*Bioinformatics*

doi.10.1093/bioinformatics/xxxxxx Advance Access Publication Date: Day Month Year ISMB 2016 Proceedings

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:** Current biomedical database interpretation approaches do not offer any support for the user to interpret content retrieved from databases. Frequently, the process is manual and the results interpretation may be biased by the researchers’ capabilities, by the sheer size and heterogeneity of the sources, or by technical limitations. We address this shortcoming by advocating a seamless integration of database and ontology content, underpinned by formal-ontological principles.

**Results:** We applied this approach to proteins and related biological process of the sulphur-based amino acid homocysteine. We are able to demonstrated the ability of using ontologies to interpret database, apply a formal validation to it, and use reasoning to retrieve hypothetical statements regarding proteins, processes, molecular activities and dysfunctional phenotypes exhibited by some organisms.

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

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

**1 Introduction**

Hypothesis and findings are generated in biomedical research based on gathering data from experiments, publications, and databases like UniProt (UniProt Consortium, 2014) or Ensembl (Cunningham *et al.*, 2014). In the biomedical field, the exploration of this contents is usually performed manually or (partly) supported by retrieval tools, e.g. by STRING (Szkla- rczyk *et al.*, 2014), or BLAST (Altschul *et al.*, 1990). Some works highlight the need and importance of database interpretation for the biomedical domain (Carnielli *et al.*, 2015; Laukens *et al.*, 2015).

Moreover, the interpretation of these results may be biased by the researchers’ capabilities, by the sheer size and heterogeneity of the sources, or by technical limitations (Triplet and Butler, 2011; Laukens *et al.*, 2015). It may result in ambiguous interpretation of data, the non-observance of records that may be crucial for a given problem, or by the user inability to query and filter a dataset, or the elevated processing costs related to massive datasets.

On the other hand, others tools like ontology-based data access (OBDA) (Poggi *et al.*, 2008) -related applications enhance data retrieval by ontologies, which serve as a query vocabulary – e.g. within SPARQL

(Harris and Seaborne, 2013) endpoints. By using, one gains the ability to integrate heterogeneous and massive data sources. Other tools rely on machine learning to interpret databases according to an ontological back- ground (Lehmann, 2009). The retrieval of OBDA-based approaches, like SPARQL endpoints, fairly supports reasoning that goes beyond what is available in current relational queries (Angles and Gutierrez, 2008).

However, such approaches are limited by the need of user interven- tion, e.g. the manual interpretation of the finally retrieved content, and the ontology population with data. Here, the interpretation problem is still pre- sent, as well as the raise of processing costs. OBDA and machine learning approaches frequently rely on the representation of data entities as indi- viduals (ABox elements in DL jargon), which results in high processing cost (Hustadt *et al.*, 2005), with or without computer-assisted reasoning.

Accordingly, the current situation regarding the (semi-)automated sup- port for database content retrieval and interpretation is characterized by a concurrent and continuous evolution of high-quality structured knowledge resources, whereas less progress can be seen regarding their usage, intero- perability and ontological grounding. Thus, the interpretation of results is frequently left to the users, and influenced by implicit background assumptions that may vary between users.

Apart from that, the usage of ontologies only to enable data integration

and retrieval restricts its applicability and justification. With a formalized

© 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” — 2016/1/14 — page 1 — #1

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

view of biomedical data provided by ontologies, we are able to integrate, retrieve, validate and derive assumptions from it, mainly due to automated classification and consistency checking delivered by Description Logics (DL) (Baader *et al.*, 2007) classifiers, like HermiT (Glimm *et al.*, 2014).

We address this shortcoming by advocating a seamless integration of database and ontology content, underpinned by formal-ontological princi- ples (Smith *et al.*, 2007), which enforce an univocal interpretation of the database content. In other words, we emphasize ontological grounding of databases as a mean enable (semi-)automated data interpretation and understanding. Ontological grounding means to identify ontology-level content in databases, and to axiomatize it under an upper level ontology. It delivers a homogeneous representation of data, linked to (parts of) existing ontologies.

