ONTOLOGIES FOR BIOMEDICINE – HOW TO MAKE AND USE THEM

SECTION I: OVERVIEW OF CURRENT APPLICATIONS OF ONTOLOGIES IN BIOINFORMATICS

Goal: *In this section, we will review current applications of ontologies in bioinformatics, with the aim of getting an implicit feel for what ontologies are and what they are used for. For each broad category of application, we will briefly describe the ontology, describe the use and then generalize that use to a broader application.*

Modern biomedical research is data-intensive and researchers seek tools to enable discovery in massive online databases. The e-science era has brought a proliferation in both data and databases, as well as an exponential growth in published literature[1]. Researchers must assimilate and integrate a growing amount of diverse information to do their work. This is clearly a challenge, and researchers are looking to computers to help them manage the information explosion[2]. In particular, they have turned to ontologies to structure their complex domain and to relate the myriad of data to shared understandings of biomedicine.

Ontologies in biomedicine span a spectrum, ranging from simple thesauri of term lists to highly expressive sources of biomedical knowledge. At the extreme of simplicity, the class names within bio-ontologies can be used as a controlled terminology for labelling data and reducing ambiguity in communication experimental results. At the other extreme, highly expressive ontologies are built containing detailed biomedical knowledge, enabling developers to create powerful computer reasoning applications that can make biomedical inferences on a massive scale. The knowledge represented in these ontologies also varies, ranging from information models and how to how to store and exchange data to encyclopaedic reference ontologies of biomedical knowledge and declarations of biomedical theory. Researchers need to be aware what domains of biomedicine are adopting ontologies and how ontologies can enable scientific work.

There are two complementary perspectives of bio-ontologies: a **content-oriented view**, concerned with the *specific ontologies* being created in biomedicine, and b) a **functional view**, dealing with *how* ontologies can be used to enable a diversity of biomedical applications. The content-oriented view has been well addressed in prior reviews [3-5], describing the activities of individuals and communities engaged in creating and improving ontologies, projects to accumulate and catalogue ontologies, and efforts to critique ontologies and to develop best practices for how ontology should be created. The functional view addresses how ontologies can be used, and can assist biomedical researcher understand the relevance of ontology to their work, as well as provide specific ideas for applications. In this review, we summarize biomedical ontology from the functional perspective, organizing the presentation according to how ontologies are used.

The range of ontology content and structure reflects the diversity of applications from the functional perspective—that ontologies are enabling a variety of types of biomedical use cases, serving as the outline for our review. Specifically, ontologies are being used in the following ways:

1. Reference for naming things
2. Representation of encyclopedic knowledge
3. Specification of information models
4. Specification of data exchange formats
5. Representation of semantics of data for information integration
6. Computer reasoning with data

We will select an example ontology to illustrate each of these use cases, expanding on each by providing:

- A general description of the ontology
- A possible application to motivate the use of the ontology
- Generalizations that may be derived based on the features of the ontology in that use case.

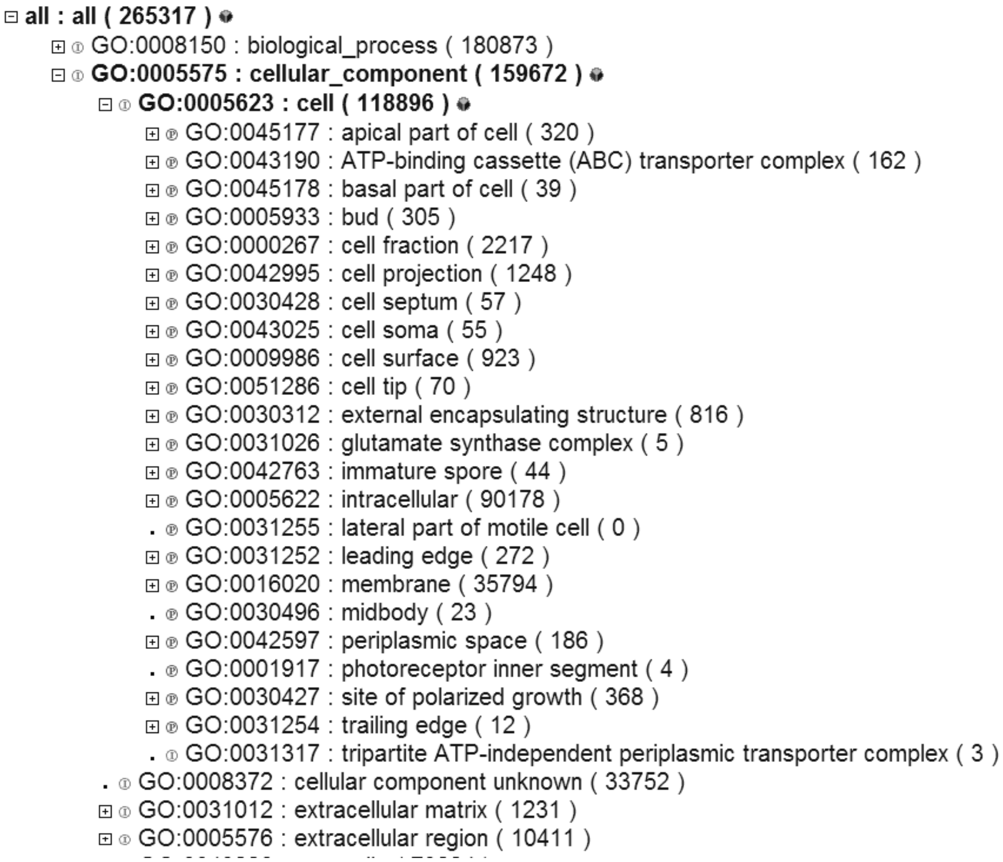
1) Reference for naming things: The Gene Ontology

The requirement of "naming things" refers to the necessity of establishing a set of controlled terms for labeling entities in databases and datasets. This is perhaps the commonest task in biomedicine to enable computers to help researchers make sense of massive online datasets and carry out their analyses. The language of biomedicine contains many synonymous terms, abbreviations, and acronyms that can refer to the same thing. For example, the process of creating glucose is referred to using a variety of synonymous terms, including "glucose synthesis," "glucose biosynthesis," "glucose formation," "glucose anabolism," and "gluconeogenesis."

It is challenging to unify diverse data sets in a consistent way when they describe similar entities that are labelled differently in different resources. An ontology provides a single name (the class name) for each entity it contains (though it can represent alternative names for that entity through the appropriate relations). The ontology can thus be used as a controlled terminology to label biomedical entities (genes, diseases, findings, etc) in a consistent way. In addition, the ontology can be augmented with terminological knowledge such as synonymy, abbreviations, and acronyms. Ontologies used in this manner enable the community to create integrated resources more easily and to contribute new terminological knowledge as the content of scientific discourse evolves.

General Description: Gene Ontology

The Gene Ontology (GO) [6] is perhaps the canonical example of an ontology created for the primary purpose of providing controlled terms for naming things. The Gene Ontology Consortium developed GO in recognition of the fact that different Model Organism Database (MODs) describe the same functions, biological processes, and cell components of gene products using different terms. In order for the MODs to describe gene products in an unambiguous manner, the Gene Ontology Consortium was established to create a standard set of names of biological entities (GO terms) and their relationships. The GO consists of three ontologies, containing entities for naming biological processes, molecular functions, and cellular components of gene products (*Figure 1*). The three ontologies provide the terms to describe what the gene products do, where and when they act, and why they perform these activities.



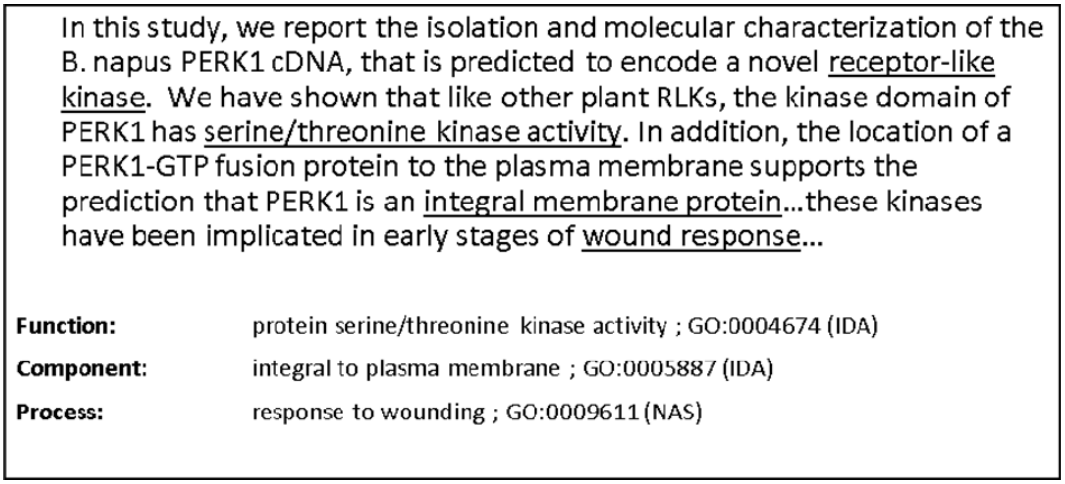
```
⊟ all : all ( 265317 )
    ⊞ GO:0008150 : biological_process ( 180873 )
    ⊟ GO:0005575 : cellular_component ( 159672 )
        ⊟ GO:0005623 : cell ( 118896 )
            ⊞ GO:0045177 : apical part of cell ( 320 )
            ⊞ GO:0043190 : ATP-binding cassette (ABC) transporter complex ( 162 )
            ⊞ GO:0045178 : basal part of cell ( 39 )
            ⊞ GO:0005933 : bud ( 305 )
            ⊞ GO:0000267 : cell fraction ( 2217 )
            ⊞ GO:0042995 : cell projection ( 1248 )
            ⊞ GO:0030428 : cell septum ( 57 )
            ⊞ GO:0043025 : cell soma ( 55 )
            ⊞ GO:0009986 : cell surface ( 923 )
            ⊞ GO:0051286 : cell tip ( 70 )
            ⊞ GO:0030312 : external encapsulating structure ( 816 )
            ⊞ GO:0031026 : glutamate synthase complex ( 5 )
            ⊞ GO:0042763 : immature spore ( 44 )
            ⊞ GO:0005622 : intracellular ( 90178 )
            . GO:0031255 : lateral part of motile cell ( 0 )
            ⊞ GO:0031252 : leading edge ( 272 )
            ⊞ GO:0016020 : membrane ( 35794 )
            . GO:0030496 : midbody ( 23 )
            ⊞ GO:0042597 : periplasmic space ( 186 )
            . GO:0001917 : photoreceptor inner segment ( 4 )
            ⊞ GO:0030427 : site of polarized growth ( 368 )
            ⊞ GO:0031254 : trailing edge ( 12 )
            . GO:0031317 : tripartite ATP-independent periplasmic transporter complex ( 3 )
        . GO:0008372 : cellular component unknown ( 33752 )
        ⊞ GO:0031012 : extracellular matrix ( 1231 )
        ⊞ GO:0005576 : extracellular region ( 10411 )
```

Figure 1 - The Gene Ontology: The Gene Ontology as displayed in the Amigo Browser (amigo.geneontology.org). The Cellular Component branch of GO is expanded, showing that it comprises a hierarchically-organized set of terms describing the components making up cells, with children being related to parent terms via is-a relations. The GO terms are used to provide a controlled terminology for annotating biomedical databases and for creating computable biomedical assertions.

The entities (represented by the terms) in the GO have *is-a* and *part-of* relations to other entities, providing the basis for representing biological knowledge. While these relations are not necessary to exploit the value of GO as a controlled terminology for names, they do enable computer reasoning applications, which can recognize that and can infer subsumption or composition by tracing the *is-a* and *part-of* relations, respectively.{OBOL}

Application

The Gene Ontology has enabled all of the Model Organism Databases (MODs) to assert the functions, processes, cellular components associated with gene products in an unambiguous manner. To make these assertions, a list of GO terms is associated with each gene product in a process referred to as “annotation[1]” (*Figure 2*).



In this study, we report the isolation and molecular characterization of the B. napus PERK1 cDNA, that is predicted to encode a novel receptor-like kinase. We have shown that like other plant RLKs, the kinase domain of PERK1 has serine/threonine kinase activity. In addition, the location of a PERK1-GTP fusion protein to the plasma membrane supports the prediction that PERK1 is an integral membrane protein…these kinases have been implicated in early stages of wound response…

Function:	protein serine/threonine kinase activity ; GO:0004674 (IDA)
Component:	integral to plasma membrane ; GO:0005887 (IDA)
Process:	response to wounding ; GO:0009611 (NAS)

Figure 2- Ontology-based Annotation: An example of use of Gene Ontology to create assertion annotations based on biomedical text. In the excerpt from biomedical text shown, the molecular function (serine/threonine kinase activity), cellular component (integral membrane protein), and biological process (wound response) of PERK1, are summarized using the appropriate GO terms from each of the three GO ontologies.

Creating such ontology-based annotations is highly valuable for both querying databases as well as analyzing high throughput data:

- **Simple queries**: researchers can search GO annotations to find all genes that have particular involved in particular biological processes, having certain molecular functions, or located in a specific cellular component.
- **GO based analysis of high throughput data:** These analyses use GO codes for gleaning biomedical insights into experimental results, and generally include the following tasks:
  - **Find over-represented GO categories** in a list of genes: If a group of genes have similar experimental results (such as sharing the same cluster in a microarray data set), the researcher can look at the GO codes associated with the genes in that cluster to establish common “biological themes” shared by that group of genes.[7]
  - **Binning**: obtain a broad view of the distribution of major GO terms in a list of genes by combining similar (but more granular) GO terms.[8]
  - **Clustering Genes** on GO terms: group together functionally related genes based on GO terms. Knowledge in the Gene Ontology guides the analysis of GO terms to perform this grouping. Gene clusters are determined by calculating an annotation-based distance between genes, taking into account all GO terms that are common to the pair and terms which are specific to each gene. The gene clusters are usually displayed using a dendrogram or a graph, based on a matrix containing the inter-gene distances.[9]

[1] The annotations created by the Model Organism Databases represent assertions about biology that hold the potential to be true for all individuals.

GENERALIZATIONS TO OTHER ONTOLOGIES AND APPLICATIONS

The use of ontologies to provide a controlled terminology for naming things has broad applications in biomedicine. In fact, many other projects are currently using ontologies precisely for this purpose:

1. **Indexing the biomedical literature**: The terms in ontologies can be used to index the literature for improving search as well as enabling text processing applications. The Medical Subject Headings (MeSH) is a terminology created by the National Library of Medicine for indexing the medical literature [10]. MeSH provides a standard set of terms that medical librarians use to describe the main topics covered in papers, the species studies, funding source, and other attributes. Originally conceived to help librarians carry out literature search, the standard names provided by MeSH have proven useful for augmenting natural language processing methods for text processing, extraction, and classification [11-14]. The of use of MeSH for providing names for biomedical entities in these applications is analogous to how names in the Gene Ontology used as annotations enable analysis of large microarray data sets: articles sharing MeSH terms have similar content, and gene products sharing GO codes have similar function, are involved in similar biological processes and are located at similar cellular locations.

2. **Integration and Access to Cancer Data**: Standard set of terms from an ontology are being used by large databases for indexing data and linking them to other resources. Cancer research and clinical practice require tight integration of large amounts of molecular and clinical data. The NCI Thesaurus, being developed by the National Cancer Institute, is desgined to integrate molecular and clinical cancer-related information [15]. It provides a controlled terminology that enables researchers to integrate, retrieve, and relate diverse data collected in cancer research. It covers topics such as cancers, findings, drugs, therapies, anatomy, genes, pathways, cellular and subcellular processes, proteins, and experimental organisms. In addition, the NCI Thesaurus represents how the entities relate to each other in a description logic framework that enables curators to maintain the integrity and extend the informational power of the terminology. The terminology enables scientists to label experimental results in a standard manner, as well as to link their research findings to disease and molecular patterns [16]. Associating research data with ontology terms also enables efficient search and retrieval of that data, by querying different levels within the ontology [17] (*Figure 3*).

