to object properties
  - Omission of categories
  - Additional axioms

3.1. KDSF and VIVO –general analysis

The first step of the implementation was the general analysis of KDSF and VIVO ontologies according to the following criteria: natural language, syntax, domain of interest, structure, and granularity. By means of the analysis mismatches and correspondences could be identified.

The main **language** of the VIVO ontology terms and their annotations is English, although the ontology also contains some instances in other languages. KDSF terms are mainly provided in German. There are also English terms such as for example "peer-reviewed", "review" in the vocabulary, which are commonly used in Germany as well.

Regarding the **syntax** it has to be noted, that the VIVO ontology is encoded in RDF, RDFS and OWL. KDSF is available in the OWL/RDFS format and as an XML Scheme which is based on the CERIF standard [14].

According to the definition of the **domain of interest**, the VIVO ontology serves the representation of scholarship with no limitations on a specific discipline or domain of knowledge [15]. Furthermore, the ontology is meant to support "*the identification, evaluation, and impact assessment of individual people and groups of people, as well as identification and reuse of the works of the people*" [15]. A set of elements delivered by the Integrated

Semantic Framework ontology modules for the VIVO-ISF ontology enables the comprehensive presentation of researchers and their work [16]. For this purpose, the VIVO ontology encompasses the following categories: *people*, *activities*, *events*, *organizations*, *equipment*, *locations*, and *research* including e.g. *publications*, *grants*, and *projects*. This set of categories is provided with every default VIVO installation.

The KDSF specification states that the standard serves the exchange of information on research activities, research personnel and their products. This information is to be treated in accordance with the principle of data compactness. The focus is on versatile core data for general institutional reporting and not on assessment of scientific performance. [17]. The vocabulary should be applicable to as many disciplines as possible and is therefore not subject to any restrictions regarding a specific domain of knowledge. KDSF delivers a set of categories for data recording and aggregating. The set includes seven superior categories for recording: persons, doctoral programs, third-party funded projects, publications, patents, spin-offs, and research infrastructure. There is an aggregation level with the following core and peripheral elements: employees, scientific promotion, third-party funds and finances, patents, spin-offs, publications, and research infrastructure [17]. In our work, we have only regarded the „Basismodell“ – the basic model for recording information.

The **structure** of both ontologies varies, too. VIVO-ISF integrates numerous classes and properties from various established ontologies. The list of ontologies includes i.a. VIVO (http://vivoweb.org/ontology/core), Basic Formal Ontology (BFO) (http://www.ifomis.org/bfo), Bibliographic Ontology (BIBO) (http://bibliographic-ontology.org/), Event Ontology (http://motools.sourceforge.net/event/event.html), Friend of a Friend (FOAF) (http://www.foaf-project.org/), Geopolitical.owl from the U.N. Food and Agriculture Organization, FRBR-Aligned Bibliographic Ontology (FaBio), (http://purl.org/spar/fabio/), VCard (http://www.w3.org/TR/vcard-rdf/) and other vocabularies [18]. KDSF refers to a few external non-OWL vocabularies, e.g., the „Access Rights Vocabulary” of the Confederation of Open Access Repositories (COAR). However, KDSF does not reuse the URIs for the access rights from the vocabulary.

Table 1: Comparison of the VIVO and KDSF Ontology entity metrics

	VIVO Ontology	KDSF
Classes	414	123
Object properties	198	36
Datatype properties	217	53
Instances	14	0

Table 1 illustrates the results of analysis made using Protégé on classes, object properties, datatype properties, and individuals included in the VIVO ontology and KDSF. According to the analysis data, VIVO-ISF counts 414 classes, 198 object properties, 217 datatype properties, and 14 individuals, whereas KDSF (version dated October 25, 2015) enumerates 123 classes, 36 object properties, 53 datatype properties and no individuals.