Ontological grounding can be applied to enhance database curation, as new statements extracted from recent publications and recorded in data- bases can be ontologically grounded and tested for adequacy inside the biological domain. As this process is supported by automated reasoning, it enables a rigorous validation supported by an expressive query language (DL Query).

A DL Query uses the semantics and reasoning procedures of DL, which allow querying across several databases at the same time, thus decreasing the costs of database integration. This "ontological grounding" is based on previous work on the formalization of tabular representations in sci- entific literature (Santana *et al.*, 2011) and models of structured clinical information (Martinez-Costa *et al.*, 2015).

In this sense, we hypothesise and demonstrate that this enables more powerful queries, supported by machine reasoning. To this end, we will (i) analyse a subset of biomedical ontologies and databases; (ii) propose an ontology-based framework that makes explicit both database content and the domain entities denoted thereby; (iii) relate this solution to cur- rent workflows in which life science data and knowledge 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 becomes simpler and more user-friendly; (vi) perform an experiment to assess the scalability of the approach.

The biological use case is addressed Competency Questions (CQs) (Gruninger and Fox, 1994) as DL queries. They range from how data related to a specific metabolism overlap in biological databases for cer- tain model organisms, to phenotypes from dysfunctional metabolism. Our examples make use of UniProt, NCBI Taxonomy (NCBI Resource Coor- dinators, 2015), Ensembl, SNOMED CT (Donnelly, 2006) , 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**

2.1 Formal ontologies

Formal ontologies present mostly extensive (poly-) hierarchies of repre- sentational units, named classes, related by a limited set of binary relations. Ontologies are centred around classes. Together with a set of binary rela- tions and logical operators in a logic-based language, e.g., DL, they constitute formal axioms. Axioms 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 all adenine molecules are nucleotide molecules.

The construction of (formal) ontologies should obey principled criteria

(Smith *et al.*, 2005), such as âŁ˜has participantâŁ™ or âŁ˜has partâŁ™. 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.

We distinguish ontology content properly from both the notions of knowledge (Schulz and Jansen, 2013) and data. Regarding knowledge1 , assertions about what is frequently associated or only true by default can- not 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 use cases embedded in the underlying sch-

ema. The distinction between what is a data element and what is its referent is often 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(Schulz *et al.*, 2011).

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, assuming that this is intuitively known by the users and interpreted accor- dingly within the expected context of use. E.g. whether “Human" denotes an individual *Homo sapiens*, the class ‘Homo sapiens’, the quality of an object belonging to the taxon *Homo sapiens* (Schulz *et al.*, 2008), or a population of humans. In a similar vein, “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. Such ambiguities underline the need to make these hidden assumptions explicit, such as by ontological grounding.

Content retrieval applications that use existing domain ontologies as vocabularies, like the ones derived from OBDA might benefit from this. 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 formalise ontology content. DL classifiers, like HermiT (Glimm *et al.*, 2014), find new subclass links, identify equivalent classes, and assure satisfiability, by spotting contradictory axioms. TBox expressions are class level axioms (e.g., “all chimps are primates"), whereas ABox contain assertions on individuals (e.g., “Washoo is a chimp").

The Semantic Web Standard OWL2 (W3C, 2012) uses the DL lan- guage *SROI Q* (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* iff 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 .

(Spear, 2006) and good practice guidelines (Schulz *et al.*, 2012). Important

principles are (i) naming conventions (Schober *et al.*, 2009); (ii) mutu- ally disjoint upper-level classes like Process or Quality as a fundamental ordering framework; and, (iii) a generally small set of canonical relations

1 In a broader sense, Cf. (Rector, 2008)

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

“main” — 2016/1/14 — page 2 — #2

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

2.4 Application background

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

5-methyltetrahydrofolate (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 regulates 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 organisms host Hcy- related bioprocesses, e.g. *Mus musculus*, *Homo sapiens*, *Gallus gallus*, *Schizosaccharomyces pombe*, and *Oryza sativa*.

