*Bioinformatics*

doi.10.1093/bioinformatics/xxxxxx Advance Access Publication Date: Day Month Year Manuscript Category

Databases, Ontologies & Text mining

**Grounding and retrieving biomedical database content with formal ontologies**

**Filipe Santana 1,∗, Fred Freitas 1 and Stefan Schulz 2**

1 Centro de Informática (CIn), Universidade Federal de Pernambuco (UFPE), Recife, 50740-560, Brazil, and

2 Institut für Medizinische Informatik, Statistik und Dokumentation, Medzinische Universität Graz, 8036, Austria

∗ To whom correspondence should be addressed.

Associate Editor: XXXXXXX

Received on XXXXX; revised on XXXXX; accepted on XXXXX

**Abstract**

**Motivation:** Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text. **Results:** Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text Text

**Availability:** <http://www.cin.ufpe.br/>integrativo

**Contact:** [fss3@cin.ufpe.br](mailto:fss3@cin.ufpe.br)

**1 Introduction**

Hypothesis generation in biomedical research depends on gathering related data from scientific publications and databases, such as UniProt (UniProt Consortium, 2014) or Ensembl (Cunningham *et al.*, 2014). The exploration of their content is performed manually or (partly) supported by retrieval tools, e.g. by STRING (Szklarczyk *et al.*, 2014), or BLAST (Altschul *et al.*, 1990). Tools like these are not able to describe the content retrieved for the user. Thus, the interpretation of these results may be biased by the researchers’ capabilities, by the sheer size and heterogeneity of the sources, or by the technical limitations of querying tools (Triplet and Butler, 2011).

*Ontology-based data access (*OBDA) (Poggi *et al.*, 2008) -related applications enables the retrieval of biomedical data with the support of ontologies, this last used only as a query vocabulary. Examples are SPARQL (Harris and Seaborne, 2013) endpoints, frequently delivered as the result of a simplified integration solution. Other tools rely on machine learning to interpret databases according to an ontological background (Fanizzi *et al.*, 2008; Lehmann, 2009).

Such approaches are limited by the need for user intervention, e.g.the manual interpretation of the final retrieved content, and the ontology popu- lation with data. Furthermore, these approaches model all data entities as individuals (ABox elements), which conflicts with formal ontological principles (Smith *et al.*, 2007) and entails high processing cost (Hustadt *et al.*, 2005).

The current situation is characterized by a continuous evolution of high-quality structured knowledge resources, whereas little progress can be seen regarding their usage, interoperability and ontological background. The lack of standardized interpretation from retrieved data among different actors may lead to undesired outcomes (misleading results, informal evaluation of data evaluation, among others) not only for biomedical research, but to any unsupervised retrieval tool in any application domain.

We here advocate a seamless integration of database and ontology content, underpinned by formal-ontological principles. We hypothesise and demonstrate that this enables more powerful queries, supported by reasoning procedures. We refer to this as "ontological grounding". The underlying ontological assumptions are based on previous work on tabular biological content Santana *et al.* (2011) and clinical documentation Martinez-Costa *et al.* (2015).

Ontological grounding means to identify ontology-level content in bio- logical databases, and to axiomatize it under an upper level ontology. This approach aims at delivering a homogeneous ontological representation of data, linked to (parts of) existing biomedical ontologies. Ontological grounding can be applied to support database curation with automated reasoning. It may enable the validation of database content in the ontological level.

When supported by reasoning, the validation process may become simple and more powerful, with a more expressive query language (DL Query). DL Query uses the semantics and reasoning procedures of Description Logics (DL) (REF) which allow querying a broad range of biological databases at the same time. Thus, decreasing the costs of database integration, if it were done in a static database structure.

To this end, we will (i) analyse a subset of biomedical ontologies and

databases; (ii) propose an ontology-based framework that makes database

© The Author 2016. Published by Oxford University Press. All rights reserved. For permissions, please e-mail: [journals.permissions@oup.com](mailto:journals.permissions@oup.com) **1**

“main” — 2016/1/3 — page 1 — #1

**2** *F. Santana et al.*

content explicit by distinguishing it from the entities denoted thereby; (iii) relate this solution to current workflows in which life science data and kno- wledge are acquired and processed; (iv) implement an example ontology from real data as an exemplar for data integration across ontologies and databases; (v) validate this example by demonstrating how querying beco- mes simpler and more user friendly in terms of user interface and query results; (vi) discuss limitations and prospects for scaling this approach up to allow processing of large quantities of data.

The biological use case is addressed by queries created as Competency Questions (CQs) (Gruninger and Fox, 1994) in DL. They range from how data related to a specific metabolism overlap in biological databases for a set of selected model organisms, to the phenotypes from dysfunctional metabolism. Our examples make use of UniProt, NCBI Taxonomy (REF), Ensembl, GO (The Gene Ontology Consortium, 2014), ChEBI (Hastings *et al.*, 2013) and PR (Natale *et al.*,

2014), organized under the upper domain ontology BioTopLite (BTL2) (Schulz and Boeker, 2013).

**2 Background**

In the following, we will introduce what are formal ontologies, the interpretation of databases grounded with ontologies, subtleties regarding the language used, and some application background. These topics highlights some (i) historical background, (ii) major use cases, and (iii) a specific example on the metabolism of a biomolecule across species.

2.1 Formal ontologies

Formal ontologies are mainly characterised by classes, identified by codes and human-readable labels. Together with a limited set of binary relations, classes form axioms that state what is universally true for all members of a class or a class-like expression. For instance, all nucleic acid molecules contain nucleotides, or that all adenine molecules are nucleotide molecules. Ontology axioms are typically expressed in a logic-based language, e.g. some kind of DL.

The construction of (formal) ontologies should obey principled criteria (Spear, 2006) and good practice guidelines (Schulz *et al.*, 2012). Important principles are (i) naming conventions that guide unambiguous labelling of classes and relations (Schober *et al.*, 2009); (ii) mutually disjoint upper-level classes like *Process* or *Quality* as a fundamental ordering framework; and, (iii) a generally small set of canonical relations (Smith *et al.*, 2005), such as **has participant**’ or **has part**’. Both top-level classes and canonic relations are usually supplied by top-level ontologies, such as BFO (Spear, 2006), RO (Smith *et al.*, 2005), or BTL2 (Schulz and Boeker, 2013).

We distinguish ontology content proper from both the notions of *kno- wledge* (Schulz and Jansen, 2013) and *data*. Regarding knowledge (in a broader sense, cf. (Rector, 2008)), assertions about what is frequently associated or only true by default cannot be straightforwardly expressed by formal ontologies. Regarding data, interpretation is often blurred by use- mention confusion, i.e. mixing data items with the things they denote. In databases, the interpretation of data elements is determined by the usecases embedded in the underlying schema. The distinction between what is a data element and what is its referent is implicit. Formal ontologies should enforce this distinction: data items are instances of information content entities, whereas the things data items denote are manifold: individuals, classes, or even nothing.