In terms of axioms, VIVO ontology includes, as shown in Table 2, 623 subsumptions, 40 equivalences, and one disjunctness axiom on classes. Among the object property axioms, there are 30 subsumptions, 57 inverse relations, two functional relations, two transitive, and one symmetric relation. There are seven object properties with domain intersections and nine object properties for which restriction axioms apply. 35 subsumptions apply to datatype properties. Nine of the properties have domain intersections.

KDSF contains 91 subsumptions and one disjunctness axiom on classes. Besides, there are one functional object and eleven datatype properties. The domain intersection axioms apply for four object properties. Two object properties have an intersection of classes as a range. Three datatype properties use the enumerated datatype (rdf:List) as a range.

Table 2: Comparison of the VIVO and KDSF Ontology Axiom Metrics

	VIVO Ontology	KDSF
Class hierachies	623	96

Class equivalences	40	1
Class disjunctness	1	1
Property hierachies	65	0
Functional relations	2	12
Transitive relations	2	0
Symmetric relations	1	0
Inverse relations	57	0
Domain intersections	16	4
Range intersections	0	2
Restrictions	9	0
Enumerated datatypes	0	3

The comparison of the related classes to publications provided by the ontologies, shown in Fig. 1, and Fig. 2(a) and (b) reveals that the VIVO ontology has a much higher **granularity**, as it contains more classes and subclasses for various types of documents. Fig. 1 states that the VIVO categories related to publications do not only have a scientific character but also refer to legal writing, arts, and humanities.