**3 Ontology grouding process**

The grounding process requires in-depth biology knowledge, insight into the way biological databases are populated, as well as ontology engineering skills, based on the understanding of upper level ontology principles and description logics.

Aware that a straightforward, automatized “ontologization" of a data- base schema is not possible, the ontology engineer has to critically assess the pros and cons of competing modelling strategies. This must be perfor- med 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.

The workflow is depicted in figure 1.

*•* Domain queries are defined as Competency Queries (Gruninger and Fox, 1994) (queries are introduced in section 5.4, explained and tested in section 6.4);

*•* One or more databases that cover the content related to the queries is selected. We selected UniProt, Ensembl and NCBI Taxonomy for exemplification;

*•* One or more ontologies are selected according to the content from database that denotes ontology classes;

*•* Ontology annotations or simbolic entries that denote ontology clas- ses are identified in database records. These are going to be the introductory links from database to ontology interpretation;

*•* Top-level classes and relations of the domain ontologies are aligned with the upper level ontology. Mappings are generated from database to the ontology aligned structure.

*•* 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.

*•* Consistency of the alignments and maps are assured by a DL reasoner.

*•* Repetitive structures, like the description that a given protein is able to perform a given intracellular process, are identified in database records according to the ontologies organization and described as general patterns:

– 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.

*•* General patterns are propagated to the overall database or subset under consideration;

*•* Content generated needs to be evaluated with the CQs defined

beforehand.

Begin

Define domain queries to the intended database(s)

Select one or more databases that cover the query content

Select domain ontologies that convey data elements, and an upper level ontology to mediate alignments

Identify ontology annotations or simbolic entries that denotes ontology classes in data

Map ontologies

and databases

End

Evaluate generated content with queries created beforehand

Propagate axioms generated towards the records under consideration

Abstract database records as axioms patterns

Identify repetitive structures and relations in database

Check consistency and derived entailments

– If the content generated meets consistency and is able to answer the user needs, the process reaches the end, and the user have as output an ontologized interpretation of the database.

– Otherwise, more iterations and evaluations are required in order to meet domain validity and user spectations.

According to the workflow, we introduce the CQs used for exemplifi- cation:

1. Which kinds of biological processes related to Hcy can be found in mice?

2. Which are the proteins that exhibit ‘*methyltransferase activity*’?

3. Which are the kinds of biological processes in which proteins of the type ‘*cystationine gama lyase*’ lyaseâŁ™ participate, exhibiting

‘*carbon-sulfur lyase activity*’?

4. Which dysfunctional biological processes entail a risk of ‘*Atheroscle- rosis*’?

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

6. Which proteins found in organisms of the kind *Bos taurus* have the capability of methionine biosynthesis?

CQs were selected to provide examples that explore entity types expres- sed in data, like phenotypes, proteins, molecules and biological processes from several organisms. From these CQs, we identified the ontological and database resources (Section 4).

**Fig. 1.** Ontology grounding process.

Next, we describe the workflow:

“main” — 2016/1/14 — page 3 — #3

**4 Resources**

4.1 Biomedical Ontologies Involved

*•* The **Gene Ontology** (GO) was created in 1998 to address biomedical information integration through standardization of terms for the anno- tation of DNA sequences and their respective characteristics. GO has become a crucial resource for functional genomics, as an ongoing col- laborative effort that delivers a controlled vocabulary underpinned by an ontology language. GO provides class hierarchies under ‘*Cellular component*’, ‘*Biological process*’, and ‘*Molecular function*’ (ontolo- gically better described as molecular activities or processes), together with the relations between them.

*•* **Chemical Entities of Biological Interest** (ChEBI) describes low- molecular-weight chemical entities for understanding and intervening in biological functioning. Each ChEBI entry denotes a chemical stru- cture 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) is held by the Pro- tein Information Resource (PIR), integrating several databases and responsible the cur- rent structure for the UniProt database. It represents 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 and clinical procedures, among others.