2.2 Interpreting databases with ontologies

Databases and less-principled domain ontologies leave the real nature of the entities as well as and the circumstances of denotation underspecified, because their authors assume that this is intuitively known by the users and interpreted accordingly within the expected context of use. E.g., *"Human"* in a biological database entry, or database schema, could be assumed to denote an individual *Homo sapiens* organism, or the class *Homo sapiens*, the quality of an object belonging to the taxon *Homo sapiens* (Schulz *et al.*, 2008), or a population of humans. The class *"Animal"* could be interpreted as including the class *Homo sapiens*, in the context of Biology or excluding it, e.g. in the context of Law. This underlines the need to make these hidden assumptions explicit. We postulate that ambiguous interpretations are best avoided if both database schemas and content are rooted in expressive domain ontologies that inherit constraints from an upper level ontology.

Content retrieval applications that use existing domain ontologies as vocabularies, like the ones derived from OBDA would possible benefit from ontology interpretation. As ODBA tools are not able to retrieve and generate new ontology content, the interpretation goes beyond what the user has specified as task-specific mappings between databases and ontologies. This requires that databases do not only use domain ontologies as standardized vocabularies, but that the meaning of their entire structure and content is described by ontology axioms and assertions. This is what we propose.

2.3 Description Logics and OWL2

Description Logics (DLs) are representation languages used to forma- lize ontology content1 . DL classifiers, like HermiT (Glimm *et al.*, 2014), compute additional subclass axioms, find equivalent classes, and assure satisfiability, by spotting contradictory axioms. DLs distinguish between TBox and Abox. TBoxes describe class level axioms (e.g., "all chimps are primates"), whereas A-boxes describe assertions on individuals, (e.g., "Washoo is a chimp").

The Semantic Web Standard OWL2 (W3C, 2012) uses the DL sublan- guage *S ROIQ* (Horrocks *et al.*, 2006), with limited expressiveness but complete and finite reasoning. OWL2 supports classes, binary relations (called object properties), and individuals, together with related axioms and assertions. For instance, the OWL2 class *Drosophila melanogaster*’ has all individual drosophila as members. As all individual drosophila are members of *Organism*’, we can infer taxonomic subsumption: ’*Drosophila melanogaster*’ forms a subclass of *Organism* if and only if all particular drosophila are equally members of *Organism*.

Such class statements are constructed by the combination of opera- tors specified in OWL2, *viz.* ’and’ for conjunctions, ’or’ for disjunctions, ’some’ for existential restrictions, and ’only’ for value restrictions, under the Manchester Syntax for OWL2 (Horridge and Patel-Schneider, 2009)2

2.4 Application background

For the use case, we use data and ontologies related to the metabolism of homocysteine (Hcy). Hcy is an amino acid that plays a key role in vitamin and cofactor metabolism, neuronal metabolism, and in the biological oxidation of enzymes. Hcy is also involved in the metabolism of sulphur-based amino acids, where it can be converted into methinine or cysteine. (Vitamin B6 dependent). When converted into methionine, the reaction depends cobalamin (Vitamin B12) and requires 5-methyltetrahydrofolate

1 For DL syntax and semantics, cf. Baader *et al.* (2007).

2 In this work, classes are written in *italic*, binary relations (OWL object properties) in **bold** case.

“main” — 2016/1/3 — page 2 — #2

*Grounding and retrieving biomedical database content with ontologies* **3**

(5-methyl-THF) (Selhub, 1999). The latter is result of the reduction of

5,10-methyl-THF via 5,10-methyl-THF reductase, an enzyme that regula- tes Hcy levels (Selhub, 1999). High levels of Hcy are reported to play a role in the pathogenesis of atherosclerosis (Muniz *et al.*, 2006), and of hepatic steatosis in hepatitis C infected subjects (Siqueira *et al.*, 2011). Many orga- nisms host Hcy-related bioprocesses, e.g. *Mus musculus*, *Homo sapiens*, *Gallus gallus*, *Schizosaccharomyces pombe*, and *Oryza sativa*.

**3 Resources**

3.1 Biomedical Ontologies

*•* The **Gene Ontology** (GO) (The Gene Ontology Consortium, 2014) was created in 1998 to address biomedical information integration through standardization of terms for the annotation of DNA sequences and their respective characteristics. GO has become a crucial resource for functional genomics, as an ongoing collaborative effort that delivers a controlled vocabulary underpinned by an ontology language. GO provides class hierarchies under ’*Cellular component*’,

’*Biological process*’, and ’*Molecular function*’ (ontologically better described as molecular activities or processes), together with the relations between them.

*•* **Chemical Entities of Biological Interest** (ChEBI) (Hastings *et al.*,

2013) describes low-molecular-weight chemical entities for under- standing and intervening in biological functioning. Each ChEBI entry denotes a chemical structure in a graphical form, together with ontological axioms. The ontology is subdivided into ‘*Molecular structure*’ and ‘*biological role*’ Whereas the further represents the structure of small molecules and their constituents, the latter is used to classify molecules depending on their disposition of participating in biological processes.

*•* The **Protein Ontology** (PR) (Natale *et al.*, 2014) is held by the *Pro- tein Information Resource* (PIR), integrating several databases and responsible the current structure for the UniProt database. It repre- sents modified forms, isoforms and protein complexes from living organisms and provides relations between them.

*•* **SNOMED CT** (Donnelly, 2006) is a large clinical terminology for human and veterinary medicine, containing formal definitions, which can be transformed into an OWL-EL ontology. SNOMED CT covers clinical findings and disorders, body parts, devices, drugs, substances, organisms, clinical procedures, among others.

*•* **BioTopLite 2** (BTL2) (Schulz and Boeker, 2013) is a lightweight and redesigned version of BioTop, created in 2006 as an upper-domain ontological layer to enable the representation of general aspects of biology and medicine. BTL2 offers highly constrained classes, using a small set of relations. Classes like *organism*, ’mono *molecular entity*’, and

’*body part*’ facilitate the alignment with other ontologies like GO, PR and ChEBI. BTL2 can be aligned with most of BFO and RO. Available biomedical ontologies compliant with these two sources can easily be integrated with BTL2.

3.2 Biological Databases

*•* The **Universal Protein Resource** (UniProt) was created in order to enable a quick understanding of the field of proteomics (UniProt Con- sortium, 2014). It provides a comprehensive, open-access resource of protein sequences and functional information. UniProt is mainly composed by a Knowledge Base (UniProtKB), subdivided in Swis- sProt (manually curated) and TrEMBL (generated and maintained by automated tools). Other parts are databases for sequences, closely related protein sequences, protein information from fully sequenced organisms, and metagenomics.

Data from literature and available in UniProt are organized and stored according protein and gene names, function, catalytic activity, cofactors, pathway information, subcellular location, among others. UniProt embeds NCBI Taxonomy IDs directly throughout its structure, as well as GO annotations (Camon *et al.*, 2003; Huntley *et al.*, 2014), together with mappings to several biological databases including Ensembl.

*•* The **Ensembl** project was launched in 1999 in order to automatically annotate genomes and to integrate this data with other biological data sources, thus creating a freely available online source. Ensembl processes and summarizes large-scale genomic data for chordates and model organisms. Its content is related to the annotation of gene and transcript locations, gene sequence evolution, genome evolution, sequence and structural variants and regulatory elements.

*•* The **NCBI Taxonomy** (NCBI Resource Coordinators, 2015) was derived from a project on the taxonomy of biological organisms that aimed at extracting sequences not available in dedicated databases from genomic literature. This coincided with the collection of data about taxonomic classifications. The goal of NCBI Taxonomy is to combine existent, distributed organism taxonomies into a single one that is included in NCBI GenBank.