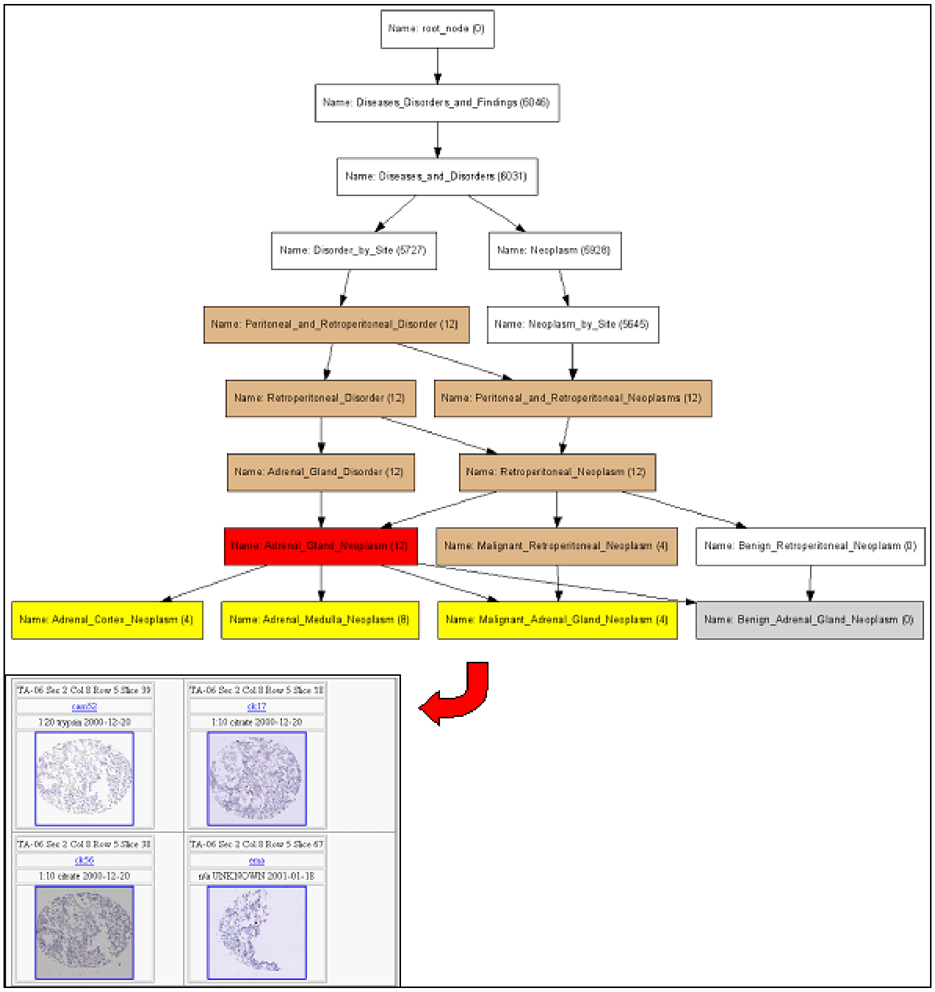

Name: root_node (0)
Name: Diseases_Disorders_and_Findings (6046)
Name: Diseases_and_Disorders (6031)
Name: Disorder_by_Site (5727)
Name: Neoplasm (6928)
Name: Peritoneal_and_Retroperitoneal_Disorder (12)
Name: Neoplasm_by_Site (5645)
Name: Retroperitoneal_Disorder (12)
Name: Peritoneal_and_Retroperitoneal_Neoplasms (12)
Name: Adrenal_Gland_Disorder (12)
Name: Retroperitoneal_Neoplasm (12)
Name: Adrenal_Gland_Neoplasm (12)
Name: Malignant_Retroperitoneal_Neoplasm (4)
Name: Benign_Retroperitoneal_Neoplasm (0)
Name: Adrenal_Cortex_Neoplasm (4)
Name: Adrenal_Medulla_Neoplasm (8)
Name: Malignant_Adrenal_Gland_Neoplasm (4)
Name: Benign_Adrenal_Gland_Neoplasm (0)
TA-06 Sec 2 Col 8 Row 5 Slice 39
cam52
1:20 trypsin 2000-12-20
TA-06 Sec 2 Col 8 Row 5 Slice 18
ck17
1:10 citrate 2000-12-20
TA-06 Sec 2 Col 8 Row 5 Slice 38
ck56
1:10 citrate 2000-12-20
TA-06 Sec 2 Col 8 Row 5 Slice 67
ema
n/a UNKNOWN 2001-01-18

Figure 3 Querying at different levels using an ontology: The figure shows a zoomed in region of the directed acyclic graph view resulting from searching for the term Adrenal gland neoplasm. The red node is the term that has been searched for and then clicked by the user, the yellow nodes are the child terms that have at least one sample in the database assigned to that term, grey nodes are child terms with no corresponding samples in the database and burlywood nodes are parent terms with less than 50 samples. Samples can be retrieved for the selected node.

3. **Encoding Clinical Data in Electronic Medical Records:** A common use of ontologies in clinical medicine is to provide a common way to describe patient information in health records. A comprehensive source for clinical terminology is the Unified Medical Language System, a large collection of biomedical vocabularies developed by the US National Library of Medicine [18]. The UMLS integrates many different biomedical vocabularies, such as the NCBI taxonomy, the Systematized Nomenclature of Medicine (SNOMED), and the International Classification of Disease (ICD). Many of the source terminologies in UMLS are being used in health information systems to provide terms for patient findings, diagnoses, laboratory results, and pathologies [19].

4. **Standardizing the Language for Describing Biomedical Images**: Biomedical images are a challenging type of information to exploit in large repositories because their contents are not explicit. Ontologies can be useful for providing names for the anatomy, pathology, and observations in images. By associating

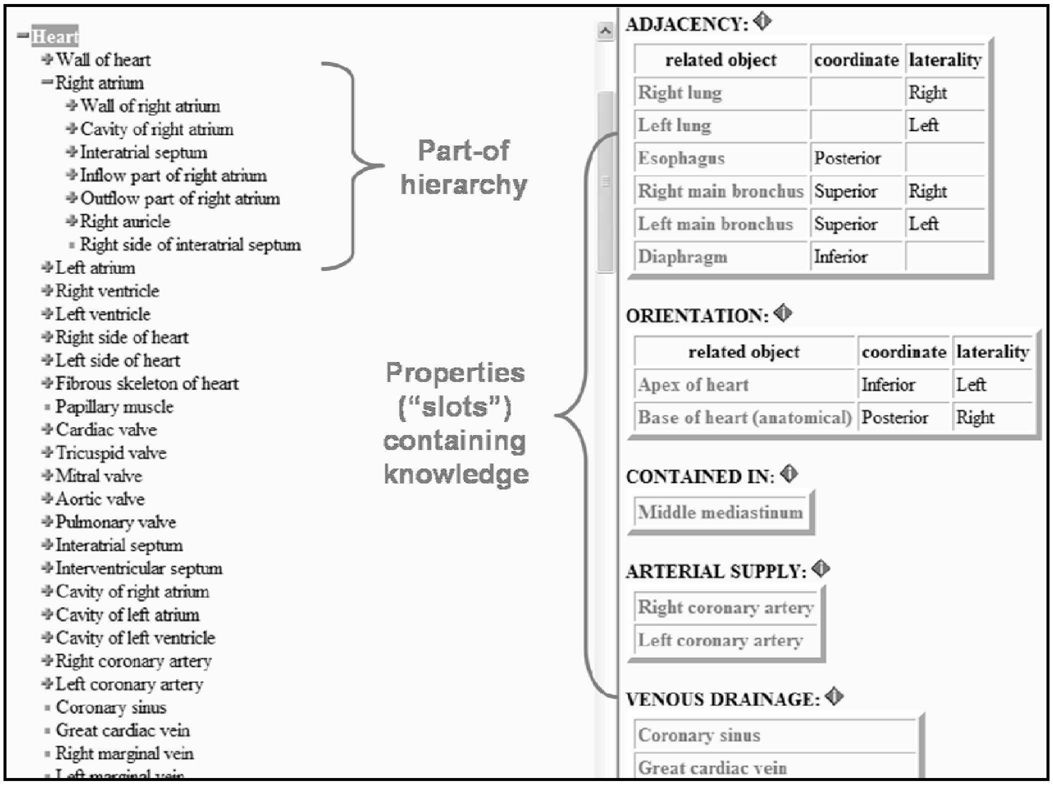
these names with images (or regions within images), it is possible to analyze large image repositories in terms of these explicit image characteristics. Several ontologies are in development to provide terms for naming entities associated with images. RadLex is a controlled terminology for radiology, providing terms for the techniques, findings, and diseases associated with medical images [20]. The Biomedical Informatics Research Network (BIRN) is creating ontologies to provide the necessary names for interrelated concepts contained in images as well as in distributed online databases [21]. The Open Microscopy Environment (OME) is an open source platform for image microscopy that coordinates the organization, storage, and analysis of the large volumes these data, enabled by the standard terms being drawn from UMLS [22]. The use of ontologies for naming entities in the images permits these projects to unify image data and non-image data, as well as streamline search in large image repositories.

2) Representation of encyclopedic knowledge

Using bio-ontologies as a source for standardized names is perhaps the simplest use of this technology, but it does not utilize the expressive power of ontologies for representing knowledge through rich relationships. Many applications need to access the knowledge-rich content of biomedicine. Many textbooks have been written to describe the components making up living systems (the entities) and how they work and interact with other components (the relations). Describing complex knowledge in texts makes that knowledge accessible to humans, but not to machines. Ontologies are increasingly being used to structure and make explicit encyclopedic biomedical knowledge in a form that is accessible to both researchers and machines.

General Description: Foundational Model of Anatomy

The Digital Anatomist Foundational Model of Anatomy (FMA) [23] is a comprehensive ontology of human anatomy. The FMA contains more than 70,000 entities that describe the elements of canonical human morphology, providing declarative descriptions of detailed anatomic structures. It is a “reference ontology” in that it specifies canonical knowledge for the domain of anatomy, in the form of a comprehensive set of entities and a large set of rich relationships (*Figure 4*).



- Heart
  - Wall of heart
  - Right atrium
    - Wall of right atrium
    - Cavity of right atrium
    - Interatrial septum
    - Inflow part of right atrium
    - Outflow part of right atrium
    - Right auricle
    - Right side of interatrial septum
  - Left atrium
  - Right ventricle
  - Left ventricle
  - Right side of heart
  - Left side of heart
  - Fibrous skeleton of heart
  - Papillary muscle
  - Cardiac valve
  - Tricuspid valve
  - Mitral valve
  - Aortic valve
  - Pulmonary valve
  - Interatrial septum
  - Interventricular septum
  - Cavity of right atrium
  - Cavity of left atrium
  - Cavity of left ventricle
  - Right coronary artery
  - Left coronary artery
  - Coronary sinus
  - Great cardiac vein
  - Right marginal vein
  - Left marginal vein

ADJACENCY:

related object	coordinate	laterality
Right lung		Right
Left lung		Left
Esophagus	Posterior	
Right main bronchus	Superior	Right
Left main bronchus	Superior	Left
Diaphragm	Inferior	

ORIENTATION:

related object	coordinate	laterality
Apex of heart	Inferior	Left
Base of heart (anatomical)	Posterior	Right

CONTAINED IN:

Middle mediastinum

ARTERIAL SUPPLY:

Right coronary artery
Left coronary artery

VENOUS DRAINAGE:

Coronary sinus
Great cardiac vein

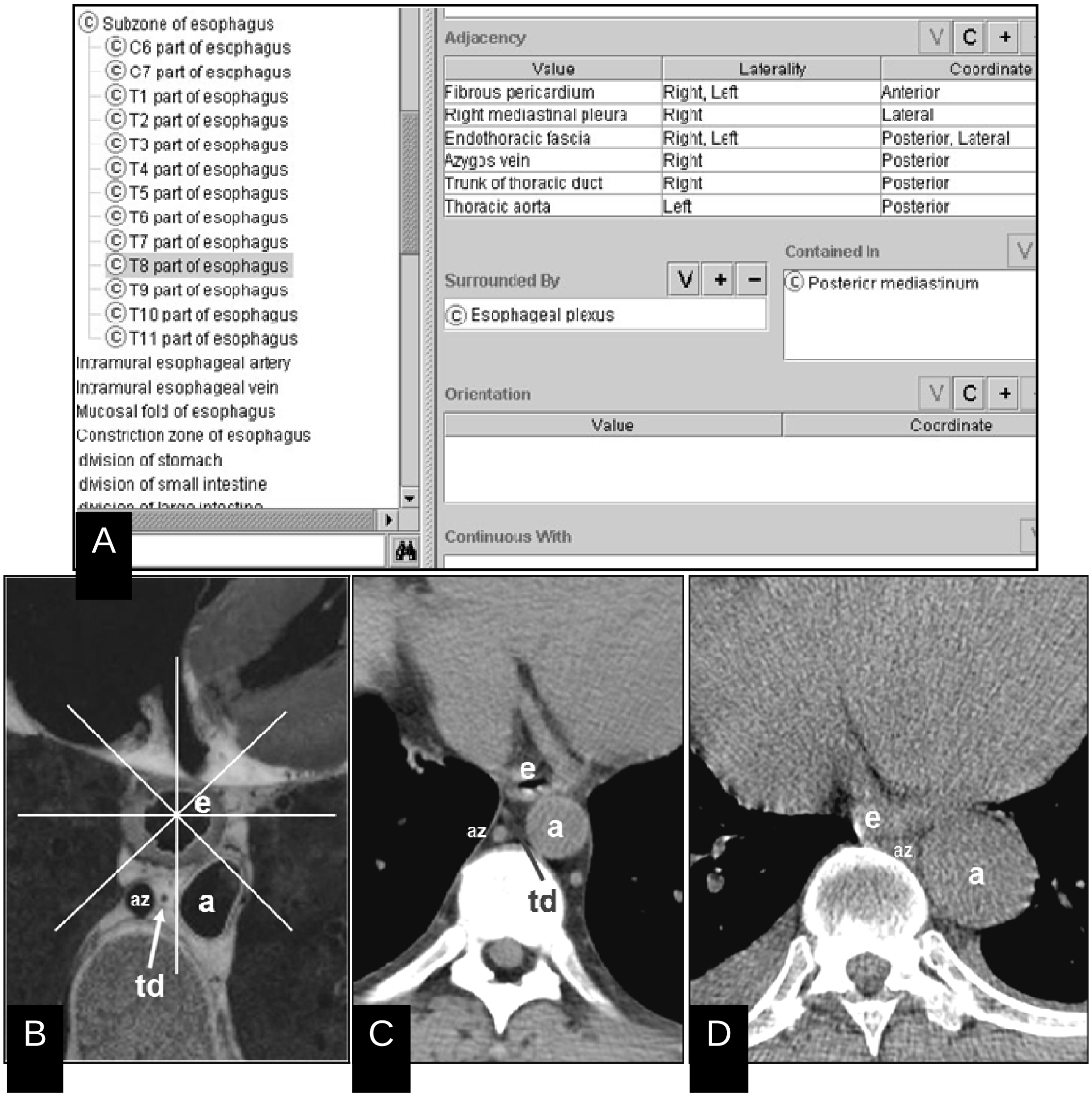
Figure 4- Foundational Model of Anatomy: The FMA is an ontology representing detailed anatomic knowledge. A screen shot of the FMA (accessible by the FM Explorer on the Web at http://fme.biostr.washington.edu:8089/FME/index.html) shows that anatomic knowledge is modeled by specifying a large set of rich relations among the anatomic entities. For example, it can be seen that the heart (left) has many relationships to other entities in the FMA (right), such as adjacency, orientation, containment, and vascular supply. Specifically, FMA tells us that the heart is contained in the middle mediastinum, and that it is supplied by left and right coronary arteries.