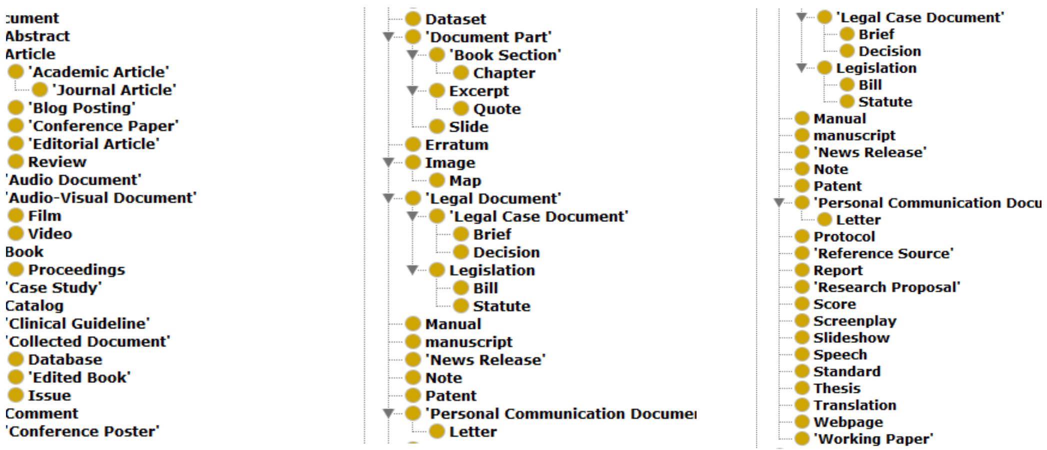

Fig. 1: classes of documents in VIVO

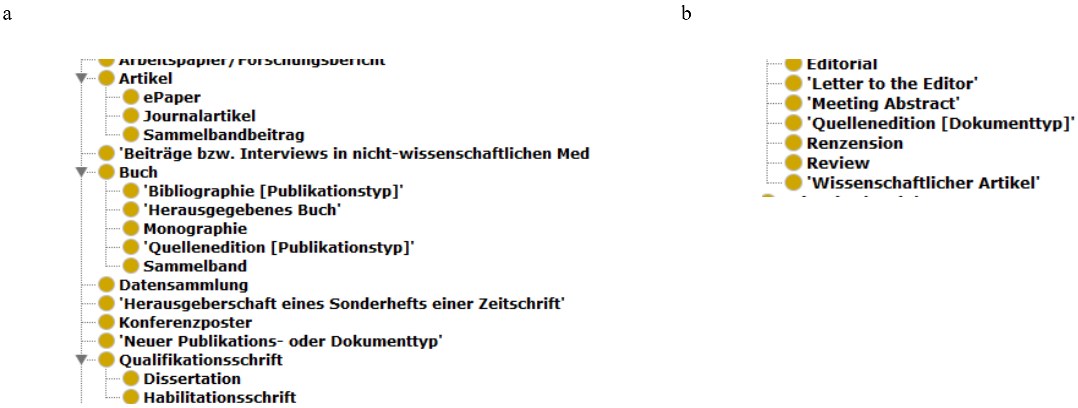

Fig. 2(a): types of publications in KDSF; (b): types of documents in KDSF

As depicted in Fig. 2(a) and (b), KDSF classifies the terms into publication types and document types. The first one refers to the forms of publication, while the latter one implies the content-related subcategorization of articles.

The analysis shows that the VIVO and KDSF ontologies have a similar domain and cover similar categories, but serve different purposes. The ontologies are available in two different natural languages. In terms of structure, VIVO Ontology uses many external terms and has a significantly higher complexity. With regard to the granularity, VIVO provides more categories for the publication sub-domain and has more hierarchies in it. KDSF uses a different structure for categorization of publications in terms of content and manifestation. The granularity and coverage vary also for the other categories of the ontologies.

3.2. Matching approaches

Due to the disparities listed above a matching device would not deliver satisfying results. This assumption was proven while matching by means of the LogMap matching system [19]. Since the alignment did not have to be created during runtime, further selection and testing of a capable matching tool was not considered necessary. Pragmatic goals and the timeline of the project were also the reason why we decided in favor of an intellectual matching.

3.2.1. Matching with LogMap

LogMap is a modular ontology matching device developed at the Oxford University. The tool delivers mappings of entities which previously have been parsed on their lexical and structural similarity. LogMap also applies an inference check and involves the user into the process of matching [20].

The output included, as shown in Table 3, mappings on only four classes. The only type of correspondence stated was equivalence. It is obvious that only the classes with the same labels could be matched. Both in VIVO and in KDSF there is a concept named “Review”. However, there is no equivalence between kdsf:Review and vivo:Review, as the meaning of the concepts differs, which can be observed exemplarily in the definitions of a review in vivo:Review and kdsf:Review.

Table 3: Results of matching with LogMap

KDSF	Correspondence	VIVO
kdsf:Patent	≡	bibo:Patent
kdsf:Review	≡	vivo:Review

kdsf:Person	≡	foaf:Person
kdsf:Software	≡	obo:ERO_0000071 (software)

Equivalence between other entities with similar meaning such as for example vivo:ConferencePoster and kdsf:Konferenzposter (English: “conference poster”) was not detected. Subsumptions were not stated either. The outputs can hardly be considered satisfactory.

3.2.2. Manual Matching

In process of matching we applied two approaches: 1. full integration of KDSF into the VIVO ontology; 2. partial integration of KDSF into the VIVO ontology.

The initial approach of matching was to identify correspondences between KDSF categories and those of the vocabularies provided in VIVO and apply them, trying to produce a complete matching. The correspondences were based on the human readable description of the elements, and their intentions. For matching, we applied equivalence and subsumption relations.

Fig. 3 illustrates the equivalences and subsumptions stated between the classes of the VIVO ontology and KDSF. As shown in Fig. 3, we have omitted the KDSF intermediate categories kdsf:Publikationstyp (English: “publication type”) and kdsf:Dokumenttyp (English: “document type”), since the categorization of publications in VIVO is already done by assigning categories to the types of documents without differentiation of form and content. Classes with a partial correspondence were inserted as sub-classes.

Fig. 3: Publication-related classes in the first version of the kdsf-vivo-alignment.owl

According to the OWL Web Ontology Language Reference, equivalence assertions state that two classes declared as equivalent have the same class extension. Furthermore, the class extensions are involved in the domain and range assertions of properties [21]. This would cause a lot of unintended inferences and affect the application in terms of reasoning and performance.