**4 Methods**

This section describes data acquisition and the methodology developed for converting database content into ontology axioms. Content and related files, such as spreadsheets, scripts, and ontology files are available in the project website [(http://www.cin.ufpe.br/~](http://www.cin.ufpe.br/)integrativo).

4.1 Sampling

Data related to 21 organisms3 , together with processes and by-products from Hcy metabolism were retrieved from the UniProt and Ensembl4 websites. The ontologies GO, ChEBI and BTL2 were downloaded in OWL2 format5 .

For the creation of a subset from UniProt and Ensembl, UniProt data were filtered by the string “homocysteine", thus retrieving all Hcy-related data from UniProt/SwissProt+Trembl. From the obtained 212,156 records, the ones with GO annotations, specified gene names and proteins descri- bed by Selhub (1999) were selected. From the resulting 1,716 records, fragments, isoforms or homologue entries are excluded. The resulting set included the proteins Methionine synthase (MS) , Methylenetetrahy- drofolate reductase (MTHFR) , Cystathionine beta-synthase (CBS) , and Gamma-cystathionase (CSE).

From this dataset with 64 records without Ensembl IDs were removed, thus obtaining a final sample with

46 Hcy-related records, made available as a Microsoft Excel spreadsheet with the following tabular structure:

*•* One Protein (e.g. *CBS)*;

*•* One Taxon (e.g. *Rattus norvegicus*);

*•* One to many GO biological processes (e.g., *Blood vessel remodeling*)

*•* One to many GO molecular functions (e.g., *Cystathionine beta- synthase activity*)

3 Giant panda, Bovine, white-tufted marmoset, dog, zebrafish, chicken, human, West Indian ocean coelacanth, African elephant, mouse, European domestic ferret, Nile tilapia, rabbit, chimpanzee, Sumatran orangutan, rat, Tasmanian devil, pig, Japanese pufferfish. Western clawed frog

4 UniProt: Release 2015\_04, Ensembl Release 79, NCBI Taxonomy

2015AA.

5 GO Revision 25527, ChEBI Release 127, BTL2 Release 8th march 2015, PR release 22nd may 2015

“main” — 2016/1/3 — page 3 — #3

*•* One to many GO cellular components (e.g., *Cytoplasm*)

*•* Zero to many phenotypes (e.g., *Endocrine pancreas increased size*)

4.2 Ontological grounding and ontology generation

The grounding process is intellectually challenging, because it requires in-depth biology knowledge, insight into the way biological databases are populated, as well as ontology engineering skills guided by the rigour of a rich upper level ontology.

Aware that a straightforward, automatized "ontologization" of the data- base schema is not possible, the ontology engineer has to critically assess the pros and cons of competing modelling strategies. This must be performed in a way that correctly accounts for the underlying biological reality on the one hand, and that provides enough expressiveness to address the use cases (formulated as competency questions), on the other

hand.

1. One or more references to GO classes in UniProt annotations for GO classes, such as go:’*Biological process*’ and go:*Methylation* included as annotation for ’Methionine synthase’ in UniProt, as well as go:’*Cellular component*’ and go:’*molecular\_function*’ classes;

2. Reference to Proteins from PR directly as protein names in UniProt;

3. Database IDs from Ensembl in UniProt, and vice-versa;

4. Organism names as described in NCBI Taxonomy inside Ensembl and

UniProt;

5. Phenotypes according to Ensembl and included as a list of subclasses of BTL2 *Situation*, alinged with *Clinical finding* in SNOMED CT.

Figure 1 illustrates the final mapping structure of GO and ChEBI with

BTL2.

btl2:‘*particular at some time*’

The workflow can be described as follows:

*•* Top-level classes and relations of the domain ontologies are ali-

btl2:*disposition*

btl2:‘*Material object*’

so:‘*sequence feature*’

btl2:‘*Process*’

gned with the upper level ontology. Additional content relevant for a complete representation is added.

chebi:*role*

btl2:*compound*

go:*biological*

*process*

go:*molecular*

*function*

*•* Consistency of the maps is assured by a DL reasoner.

*•* Database objects are subjected to ontology-inspired scrutiny: while generally categorized as information entities, data objects have to be connected to their referents in the domain, for which it is decided whether they are individuals or classes.

chebi:‘*chemical entity*’

chebi:‘*molecular entity*’

pr:*protein*

btl2:‘*poly molecular composite entity*’

btl2:‘*structured biological entity*’

go:*cellular*

*•* The interdependencies and relationships between the referents and/or

their types are analysed, based on domain knowledge.

*•* The need for newly defined subclasses is assessed.

btl2:*organism*

*component*

*•* Prototyping is done by manual creation of a small OWL file based on a limited number of database records. It is submitted to repeated sati- sfiability testing and discussions among experts for representational richness and adequacy, looking at computed entailments and results of sample DL queries.

*•* Numerous iterations are done, regarding choice of object properties, quantifiers, nesting of logical expressions, reference to implicit know- ledge. Whenever a design decision leads to reasoning errors it has to be revised.

*•* Recurring structures are identified, which results in the abstraction of representational ontology patterns to be used for the whole database content.

*•* The sample ontology is translated into OWL/XML format and manu- ally dissected in order to identify variable and fixed elements. All variable elements are identified by placeholders.

*•* Both database extract and ontology patterns with placeholders are represented as spreadsheets.

*•* The target ontology is generated by transforming database content with ontology patterns, using a customized script that takes spreadsheet content and generates OWL code.

For this work, code generation (OWL2) was performed with a simple script in VBA6 .

4.3 Alignments and mappings

Top classes from GO, PR, and ChEBI were included as subclasses of BTL2 leaf nodes and tested for logical consistency using HermiT. In particular, the following alignments were done:

6 *Visual Basic for Application*

**Fig. 1.** Alignment of GO, ChEBI and PR under BTL2

4.4 Ontology modules extraction

To optimize performance during exemplification, we modularized GO, ChEBI and PR to include the classes strictly mentioned in data. These classes were manually identified and listed as a signature file. From it, we used the plug-in Ontology Modularity (Jiang *et al.*, 2011), together with Protégé.With this simple procedure, we were able to generate a representative, complete and satisfiable subset of each ontology. For more information concerning the completeness and validity of ontology modules, please C.f. Parsia *et al.* (2010).

4.5 Evaluation Methodology

The ontological content is evaluated by a set of competency questions (CQs). These are first formulated in English by the first author, a biologist. CQs are shaped according to how domain experts would query a biological database, and not how ontology engineers would interpret it, in order to be neutral regarding the internal structure of the ontology.

Similarly to the grounding process, the translation of queries, origi- nally formulated in natural language, into DL queries relies on the correct identification of query components that denote relations, referents, and the way how domain entities are related to one another. CQs, explanation and description about expected results are provided in section 5.4.2.

The following CQs were formulated:

1. Which kinds of biological process are included in organisms of a specific type of *Mus musculus*?

2. Which are the proteins that are able to perform certain molecular

functions of type ‘*methyltransferase activity*’?

“main” — 2016/1/3 — page 4 — #4

3. Which are the kinds of biological processes in which a specific protein of type ‘*cystationine gama lyase*’ participates, and has the function of performing molecular activities of type ‘*carbon-sulfur lyase activity*’?