*•* **BioTopLite 2** (BTL2) is a lightweight and redesigned version of Bio- Top, 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.

4.2 Biological Databases

*•* The **Universal Protein Resource** (UniProt) was created in order to enable a quick understanding of the field of proteomics. It provi- des a comprehensive, open-access resource of protein sequences and functional information. UniProt is mainly composed by a Knowledge Base (UniProtKB), subdivided in SwissProt (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, sub-cellular location, among others. UniProt embeds NCBI Taxonomy identifiers directly throughout its structure, as well as GO annotations (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 proces- ses 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** was derived from a project on the taxonomy of biological organisms that aimed at extracting sequences not avai- lable 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.

**5 Methods**

In the following, data acquisition and the methodology for converting database content into ontology axioms are described. Content and related files, such as spreadsheets, scripts, and ontology files are available in the project website [(http://www.cin.ufpe.br/˜integrati](http://www.cin.ufpe.br/%CB%9Cintegrativo))v[o).](http://www.cin.ufpe.br/%CB%9Cintegrativo))

5.1 Sampling

Data related to 21 organisms3 , together with processes and by-products related to Hcy metabolism were retrieved from the UniProt website and Ensembl4 . The ontologies GO, ChEBI and BTL2 were downloded 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 includes the proteins Methionine synthase (MS), Methylenetetrahydrofolate reductase (MTHFR), Cystathionine beta- syn- thase (CBS) , and Gamma-cystathionase (CSE). After removing records without Ensembl IDs, a final sample with 46 Hcy-related records, is 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*)

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

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

5.2 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:

1. One or more references to GO classes in UniProt annotations for GO classes, such as go:‘*Biological process*’ go:*Methylation* inclu- ded 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;

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, SNOMED CT July 31st 2014.

“main” — 2016/1/14 — page 4 — #4

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 2 illustrates the final mapping structure of GO, SNOMED CT, PR and ChEBI with BTL2.

5.3 Technical aspects for ontology grounding

The following technical aspects were applied:

*•* Top-level classes and relations of the domain ontologies were aligned manually with the upper level ontology (cf. 2) with the ProtÃ©gÃ© v5 beta 17 ontology editor;

*•* Consistency of the maps is assured with the DL reasoner HermiT 1.3.8;

*•* For performance optimization, modules of external ontologies (GO, ChEBI and PR) are created, with the classes referred in the selected database content as signature, using the ProtÃ©gÃ© plug-in Ontology Modularity (Jiang *et al.*, 2011).

*•* Database objects were subjected to ontology-inspired scrutiny with the support of spreadsheets;

*•* Prototyping was performed manually in ProtÃ©gÃ©, to test general axioms regarding ontological formalization;

*•* 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 also 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 using VBA scripts in MS Excel.

btl2:‘*particular at some time*’

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

5.4 Methodology for CQ evaluation

The ontological content is evaluated by a set of competency questions (CQs). Formulated in English by the first author, a biologist, they are sha- ped 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.

The translation of these CQs 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 (cf. section 6.4.1). To avoid biased judgements, the grounding process is assessed by means of DL 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.

5.5 Methodology for evaluation of scalability

Scalability is evaluated by artificially increasing the size of the ontology by the factors f ∈ 1, 3, 10, and 30. This is done programmatically, creating new classes suffixed by i ∈ 1, …, f. Tangledness of these experimental ontologies is guaranteed by random assignment of *i* in the axioms.

The different ontology sizes are submitted to consistency checking test in order to verify reasoning performance (in hours). A second test performed is the satisfiability time (miliseconds, ms), as it accounts for the time a CQ is validated against the derived model from the test ontology.

All tests are performed in an Intel-based core i5 3210M laptop, with

8Gbytes of RAM, Windows 10 x64 and Java JDK 8 (built 66).