The same class extension of the equivalent classes means, that the equivalent classes have also the same set on individuals. Interpreted by the VIVO application, one and the same set on entities would be displayed in the interface more than once, each time assigned to another class.

Due to the same class extension of the equivalent classes, the domain and range assertions for the properties of the VIVO ontology would apply for the properties of the KDSF and vice versa, although it is not always intentional.

One more negative effect is the prolonged data import. While importing data into VIVO, the reasoner has to check every assertion and draw conclusions from them. When we load a large scale data modeled according to an ontology alignment, where all the entities of the KDSF are matched per equvalences or subsumptions to the entities of the VIVO ontology, the inferencing would take much longer to complete. As a result, the loading process would also take considerably longer.

For the above mentioned reasons, in contrast to our initial approach, the second approach was to implement only KDSF concepts which do not have equivalent counterparts in the VIVO ontology. These entities were integrated as (strict) subsumptions.

Relations were matched in a similar way. We adopted only properties that could not been represented by VIVO properties.

For example, kdsf:hatVerlag (“has Publisher”) has an equivalent correspondence in VIVO with vivo:publisher. Although the range definition of the two relations differs – the range of kdsf:hatVerlag is limited to publishers, whereas the range of vivo:publisher covers foaf:Organization allowing all kinds of organizations to act as a publisher – the meaning of statements would not be changed if using vivo:publisher. The relation kdsf:hatVerlag can be therefore represented by its VIVO counterpart and does not have to be adopted explicitly.

The results of the computational matching confirmed that human estimation and background knowledge in matching is necessary. The first approach of intellectual matching included setting both equivalence and subsumption correspondences. The second approach was limited to integrating KDSF entities by using subsumptions.

3.3. Transformation and Extension

Although KDSF is available in OWL, its classes and properties are not necessarily designed for use within Linked Data applications. To make KDSF more applicable in VIVO we had to modify several entities and to extend the ontology by named individuals and additional axioms. The types of modifications can be divided into: classes to instances, datatype properties to object properties, omission of categories, and additional axioms.

First there are modifications of the type classes-to-instances. KDSF contains several classes the specializations of which are rather instances than classes as they represent concrete objects and cannot be specified further. As examples, kdsf:DFG (English: “German Research Foundation”) and, kdsf:EU (“European Union”), which are modeled as classes in KDSF but represent concrete organizations, can be listed here. According to the OWL Web Ontology Language Reference, classes are used to group objects with similar properties [21]. The above mentioned classes were therefore transformed into instances.

Another modification would be from datatype properties to object properties. Some properties in KDSF that are currently designed as datatype properties would be more valuable if they could be used for interlinking entities instead of attributing them with strings. These include, for instance, kdsf:Zugriffsrechte (English: “access rights”), kdsf:hatPromotionsberechtigungAus (English: “has doctoral authorization from”), kdsf:KoordinatorEinrichtung (English: “coordinating organization”), etc. These properties were changed into object properties so that consequently, it was necessary to adapt their ranges as well. To do this, we selected suitable VIVO classes or created new ones and filled them with instances if necessary. The changed properties, new classes and instances have been defined in a new vocabulary: KDSF-VIVO-Extension. In the case of kdsf:Zugriffsrechte, we had to create a new class kdsf-vivo:Zugriffsrechte and fill it with named individuals: *open access*, *restricted access*, *embargoed access*, and *metadata only access*. Since every entity in VIVO must be referenced with a URI, we have reused the URIs of the concepts from the Controlled Vocabulary for Access Rights (COAR) [22].