The FMA was created through disciplined modeling of the structural organization of the human body in collaboration with anatomists and knowledge engineers. It was not created with a particular application in mind; rather, the goal was to provide an electronically-accessible encyclopedic reference for anatomic knowledge.

APPLICATION

The knowledge in FMA can be exploited in applications that require more detailed information about entities beyond their name. Software applications that need anatomical knowledge about particular organs can access FMA as a "reference ontology," looking up entities and their relations in FMA to determine canonical facts about anatomic structures needed by an application, such as organ composition, continuity, and adjacency (*Figure 4*).

For example, an application could be created to use anatomic reference knowledge in the FMA to help radiologists interpret imaging studies, informing them about the anatomic structures affected by abnormalities in adjacent organs (*Figure 5*). Radiological imaging interpretation is a problem area where anatomic knowledge is needed, but access to that information is limited, because anatomy is incompletely visualized in imaging procedures. Some anatomic structures are not visible in radiology images due to limited spatial resolution or to individual patient characteristics (Compare *Figure 5*C and *Figure 5*B). A software application can use FMA to recognize that small anatomic structures such as the thoracic duct are adjacent to larger, visible structures such as the esophagus (*Figure 5*A), and inform the radiologist that an abnormality such as a mass in the esophagus may be affecting the adjacent thoracic duct, even though the latter is not visible in the radiographic image (*Figure 5*D). Such detailed anatomic knowledge is useful in informing practitioners about diagnostic possibilities that might have been overlooked.



- Ⓒ Subzone of esophagus
  - Ⓒ C6 part of esophagus
  - Ⓒ C7 part of esophagus
  - Ⓒ T1 part of esophagus
  - Ⓒ T2 part of esophagus
  - Ⓒ T3 part of esophagus
  - Ⓒ T4 part of esophagus
  - Ⓒ T5 part of esophagus
  - Ⓒ T6 part of esophagus
  - Ⓒ T7 part of esophagus
  - Ⓒ T8 part of esophagus
  - Ⓒ T9 part of esophagus
  - Ⓒ T10 part of esophagus
  - Ⓒ T11 part of esophagus
- Intramural esophageal artery
- Intramural esophageal vein
- Mucosal fold of esophagus
- Constriction zone of esophagus
- division of stomach
- division of small intestine
- division of large intestine

Adjacency

Value	Laterality	Coordinate
Fibrous pericardium	Right, Left	Anterior
Right mediastinal pleura	Right	Lateral
Endothoracic fascia	Right, Left	Posterior, Lateral
Azygos vein	Right	Posterior
Trunk of thoracic duct	Right	Posterior
Thoracic aorta	Left	Posterior

Surrounded By: Ⓒ Esophageal plexus

Contained In: Ⓒ Posterior mediastinum

Orientation

Value	Coordinate

Continuous With

Figure 5- Using ontologies to enable knowledge-based applications: Among the rich relations contained in the FMA is knowledge about anatomic adjacency—specifications about which anatomic structures are adjacent. Knowledge about adjacency can be used by a computer application to determine which anatomic structures in the vicinity of abnormality may be affected. (A) A portion of FMA in Protégé showing that detailed adjacency information is represented; in particular, it can be seen that at the T8 level of the esophagus (highlighted class in left panel), the thoracic duct is located to the right and posterior to the esophagus (value for "adjacency slot" shown in right panel). (B) an image from Visible Human at the T8 level of esophagus showing how adjacency in FMA is established using a relative

coordinate system (td=thoracic duct, az=azygous vein, e=eshophagus, a=aorta; this cross section is viewed from below, such that the left side of the patient is on right side of the image). (C) An axial Computed Tomography (CT) scan at the same level as (B) showing similar adjacency relations as represented in (A). (D) In this CT scan in a different patient from (C), the thoracic duct is not visible, but its presence and location can be deduced from the FMA (A), and this knowledge used to infer that it may be affected by an abnormality (such as a mass) in the adjacent esophagus.

It has been previously shown that FMA can be useful as a reference knowledge source to predict the anatomic consequences of penetrating injury [24]. In that work, an application was developed to deduce all the anatomic structures that could be injured consequent to penetrating trauma—whether those structures were directly in the path of injury or very close to it. In order for these inferences to be made, the software application queried FMA to find the classes associated with organs that were directly on the path of injury as well as those adjacent to it, informing care takers about organs that are injured as well as potentially injured. Such inferences are made directly from the canonical anatomic knowledge contained in the FMA reference ontology. Another application of FMA's rich knowledge about the anatomic structure of biological systems is as a reference ontology for organizing other biomedical information, including normal and abnormal functions. [25, 26].

GENERALIZATIONS TO OTHER ONTOLOGIES AND APPLICATIONS

The use of ontologies as a reference for applications to obtain knowledge has general applicability to many biomedical domains, ranging from helping basic research, to assisting with medical decision support and teaching. For example, Dameron and colleagues developed a numeric and symbolic representation of brain cortex anatomy as a reference ontology [27]. Their ontology and knowledge base could be reused in various application contexts such as teaching, decision support in neurosurgery, and sharing of neuroimaging data for research purposes. Similar to FMA as described above, their reference ontology can drive a broad range of applications that would require detailed knowledge about brain morphological features—that knowledge being obtained by querying the various relations connecting classes in the ontology.

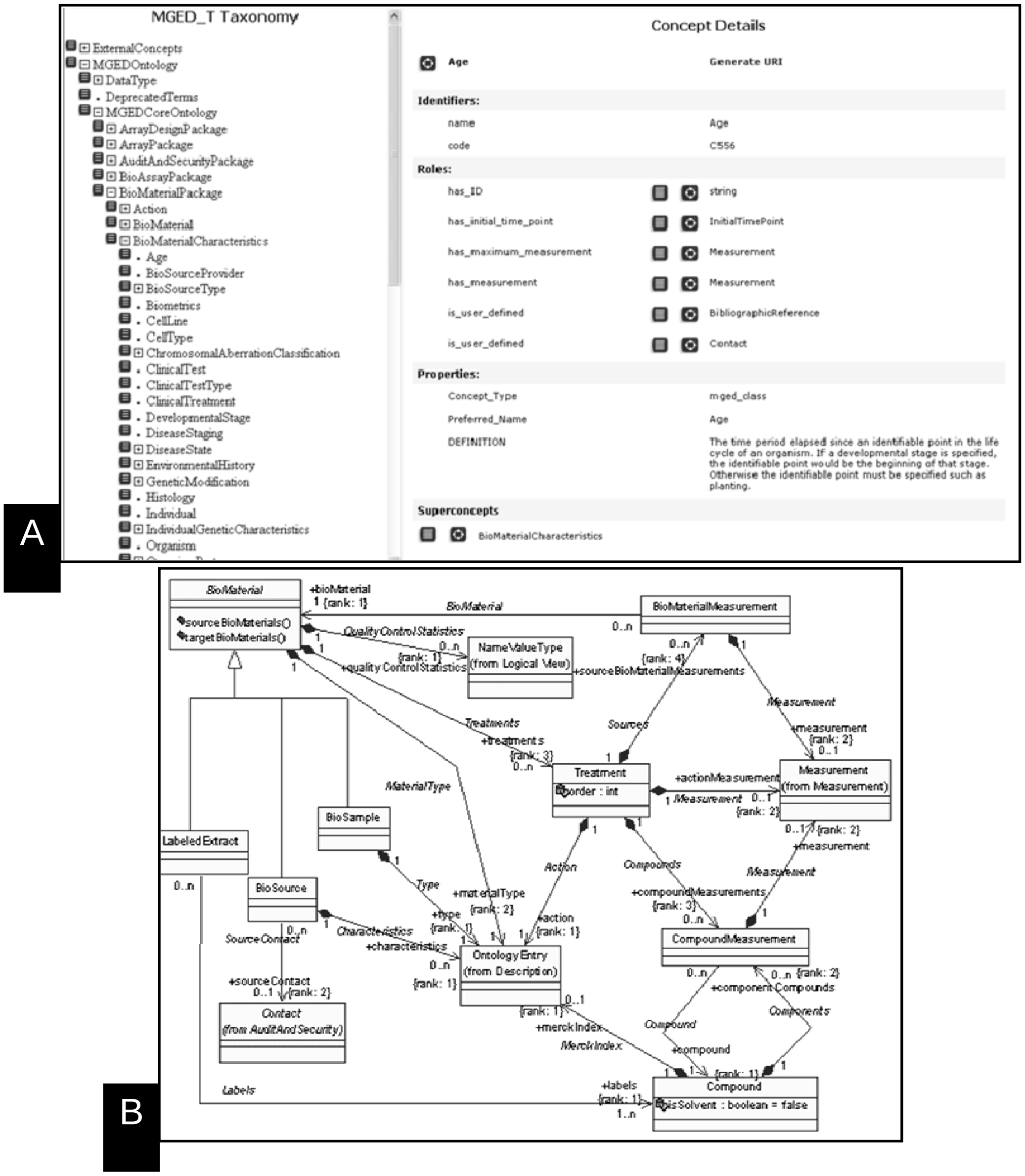
In recent work Kahn and colleagues, created a reference ontology for radiology imaging procedures [28]. This reference ontology contains detailed knowledge about how radiology imaging procedures are performed and the types of images that are produced. Their work demonstrated that diverse intelligent applications could be created using the same ontology as a knowledge source.

3) SPECIFICATION AN INFORMATION MODEL (DATABASE/KNOWLEDGEBASE SCHEMA)

Information models outside of the biomedical domain are rarely specified using ontologies; often UML models, entity-relation diagrams, or database schema diagrams are used. However, the use of ontology for the building of models of biomedical information and databases offers several advantages. First, ontologies provide an explicit specification of the terms used to express information in the biomedical domain. Secondly, ontologies enable additional capabilities, such as making relationships among data types in databases explicit, and supporting automated reasoning, such as deducing subsumption among classes. In addition, complex database and knowledgebase schemas are often viewed in a more intuitive manner in ontologies, especially since some ontology tools such as Protégé (http://protege.stanford.edu) contain visualization tools that enable developers to create graphical visualizations of schemas [29]. A particular benefit of using ontologies for creating information models is the ability to reuse existing ontologies by inclusion in creating those models. Finally, representations of information models using ontologies can be published on the Semantic Web if the ontology is in the format of the Web Ontology Language (OWL) [30].

General Description: MAGE-OM, MAGE-ML, MGED Ontology

Microarrays are currently a very popular experimental method being used to generate molecular-level biomarkers for a variety of biological states and medical diseases. The creation of large amounts of microarray data and the creation of databases to enable sharing of these data quickly raised the need for standards in describing microarray experiments and results. The MIAME standard specifies the minimum information needed to describe a microarray experiment, and the Microarray Gene Expression Object Model (MAGE-OM) and markup language MAGE-ML provide a mechanism for standardized representation of microarray data for data exchange [31]. The MGED Ontology is being created to provide a common terminology and an information schema for annotating microarray experimental results (*Figure 6*). The MGED Ontology provides terms for annotating all aspects of a microarray experiment -- including the design of the experiment and array layout, the preparation of the biological sample and the methods used to analyze the data – as well as provides a structure for relating these aspects with one another.



MGED_T Taxonomy

- ⊞ ExternalConcepts
- ⊟ MGEDOntology
  - ⊞ DataType
  - DeprecatedTerms
  - ⊟ MGEDCoreOntology
    - ⊞ ArrayDesignPackage
    - ⊞ ArrayPackage
    - ⊞ AuditAndSecurityPackage
    - ⊞ BioAssayPackage
    - ⊟ BioMaterialPackage
      - ⊞ Action
      - ⊞ BioMaterial
      - ⊟ BioMaterialCharacteristics
        - Age
        - BioSourceProvider
        - ⊞ BioSourceType
        - Biometrics
        - CellLine
        - CellType
        - ⊞ ChromosomalAberrationClassification
        - ClinicalTest
        - ClinicalTestType
        - ClinicalTreatment
        - DevelopmentalStage
        - DiseaseStaging
        - ⊞ DiseaseState
        - ⊞ EnvironmentalHistory
        - ⊞ GeneticModification
        - Histology
        - Individual
        - ⊞ IndividualGeneticCharacteristics
        - Organism

Concept Details

Age	Generate URI
Identifiers:	
name	Age
code	C556
Roles:	
has_ID	string
has_initial_time_point	InitialTimePoint
has_maximum_measurement	Measurement
has_measurement	Measurement
is_user_defined	BibliographicReference
is_user_defined	Contact
Properties:	
Concept_Type	mged_class
Preferred_Name	Age
DEFINITION	The time period elapsed since an identifiable point in the life cycle of an organism. If a developmental stage is specified, the identifiable point would be the beginning of that stage. Otherwise the identifiable point must be specified such as planting.
Superconcepts	
BioMaterialCharacteristics	

Figure 6- Using ontologies for specifying an information model: (A) This figure shows the MGED Ontology (http://mged.sourceforge.net/ontologies/MGEDontology.php). This ontology contains terms that are relevant for describing the design of an experiment, and are used in specifying information models for storing and exchanging microarray data. (B) This figure demonstrates an example information model (MAGE-OM) specified using terms from the MGED Ontology, the latter providing the semantics of entities in the information model.

MAGE-ML is an XML-based markup language that is derived from the MAGE object model, MAGE-OM. MAGE-ML is used to describe and communicate information about microarray based experiments among researchers and microarray databases. MAGE-ML describes microarray designs, microarray manufacturing information, microarray experiment setup and execution information, gene expression data and data analysis results. MAGE-ML is developed and described using the Unified Modeling Language (UML). MAGE-OM is the primary information model for describing microarray experiments.

APPLICATION

In case of microarray data, not having knowledge of the context in which the experiment is done severely limits the ability to interpret the numbers in the data file. The MAGE-OM, MAGE-ML, and MGED Ontology address the need to specify experimental context by providing a representation of the information models that convey this information. The MAGE-OM describes the information model, and the MAGE-ML conveys the instance data (*Figure 6*). MGED Ontology is used to provide applications with the terms needed to annotate or query microarray data, especially by biologists who may have little knowledge of the ontology structure.

The key advantages of using ontology for specifying information models in the microarray community are to:

- provide standard terms for annotation of microarray experiments when they are submitted to public repositories
- enable unambiguous descriptions of how microarray experiments are performed
- enable structured queries of elements of the experiments, enabling use of MGED ontology to expand queries using the subsumption relations in the ontology structure.

Community microarray database resources such as Stanford Microarray Database, Array Express, and the Gene Expression Omnibus currently require that the submitters of microarray data provide the minimum set of information declared by the MIAME standard.

GENERALIZATIONS TO OTHER ONTOLOGIES AND APPLICATIONS

The need to represent information models for structured capture of experimental context (under which the experiment was performed), as well as to convey the actual results of experiments to researchers and databases, is not unique to microarray studies. Many other biomedical areas of work could benefit from similar ontology-based structured approaches to describing their data. A large group of researchers have come together to create the Ontology of Biomedical Investigation (OBI), which is creating an ontology for the description of biological and medical experiments and investigations. The ontology will enable the consistent annotation of biomedical investigations. It will model the design of an investigation, the protocols and instrumentation used, the material used, the data generated and the type analysis performed on it; enabling data exchange between researchers and databases similar to the ontology efforts being undertaken in the microarray community.