**6 Results**

6.1 Basic assumptions

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

btl2:*disposition*

btl2:‘*Material*

*object*’

so:‘*sequence*

*feature*’

btl2:‘*Process*’

Each DB record implicitly introduces a series of defined subclasses

chebi:*role*

btl2:*compound*

go:*biological process*

go:*molecular function*

of a biological process class. Each of these subclasses is defined by having proteins of a certain type as well as the small molecule Hcy as participants, occur within certain cell components, are parts of certain

chebi:‘*chemical entity*’

chebi:‘*molecular entity*’

pr:*protein*

btl2:‘*poly molecular composite entity*’

btl2:‘*structured biological entity*’

btl2:*organism* go:*cellular component*

biological processes. In addition, only dysfunctional processes lead to the risk of developing the pathological phenotypes mentioned in the source.

Thus, the content of the biological databases under scrutiny is entirely expressed at class level. It makes the assumption that all these defined subclasses are non-empty, as otherwise there would not have been any

**Fig. 2.** Alignment of GO, ChEBI, SNOMED CT and PR under BTL2

Figure 2 can be summarized as follows:

*•* GO classes that are included in UniProt as annotations were aligned like: go:‘*Biological process*’ and ‘*Molecular function*’ as subclass of *Process*; and go:‘*Cellular component*’ as subclass of btl2:‘*Structured biological entity*’;

*•* Reference to *Protein* from PR mapped directly as protein names in

UniProt, and aligned with chebi:*Protein*;

*•* Small molecules included according to ChEBI. ChEBI is aligned as sublass of BTL2 *Compound*

*•* Database IDs mapping from Ensembl in UniProt, and vice-versa;

*•* Organism names as described in NCBI Taxonomy inside Ensembl and

UniProt, included as subclasses of btl2:*Organism*;

experimental evidence manifested as a curated database entry. Additio- nally, we assume that no wrong data occur (data instances that do not have any referent in reality). This interpretation allows us to refrain from reasoning about individuals, which avoid known scaling problems.

Each record from UniProt and Ensembl is interpreted as unique; one or more similar experiments may have resulted in the population of a single DB record.

6.2 Ontological Grounding

In the following, 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). 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:

“main” — 2016/1/14 — 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. IDs from UniProt and Ensembl are used only for mapping purposes

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.

*•* 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* is capable of performing one or more molecular functions (processes) *Mf* ;

*• Bp* processes 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* have dispositions to be realized by *Bp*

processes;

*•* All types of protein *P* in *O* are able to perform *Mf* processes;

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

*•* Each *Bp* referred to by in a DB record may happen within the same cellular structure in several organisms *O*, including organism-specific proteins *P* and molecules *M*. However, each DB record denotes an exclusive occurrence of *Bp*. In this sense, each DB record repre- sents 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.

*•* The DB structure is not explicit enough to connect a *Bp* subclass to a specific *Mf* process. Therefore in the definition of each *Bp* subclass the *Mf* processes are attached to the protein agent of that *Bp* subclass as possible realisations of the related disposition class (‘**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.

6.3 Ontology patterns

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

Table 3. Defined Subclasses of proteins P.

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

*P* subclassOf pr:*Protein*

*P\_sensu\_O* subclassOf *P*

Table 3 shows how proteins of any type *P* within a single database record are introduced as defined 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 subclasses 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 clas- ses type *C*1 or *C*2 or …or *Cn* are created under go:‘*Cell component*. This is due to the fact that the DB structure is not explicit enough to connect specific cell components to specific process subclasses.

Table 6 shows the axioms for *Bp\_in\_O\_with\_P\_and\_M* classes.

“main” — 2016/1/14 — page 6 — #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 processes.

Some *Bp\_in\_O\_with\_P\_and\_M* processes are dysfunctional and then bear the risk of pathological phenotypes (represented as SNOMED CT findings, ontologically btl2:*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. They are realizations of a risk (a sort of disposition) of causing the dysfunctional phenotype stated in the DB.