4. Which are the kinds of biological processes that entail some risk of causing a specific dysfunctional state of ‘*Atherosclerosis*’?

5. Which kinds of organisms are capable of performing a specific biological process of type of ‘*cysteine biosynthetic process*’?

6. Which are the proteins that are found in organisms of the kind ‘*Bos taurus*, which bears the capability to perform a specific biological process of the kind ‘*methionine biosynthetic process*’?

CQs were selected to provide examples that explore entity types expres- sed in data, like phenotypes, proteins, molecules and biological processes from several organisms. To avoid biased judgements, the grounding pro- cess is assessed by means of description logics classification and retrieval of content from the axioms generated. This is possible as CQs are rendered as DL queries and submitted to the final ontology.

**5 Results**

5.1 Basic assumptions

The database content inspection was performed according to the following interpretation:

Each record introduces a series of defined subclasses of a biological process. Each process subclass is defined by having proteins of a certain type and from a certain species as participants, have hcy as output, occur in defined cell components and are parts of certain biological processes.

This interpretation express the knowledge denoted by biological data- bases at class level. It introduces a large number of defined biological process subclasses and makes the important assumption that they are non- empty (otherwise there would not have been any experimental evidence). This interpretation allows us to leverage reasoning about individuals from the A-box to the T-box.

E.g. reasoning is used to classify and check consistency of classes, which in turn are used to enable relations among individuals. Considering relationships among individuals are limited by the relations from class description, and in most use case scenarios 7 individuals must follow their class -based descriptions, all consistency checking and classification may be performed only according the T-box. This avoid the high computational cost of reasoning with A-box individuals.

With our interpretation, a record subtlety is represented as a new subclass. In other words, a record that introduces a human being able to develop diabetes due to high sugar diet is a type of human that reacts to a high sugar diet as being able to develop diabetes. This approach, at the same time, is able to represent ontologically the record content and do not entail performance issues. Thus, avoid the creation of misleading class definitions to commit to the world interpretation embedded on data.

The construction of the appropriate ontology pattern had to consider several implicit facts. Apart from the general assumption that hcy takes place in all of the described processes, we have to assert that the they are related to both cell components and organisms via the BTL2 relation

‘**is included in**’, and that only dysfunctional processes lead to the risk of developing certain pathological phenotypes.

Each record found in UniProt and Ensembl is interpreted as unique; one or more similar publications may result in one record, and every variation in an experiment results in a new record. Fields can be filled by zero to many references to ontology classes, the type of which is given by the column

7 ontology-based database integration, intelligent agents, information retrieval, among others

header, e.g., UniProt *“Protein”* for classes of pr:*Protein*, *“Molecule”* for btl2:‘*Mono molecular entity*’, and *‘Organism’* for btl2:*Organism*.

5.2 Ontological Grounding

The ontological grounding steps of the selected database content is described. Table 1 shows a subset of the table created as a view from UniProt and Ensembl(Table 1) (IDs from UniProt and Ensembl are used only for mapping purposes). We translate the content from table 1 as an example table 2 for interpretation.

As a first task, we interpreted how data in Table 2 can be represented from an ontological point of view:

*•* There exist biological processes of the type *Bp* in organisms of the type

*O* that have the protein *P* and the small molecule *M* as participants;

*•* In each *Bp*, the protein *P* bears the capability to perform one or more molecular functions *Mf* ;

*• Bp* occur in one or more types of cellular components *C*;

*•* There exist biological processes of the type *Bp* that are dysfunctio- nal and therefore bear the risks of causing one or more pathological phenotypes of the type *Ph*;

*•* All organisms of the type *O* bear dispositions to be realized by types of *Bp*;

*•* All types of protein *P* in *O* bear one or more types of *Mf*;

*•* Proteins of class *P* are not organism specific. However, the records refer to organism specific proteins, and we introduce subclasses *P\_sensu\_O* for each record (Protein *P* from Organism type*O*).

*•* Each *Bp* referred to by a record may happen with the same structure in several organisms *O*, including organism-specific proteins *P* and molecules *M*. However, each record an exclusive occurrence of *Bp*. In this sense, a record represents specific subclasses of *Bp*, identified as *Bp\_in\_O\_with\_P\_and\_M*, generated as a combination of biological process, organism, protein and small molecule.

*•* The database structure leaves open in which cellular component *C* a given *Bp* subclass is located, when there is more than one entry in the cellular component field. For this reason, we generate union classes of the type *C*1 or *C*2 or …or *Cn* to which the process locations can be safely assigned.

*•* It is not possible to specify that a *Bp* subclass entails the realization of an specific molecular function of the type *Mf*. Therefore in the definition of each *Bp* subclass the molecular function from the records are attached to the protein agent of that *Bp* subclass (‘**is realized by**’ only *Mf* ) .

*•* If there are phenotype entries *Ph*, a new class of the type *Dysfunctional\_Bp\_in\_O\_with\_P\_and\_M* is generated for every *Bp\_in\_O\_with\_P\_and\_M*, and all phenotypes are referred to as being the realizations of risks.

In all cases, when querying any ontology derived from an interpreta- tion process like this, we must perform a two step query. This is required because we may retrieve the specific classes generated from table conver- sion, like a *Bp\_in\_O\_with\_P\_and\_M*. However, its superclass (e.g. the related *Bp*) denotes the real ontological entity and should be shown as result.

5.3 Ontology patterns

This analysis allowed us to identify ontology patterns. First, we present the axiomatic definitions of subclasses of *P*(table 3), *Bp* (table 4) and *C* (table 5) .

Table 3. Defined Subclasses of proteins P.

*P* subclassOf pr:*Protein* ; *P\_sensu\_O* subclassOf *P*

“main” — 2016/1/3 — page 5 — #5

Table 1. Uniprot and Ensembl table view.

Entry Protein Organism GO (bp) GO (mf) GO (cc) Ensembl ID Ensembl Phenotype

F1MEW4 CBS *Bos taurus* blood vessel remodeling; …

cystathionine *β*- synthase activity

…

cytoplasm … ENSBTAT00000000184; … No phenotype associated

Q99707 MS *Homo sapiens*

cobalamin meta- bolic process; …

cobalamin binding; …

cytoplasm … ENST00000366577; ENST00000535889

Neural tube defect; Mega- loblastic anemia; …

UniProt entries in the left, and Ensembl in the right. GO (bp) , GO (mf) and GO (cc) represents rows from UniProt that include annotations for GO

classes ’*Biological process*’, ’*Molecular function*’ and ’*Cellular component*’ respectively.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Table 2. Template table. |  | | | | | | | |
|  | # | *P* | *O* | *Bp* | *Mf* | *C* | *Ph* | *M* |
|  | *k* | *Pk* | *Ok* | *Bp*1 *. . . n* | *M f*1 *. . . n* | *C*1 *. . . n* | *P h*1 *. . . n* | *M*1 *. . . n* |
|  | … | … | … | … | … | … | … | … |
|  | *l* | *Pl* | *Ol* | *Bp*1 *. . . n* | *M f*1 *. . . n* | *C*1 *. . . n* | *P h*1 *. . . n* | *M*1 *. . . n* |
|  | … | … | … | … | … | … | … | … |
|  | *m* | *Pm* | *Om* | *Bp*1 *. . . n* | *M f*1 *. . . n* | *C*1 *. . . n* | *P h*1 *. . . n* | *M*1 *. . . n* |

The symbol # represents record IDs; *P* proteins; *G* genes; *O* organisms; *Bp* biological processes; *Mf* molecular function; *C* cellular component; *Ph*

phenotype; and, *M* the associate molecules.