4) SPECIFICATION OF A DATA EXCHANGE FORMAT: BIOPAX

Exchanging data is vital in several biomedical domains where multiple different, yet related databases have emerged over time and they wish to share and link their data. For example, several pathway databases have arisen that contain rich information about metabolic, signal transduction and gene regulatory pathways from particular species. Ontologies, by specifying data exchange formats, can greatly facilitate the process of sharing data amongst biomedical resources such as pathway databases.

General Description

Knowledge about biomedical pathways is central to scientific research. There are more than 200 biomedical databases contain information pertinent to pathways. BioPAX is an emerging format for sharing pathways that aims to provide a standard for representing metabolic, biochemical, transcription regulation, protein synthesis, and signal transduction pathways[32]. Pathways in BioPAX are composed of a set of interactions. The top-level BioPAX definition of pathway is general enough to capture the many kinds of pathways used by biologists; figure 7 shows the different kinds of pathway information that can be represented in BioPAX. Pathways in the BioPAX format can be represented as objects using the web ontology language (OWL), which is a language recommended by the World Wide Web consortium (W3C) to create an ontology. OWL can be expressed in RDF/XML syntax or using the N3 triple format.

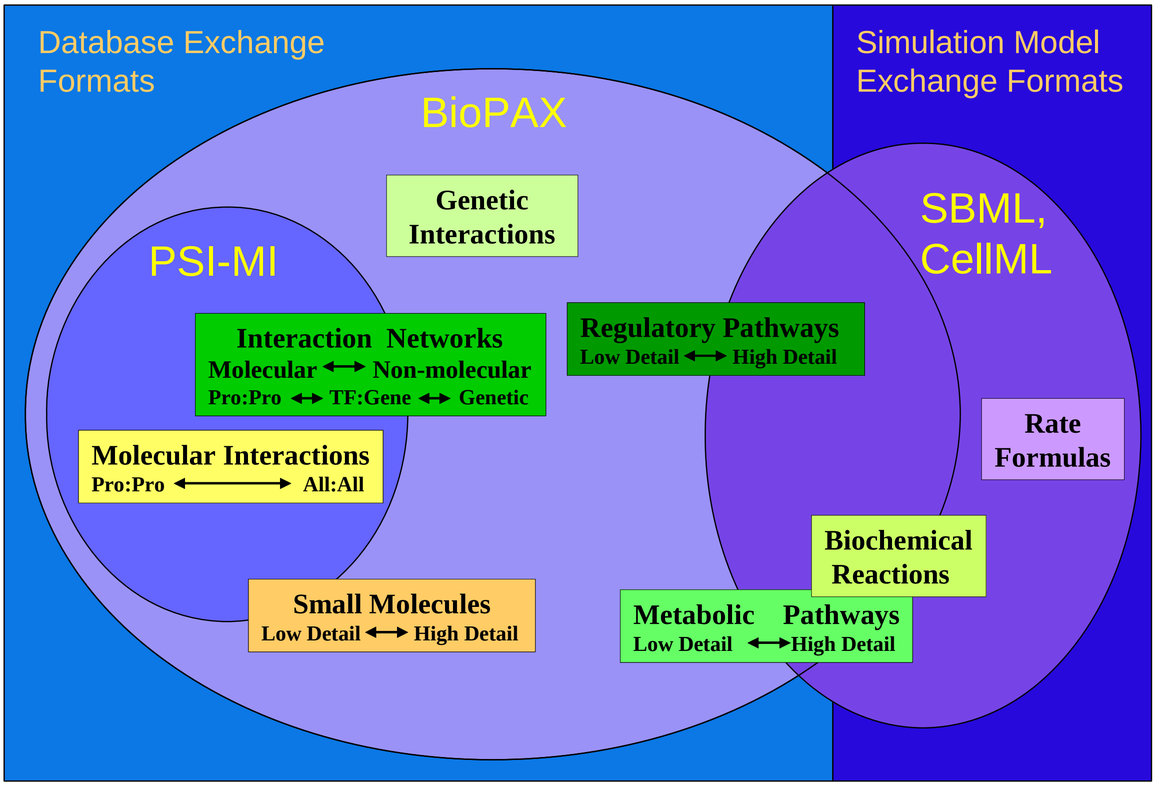


Figure 7- Using ontologies for specifying a data exchange format: The figure shows the scope of BioPAX and the kinds of pathways it aims to represent. At the left are simple binary interactions and at the right are complex models of biological reaction systems that are used for simulation. (Figure courtesy of Gary Bader and Mike Carey)

Application

Currently, leading pathway resources such as the Kyoto Encyclopedia of Genes and Genomes (KEGG)[33], BioCyc[34], and Reactome[35] make their data available in BioPAX format and BioPAX viewers are available as additional modules in pathway analysis tools such as PATIKA[36] and Cytoscape[37]. This provides an opportunity to construct unified pathway resources such as the Pathway Knowledge Base project (PKB) that enables querying across different species and across multiple pathway resources simultaneously. It also enables comparison of the degree of complementary across different pathway sources.

Generalizations to other ontologies and applications

From the ontological perspective, BioPAX is a very “light” ontology, but therein lays its power. It allows diverse information sources to speak in a simple common language enabling interoperability. In future such common exchange formats will be needed for increasingly exchanging complex artefacts such as simulation models.

5) Representation semantics of data for information integration: TAMBIS

The amount of biomedical data available online is enormous, and biomedical discovery commonly occurs by integrating related, yet diverse data from different sources. Performing such integration is laborious, and it does not scale in the setting of the current information explosion.

Ontologies can streamline the process of integrating and accessing data across diverse resources. As described earlier, ontologies provide a means to make the semantics of a domain explicit by providing rich relations among its entities. Specifying the semantics of data in a variety of databases can enable researchers to integrate heterogeneous data across different databases. While it might be simplest to link objects with the same name (syntactical equivalence) in different databases, this is not necessarily a good approach, because names of biological entities (e. g. genes, proteins, pathways) are not the same across databases.

A more robust approach to integrating data is to link based on shared characteristics of biological entities in databases. Ontologies provide a common declarative foundation for describing the content of biomedical databases. Computer reasoning programs can be applied to ontologies to determine if two objects in different databases refer to the same biomedical entity. Thus, an ontology-based framework can facilitate the exchange, integration and validation of information.

General Description

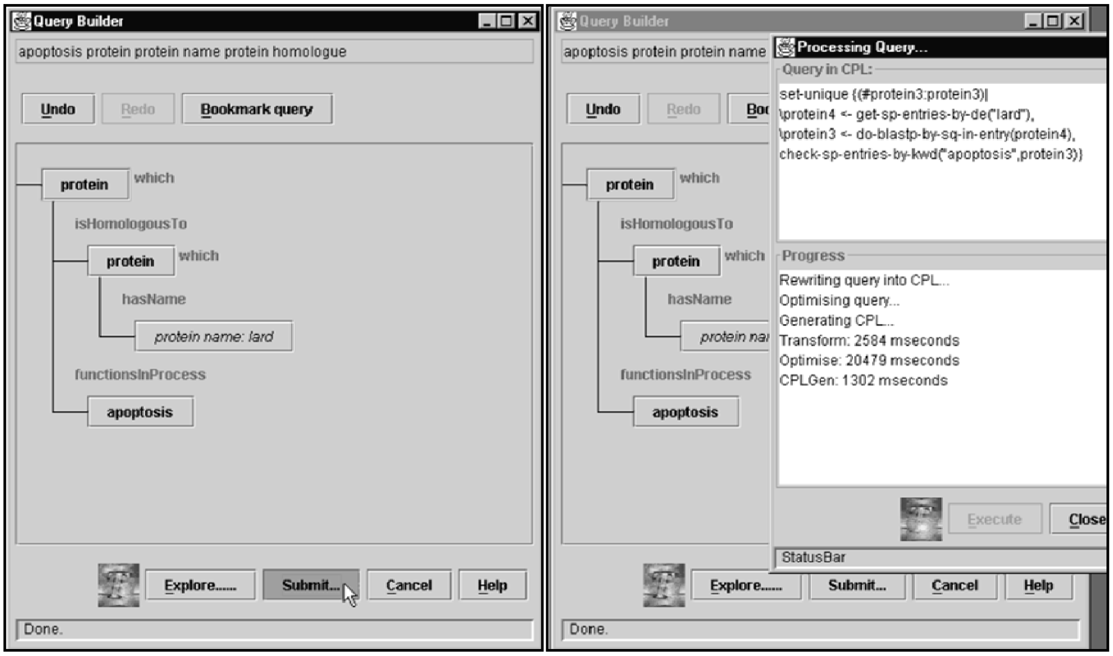
TAMBIS is a project that aimed to provide transparent access to disparate biological databases and analysis tools, enabling users to access and virtually integrate a wide range of biomedical resources [38]. Their system includes an ontology (the TAMBIS ontology[2]), a knowledge base of biological terminology (the biological Concept Model), a model of the underlying data sources (the Source Model) and a user interface. The Concept Model provides the user with the concepts necessary to construct queries, and shields the user from the details of the various database sources. The Source Model provides a description of the underlying sources and mappings between terms used in the sources and terms in the biological Concept Model. The TAMBIS ontology serves as a single access point for multiple biological information sources. Queries are phrased in terms of the ontology, and TAMBIS converts them to requests to appropriate sources.

Application

The creators of TAMBIS developed an application that uses the TAMBIS ontology to enable users to formulate a query across a set of diverse biomedical sources, providing a means to virtually integrate

[2] http://www.ontologos.org/%5COntology%5CTAMBIS.htm

these resources (*Figure 8*). Source-independent, declarative queries formed from terms in the Concept Model are transformed into a set of source dependent, executable procedures.



```
set-unique {(#protein3:protein3)|
\protein4 <- get-sp-entries-by-de("lard"),
\protein3 <- do-blastp-by-sq-in-entry(protein4),
check-sp-entries-by-kwd("apoptosis",protein3)}
```

```
Rewriting query into CPL...
Optimising query...
Generating CPL...
Transform: 2584 mseconds
Optimise: 20479 mseconds
CPLGen: 1302 mseconds
```

Figure 8- Using ontologies for integrating data resources: Screen shots from the TAMBIS application are shown. A user query is formulated in terms of entities from the TAMBIS ontology (left). In the figure, the query illustrated is: Find proteins which are homologous to the lard protein and functions in the apoptosis biological process." This query is based on a overarching Concept Model that subsumes all the information models reflected in the databases that the TAMBIS system covers. The Concept Model-based query is translated into appropriate database-specific queries by the application, issued to each appropriate database, and the results collected and returned to the user (right).

The query process in the application proceeds in the following phases:

1. **Query formulation**: the user formulates a query in terms of concepts & relationships in the TAMBIS ontology, constructing a concept describing information of interest using ontology terms. The output of this phase of the application is a source-independent ontology-based conceptual query.
2. **Query transformation**: The application examines the ontology terms comprising the query to identify biomedical database sources needed to answer query, and it then constructs a query plan tailored to the requirements of each source database.
3. **Query execution**: The application submits the individual queries to the relevant source databases and collects the results, returning them to the user. The TAMBIS developers created wrappers for each source database so the latter can be accessed in syntactically consistent manner.

GENERALIZATIONS TO OTHER ONTOLOGIES AND APPLICATIONS

The need to integrate diverse data is great in biomedicine, and the methods pioneered in the TAMBIS project are being applied in other domains in biomedicine. Researchers part of the Biomedical Informatics Research Network (BIRN) have recently been undertaking activities to use ontologies to integrate image- and non-image data pertaining to the neurosciences [21]. BIRN employs a "mediator architecture" to link multiple databases together, each maintaining their specific local schema, into an accessible federated platform. When a researcher queries the BIRN database, the query is processed by the mediator, which uses ontologies to relate and integrate the various source databases. The mediator parses the user query and subsequently submits database-specific queries to the relevant data sources. The approach is thus similar in principle to the approach used by TAMBIS. However, BIRN is currently working on enriching the knowledge contained in the ontology [39], migrating the ontology to OWL, and creating defined classes as well as rich relations that will provide the knowledge necessary for computer reasoning with the integrated data sources (see section on Computer Reasoning below) [40].

6) Computer reasoning with data: Hybrow

Perhaps one of the most compelling advantages ontologies can provide in helping researchers exploit the vast amounts of biomedical knowledge available in electronic form is *computer reasoning*. Computer reasoning encompasses methods that use ontologies to make inferences based on the knowledge they contain as well as any additional contextual information or asserted facts. There is a tremendous amount of biomedical knowledge currently available in online databases and in the published literature. As a result, on the one hand there is an abundance of individual data types such as gene and protein sequences, gene expression data, protein structures, protein interactions and annotations. On the other hand there is a shortage of tools and methods that can handle this deluge of information and allow a scientist to draw meaningful inferences.

Currently, a significant amount of time and energy is spent in merely locating and retrieving information rather than thinking about what that information means. It is extremely difficult to integrate current knowledge about biological systems and formulate hypotheses (or "Models") spanning a large number genes and proteins [41]. It is difficult to determine whether the hypotheses are consistent internally or with data, to refine inconsistent hypotheses and to understand the implications of complicated hypotheses [42]. It is obvious that this situation needs to be rectified and tools need to be developed that utilize formal methods to query and interpret the information at hand [43]. As suggested by Fedoroff et al, we need *tools for thought*: formal representational systems appropriate for representing models of biological systems, and computational tools that can manipulate, check, and use these models to make predictions and form explanations.

General Description

HyBrow (Hypothesis Browser) is a system for the representation, manipulation and integration of diverse biological data - such as gene expression, protein interactions & annotations - with prior biological knowledge for the purpose of evaluating alternative hypotheses. Hybrow's purpose is to evaluate and rank hypotheses based on user-defined 'rules', and consistency with all information available to it.[44]

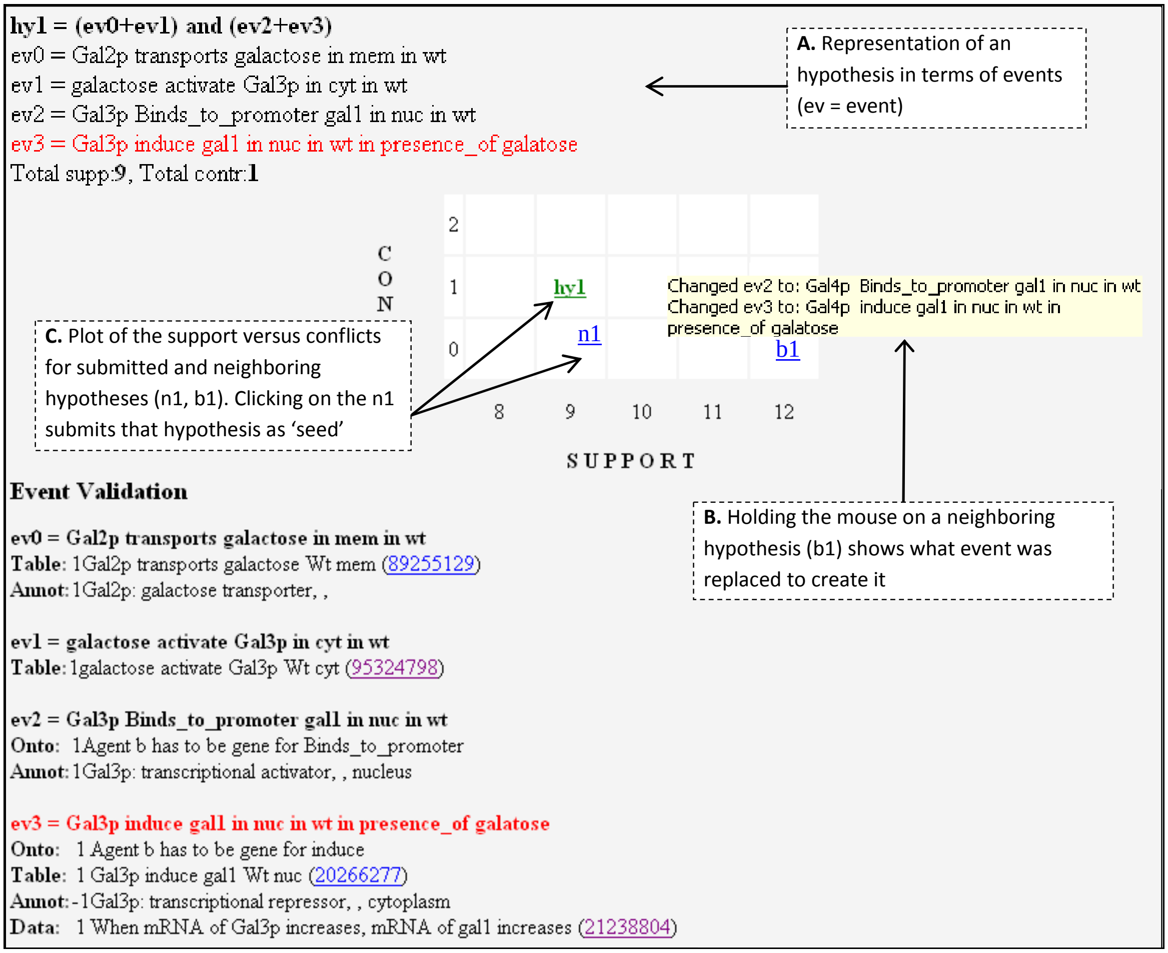
Application

HyBrow consists of an event-based ontology for representing hypotheses about biological processes at different levels of detail, a knowledgebase that stores diverse biological information sources such as gene expression, protein interactions & annotations, and programs to perform hypothesis design and evaluation.