Following, the axioms required to represent *P\_sensu\_O* (Table 8) , i.e. the protein type of an specific organism from a DB 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 *Mf* processes specified in the DB records 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.

6.4 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.

**6.4.1 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 processes related to Hcy can be found in mice? (Cf. table 10)* 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 in DL.

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

34 classes (including ancestors) are retrieved. After filtering out the system-defined subclasses we obtain: âŁ˜amino acid betaine catabolic process’, ’blood vessel remodeling’ âŁ¦

These results are expected as they match the content represented in data.

*CQ2: Which are the proteins that exhibit ‘methyltransferase activity’? (Cf. table 11)* This query is meant to retrieve classes of proteins that are able to perform certain *Mf* processes. It is important to highlight that some proteins are capable to act in a specific way, like polymorphisms in gene MS leads to methionine synthase deficiency, which leads to higher homocysteine levels together with dysfunctional phenotypes in humans and mice.

To illustrate, we rewrite CQ2 in DL in table 11.

Table 11. Competency Question #2 in DL.

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

(‘**has realization**’ only ‘*methyltransferase activity*’))

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

– Results:

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

*•* ‘*Methionine synthase sensu Mus musculus*’;

*•* and 7 more classes.

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

*CQ3: Which are the kinds of biological processes in which proteins of the type ‘cystationine gama lyase’ participate, exhibiting ‘carbon-sulfur lyase activity’? (Cf. table 12)* This query is related to the identification of biological processes (e.g. reactions) that involve a specific protein, a 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 12)

Table 12. Competency Question #3 in DL.

‘*biological\_process*’ and

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

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

‘*carbon-sulfur lyase activity*’)))))

34 classes (including ancestors) are retrieved, of which the follow- ing are displayed after filtering âŁ˜cellular nitrogen compound metabolic process’ , âŁ˜cysteine biosynthetic process’ âŁ¦

*CQ4: Which dysfunctional biological processes entail a risk of ‘Athero- sclerosis’? (Cf. table 13)* This query retrieves biological processes that

“main” — 2016/1/14 — page 7 — #7

entail the risk of developing a dysfunctional phenotype. This query is rele- vant 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 13).

Table 13. Competency Question #4

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

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

10 classes (including ancestors) are retrieved, of which the following are displayed after filtering: âŁ˜Dysfunctional homocysteine metabolic process’, âŁ˜Dysfunctional response to interleukin 1’ âŁ¦

*CQ5: Which kinds of organisms are capable of performing cysteine biosyn- thesis? (Cf. table 14)* 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. Organisms from different species that include for which the same proteins under the same conditions are described may not include similar processes.

CQ5 is available below (table 14).

Table 14. Competency Question #5

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

(‘**has realization**’ only *‘Cysteine biosynthetic process*’)))

The following organism classes were retrieved: âŁ˜Homo sapien- sâŁ™; âŁ˜Mus musculusâŁ™, and 14 more.

*CQ6: Which proteins found in organisms of the kind âŁ˜Bos Taurus have the capability of methionine biosynthesis? (Cf. table 6.4.1)* The aim of this query is to retrieve specific proteins that are related to the process ‘*methi- nine 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 6.4.1).

Table 15. Competency Question #6.

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

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

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

x classes (including ancestors) are retrieved, of which the following are displayed after filtering: âŁ˜Betaine homocysteine S-methyltransferase 1", âŁ˜Cystathionine beta synthase’ âŁ¦ COMPLETE

**6.4.2 Computational performance**

The output of the ontology grounding process is a OWL file with the interpreted axioms propagated towards the database or database subpart. In order to evaluate the costs of our approach, we compare the generated and aligned ontologies with an artificially generated ontology with the same configuration.

Table 16 describes the reasoning performance of artificially generated files when compared to the same queries (CQ1-CQ6) presented previously. Accordingly, it is possible to verify that, under DL expressivity *ALC* , the classification time increases proportionally with classes and axioms numbers, without sacrificing satisfiability of queries and classes retrieval.