Table 3 shows how proteins of any type *P* within a single database record are described as subclasses of pr:*Protein*. The composed name concerning the protein of a specific organism *P\_sensu\_O* is a type of *P*.

Table 4. Defined Subclasses of biological process Bp.

*Bp* subclassOf go:‘*biological\_process*’ *Bp\_in\_O\_with\_P\_and\_M* subclassOf *Bp Dysfunctional\_Bp\_in\_O\_with\_P\_and\_M* subclassOf

*Bp\_in\_O\_with\_P\_and\_M*

Biological processes *Bp* are subclasses of go:‘*Biological process*’, and the combination of a specific biological process of a record with the rela- ted organism, protein and small molecule(s) determines the creation of *Bp\_in\_O\_with\_P\_and\_M* as subclass of *Bp*.

Cellular components of any type *C* (within a single record) are put as subclasses of go:’*cell\_component*’ (Table 5).

Table 5. Cellular component C union classes.

*C* subclassOf go:‘*Cell component*’ *C*1 *\_or\_Cn* subclassOf go:‘*cell\_component*’ *C*1 \_*or*\_*Cn* equivalentTo (*C*1 or *C*2 or …or *Cn* )

When a record includes more than one cellular component, union classes type *C*1 or *C*2 or …or *Cn* are created under go:*cell\_component*.

Next, the axioms for *Bp\_in\_O\_with\_P\_and\_M* classes (Table 6.)

Table 6. Bp\_in\_O\_with\_P\_and\_M.

*Bp\_in\_O\_with\_P\_and\_M* equivalentTo *Bp*

and (‘**has participant**’ some *M* )

and (‘**has participant**’ some (*P* and

(‘**is bearer of** ’ some (btl2:*Function* and

(‘**is realization of** ’ only *Mf* )))

and (‘**is included in**’ some *C*1 or *C*2 or …or *Cn* )

and (‘**is included in**’ some *O*)

Axioms for *Bp\_in\_O\_with\_P\_and\_M* (Table 6) describes that a bio- logical process from a single record has as participants one or more small molecules; the process is included in the combination of one or more cel- lular component; included in a specific organism; and, the protein from the record is a participant in the process, and bear some function to perform molecular functions.

Some *Bp\_in\_O\_with\_P\_and\_M* are dysfunctional, i.e. denotes abnor- mal organisms’ situations.

Table 7. Dysfunctional phenotypes of Bp\_in\_O\_with\_P\_and\_M.

*Dysfunctional\_Bp\_in\_O\_with\_P\_and\_M* equivalentTo

*Bp\_in\_O\_with\_P\_and\_M*

and (‘**is bearer of** ’ some ‘*Dysfunctional Quality*’)

*Dysfunctional\_Bp\_in\_O\_with\_P\_and\_M* subClassOf

*Bp\_in\_O\_with\_P\_and\_M*

and (‘**is realization of** ’ only (*Risk* and (**causes** some *Ph*)))

*Dysfunctional\_Bp\_in\_O\_with\_P\_and\_M* are processes that bear the quality of being dysfunctional; and, is the realization of a risk (disposition type) of causing the dysfunctional phenotype.

Following, the axioms required to represent *P\_sensu\_O* (Table 8) , i.e. the protein type of an specific organism from a record.

Table 8. Subclasses created for the organism specific protein (P\_sensu\_O)

classes in database records

*P\_sensu\_O* equivalentTo *P* and (‘**is included in**’ some *O*)

*P\_sensu\_O* subClassOf *P* and (‘**is bearer of** ’ some (*Function* and

(‘**has realization**’ only *Mf* )))

Definitions that follow the pattern from Table 8 describe organism- specific protein molecule classes. In addition, the molecular functions specified in the tables are added.

The last axiom required is about organisms as bearers of dispositions related to performing biological processes (Table 9) .

Table 9. Axioms generated for organisms O in database records

*O* subClassOf btl2:*Organism* and

(‘**is bearer of** ’ some (*Disposition* and

‘**has realization**’ only *Bp*)))

Table 9 attaches dispositions realized by specific biological processes to organisms.

“main” — 2016/1/3 — page 6 — #6

5.4 Evaluating the content generated

The analysis of the content of database entries has resulted in a set of OWL T-Box axioms for each database record as specified above. We recall two basic assumptions made, *viz.* (i) non-emptiness of classes: i.e. each of the newly defined classed corresponds to at least one fact described in literature, and (ii) the veracity of database entries, i.e. each informa- tion is considered a statement of truth. Given these boundary conditions, evaluation of the generated TBoxes will address the aspects: (i) logical satisfiability when importing all constraints from the upper-level-ontology BTL2; (ii) adequacy (correctness and completeness) of entailments against CQs; and, (iii) computational performance.

**5.4.1 Evaluation of Satisfiability**

This step is evaluated automatically with the support of HermiT OWL2 reasoner. When a new version of the ontology is created and new subclasses are generated from records, the reasoner is used to verify if the content is satisfiable (logically sound). To exem- plify the procedure, we evaluate with the reasoner if a class of the type *Bp\_in\_O\_with\_P\_and\_M* is (or is not) equivalent to the original *Bp* superclass. To be more especific, such as if ‘*methyla- tion\_in\_Homo\_sapiens\_with\_Methionine\_synthase\_and\_Homocysteine*’ is a *methylation*.

Other type of logical satisfiability performed is the evaluation of the correct usage of relations inside the axioms created, for instance, when applying the relation ‘**has realization**’ between an organism of the type *O* and a biological process of type *Bp*. According to BTL2, material entities are bearers of the capability (*Role*, *Function* or *Disposition*) that culminates with the realization of a specific process.

**5.4.2 Evaluation of Competency Questions (QCs)**

In the following, each competency question is translated into a DL query. The result is analysed and discussed.

*CQ1: Which kinds of biological process are included in organisms of a specific type of Mus musculus?* This query is intended to retrieve all biological process classes that takes place in organisms. CQ1 is translated and presented in table 10 in DL.

Table 10. Competency Question #1

‘*Biological process*’ and (‘**is included in**’ some ‘*Mus musculus*’)

The following classes are retrieved from the ontology.

– Results:

*•* ‘*amino acid betaine catabolic process in Mus musculus with Betaine homocysteine S methyltransferase 1 and Homocysteine*’;

*•* ‘*blood vessel remodeling in Mus musculus with Cystathionine beta- synthase and Homocysteine*’;

*•* ‘*cartilage development involved in endochondral bone morpho- genesis in Mus musculus with Cystathionine beta-synthase and Homocysteine*’; and 36 more classes.

~~However, as we previously described, we must retrieve the superclasses from these results. These are the classes included in databases, and might be seen as the result. To this last step, we have to create a second query (table 11)~~

Table 11. CQ1 second step query.