- Ontology for representing hypotheses: The ontology used in HyBrow allows the representation of knowledge about the Galactose Metabolism in yeast (GAL system) in a manner compatible with the event-based conceptual framework used for reasoning within HyBrow[45]. An event consists of an acting agent (the "subject," such as gene, RNA, protein), a target agent (the "object," such as a gene, protein, complex), a relationship (the "verb," such as induce, repress, bind), a context in which the event takes place, and an optional set of associated conditions (such as the presence or absence of other agents) which accompany the event. The construction of events from elements of the ontology, event sets from events, and hypotheses from event sets is governed by a context-free grammar. Events that occur in the same context are combined to form *event sets* and an *hypothesis* consists of event sets linked by logical and temporal operators. An hypothesis must contain at least one event set, which must contain at least one event [44]. Contexts specify *where* events occur in the cell and *under what genetic conditions* they occur. The *contexts* are derived from established ontologies. For example, terms for specifying physical locations in the cell come from the cellular component division of the Gene Ontology. The current hypothesis ontology allows representation of events such as: 'Gal4p binds to the promoter of the gal1 gene in the presence of galactose in wild type *S. cerevisiae*'.

- HyBrow's knowledge base stores the different finds of information as well as existing knowledge about the GAL system. The knowledgebase accommodates available literature, curated primarily from YPD[46] at a coarse level of resolution. The knowledgebase was populated by manual curation using loading forms like the EcoCyc database[25] as well as PERL scripting to access the existing public repositories (such as the Saccharomyces Genome Database) to retrieve desired information.

- Hypothesis design and evaluation: There are two interfaces for constructing hypotheses: The diagrammatic interface allows users to draw diagrams using a visual notation constructed in accordance with proposed conventions[27], which are then automatically translated into hypotheses. The widget interface allows the user to construct hypotheses using subject/verb/object selection menus. Hypotheses are saved to local files and then submitted for evaluation via the web. When HyBrow receives an hypothesis, it checks the connections between events and event sets for conformity with the hypothesis grammar. If the hypothesis passes these tests for syntax, each event is then checked for validity using the appropriate rule corresponding to the relationship proposed in the event. For each event, a support, conflict or cannot comment result is returned. Finally, the support and conflict calls are tallied based upon the logical structure of the hypothesis and presented to the user in a web interface [Figure 9]



hy1 = (ev0+ev1) and (ev2+ev3)
ev0 = Gal2p transports galactose in mem in wt
ev1 = galactose activate Gal3p in cyt in wt
ev2 = Gal3p Binds_to_promoter gal1 in nuc in wt
ev3 = Gal3p induce gal1 in nuc in wt in presence_of galatose
Total supp:**9**, Total contr:**1**

A. Representation of an hypothesis in terms of events (ev = event)

Changed ev2 to: Gal4p Binds_to_promoter gal1 in nuc in wt
Changed ev3 to: Gal4p induce gal1 in nuc in wt in presence_of galatose

C. Plot of the support versus conflicts for submitted and neighboring hypotheses (n1, b1). Clicking on the n1 submits that hypothesis as 'seed'

B. Holding the mouse on a neighboring hypothesis (b1) shows what event was replaced to create it

Event Validation

ev0 = Gal2p transports galactose in mem in wt
Table: 1Gal2p transports galactose Wt mem (89255129)
Annot: 1Gal2p: galactose transporter, ,

ev1 = galactose activate Gal3p in cyt in wt
Table: 1galactose activate Gal3p Wt cyt (95324798)

ev2 = Gal3p Binds_to_promoter gall in nuc in wt
Onto: 1Agent b has to be gene for Binds_to_promoter
Annot: 1Gal3p: transcriptional activator, , nucleus

ev3 = Gal3p induce gall in nuc in wt in presence_of galatose
Onto: 1 Agent b has to be gene for induce
Table: 1 Gal3p induce gal1 Wt nuc (20266277)
Annot: -1Gal3p: transcriptional repressor, , cytoplasm
Data: 1 When mRNA of Gal3p increases, mRNA of gal1 increases (21238804)

Figure 9- Using ontologies for computer aided reasoning: The figure shows a screen shot from HyBrow's result page that illustrates the evaluation of a simple hypothesis: "Gal2p transports galactose into the cell at the cell membrane. In the cytoplasm, galactose activates Gal3p. Gal3p binds to the promoter of gal1 gene and induces its transcription in the presence of galactose." This hypothesis was decomposed into events (shown in Figure 8A). In its assessment, HyBrow reported support from literature and GO annotation for event number 0 (ev0); support from literature for ev1; support from ontology constraints and annotation for ev2; and support from the ontology, literature and data divisions for ev3. HyBrow discovered a conflict for ev3 (marked in red) from the annotation rule division since Gal3p is annotated to be primarily in the cytoplasm in presence of galactose. HyBrow then searched for variant events. For ev2 it found an event (Gal4p binds to promoter of gal1) with higher support and for ev3 it found the more meaningful event (Gal4p induces gal1 in nucleus in wt in presence of galactose) with the same support but no conflict. These events were inserted in place of the original events to create a neighbouring hypothesis that is better than the original hypothesis (Figures 8B and 8C). HyBrow is able to present to the user its rankings, explanations for them, and references to conflicting and supporting data in a summary page.

GENERALIZATIONS TO OTHER ONTOLOGIES AND APPLICATIONS

The approaches to using computer reasoning with biomedical knowledge with ontologies can be generalized to other problem areas, though the underlying methods remain similar. Dameron and colleagues developed a method for managing the semantic consistency of an ontology of brain-cortex anatomy that exploits the explicit representation of knowledge in ontologies to support the reasoning process [47].

The FMA ontology of anatomy mentioned earlier has been used in a reasoning application to deduce the physiological consequences of injury to the arteries supplying the heart [48]. In this work, knowledge about which arteries supply different regions of the heart was represented using an OWL ontology, in which the parts of the heart were defined classes, based on specification of the arteries supply each heart region. Connectivity among different arterial segments was specified via continuity relations, and the concepts of arterial occlusion and heart ischemia were defined. A computer reasoning service was implemented that posed the problem of inferring heart injury as a classification problem, based on asserting arterial injuries, classifying the resulting ontology, and reading off newly-classified anatomical entities that reflected the inferred heart injuries [48].

Another way computer reasoning with ontologies has been generalized to other domains is encoding classification criteria in explicit ontology-based form. Criteria for classification abound in biomedicine, and they are generally applied by hand to case data. Ontologies can be used to represent the classification criteria explicit using logic formalisms that some ontology languages provide, such as OWL. In recent work, for example, the classification criteria for interpreting mammography images was represented using OWL to enable creation of applications that automatically classify the interpretation of these studies [49].

In theory, computer reasoning could be applied to knowledge bases created from natural language processing (NLP) methods applied to the biomedical literature to help researchers make sense of it. At the current time, most work has focused on using ontologies to guide NLP according to pre-determined knowledge models or to infer ontologies from text. The GeneWays project developed a knowledge model that enables analysis of signal-transduction pathways in eukaryotes [50]. This system uses the ontology as a knowledge model to analyze the interactions between molecular substances, integrating information extracted from multiple sources by NLP techniques to infer a consensus view of molecular networks. The application provides automatic retrieval of signal transduction data from electronic versions of scientific publications using natural language processing techniques. It also provides visualization and editing of representations of regulatory systems. As the accuracy of information extraction in such systems improves, we will likely see computer inference applications with the biomedical literature.

Discussion

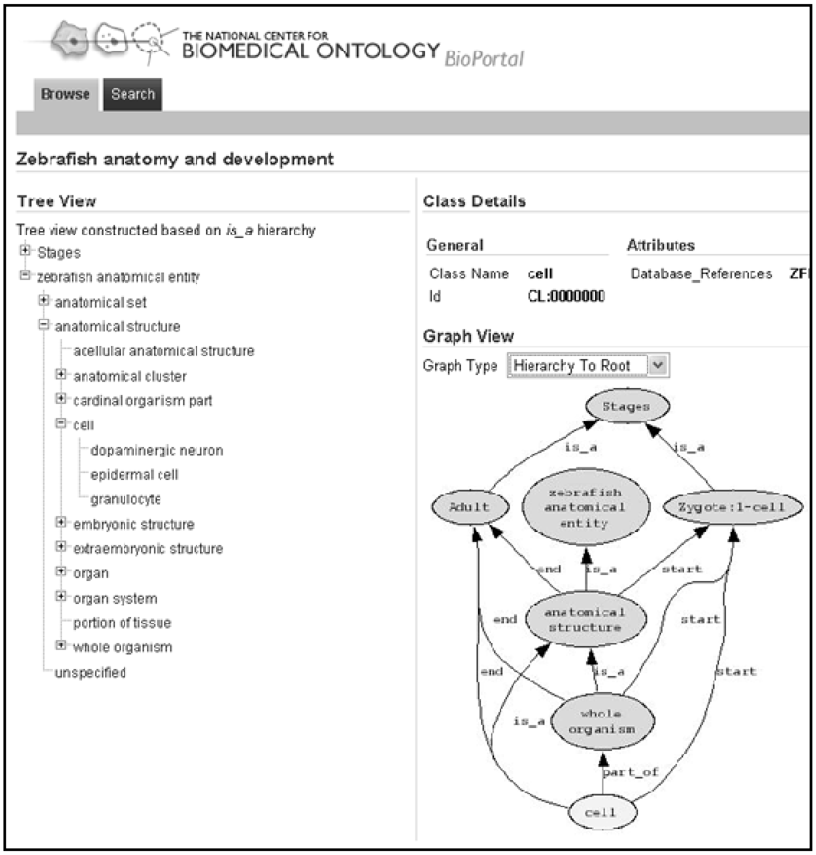
In this part of the tutorial, we have argued that it is useful to survey biomedical ontologies from a functional perspective—in terms of how they are used. We believe that this approach is helpful to those coming into the field to get a sense for the spectrum of potential use cases and to recognize opportunities for ontology to help with their particular use case. However, researchers and potential ontology developers may face challenges in getting started using ontologies in their work.

Finding ontologies

The first challenge is that one needs to find existing ontologies that may be suitable for a project, or to select among available ontologies. Over time, the number of ontologies has expanded tremendously. While a few years ago, researchers in particular biomedical domains needed to keep track of one or two ontologies, currently the number of ontologies pertinent to researchers has expanded greatly. Even if an ontology is in a different domain, it could contain much content that is similar to other domains; for example, mouse anatomy is relevant to workers in the human anatomy domain because there are similarities among many anatomic structures. The proliferation in biomedical ontologies with little concomitant infrastructure development to enable the community to access them has fragmented the field, and the biomedical community is finding it difficult to effectively access and use these valuable knowledge resources.

The National Center for Biomedical Ontology has recently created BioPortal, a Web portal to biomedical ontologies that provides users and software agents comprehensive access to a virtual library of ontologies [51]. The BioPortal ontology library contains over 50 ontologies, including those from the biological and medical domains. The BioPortal Ontology Library unifies ontologies and provides a common access mechanism to ontologies regardless of their original format.

The BioPortal enables users to browse the ontology library to quickly find groups of ontologies related to a variety of domains of interest as well as enables searching of the BioPortal ontology library. Once particular ontologies of interest are located, individual ontologies can be viewed as an expandable tree or graph view (*Figure 10*).



Zebrafish anatomy and development

Tree View

Tree view constructed based on *is_a* hierarchy

- Stages
- zebrafish anatomical entity
  - anatomical set
  - anatomical structure
    - acellular anatomical structure
    - anatomical cluster
    - cardinal organism part
    - cell
      - dopaminergic neuron
      - epidermal cell
      - granulocyte
    - embryonic structure
    - extraembryonic structure
    - organ
    - organ system
    - portion of tissue
    - whole organism
  - unspecified

Class Details

General

Class Name cell

Id CL:0000000

Attributes

Database_References ZF

Graph View

Graph Type Hierarchy To Root

Figure 10- BioPortal: Ontology access and visualization: In BioPortal, ontologies are shown both as an expandable tree (left) as well as a local neighborhood graph (right; selected class is highlighted in yellow).

The BioPortal ontology library services are provided to external clients as a suite of Web services, enabling developers to access functionality such as ontology categorization and description, graphical ontology browsing, and term search in their applications. This Web portal approach appears promising for unifying and disseminating ontology content, while reducing community fragmentation and providing them the tools needed to exploit these rich resources.

Key Points

1. There is a growing community interested in using and producing biomedical ontologies.

2. The diversity of ontologies and their content can bewilder those not already deeply familiar with the field; it is helpful to organize ones thinking about bio-ontologies from a functional perspective.

3. Ontologies are used in biomedicine for naming entities, providing a reference to encyclopedic knowledge, specifying information models, data exchange formats, and semantics of data for information integration, and enabling computer reasoning with biomedical data.

4. The growth in biomedical ontologies has created new paradigms for people to work with them, as well as created the need for new tools to enable collaborative ontology development.

SECTION II: ONTOLOGIES – WHAT THEY ARE AND WHAT THEY ARE NOT

Goal: *In this section, we will discuss "ontology" as understood in philosophy, computer science and information science to explain how the computer/information science meanings are different - but related to - the philosophical meaning of ontology. In practice, ontologies provide standardized labels which are used to annotate different experimental data. We will discuss the implications of this annotation-based view of ontology to clarify the difference between terminologies, taxonomies, application ontologies and reference ontologies.*

WHAT IS AN ONTOLOGY?

Ontology means different things to different people and we spend a considerable amount of time reconciling them. Here are the most common claims to the "meaning" of ontology.

Philosophy: Ontology is the study of what entities and what types of entities exist in reality. [Alt - An ontology is a declaration of the entities & relationships that can exist in a portion of reality (which is of interest to us)]

AI: An ontology is a explicit specification of concepts & relationships that can exist in a domain of discourse

Formal Methods: An ontology is the statement of a logical theory

CS: an ontology is a data model that represents a domain and is used to reason about the objects in that domain and the relations between them

The common ground is that an ontology is a specification of entities (or concepts), relations, instances and axioms in an area of study. Though it fun to argue about these different points of view, in practice that quite pointless unless the question of: **What is an Ontology for and what are you going to do with it,** is answered. Several artifacts are collectively referred to as "ontologies", the figure below illustrates that there is a continuous spectrum from glossaries at one end and general logic at the other.