**7 Discussion**

A problem our work addresses is the formal interpretation of biological database content to enable content retrieval using a richer query paradigm. We applied this approach to proteins and related biological process of the sulphur-based amino acid hcy and demonstrated the ability of using

ontologies to interpret database, apply a formal validation to it, and use reasoning to retrieve hypothetical statements regarding proteins, proces- ses, molecular activities and dysfunctional phenotypes exhibited by some organisms.

All queries target DL Tboxes, thus avoiding the import and inclusion of individuals. Reasoning is finite and complete, in contrast to 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 whether individuals are member, as observed in biological experiments. We ontologically ground database content, submitting the generated content to a formal scrutiny. The process is guided by ontological principles and CQs written in DL.

The output is an "ontologized" database representation, which includes the particular view of the world embedded. This vision can be updated with as more queries are created. However, the ontological interpretation of the database does not entail changes, e.g. due to the fixed database organization. In other words, the interpretation given to "what is a protein" from a single database may vary only if its schema changes.

In this scenario, when reasoning supported by upper domain and domain ontologies is applied, we are able to derive queries simulating the applicability of experiments summarized in database records. Valida- tion is performed with a more realistic view of the world embedded in ontologies. In this sense, hypothetical statements derived from literature can be tested for satisfiability according to what is actually available.

The fact that typical queries (sometimes) target possibilities “*Are mem- bers of the class A able to do B*?" is addressed by two mechanisms. Firstly, the inclusion 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 (at least virtually) populated, i.e. there are database records that supports the creation of one or more classes. In this sense, the question “*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 (Lehmann, 2009; Kanehisa *et al.*, 2012; Carnielli *et al.*, 2015). The DL-Learner system Lehmann (2009)is grounded on the requirement for schema acquisition methods, as they intend to address the problem with class learning techniques. DL-Learner is designed to find logical explanations for individuals 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.

In the biomedical domain, Kanehisa *et al.* (2012) highlighted the importance of data interpretation from biomedical databases. Interpretati- ons strategies are embedded in KEGG by set of mapping operations among its internal modules, allowing the identification of organism-specific path- ways. QueryGen (Bobed and Mena, 2016) presented a system that, given user keywords, propose formal queries to retrieve data from repositories. With our approach, this is performed beforehand when the user interpret the database, and reasoning to resolve queries and misinterpretations.

The usage of DL reasoning may be considered a costly approach. However, we were able to demonstrate that in small scenarios, reaso- ning has a reasonable performance. New DL calculus methods are under development, which may decrease reasoning time (Freitas, 2011).

An improvement of our work could be 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 relatio- nal queries (Angles and Gutierrez, 2008). Our approach allow evaluating databases from the ontological level, e.g. computing class-subclass rela- tions, consistency checking and subsumption. This reduces the need to

“main” — 2016/1/14 — page 8 — #8

Table 16. Brief description of ontological sources, expressivity and reasoning performance.

Ontology Class (n) Sub. Axioms Equi. Axioms DL Classification (h) CQ1 CQ2 CQ3 CQ4 CQ5 CQ6 x1 903 1053 647 *ALC* 1998ms 6ms 9ms 15ms 17ms 20ms 20ms x3 2512 2957 1945 *ALC* 6448ms 5ms 5ms 18ms 23ms 23ms 23ms x10 8599 10268 6707 *ALC* 48259ms 12ms 14ms 19ms 20ms 21ms 22ms x30 25945 31072 20398 *ALC* 395648ms 7ms 8ms 16ms 19ms 21ms 21ms Modularized

manually filter/interpret data, without compromising the capability to be queried with SPARQL endpoints.

For instance, to retrieve a protein that has methylation capability, 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 behaves and leave the querying and computing complexity to the machine.

As mentioned by Laukens *et al.* (2015), one may argue that a reaso- ning based approach like ours generates more complexity for the user, as he needs to have some DL and ontological background, as well as the inabi- lity (from the representation) to handle updates. However, the complexity behind DL does not go too far from the basics required for relational alge- bra, like set theory. We do not consider updates a major drawback because the interpretation process rely, at the same time, on schema and content.