*amino acid betaine catabolic process in Mus musculus with Betaine homocysteine S methyltransferase 1 and Homocysteine* and *blood vessel remodeling in Mus musculus with Cystathionine beta synthase and Homocysteine* and … and *superoxide metabolic process in Mus musculus with Cystathionine beta synthase and Homocysteine*

~~That brings the following result.~~

~~– Results:~~

*•*

These results are expected as they match the content represented in data, without changing any domain or upper domain ontology. Here we see how a query on potentialities can be expressed as a simple DL query. However, the correct interpretation hinges on the assumption that none of these specific subclasses is empty.

~~From this point, we do not show the results from the first step query, and present directly the result as the reader may have understood the notion behind the two-step query approach.~~

*CQ2: Which are the proteins that are able to perform certain molecu- lar functions of type ‘methyltransferase activity’?* This query is meant to retrieve classes of proteins that are able to perform certain molecular functions. It is important to highlight that some proteins are capable to act in a specific way, like polimorphisms in gene MS leads to methionine synthase deficiency, which leads to higher homocysteine levels together with dysfunctional situations in humans and mice.

To illustrate, we rewrite CQ2 in DL in table 12.

Table 12. Competency Question #2

*protein* and (‘**is bearer of** ’ some *function* and

(‘**is realized by**’ only ‘*methyltransferase activity*’))

Using DL query and reasoning, we obtain the following results:

– Results:

*•* ‘*Betaine homocysteine S-methyltransferase 1 sensu Homo sapiens*’;

*•* ‘*Cystathionine beta-synthase sensu Homo sapiens*’;

*•* ‘*Cystathionine gamma lyase sensu Homo sapiens*’;

*•* ‘*Methionine synthase sensu Homo sapiens*’;

*•* ‘*Methylenetetrahydrofolate reductase sensu Homo sapiens*’.

These are the proteins able to perform for the ‘*methyltransferase activity*’ molecular function.

*CQ3: Which are the kinds of biological processes in which a specific protein of type ‘Cystationine gama lyase’ participates, and has the function of performing molecular activities of type ‘carbon-sulfur lyase activity’?* This query is related to the identification of biological processes (e.g. reactions) that involve a specific protein that should be able to performing this reaction. The relevance of this query is related to the capability of retrieving specific biological processes by means of proteins from specific reactions.

CQ3 is available below (table 13)

Table 13. Competency Question #3

‘*biological\_process*’ and

‘**has participant**’ some (‘*Cystationine gama lyase*’ and

(‘**is bearer of** ’ some (*Function* and (‘**has realization**’ only

‘*carbon-sulfur lyase activity*’)))))

Below, the results.

“main” — 2016/1/3 — page 7 — #7

– Results:

*•* ‘*cellular nitrogen compound metabolic process in Homo sapiens with cystathionine gamma lyase and Homocysteine*’;

*•* ‘*cysteine biosynthetic process in Homo sapiens with Cystathionine gamma lyase and Homocysteine*’;

*•* ‘*cysteine metabolic process in Homo sapiens with Cystathionine gamma lyase and Homocysteine*’;

*•* and 14 more classes.

*CQ4: Which are the kinds of biological processes that entail some risk of causing a specific dysfunctional state of ‘Atherosclerosis’?* This query retrieves biological processes that entail the risk of developing a dysfu- nctional phenotype. This query is relevant in the sense whether it enables the identification of any abnormal situations regarding an specific process from an organism.

CQ4 is available in the following page (table 14).

Table 14. Competency Question #4

‘*biological\_process*’ and (‘**is realization of** ’ only

(*Risk* and (**causes** some ‘*Atherosclerosis*’)))

Next, CQ4 results.

– Results

*•* ‘*Dysfunctional homocysteine metabolic process in Rattus norvegicus with Methylenetetrahydrofolate reductase and Homocysteine*’;

*•* ‘*Dysfunctional methionine biosynthetic process in Rattus norvegicus with Methylenetetrahydrofolate reductase and Homocysteine*’;

*•* ‘*Dysfunctional one carbon metabolic process in Rattus norvegicus methylenetetrahydrofolate reductase and Homocysteine*’;

*•* and 12 more classes.

*CQ5: Which kinds of organisms are capable of performing a specific bio- logical process of type of ‘Cysteine biosynthetic process’?* This query retrieves organisms that are capable of performing specific biological pro- cesses. This query is relevant because not all biological processes for organisms are fully described. Even two different organisms that include same proteins under same conditions may not include similar processes.

CQ5 is available below (table 15).

Table 15. Competency Question #5

*Organism* and (‘**is bearer of** ’ some (*Disposition* and

(‘**is realized by**’ only *‘Cysteine biosynthetic process*’)))

The results are displayed below.

– Results:

*•* ‘*Homo sapiens*’; ‘*Mus musculus*’.

*CQ6: Which are the proteins that are found in organisms of the kind ‘Bos taurus, which bears the capability to perform a specific biological pro- cess of the kind ‘methionine biosynthetic process’?* The aim of this query is to retrieve specific proteins that are related to the process ‘*methinine biosynthetic process*’, when performed by a organism of the type ‘*Bos taurus*’. In other words, we are able to identify specific proteins by means of organisms and biological processes among the content embedded in databases.

CQ6 is written in DL as follows (table 5.4.2).

Table 16. Competency Question #6.

*protein* and (‘**is included in**’ some (‘*Bos taurus*’ and

(‘**is bearer of** ’ some (disposition and (‘**has realization**’ only

‘*methionine biosynthetic process*’)))))

The results from CQ6 are displayed below.

*•* Result:

*•* as

*•* as

**5.4.3 Computational performance**

Data, ontologies and queries were manipulated with an Intel core i7 4510U with 8gb of RAM. The raw ontology presents expressivity ALC and took half second for classification and consistency checking.

When including modules from GO

When reasoning with query X, it took less than XXX seconds.

**6 Discussion**

A problem our work addresses is the formal interpretation of database content. It is responsible to enable retrieval of biological databases with a richer query paradigm. With our approach, queries can be performed directly to the ontology(ies), excluding the requirement for ontology population. With our proposal, we reduce query complexity to ontology level, with complete and finite reasoning, in comparison with the costs of rich TBoxes together with populated Aboxes (Motik and Sattler, 2006). The interpretation supports the notion of representing the content according to the classes the individuals are member, as observed in biological experiments.

The fact that typical queries target possibilities “*Are members of the class A able to do B?*” is addressed by two mechanisms. Firstly, the inclu- sion of dispositions as first-class entities in our ontology, and secondly the definition of specific subclasses. The latter case, however, requires the assumption that all of these classes are populated. In this sense, the que- stion “*Are members of the class A able to do B?*” is therefore translated into the question “*Does A have a subclass A’ all members of which actually do B?*”, to highlight the existence of classes, and not its direct assertion.

Interpreting database content under an ontological perspective has been a topic of research interest. Fanizzi *et al.* (2008) present a tool called DL- FOIL, which relies on refinement operators for class-learning. In DL-FOIL grounding is presented as learning by search of class definitions in an indu- ced search space. A limitation of DL-FOIL is the treatment of individuals that do not belong to a specific class, as well as the incompleteness of the refinement operator.

The DL-Learner system (Lehmann, 2009) is grounded on the require- ment for schema acquisition methods, as they intend to address the problem with class learning techniques. DL-Learner is designed to find logical explanations for indi- viduals inside the ontology. It is limited by the fact that positive and negative examples must be provided, and individuals must be included directly in the ontology. As previously mentioned, querying axiomatized and populated ontologies is costly.