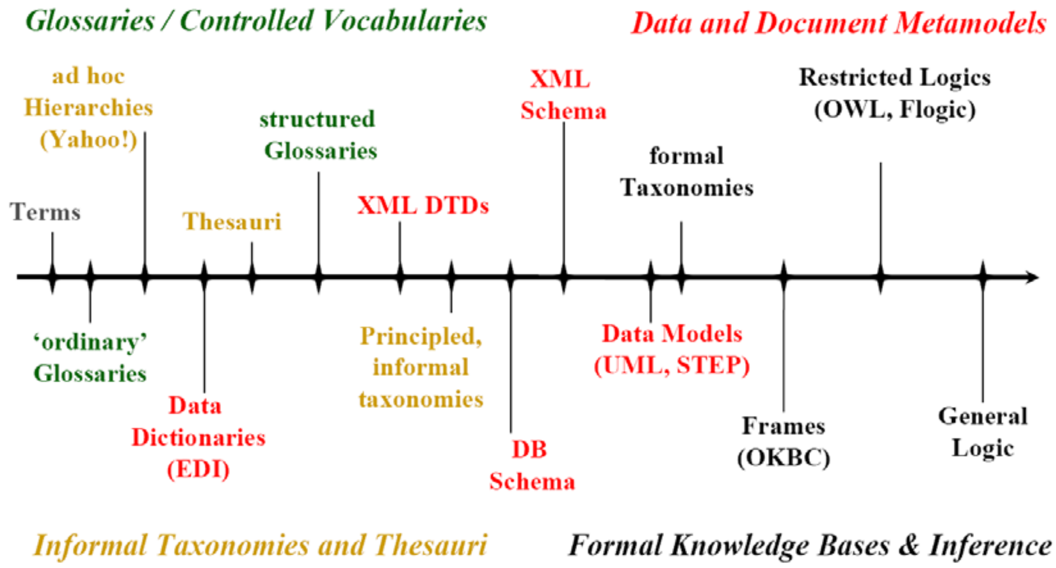

Glossaries / Controlled Vocabularies
Data and Document Metamodels
ad hoc Hierarchies (Yahoo!)
structured Glossaries
XML Schema
Restricted Logics (OWL, Flogic)
formal Taxonomies
Terms
Thesauri
XML DTDs
'ordinary' Glossaries
Principled, informal taxonomies
Data Models (UML, STEP)
Data Dictionaries (EDI)
DB Schema
Frames (OKBC)
General Logic
Informal Taxonomies and Thesauri
Formal Knowledge Bases & Inference

Figure 11: the spectrum of knowledge artefacts. (Originally by Michael Uschold, used with permission)

Application Ontology vs. Reference Ontology

A reference ontology is analogous to a scientific theory, it consists of representations of biological reality which are correct according to our current understanding.

An application ontology is a software artifact for structuring data according to some hierarchy of classes, for the purpose of managing and manipulating that data, supporting interoperability of various resources.

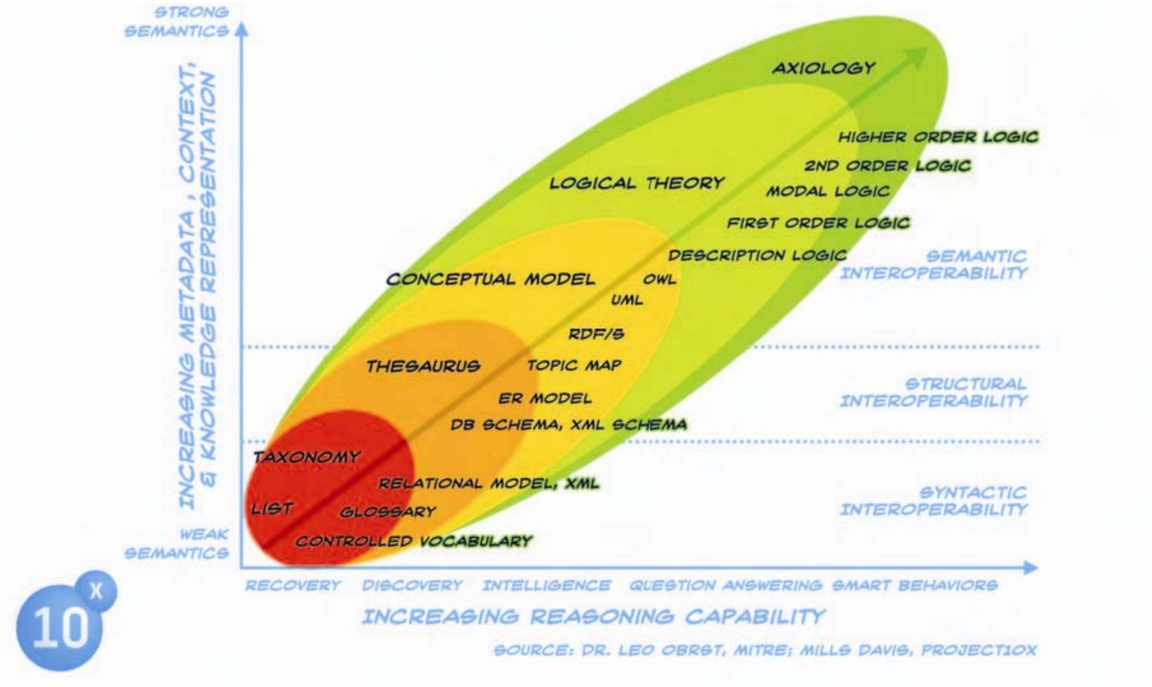
We believe that as far as possible, one should focus on developing scientific information models, data-models, and process-models etc that are as close as possible to and that refer to *reference ontologies*.

What an ontology is NOT

- An ontology is not the same as a knowledgebase:
  - Ontology (types) + Instances = KB
- An ontology is not the same as a database schema
  - A database schema is designed to store the instances conforming to an ontology
- An ontology is not the same as an XSD
  - An XSD tells you how to structure the information that describes the instances

Tradeoff between semantic rigor and complexity of reasoning

While it is tempting to be as rigorous as possible while creating an ontology, we would like to stress again that the key question is: **What is an Ontology for and what are you going to do with it.** Developing knowledge structures with rigorous semantics is hard, time consuming and expensive. The need to do that has to be weighed against the complexity of reasoning that is required for the pertinent use case. For example, if the end goal is database cross-linking then it is probably a waste of time to build in strong semantics that support modal logics. The figure below, illustrates the balance between the complexity of reasoning possible and the semantics needed for it



SOURCE: DR. LEO OBRST, MITRE; MILLS DAVIS, PROJECT10X

Figure 12: Relationship between semantic rigor and complexity of reasoning. The red zone indicates the most 'bang for the buck' zone where there is maximum gain with the least effort. Developing strong semantics that support first order logic (or more expressive logics) has to be justified by demonstrating a strong need for that.

Section III: Building ontologies using OWL

Goal: *We will provide an overview of the various constructs available in OWL and how they can be used to represent biomedical knowledge in an ontology. We will demonstrate an example of how DL-reasoners can be used to verify ontologies created in OWL and how such reasoning will help in reducing errors in biomedical ontologies. We will provide indications of some of the problems facing ontology developers using OWL in the bioinformatics domain.*

Weaving the
Biomedical semantic w

Participants are requested to read the included published paper, *weaving the Biomedical semantic web with protégé-owl*, for details on this topic

Weaving the Biomedical Semantic Web with the Protégé OWL Plugin

Holger Knublauch Olivier Dameron Mark A. Musen

Stanford Medical Informatics, Stanford University, Stanford, CA (`http://protege.stanford.edu`)

Abstract

In this document we show how biomedical resources can be linked into a Semantic Web using Protégé. Protégé is a widely-used open-source ontology modeling environment with support for the Web Ontology Language (OWL). With the example domain of brain cortex anatomy we demonstrate how Protégé can be used to build an OWL ontology and to maintain ontology consistency with a description logic classifier. We also show how Protégé can be used to link existing Web resources such as biomedical articles and images into a Semantic Web.

INTRODUCTION

Biomedical Web resources in the existing internet are mainly optimized for use by humans. For example, researchers need to know the "correct" keywords to do a meaningful search using an online publications database. The vision of the Semantic Web [3] is to extend the existing Web with conceptual metadata that are more useful to machines, revealing the intended meaning of Web resources. This meaning could be used by software agents to perform tasks that are difficult with the current Web architecture. For example, an intelligent agent could retrieve semantically related publications, even if they don't contain the "correct" keyword.

Ontologies are a central building block of the Semantic Web. Ontologies define domain concepts and the relationships between them, and thus provide a domain language that is meaningful to both humans and machines. Ontologies are being defined for many biomedical domains, such as anatomy, genetics, and cancer research. The concepts from these ontologies can be used to annotate Web resources. The Web Ontology Language (OWL) [13] is widely accepted as the standard language for sharing Semantic Web contents.

Protégé [4, 7] is an ontology development environment with a large community of active users. Protégé has been used for more than a decade to build large-scale biomedical applications. Rather recently, Protégé has been extended with support for OWL, and has become one of the leading OWL tools.

Our goal in this document is to help biomedical projects get started with Semantic Web technology. We first describe the architecture of a typical biomedical Semantic Web application from the domain of brain cortex anatomy. Then we give a short overview of Protégé and its OWL support. We describe how Protégé can be used to define domain classes and properties, and how to use features such as a classifier to maintain semantic consistency. We also briefly introduce the essential features of OWL and their representation in Protégé. Then we show how to link existing Web resources into the Semantic Web, so that they can be accessed by intelligent agents. We end this document with discussion and conclusions.

A BIOMEDICAL SEMANTIC WEB

The current Internet already contains vast amounts of biomedical information resources, such as research articles, images, clinical guidelines, and drug catalogues. Making these resources available in a more structured way is one of the goals of several large-scale ontology development efforts. For example, the goal of the National Cancer Institute's Thesaurus project [5] is to provide a well-defined conceptual model so that cancer-related resources can be structured in a machine-readable way. This conceptual model is an OWL ontology with tens of thousands of classes and dozens of properties.

For the purpose of this paper, we start with a less ambitious example ontology of brain cortex anatomy. Potential use cases of this ontology are teaching, decision support for clinical practice, sharing of neuroimaging data, or semantic assistance for data processing tools. The ontology defines concepts such as `FrontalLobe` and `LeftCentralSulcus`, and specialization, composition and spatial neighborhood relationships. In addition, the ontology also defines the logical characteristics of the concepts. For example, it states that a brain `Hemisphere` is composed of exactly five distinct lobes: one `FrontalLobe`, one `ParietalLobe`, one `TemporalLobe`, one `OccipitalLobe` and one `LimbicLobe`. These concepts and relationships are implemented as OWL classes and properties. They are stored in an OWL file which resides on a publicly accessible Web server. After the ontology has been published on the Web, other OWL ontologies, resources, agents, and services can link to this file and use the ontology's concepts. For example, a Web repository of MRI scans could provide a collection of image metadata objects that would represent the attributes of the single scans (dimensions, resolution, contents), so that the best images for a specific topic can be retrieved automatically. If the image repository is loosely coupled and distributed over multiple hosts (e.g., multiple hospitals), then each of the servers could provide its own metadata objects. A user searching for a particular scan of a frontal lobe could then invoke an intelligent agent that would crawl through the various repositories to search for the best matches.

Another example of a Semantic Web application would be a context-sensitive search function for research articles. A publication database such as PubMed could provide a Web service that would refer to a conceptual model when providing metadata about articles. It could also rely on this conceptual model to guide and assist query processing. Users could invoke this Web service through a simple client application. The Web service could exploit the definitions from the ontology to widen or narrow the search into concepts that are substantially related to the terms the user has asked for. For example, it could deliver papers about glioma located in the precentral gyrus although the user has only asked for tumors of the frontal lobe, exploiting the background knowledge that a glioma is a kind of tumor and that the precentral gyrus is a part of the frontal lobe.

One of the advantages of shared conceptual models is that they can be reused in various contexts, even some that have not been imagined yet. Finally, the Semantic Web could even be used to point researchers and domain experts into new directions and to reveal cross-links between domains.

These examples illustrate the central role of *ontologies* in Semantic Web applications. Ontologies should adequately represent a domain and allow some kind of formal reasoning. They should be both understandable by humans and processable by software agents. Furthermore, since ontologies will evolve over time, they need to be maintainable. This demands for ontology modeling tools that provide a user-friendly view on the ontology and support an iterative working style with rapid turn-around times. Tools should also provide intelligent services that reveal inconsistencies and hidden dependencies among definitions.

PROTÉGÉ AND THE OWL PLUGIN

Since its beginning in the 1980's, Protégé has been driven by biomedical applications. Protégé started as a rather specialized tool for a specific kind of problem solving [4], but evolved into a very generic and flexible platform for many types of knowledge-based applications and tools from all kinds of domains.

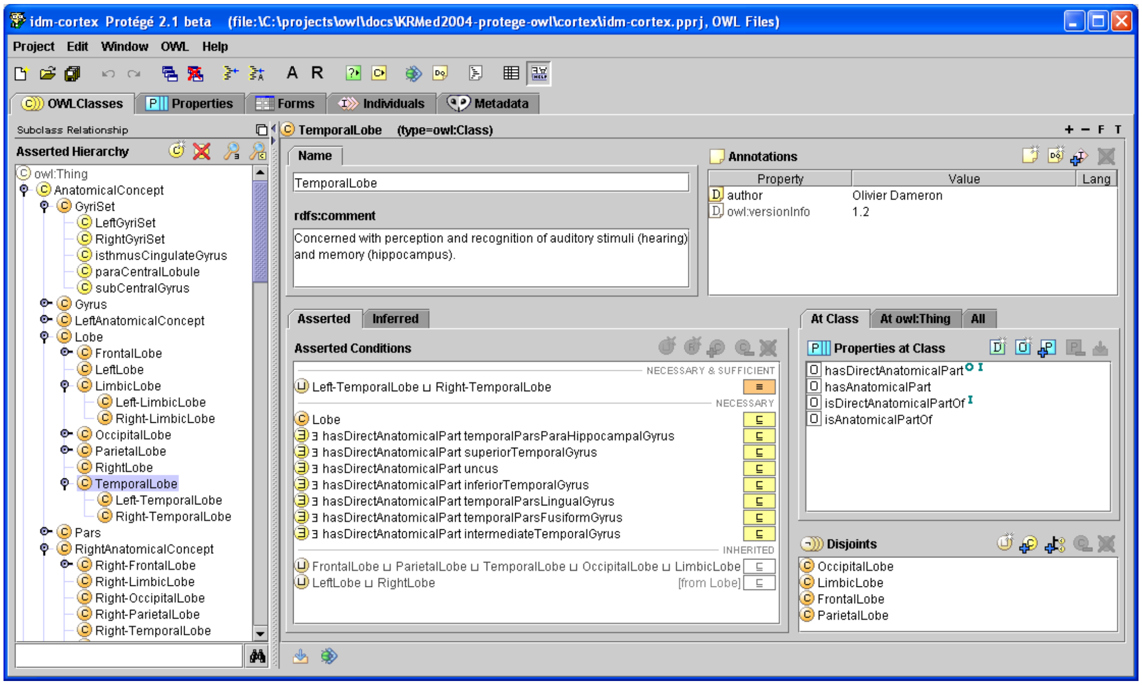
Protégé can be characterized as an ontology development environment. It provides functionality for editing classes, slots (properties), and instances. One of its strengths is that it can automatically generate a user interface from class definitions, and thus can support rapid knowledge acquisition. Protégé supports database storage that is scalable to several million concepts, and provides multi-user support for synchronous knowledge entry.