Furthermore, we had to omit categories. KDSF provides a number of intermediate classes and related properties such as, e.g., kdsf:Professorenbezeichnung (“professor’s designation”), kdsf:hatBezeichnung (English: “has designation”), kdsf:SonstigeProfessoren (English: “other professors”) etc. which we considered unnecessary. The different types of professorship including full professors and their varieties as well as other non-ordinary professors can be represented by subclasses of kdsf:Professoren (English: “professors”). These entities were therefore omitted. Other candidates for omission are the categories mentioned in Chapter 3.2.2 that can be represented by equivalent VIVO concepts.

On the other hand, some new and additional axioms proved necessary. KDSF does not contain any inverse relations. Inverse relationships have the advantage of defining a relationship in both directions: persons contribute to projects – projects have contributors [21]. In VIVO inverse assertions that are interpreted by the reasoner and displayed in the user interface reduce efforts on data entry. For this reason, we regard inverse assertions as useful and worth to be added to the KDSF-VIVO-Extension.

4. Conclusion and future plans

The processes described above have resulted so far in a first draft of a KDSF-VIVO-Alignment and a KDSF-VIVO-Extension [23]. Both drafts have been successfully tested in VIVO 1.9.

The VIVO ontology and KDSF specification have overlapping parts but serve different tasks. The ontologies are also different in terms of terminology, structure, and intension of concepts. When applying a fully automated ontology matching these discrepancies would cause insufficient results.

With regards to our approaches, the full integration of KDSF into VIVO by assigning both equivalences and subsumptions proved to be unpractical, since it has negative effects on reasoning and performance. For this reason, we had to revise our approaches and apply other methods than initially planned. We currently work on the partial integration of KDSF concepts that do not overlap with other concepts in VIVO. We also recognized that an alignment alone would not suffice to make KDSF applicable in VIVO. A number of modifications on the concepts and structure of KDSF were and are still necessary to enable an optimal use of KDSF within VIVO. We do not exclude that the alignment and extension we have created could be designed in a different way. It is well-known that there are many ways to model a domain of interest and each model has its pros and cons. These advantages and disadvantages become especially visible through reasoning in a Linked Data application like VIVO.

Currently, the next version of the KDSF-VIVO-Alignment and KDSF-VIVO-Extension is being developed. We aim to work on this version in collaboration with other German organizations. We put our work up for discussion and are open to suggestions and criticism. The outcomes of this work are going to be applied in a reporting tool based on VIVO software for the German National Library of Science and Technology (TIB) and will be made available to other interested users. Since the VIVO and the KDSF ontology are constantly evolving, we will put a lot of effort into keeping the alignment up-to-date. We will also continue to evaluate semi-automated methods and tools in order to master that challenge.

References

[1] Wissenschaftsrat, "Empfehlungen zur Spezifikation des Kerndatensatz Forschung," 22 January 2016. [Online]. Available: https://www.wissenschaftsrat.de/download/archiv/5066-16.pdf. [Accessed 11 June 2018].

[2] Deutsches Zentrum für Hochschul- und Wissenschaftsforschung GmbH (DZHW), "Kerndatensatz Forschung," 2018. [Online]. Available: https://www.kerndatensatz-forschung.de/. [Accessed 19 October 2018].

[3] Institut für Forschungsinformation und Qualitätssicherung (iFQ); Fraunhofer-Institut für Angewandte Informationstechnik; Geschäftsstelle des Wissenschaftsrates, "Kerndatensatz Forschung : Dokumentation der Ergebnisse ; 13 Oktober 2015," [Online]. Available: http://www.kerndatensatz-forschung.de/version1/Ergebnisbericht_Projekt_KDSF.pdf.

[4] Duraspace, "VIVO," [Online]. Available: http://vivoweb.org/info/about-vivo. [Accessed 28 Februar 2018].

[5] L. Otero-Cerdeira, F. J. Rodríguez-Martínez and A. Gómez-Rodríguez, "Ontology matching: A literature review," *Expert Systems with Applications,* no. 42, pp. 949-971, 2015.