The complexity âŁ“ in the sense of manual creation of ontology patterns, however, appears necessary in order to produce a precise, ontology-based picture of what a database really represents, and how the informal database-ontology links are to be interpreted. The solution com- pletely refrains from representing the denoting entities (data instances), setting the full focus on the denotations, hence its fulfils its goals without ABox reasoning.

Current solutions to acquire and process life science data have mainly focused on the analysis of networks, pathways, and sequences. Recent work has been concerned with the functional analysis of data, mostly limited to syntactic approaches, without sufficiently analysing the partly implicit meaning under the surface of data structures. Several limitations need to be highlighted:

*•* Considerable scaling problems issues could be demonstrated when increasing the size of data. This can partially be mitigated by narrowing down the database content of interest, such as done in this example, where the content was filtered by one chemical entity, viz. Hcy.

*•* Another strategy for addressing computational issues would be to decrease the ontology expressiveness. The ontology patterns presented in this paper used disjunctions and value restrictions, both of which are not supported by the computationally ideal OWL EL profile. This considerably increases classification and consistency checking time. However, satisfiability of CQs were performed in a reasonable time.

*•* DL Query is simple to create, but users have to change the way cur- rent query strategies from the relational basis to a more expressive paradigm.

*•* Users may have to know how the ontologies are organized and cre- ated, as well as for creating relational queries, in order to handle query creation, even if ontologies adheres to one or more real world interpretations.

*•* Databases may adhere to one or more ontological representation in order to make clear to the user whether a record from a DB refers to an ontology class, or a given DB entry refer to an ontology class.

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. In this case,

the interpretation procedure must be recreated to include possible upda- tes. We are currently 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. A fully automated approach may hinder some onto- logical assumptions that are not able to be represented logically, or is epistemologically controversial.

**8 Conclusion**

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

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 database is extensive by nature, its organization under real world settlements certainly would be improved when accompanied by formal ontologies. Additionally, it may enable further integration capabi- lities 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 CQs for- mulated as DL queries. We query the data ontologically, without requiring any additional database processing or user intervention. If the data inter- pretation 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.

**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.

“main” — 2016/1/14 — page 9 — #9

Bobed, C. and Mena, E. (2016). QueryGen: Semantic interpretation of keyword queries over heterogeneous information systems. *Inf. Sci. (Ny).*, **329**, 412–433. Carnielli, C. M., Winck, F. V., and Paes Leme, A. F. (2015). Functional annotation

and biological interpretation of proteomics data. *Biochim Biophys Acta*, **1854**(1),

46–54.

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

*Acids Res.*, **43**(D1), D662–D669.

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

Freitas, F. (2011). A Connection Method for the Description Logic ALC. In R. Rosati, S. Rudolph, and M. Zakharyaschev, editors, *DL 2011*, Barcelona. CEUR-WS. 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.

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.

Kanehisa, M., Goto, S., Sato, Y., Furumichi, M., and Tanabe, M. (2012). KEGG for integration and interpretation of large-scale molecular data sets. *Nucleic Acids Res.*, **40**(D1), 1–6.

Laukens, K., Naulaerts, S., and Berghe, W. V. (2015). Bioinformatics approaches for the functional interpretation of protein lists: From ontology term enrichment to network analysis. *Proteomics*, pages n/a–n/a.

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

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

*LNCS*, **4900**, 133–173.

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

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

Schulz, S., Brochhausen, M., and Hoehndorf, R. (2011). Higgs Bosons, Mars Mis- sions, and Unicorn Delusions: How to Deal with Terms of Dubious Reference in Scientific Ontologies. In *ICBO 2011*, pages 183–189, Buffalo.

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: coor- dinated evolution of ontologies to support biomedical data integration. *Nature biotechnology*, **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” — 2016/1/14 — page 10 — #10