The current version of Protégé (2.1) is highly extensible and customizable. At its core is a frame-based knowledge model [9] with support for metaclasses. These metaclasses can be extended to define other languages on top of the core frame model [10]. For these other languages, Protégé can be extended with back-ends for alternative file formats. Currently, back-ends for Clips, UML, XML, RDF, DAML+OIL, and OWL are available for download.

Protégé not only allows developers to extend the internal model representation, but also to customize the user interface freely. As illustrated in Figure 1, Protégé's user interface consists of several screens, called *tabs*, which display different aspects of the ontology in different views. Each of the tabs can be filled with arbitrary components. Most of the existing tabs provide a tree-browser view of the model, with a tree on the left and details of the selected node on the right hand side. The details of the selected object are typically displayed by means of *forms*. The forms consist of configurable components, called *widgets*. Typically, each widget displays one property of the selected object. There are standard widgets for the most common property types, but ontology developers are free to replace the default widgets with specialized components. Widgets, tabs, and back-ends are called *plugins*. Protégé's architecture allows developers to add and activate plugins arbitrarily, so that the default system's appearance and behavior can be completely adapted to a project's needs.

The OWL Plugin[1] [8] is a complex Protégé plugin with support for OWL. It can be used to load and save OWL files in various formats, to edit OWL ontologies with custom-tailored graphical widgets, and to provide access to reasoning based on description logic. As shown in figure 1, the OWL Plugin's user interface provides various default tabs for editing OWL classes, properties, forms, individuals, and ontology metadata. The following section explains how to use the Classes, Properties and Metadata tabs for the de-

[1] `http://protege.stanford.edu/plugins/owl`



Name: TemporalLobe

rdfs:comment: Concerned with perception and recognition of auditory stimuli (hearing) and memory (hippocampus).

Property	Value	Lang
author	Olivier Dameron	
owl:versionInfo	1.2	

Asserted Conditions

NECESSARY & SUFFICIENT

Left-TemporalLobe ⊔ Right-TemporalLobe ≡

NECESSARY

Lobe ⊑
∃ hasDirectAnatomicalPart temporalParsParaHippocampalGyrus ⊑
∃ hasDirectAnatomicalPart superiorTemporalGyrus ⊑
∃ hasDirectAnatomicalPart uncus ⊑
∃ hasDirectAnatomicalPart inferiorTemporalGyrus ⊑
∃ hasDirectAnatomicalPart temporalParsLingualGyrus ⊑
∃ hasDirectAnatomicalPart temporalParsFusiformGyrus ⊑
∃ hasDirectAnatomicalPart intermediateTemporalGyrus ⊑

INHERITED

FrontalLobe ⊔ ParietalLobe ⊔ TemporalLobe ⊔ OccipitalLobe ⊔ LimbicLobe ⊑
LeftLobe ⊔ RightLobe [from Lobe] ⊑

Properties at Class

hasDirectAnatomicalPart
hasAnatomicalPart
isDirectAnatomicalPartOf
isAnatomicalPartOf

Disjoints

OccipitalLobe
LimbicLobe
FrontalLobe
ParietalLobe

Figure 1: The class editor of the Protégé OWL Plugin.

sign of a biomedical ontology. The section after that introduces how to use the Individuals and Forms tabs for the acquisition of Semantic Web contents.

BUILDING OWL ONTOLOGIES WITH PROTÉGÉ

An OWL ontology can be regarded as a network of classes, properties, and individuals. *Classes* define names of the relevant domain concepts and their logical characteristics. *Properties* (sometimes also called slots, attributes or roles) define the relationships between classes, and allow to assign primitive values to instances. *Individuals* are instances of the classes with specific values for the properties. The Semantic Web can be regarded as a network of ontologies and other Web resources. OWL ontology concepts can have references to concepts in other ontologies. The basic mechanism for this capability is ontology import (i.e., an ontology can import resources from existing ontologies and create instances or specializations of their classes).

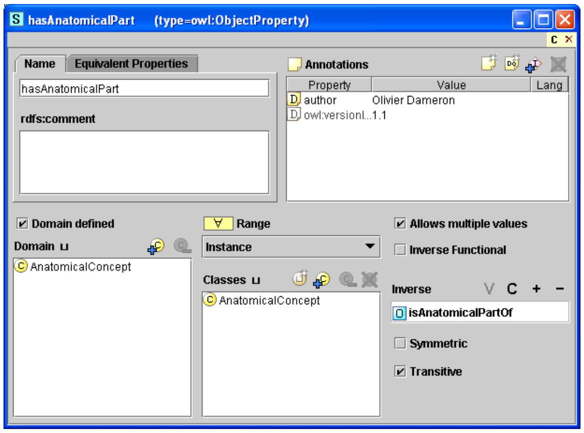
In our biomedical example ontology, we have a class called `CentralSulcus` which is defined as a kind of `AnatomicalConcept` that has a measured average depth. Individuals from this ontology would describe specific case data (e.g., a specific left central sulcus of an individual with the value of 23 mm for its depth). For the example ontology, we can import an existing ontology about units, and thus reuse the concepts from other files and support knowledge sharing. Let's take a look at how these elements can be defined in Protégé.

Classes

The most important view in the Protégé OWL Plugin is the OWLClasses tab (Figure 1). This tab displays the tree of the ontology's classes on the left, while the selected class is shown in a form in the center. The upper region of the class form allows users to edit class metadata such as name, comments, and labels, in multiple languages. The widget in the upper right area of the form allows users to assign values for *annotation properties* to a class. Annotation properties can hold arbitrary values such as author and creation date. Ontologies can define their own annotation properties or reuse existing ones such as those from the Dublin Core ontology. In contrast to other properties, annotation properties do not have any formal meaning for external OWL components like reasoners, but they are an extremely important vehicle for maintaining project-specific information. A typical use case for annotation properties in a biomedical field is to assign standardized identifiers such as ICD codes for concepts that describe a disease. Annotation properties, such as the predefined *`rdfs:seeAlso`*, can also be used to define cross-references between concepts. The OWL Plugin also uses annotation properties to store Protégé-specific information, and to manage "to-do" lists for ontology authors.

Properties

The *Properties* widget of the OWLClasses tab allows users to view and create relationships between classes. It provides access to those properties that could be used by the instances of the current class. The characteristics of a property are edited through the form shown in Figure 2. This form provides a metadata area in the upper part, displaying the property's name, annotations, and so on, similar to the presentation in the class form.



hasAnatomicalPart (type=owl:ObjectProperty)

hasAnatomicalPart

rdfs:comment

Property	Value	Lang
author	Olivier Dameron	
owl:versionI...	1.1	

☑ Domain defined

Domain ⊔

AnatomicalConcept

Range

Instance

Classes ⊔

AnatomicalConcept

☑ Allows multiple values

☐ Inverse Functional

Inverse

isAnatomicalPartOf

☐ Symmetric

☑ Transitive

Figure 2: An OWL property form in Protégé.

The available choices in the *Range* drop-down box depend on whether the property is a *datatype property* with primitive values, or an *object property* with references to other classes. For datatype properties, Protégé supports enumerations of symbols (owl:oneOf), and all reasonable XML Schema datatypes, grouped into booleans, floats, integers, and string types. For example, the datatype property *hasMeasuredDepth* can only take floats as values. Object properties can store references to individuals or classes from the ontology. For example, the object property *hasAnatomicalPart* can only take instances of AnatomicalConcept as values.

Depending on whether a property is an object or a datatype property, Protégé provides widgets for other property characteristics, such as whether the property is symmetric or transitive. Symmetric properties describe bidirectional relationships (i.e., if A is related to B via property R_s, then B is also related to A). For example, the contiguity relationship is symmetric. A property R_t is transitive if when A is related to B by R_t and B is related to C by R_t, then (A is also related to C by R_t). Part/whole relationships such as *hasAnatomicalPart* are usually considered to be transitive.

The *Domain* widget can be used to restrict a property's domain (i.e., the list of classes where the property can be used). Domain restrictions are optional and often left blank in OWL ontologies, because they may slow down some reasoning processes. If a property does not have a domain restriction, then it can be used for instances of any class.

Specialization

OWL has its theoretical foundation in description logic [1]. In description logic, a class is a set of individuals. The concept corresponding to the set of all individuals is usually called *Top* ($\top$), or *Thing*. Whenever the set of the individuals of a class B is a subset of the set of the individuals of a class A, B is said to be a *subclass* of A (noted B $\sqsubseteq$ A). B is also said to be a kind of A. All classes are subconcepts of $\top$.

In other words, superclasses define *necessary* conditions for class membership. Conversely, subclasses define *sufficient* conditions for class membership. For example, being a frontal lobe is a necessary condition for being a left frontal lobe: in order to be an instance of LeftFrontalLobe, an individual has to be an instance of FrontalLobe (and most certainly has to fulfill other requirements). Conversely, being a left frontal lobe is a sufficient condition for being a frontal lobe: every instance of LeftFrontalLobe is also an instance of FrontalLobe (but there may be other instances of FrontalLobe that are not instances of LeftFrontalLobe).

It is really important to keep in mind that a subconcept is a subset of individuals. Indeed, it is a common mistake to mix specialization and composition hierarchies. However, defining UpperLobeOfLung as a subconcept of Lung is erroneous because a lobe of a lung is not a kind of lung, but a part of a lung. Correct subconcepts for lung could be LeftLung and RightLung.

The specialization principle also implies inheritance of the properties. For instance, if we say that every Sulcus has an *averageDepth* and that CentralSulcus is a subclass of Sulcus, then every CentralSulcus also has an *averageDepth*. Because subclasses are more specific than their superclasses, the range of a subclass may itself be a subclass of the range of the superclass. This is called *property restriction*. For example, we can say that every Sulcus has a side in the class Side, and that every LeftSulcus (subclass of Sulcus) has a side LeftSide (subclass of Side).

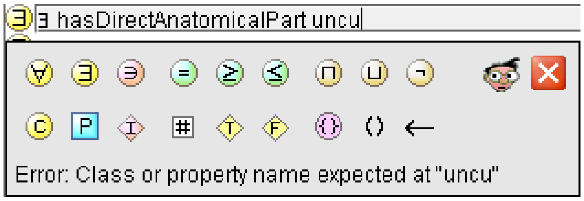
In Protégé, the tree widget of the OWLClasses tab is organized according to the subclass hierarchy. We can see that owl:Thing (which represents $\top$) is the root of the tree. Protégé users can browse, view, and edit the classes from the tree, create new subclasses, and move classes easily with drag-and-drop. Direct superclasses are also listed in the Conditions widget, which is described next. The OWL Plugin also allows to navigate and edit ontologies according to other relationships between classes, in particular to visualize the part-of relationships that are so common in biomedical domains.

Logical Class Characteristics

The *Conditions* widget of the OWLClasses tab allows to fully take advantage of OWL's description logic support, and to express conditions on the classes based on property restrictions and other expressions. The syntax used for OWL expressions in Protégé is summarized in table 1.

The key point here is to understand that an expression involving a property and its range such as "∃ *property* `Concept`" or "∀ *property* `Concept`" represents a set of individuals, and therefore can be interpreted as a concept. For example, (∃ *hasPart* `Lobe`) is the set of all the individuals related to at least one instance of `Lobe` by the *hasPart* relationship (they could also be related to instances of other concepts). Conversely, (∀ *hasPart* `Lobe`) is the set of all the individuals which are exclusively related to instances of `Lobe` by the *hasPart* relationship (or which are related to nothing by this relationship). Similarly, the union and intersection of two sets are also sets and can be interpreted as classes. For example, (`LeftAnatomicalPart` ⊓ `Gyrus`) represents the set of all left anatomical parts that are at the same time gyri, and (`LeftGyrus` ⊔ `RightGyrus`) represents the set of individuals that are instances of either concept. The ¬ operator can be used to define a class of any individual except those from a given class. For instance, ¬`LeftSide` is the set of all the individuals that are not instance of `LeftSide`. Finally, OWL also allows to define a class by exhaustively enumerating its instances.

The logical symbols used by the Protégé OWL Plugin are widely used in the description logic community [1]. Their major advantage is that they allow to display even complex class expressions in a relatively compact form. As shown in Figure 3, Protégé provides a convenient expression editor with support for either mouse or keyboard editing. However, some domain experts, especially from rather non-technical domains such as biomedicine, may require some training before they get used to these symbols. For these users, Protégé provides an English prose explanations of an OWL expression when the mouse is moved over it. Our collaborators are also working on alternative editors which support a rather template-based editing metaphor. Protégé's generic form architecture allows to quickly assemble alternative editors into the environment.



∃ hasDirectAnatomicalPart uncu

Error: Class or property name expected at "uncu"

Figure 3: Protégé provides a comfortable editor for arbitrary OWL expressions.

The formal definitions of the OWL primitives can be exploited by reasoners. They compute the specialization relationships (inheritance) between the classes based on their logical definitions. This reasoning support has shown to be a very valuable feature during ontology design, particularly in biomedical domains ([5, 11]). Ontology designers can periodically invoke a reasoner to see whether the logical class definitions meet the expectations, and to make sure that no inconsistency arise.

Necessary conditions. As mentioned above, a necessary condition for an individual to be an instance of a class is to be an instance of all the superclasses of this class. In addition to saying that a class is a subclass of its superclasses, such as `FrontalLobe` is a subclass of `Lobe`, necessary conditions allow the specify the properties that the class has to fulfill. This is an important activity when building an ontology, because, we don't want to limit ourselves to saying that a frontal lobe is a kind of lobe; we also want to represent what is specific to the frontal lobe, as opposed to the other lobes. For example, the frontal lobe has to be delimited by the central sulcus, as well as by the lateral sulcus. Therefore, to the original condition `FrontalLobe` ⊑ `Lobe`, we can add the two following necessary conditions "`FrontalLobe` ⊑ (∃ *isDelimitedBy* `CentralSulcus`)" and "`FrontalLobe` ⊑ (∃ *isDelimitedBy* `LateralSulcus`)". These conditions can also hold for other concepts, but an individual that fails to fulfill these conditions cannot be an instance of `FrontalLobe`.

Necessary and sufficient conditions. Necessary conditions can be interpreted as subset-superset relationships between sets of individuals. Similarly, we may want to represent that two classes have exactly the same instances (they are mutual subclasses of the other). For example, as the left and the right frontal lobe are two kinds of frontal lobe, we have

OWL element	Symbol	Key	Example expression in Protégé
owl:allValuesFrom	∀	*	∀ *hasPart* Lobe
owl:someValuesFrom	∃	?	∃ *hasDirectAnatomicalPart* RectusGyrus
owl:hasValue	∋	$	*hasColor* ∋ yellow
owl:minCardinality	≥	>	*hasSide* ≥ 1 (at least one value)
owl:maxCardinality	≤	<	*hasSide* ≤ 2 (at most two values)
owl:cardinality	=	=	*hasSide* = 1 (exactly one value)
owl:intersectionOf	⊓	&	LeftAnatomicalConcept ⊓ Gyrus
owl:unionOf	⊔	|	LeftGyrus ⊔ RightGyrus
owl:complementOf	¬	!	¬LeftSide
owl:oneOf	{ ... }	{ }	{yellow green red}

Table 1: Protégé uses traditional description logic symbols to display OWL expressions. Property names such as *hasSide* appear in italics. A common naming convention is to use uppercase names such as Lobe to represent classes, while individuals like yellow should be written in lower case.

the following condition: (LeftFrontalLobe ⊔ RightFrontalLobe) ⊑ FrontalLobe. But we also want to say that every frontal lobe is either a left or a right frontal lobe. Therefore, we use a necessary and sufficient condition (LeftFrontalLobe ⊔ RightFrontalLobe) ≡ FrontalLobe, which basically says that if you have a frontal lobe, then it is either a left or a right one (⊒); and that if you have a left or a right frontal lobe, then it is a frontal lobe (⊑). Classes with necessary and sufficient conditions are called *defined* classes (represented by orange icons in Protégé), while classes with only necessary conditions are called *primitive* (yellow icons). The Conditions widget allows to edit either type of conditions, and to copy or move expressions between blocks.