Interpreting data and grounding it as ontology axioms seems as a suita- ble solution to boost interoperability with the support of formal ontologies. However, the usage of principled ontologies as guidance for interpreting data is quite limited. Earlier , we described how the content of tables from

scientific publications can be interpreted using formal ontologies under a

“main” — 2016/1/3 — page 8 — #8

Table 17. Brief description of ontological sources, expressivity and reasoning performance.

Ontology Subclass Axioms Equivalency Axioms Hidden GCI Class count DL Expressivity Reasoning time (h) CQ6 processing time (ms) Raw 1054 647 643 1721 *ALC* 0.017

GO - GO Module - ChEBI - ChEBI Module - PR - PR Module - BTL2 - Modularized 3284 980 973 1960 *SRI* 1.9

rigid upper level, also using competency questions for evaluation (Santana

*et al.*, 2011).

An improvement of our work embeds is the possibility of using more expressive power to retrieve generalizable content by means of DL Query. The retrieval of OBDA-based approaches, like SPARQL endpoints fairly supports reasoning that goes beyond what is available in current rela- tional queries (Angles and Gutierrez, 2008). Our approach allow evaluating databases from the ontological level, e.g. com- puting class-subclass relations, consistency checking and subsumption. This reduces the effort from the end user to bear some extent of expertise to manually filter/interpret data, without compromising the capability to be queried with SPARQL endpoints.

In this work, we re-affirm and demonstrate the power embedded in formal ontologies to represent database content. We model real data with the support of formal ontologies queries that requires reasoning at some extent, and exempt the user to know specific domain details to optimize domain query, like when creating relational or SPARQL queries.

For insta- nce, to retrieve a protein that bear the capability to realize methylation, with relational or SPARQL (without ontological treatments), the user must cre- ate joins and filters to gather content from different tables/filters. With DL query, we only need to define how the process behave and leave the querying and computing complexity to the machine. One may argue that our approach generates more complexity to domain representation. However, we may agree that this complexity is restricted to the representational level, and it is (at some extent) already covered by most formal biomedical ontologies.

To exemplify, Huang *et al.* (2009) described that, until 2009, 68 tools were available for data retrieval and comparison in biology. These tools were mainly devoted to the structural analysis of biological networks, sequences or pathways. These tools were created because (in most cases), databases have different purposes, different methodologies on how data is maintained, and incorporate implicit assumptions.

However, even finding certain advantages, we are aware of current limitations, such as scaling problems when increasing the size of data and ontologies. Other limitation is the understanding of the ontological representation from a usability point of view. The description of the formal aspects entails a deeper understanding of the ontology(ies) used, which is not required when creating database schema or informal representation.

Further investigations are required to address the impact that database updates may generate. Database updates may generate modification in the schema level that can lead to adaptations on the interpretation procedure, such as table joins or the obsolescence of certain content. We are cur- rently developing a system to support the interpretation procedure, which would minimize the deep ontological understanding required by current approach, and address the inherent awareness of data interpretation.

**7 Conclusion**

We presented an ontology engineering framework that supports interpreta- tion of biological data, from different sources, using a highly constrained upper-level ontology (BTL2), to which GO, PR and ChEBI were aligned. The ontology analysis of the content of biological databases yielded a set of ontology patterns, which were used to translate databases’ content into formal ontologies. The resultant ontological content was presented to formal scrutiny with DL queries, answered only by means of reasoning.

As we exemplified our framework under the biological domain, it can be ported to other domains that includes highly constrained and formalized ontologies. For instance, Prestes *et al.* (2013) described an upper-domain ontology based on IEEE standards for representing intelli- gent agent systems knowledge bases, following DOLCE (Gangemi *et al.*,

2002) and SUMO (Pease *et al.*, 2002).

The ontology derivate we produce focus on TBox reasoning rather than the analysis of data itself. Following this, a databased grounded under a formal ontology might enable the identification of representa- tional flaws under real world situations, opposite to application-driven databases, as it uses DL reasoning. As databases are extensive by nature, its organization under real world settlements certainly would be improved when grounded by formal ontologies. Additionally, it may enable further integration capabilities by means of automated evaluation using reasoning, e.g. without any dependence in user support.

According to our findings, this is possible because we interpret the entries in biological databases in ways that derive generalizable statements. These are expected to reveal scientific laws and can be ascribed to all indi- viduals that are members of a given class as well as database records. Reasoning can then be restricted to a TBox level, thus avoiding high pro- cessing cost that occurs when populating highly axiomatised TBoxes with individuals.

The feasibility of the approach could be demonstrated using com- petency questions formulated as DL queries. We query data ontologically, without requiring any additional database processing. If the data interpretation is ontologically sound, by inheritance all data may be considered sound in a real world use case.

**Funding**

This work was funded by *Conselho Nacional de Aperfeiçoamento de Pessoal de Nível Superior* (CAPES) 3914/2014-03; and, *Conselho Nacio- nal de Desenvolvimento Científico e Tecnológico* (CNPq) 140698/2012-4. Conflict of Interest: none declared.

“main” — 2016/1/3 — page 9 — #9

**References**

Altschul, S. F., Gish, W., Miller, W., Myers, E. W., and Lipman, D. J. (1990). Basic local alignment search tool. *J. Mol. Biol.*, **215**(3), 403–410.

Angles, R. and Gutierrez, C. (2008). The Expressive Power of SPARQL. *LNCS*,

**5318**, 114–129.

Baader, F., McGuinness, D. L., Nardi, D., *et al.* (2007). *The Description Logics Hand- book: Theory, implementation, and applications*. Cambridge University Press, Cambridge, 2 edition.

Camon, E., Barrell, D., Lee, V., Dimmer, E., and Apweiler, R. (2003). The Gene Ontology Annotation (GOA) Database - An integrated resource of GO annotations to the UniProt Knowledgebase. *In Silico Biology*, **4**(2).

Cunningham, F., Amode, M. R., Barrell, D., *et al.* (2014). Ensembl 2015. *Nucleic*

*Acids Res.*, **43**(D1), D662–D669.

Degtyarenko, K., de Matos, P., Ennis, M., *et al.* (2008). ChEBI: a database and onto- logy for chemical entities of biological interest. *Nucleic Acids Res.*, **36**(Database issue), D344–50.

Donnelly, K. (2006). SNOMED-CT: The advanced terminology and coding system for eHealth. *Stud Health Technol Inform*, **121**, 279–90.

Fanizzi, N., Amato, C., and Esposito, F. (2008). DL-FOIL: Concept Learning in Description Logics. *LNCS 5194*, **5194**(Proceedings of the 18th International Conference, ILP 2008 Prague, Czech Republic, September 10-12, 2008), 107–121. Gangemi, A., Guarino, N., Masolo, C., Oltramari, A., and Schneider, L. (2002).

Sweetening Ontologies with DOLCE. *LNCS*, **2473**, 223–233.

Glimm, B., Horrocks, I., Motik, B., Stoilos, G., and Wang, Z. (2014). HermiT: An

OWL 2 Reasoner. *J. Autom. Reason.*, **53**(3), 245–269.