[6] M. Uschold and Gruninger, Michael, "Ontologies and Semantics for Seamless Connectivity," *SIGMOD Rec.,* vol. 33, no. 4, pp. 58-64, 2004.

[7] S. M. Falconer and M.-A. Storey, "A cognitive support framework for ontology mapping," in *The Semantic Web. Lecture Notes in Computer Science, vol. 4825*, Heidelberg, Springer, 2007, pp. 114-127.

[8] F. Scharffe, *Correspondence Patterns Representation,* Innsbruck, 2009.

[9] J. Euzenat and P. Shvaiko, Ontology Matching : Second Edition, Berlin, u.a.: Springer, 2013.

[10] F. Scharffe, O. Zamazal and D. Fensel, "Ontology alignment design patterns," *Knowledge and Information Systems,* vol. 40, no. 1, pp. 1-28, July 2014.

[11] L. Lezcano, B. Jörg, B. Lowe and J. Corson-Rikert, "Promoting International Interoperability of Research Information Systems: VIVO and CERIF," *JUCS,* vol. 19, no. 12, pp. 1854-1867, 2013.

[12] M. Kay, "Saxon XSLT and XQuery processor," 27 September 2018. [Online]. Available: http://saxon.sourceforge.net/. [Accessed 19 October 2018].

[13] A. Nogales and J. Mayor, "cerif2vivo," 11 December 2013. [Online]. Available: https://github.com/ieru/cerif2vivo. [Accessed 19 October 2018].

[14] Kerndatensatz Forschung, "XML-Schema," n.d.. [Online]. Available: http://www.kerndatensatz-forschung.de/version1/technisches_datenmodell/index-xml.html. [Accessed 8 June 2018].

[15] M. Conlon, "VIVO Ontology Domain Definition," 4 December 2017. [Online]. Available: https://github.com/openrif/vivo-isf-ontology/wiki/VIVO-Ontology-Domain-Definition. [Accessed 2 June 2018].

[16] G. Triggs, "Ontology Reference," 16 October 2016. [Online]. Available: https://wiki.duraspace.org/display/VIVODOC19x/Ontology+Reference. [Accessed 3 June 2018].

[17] Institut für Forschungsinformation und Qualitätssicherung (iFQ); Fraunhofer-Institut für Angewandte Informationstechnik; Geschäftsstelle des Wissenschaftsrates, "Spezifikation des Kerndatensatz Forschung : Version 1.0 ; 24. September 2015," [Online]. Available: https://kerndatensatz-forschung.de/version1/Spezifikation_KDSF_v1.pdf.

[18] J. Corson-Rikert, "Source Ontologies Für VIVO," n. d.. [Online]. Available: https://wiki.duraspace.org/display/VIVODOC19x/Source+ontologies+for+VIVO. [Accessed 03 06 2018].

[19] Information System Group, "LogMap Ontology Matching Tool," [Online]. Available: http://www.cs.ox.ac.uk/isg/tools/LogMap/. [Accessed 19 October 2018].

[20] E. Jiménez-Ruiz and B. Cuenca Grau, "LogMap: Logic-based and Scalable Ontology Matching," n.d.. [Online]. Available: https://www.cs.ox.ac.uk/isg/projects/LogMap/papers/paper_ISWC2011.pdf. [Accessed 7 June 2018].

[21] W3C, "OWL Web Ontology Language. Reference," 10 February 2004. [Online]. Available: https://www.w3.org/TR/owl-ref/. [Accessed 5 June 2018].

[22] COAR Controlled Vocabularies Interest Group, "Controlled Vocabulary for Access Rights (Draft V1)," 20 April 2017. [Online]. Available: http://vocabularies.coar-repositories.org/documentation/access_rights/. [Accessed 19 October 2018].

[23] T. Walther, A. Kasprzik and C. Hauschke, "VIVO-KDSF-Integration," 8 May 2017. [Online]. Available: https://github.com/VIVO-DE/VIVO-KDSF-Integration. [Accessed 19 October 2018].