Gruninger, M. and Fox, M. S. (1994). The Role of Competency Questions in Enterprise Engineering. In *IFIP WG5.7 Work.*, pages 1–17, Trondheim, Norway. Harris, S. and Seaborne, A. (2013). SPARQL 1.1 Query Language.

[http://www.w3.org/TR/sparql11-query/.](http://www.w3.org/TR/sparql11-query/)

Hastings, J., De Matos, P., Dekker, A., *et al.* (2013). The ChEBI reference database and ontology for biologically relevant chemistry: Enhancements for 2013. *Nucleic Acids Res.*, **41**(D1), D456–63.

Horridge, M. and Patel-Schneider, P. F. (2009). OWL 2 Web Ontology Language: Manchester Syntax. [http://www.w3.org/TR/owl2-manchester-syntax/.](http://www.w3.org/TR/owl2-manchester-syntax/)

Horrocks, I., Kutz, O., and Sattler, U. (2006). The Even More Irresistible SROIQ. In *KR 2006*, pages 1–36.

Huang, D. W., Sherman, B. T., and Lempicki, R. a. (2009). Bioinformatics enrich- ment tools: Paths toward the comprehensive functional analysis of large gene lists. *Nucleic Acids Res.*, **37**(1), 1–13.

Huntley, R. P., Sawford, T., Mutowo-Meullenet, P., *et al.* (2014). The GOA database: Gene Ontology annotation updates for 2015. *Nucleic Acids Res.*, **43**(D1), D1057– D1063.

Hustadt, U., Motik, B., and Sattler, U. (2005). Data Complexity of Reasoning in Very Expressive Description Logics. In *IJCAI-05*, pages 466–471, Edinburg, Scotland. Morgan Kaufmann Publishers Inc.

Jiang, G., Solbrig, H. R., Chalmers, R. J., *et al.* (2011). A Case Study of ICD-11

Anatomy Value Set Extraction from SNOMED CT. In *ICBO 2011*, pages 133–138, Buffalo.

Lehmann, J. (2009). DL-Learner: Learning Concepts in Description Logics. *J. Mach. Learn. Res.*, **10**, 2639–2642.

Lehmann, J., Auer, S., Bühmann, L., and Tramp, S. (2011). Class expression learning for ontology engineering. *J. Web Semant.*, **9**(1), 71–81.

Martinez-Costa, C., Cornet, R., Karlsson, D., Schulz, S., and Kalra, D. (2015). Semantic enrichment of clinical models towards semantic interoperability. The heart failure summary use case. *JAMIA*, pages 565–576.

Motik, B. and Sattler, U. (2006). A Comparison of Reasoning Techniques for Quer- ying Large Description Logic ABoxes. In *LPAR-06*, Phnom Penh, Cambodia. Springer-Verlag Berlin, Heidelberg.

Muniz, M. T. C., Siqueira, E. R., Fonseca, R. a., *et al.* (2006). Avaliação da relação entre o polimorfismo C677T no gene para MTHFR e a concentração plasmática de homocisteína na doença arterial coronariana.

Natale, D. a., Arighi, C. N., Blake, J. a., *et al.* (2014). Protein Ontology: A controlled structured network of protein entities. *Nucl. Acids Res.*, **42**(D1), 415–421.

NCBI Resource Coordinators (2015). Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research*, **43**(Database issue), D6–D17. Parsia, B., Sattler, U., and Schneider, T. (2010). The Modular Structure of an

Ontology : An Empirical Study. In *KR 2010*, pages 584–586. IOS Press.

Pease, A., Niles, I., and Li, J. (2002). The Suggested Upper Merged Ontology: A Large Ontology for the Semantic Web and its Applications. Technical report, AAAI.

Poggi, A., Lembo, D., Calvanese, D., *et al.* (2008). Linking data to ontologies.

*LNCS*, **4900**, 133–173.

Prestes, E., Carbonera, J. L., Rama Fiorini, S., *et al.* (2013). Towards a core ontology for robotics and automation. *Rob. Auton. Syst.*, **61**(11), 1193–1204.

Rector, A. (2008). Barriers, approaches and research priorities for integrating biomedical ontologies. Technical report, EU Semantic Health Support Action. Santana, F., Schober, D., Medeiros, Z., Freitas, F., and Schulz, S. (2011). Ontology

patterns for tabular representations of biomedical knowledge on neglected tropical diseases. *Bioinformatics*, **27**(13), i349–i356.

Schober, D., Smith, B., Lewis, S. E., *et al.* (2009). Survey-based naming conventions for use in OBO Foundry ontology development. *BMC Bioinformatics*, **10**, 125. Schulz, S. and Boeker, M. (2013). BioTopLite: An Upper Level Ontology for the

Life Sciences. Evolution, Design and Application. In U. Furbach and S. Staab, editors, *Inform. 2013*, Koblenz. IOS Press.

Schulz, S. and Jansen, L. (2013). Formal Ontologies in Biomedical Knowledge

Representation. *IMIA Yearb. 2013*, **8**(Evidence-based Health Informatics), 132–

146.

Schulz, S., Stenzhorn, H., and Boeker, M. (2008). The ontology of biological taxa.

*Bioinformatics*, **24**(13), 313–321.

Schulz, S., Grewe, N., Röhl, J., *et al.* (2012). Guideline on Developing Good Ontologies in the Biomedical Domain with Description Logics. Technical Report December, Universität Rostock, Rostock.

Selhub, J. (1999). Homocysteine metabolism. *Annu. Rev. Nutr.*, **19**, 217–246. Siqueira, E. R. F., Oliveira, C. P. M. S., Muniz, M. T. C., *et al.* (2011). Methy-

lenetetrahydrofolate reductase (MTHFR) C677T polymorphism and high plasma homocysteine in chronic hepatitis C (CHC) infected patients from the Northeast of Brazil. *Nutrition journal*, **10**(1), 86.

Smith, B., Ceusters, W., Klagges, B., *et al.* (2005). Relations in biomedical ontologies. *Genome Biol.*, **6**(5), R46.

Smith, B., Ashburner, M., Rosse, C., *et al.* (2007). The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat. Biotechnol.*,

**25**(11), 1251–5.

Spear, A. D. (2006). *Ontology for the Twenty First Century : An Introduction with Recommendations*. IFOMIS, Saarbrücken, Germany. [http://www.ifomis.org/bfo/manual.](http://www.ifomis.org/bfo/manual)

Szklarczyk, D., Franceschini, a., Wyder, S., *et al.* (2014). STRING v10: protein- protein interaction networks, integrated over the tree of life. *Nucleic Acids Res.*,

**43**(D1), D447–D452.

The Gene Ontology Consortium (2014). Gene Ontology Consortium: going forward.

*Nucleic Acids Res.*, **43**(D1), D1049–D1056.

Triplet, T. and Butler, G. (2011). Systems Biology Warehousing: Challenges and Strategies toward Effective Data Integration. In F. Laux and L. Strömbäck, editors, *DBKDA-2011*, pages 34–40, St. Maarten.

UniProt Consortium (2014). Activities at the Universal Protein Resource (UniProt).

*Nucleic Acids Res.*, **42**(Database issue), D191–8.

W3C (2012). OWL 2 Web Ontology Language Document Overview. [http://www.w3.org/TR/owl2-overview/.](http://www.w3.org/TR/owl2-overview/)

“main” — 2016/1/3 — page 10 — #10