The open world assumption. Description logic make the so-called *open world assumption*, that is what is not said denotes a lack of knowledge (whereas in other contexts such as databases, what is not said is assumed to be false). A direct consequence is that if we don't say explicitly that two classes such as LeftFrontalLobe and RightFrontalLobe are disjoint, then it is perfectly valid for them to have individuals in common. The *Disjoints* widget, in the lower right corner of the OWLClasses tab allows users to represent axioms to control this aspect.

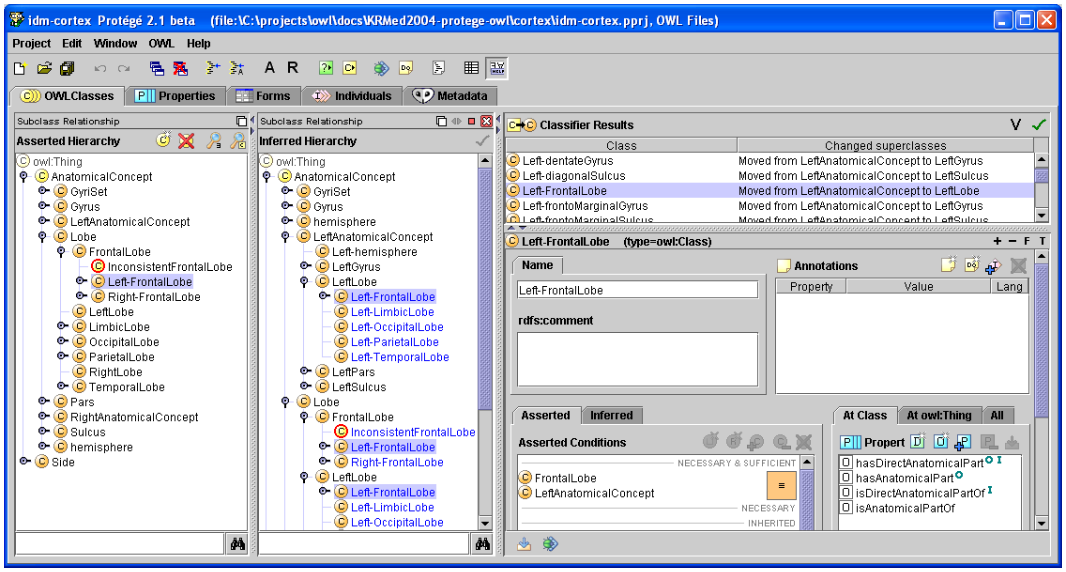
Classification and Consistency Checking

One of the major strengths of description logic languages like OWL is their support for intelligent reasoning. In our context, *reasoning* means to infer new knowledge from the statements asserted by an ontology designer. *Reasoners* are tools that take an ontology and perform reasoning with it. The OWL Plugin can interact with any reasoner that supports the standard DIG interface, such as Racer [6]. Since these reasoners are separate tools we will not discuss their details in this paper, but focus on their application oriented utility. During ontology design, the most interesting reasoning capabilities from these tools are classification and consistency checking.

Classification. Classification is used to infer specialization relationships between classes from their formal definitions. Basically, a classifier takes a class hierarchy including the logical expressions, and then returns a new class hierarchy, which is logically equivalent to the input hierarchy. As illustrated in Figure 4, Protégé can display the classification results graphically. After the user has clicked the classify button, the system displays both the asserted and the inferred hierarchies, and highlights the differences between them.

For example, we defined LeftFrontalLobe as any frontal lobe located in the left hemisphere (LeftFrontalLobe ≡ (FrontalLobe ⊓ LeftAnatomicalConcept)). Therefore, it appears as a direct child of the last two concepts in the asserted hierarchy (Figure 4). Similarly, we also defined LeftLobe as any lobe located in the left hemisphere (LeftLobe ≡ (Lobe ⊓ LeftAnatomicalConcept)). Because the definition of LeftFrontalLobe doesn't mention LeftLobe, these two concepts don't appear to be related. However, after classification, the reasoner infers from FrontalLobe ⊑ Lobe that LeftFrontalLobe is also a subclass of Leftlobe. Note: we could as well have defined LeftFrontalLobe ≡ (FrontalLobe ⊓ LeftLobe), but then we wouldn't have known that it is also a LeftAnatomicalConcept until the reasoner have found out.

This reasoning capability associated with description logic is of particular importance because it allows the



Class	Changed superclasses
Left-dentateGyrus	Moved from LeftAnatomicalConcept to LeftGyrus
Left-diagonalSulcus	Moved from LeftAnatomicalConcept to LeftSulcus
Left-FrontalLobe	Moved from LeftAnatomicalConcept to LeftLobe
Left-frontoMarginalGyrus	Moved from LeftAnatomicalConcept to LeftGyrus
Left-frontoMarginalSulcus	Moved from LeftAnatomicalConcept to LeftSulcus

Name: Left-FrontalLobe

rdfs:comment

Asserted Conditions: FrontalLobe, LeftAnatomicalConcept

Figure 4: Protégé provides access to description logic classifiers and can display both the asserted and the inferred class relationships.

user to provide intensional definitions for the classes. The specialization relationships become consequences of these definitions, and allow constraints inheritance. Without reasoning capabilities, the approach of creating an ontology is more extensional. It would require to explicitly state every specialization relationships between the concepts (e.g., in the previous example between `LeftFrontalLobe` and `LeftLobe`). This support is especially valuable in the domain of biomedicine, with its deeply nested hierarchies and multi-relationships between almost every part of the anatomy [1, 12]. Using OWL, ontology designers could just add a new concept by describing its logical characteristics, and the classifier would automatically place it in its correct position. Furthermore, it would report the side-effects of adding a new class.

Consistency checking. In addition to providing automatic classification, reasoning capabilities can be exploited to detect logical inconsistencies within the ontology. We could introduce a class `InconsistentFrontalLobe`, which is both a `LeftFrontalLobe` and a `RightFrontalLobe`. Since the last two concepts are defined to be disjoint, the reasoner reports that no individual can be an instance of this class. Clearly, these consistency checks can help tremendously in the construction and maintenance of large biomedical terminologies [12].

OWL Full and OWL DL An important issue with reasoning in OWL is that many reasoners are not able to handle the full expressivity of OWL. The OWL specification distinguishes between OWL Full and OWL DL to indicate which language elements are typically tractable for reasoners. Ontologies that use OWL Full elements such as metaclasses cannot be classified. Protégé allows users to edit some OWL Full concepts and provides features to help convert the ontology into OWL DL when a classifier is to be used. However, since OWL Full ontologies can state anything about anything, Protégé does not support the complete OWL Full syntax.

LINKING BIOMEDICAL RESOURCES INTO THE SEMANTIC WEB

This section demonstrates how to use OWL to link biomedical resources into the Semantic Web. In our scenario, OWL ontologies provide the vocabulary for describing the contents of images and scientific articles.

In order to describe biomedical images, we have defined a small image ontology, which basically only defines a single class `Image`, and defines four properties for each image: the integer properties *`hasWidth`* and *`hasHeight`* provide the dimensions of the image, the property *`hasURI`* stores a reference to the image's location, and the property *`hasContents`* can link an `Image` to an OWL class, such as those defined in the brain cortex ontology. These content concepts can later be used by intelligent agents for search purposes.

Protégé can now be used to create a new ontology `cortex-images.owl`, which imports the cortex ontology and the images ontology. The new ontology basically contains instances of the Image class, and uses the classes from the cortex ontology as contents values. Whenever concepts are imported from another ontology, Protégé displays them with a prefix such as cortex:.

Protégé provides excellent support for the acquisition of instances. As illustrated in Figure 5, the OWL Plugin makes this functionality available through the *Individuals* tab. For each class in an ontology, Protégé generates forms with appropriate widgets to acquire instances of the class. The Individuals tab shows the classes, their instances, and a form for the selected instance. By default, this form will contain default widgets, such as a numeric text field for integer properties and a clickable list for object properties. For example, Protégé has selected a list widget with create, add and remove buttons for the *hasContents* property. However, for the *hasURI* property, the system has selected a simple text field widget, which is not optimized for displaying images.

Fortunately, Protégé provides a *Forms* tab, which can be used to customize the forms. The Forms tab allows users to move and resize the widgets, and to replace widgets with other suitable ones. In our example, we have replaced the default text field widget for *hasURI* with an image widget, so that a preview of the image can be shown below the URI. Protégé's open architecture allows users to add arbitrary Java components as widgets, if the catalogue of default widgets is not sufficient. With a little bit of programming, we could provide a widget that allows users to select an image, and then fills the values of width and height automatically.

After the instances/individuals have been edited, they can be exported onto a Web server, so that agents can find and process them. A simple search agent would crawl through multiple image repositories, and analyze the image ontologies using an OWL parsing library such as Jena[2]. Supplied with a search concept such as FrontalLobe, an agent could then retrieve and filter images by their semantic proximity. A very similar approach can be used to implement a repository of scientific articles.

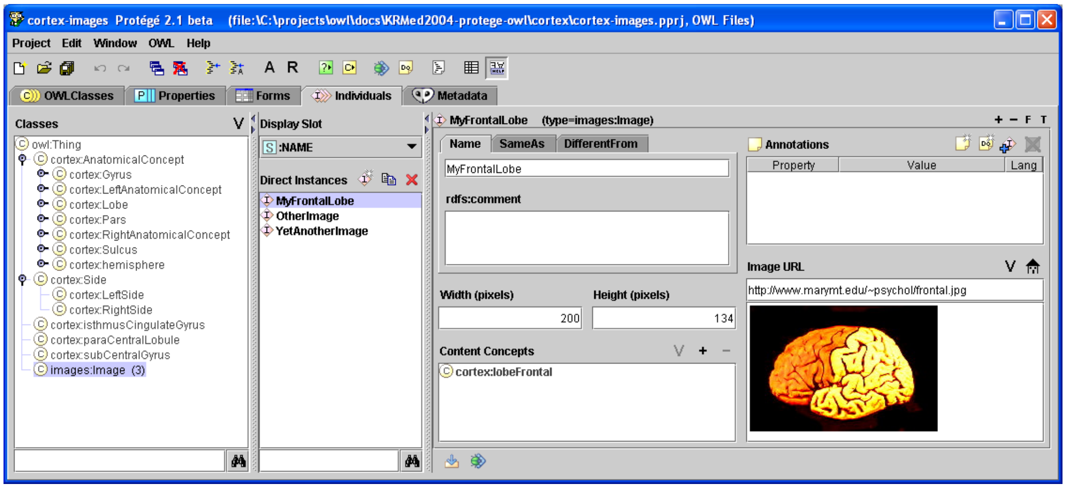
DISCUSSION AND CONCLUSION

Our main goal in this paper was to introduce the Protégé OWL Plugin, and to show that it provides a promising platform for biomedical ontology and Semantic Web projects. The OWL Plugin pioneers user-friendly components for building and reasoning with description logic ontologies. While researchers from the description logic community have managed to create deeply studied maps of their theoretical terrain, we believe it is now time to put languages such as OWL into practice, and thus reveal the strengths and weaknesses of these languages for particular domains in everyday use. Some issues of how to handle description logic in the development of large clinical terminologies have already been discussed by others (e.g., [5, 12, 11]). However, more work is necessary, in particular in training biomedical domain experts to use the rich semantics of OWL.

Some of the advantages of OWL are already obvious. Descriptions logic rely on a well defined semantics which makes modeling not only the structure, but also the meaning of a domain possible. As opposed to other formalisms such as frames [9], description logic allow users to provide intensional definitions for the concepts. As a consequence, ontologies are more compact, less error-prone, and easier to maintain. The precise semantics of description logic makes it possible to perform automatic reasoning. The intensional definitions of the concepts can be exploited by classifiers. Therefore, when adding a new class, one doesn't have to worry anymore about putting it in the right place in the taxonomic hierarchy. Moreover, multiple inheritance is automatically detected and dealt with. Classifiers can detect any logical inconsistencies in a class definition, that would prevent it of having instances. Eventually, reasoners can infer the correct relationships when combining ontologies of related domains, or extending an ontology with context-specific features. This point favors the sharing of common semantic references and their reuse in various contexts. Therefore, we expect OWL to play a key role not only for the Semantic Web, but also for the evolution and sharing of biomedical knowledge.

A final note about other ontology modeling tools. Given the short history of the Semantic Web, there are few other tools available with OWL support. One of the most popular ontology editors beside Protégé is OilEd [2]. From the beginning on, OilEd has been optimized for reasoning with description logic, and has been successfully used for various biomedical ontology projects. However, OilEd's authors never intended it as a full ontology development environment, but rather as a platform for experiments. As a result, OilEd's architecture is neither scalable to really large ontologies, nor sufficiently flexible to support customized user interface widgets. Furthermore, it suffers from a rather complicating user interface for editing logical expressions. The developers of Protégé and the OilEd team have recently joined forces in a transatlantic project called CO-ODE, which leads to a growing number of extensions for the Protégé OWL

[2] http://jena.sourceforge.net/


cortex-images Protégé 2.1 beta (file:\C:\projects\owl\docs\KRMed2004-protege-owl\cortex\cortex-images.pprj, OWL Files)
Project Edit Window OWL Help
A R
OWLClasses
Properties
Forms
Individuals
Metadata
Classes
Display Slot
Direct Instances
Name
SameAs
DifferentFrom
Annotations
Property
Value
Lang
+ − F T

owl:Thing
cortex:AnatomicalConcept
cortex:Gyrus
cortex:LeftAnatomicalConcept
cortex:Lobe
cortex:Pars
cortex:RightAnatomicalConcept
cortex:Sulcus
cortex:hemisphere
cortex:Side
cortex:LeftSide
cortex:RightSide
cortex:isthmusCingulateGyrus
cortex:paraCentralLobule
cortex:subCentralGyrus
images:Image (3)

:NAME

MyFrontalLobe
OtherImage
YetAnotherImage

MyFrontalLobe (type=images:Image)

MyFrontalLobe

rdfs:comment

Width (pixels): 200
Height (pixels): 134

Content Concepts
cortex:lobeFrontal

Image URL
http://www.marymt.edu/~psychol/frontal.jpg

Figure 5: Protégé generates user interfaces to acquire individuals of ontology concepts. This can be used to annotate Web resources such as images for a clinical online repository.

Plugin. Many other groups from around the world are also developing Protégé plugins, including tools which can be used to edit OWL classes and relationships in a visual UML-style diagram. Other large-scale Protégé plugins are being optimized for the OWL Plugin. With its large and rapidly growing community of thousands of users, Protégé has the potential to maintain its position as one of the leading open-source ontology development environments for the Semantic Web.

Acknowledgements

This work has been funded by a contract from the US National Cancer Institute and by grant P41LM007885 from the National Library of Medicine. Olivier Dameron is funded by INRIA. Additional support for this work came from the UK Joint Information Services Committee under the CO-ODE grant. Several colleagues and students at SMI were involved in the development of the OWL Plugin, in particular Ray Fergerson and Prashanth Ranganathan. Our partners from Alan Rector's team at the University of Manchester have made very valuable contributions.

References

[1] Franz Baader, Diego Calvanese, Deborah McGuineness, Daniele Nardi, and Peter Patel-Schneider, editors. *The Description Logic Handbook*. Cambridge University Press, 2003.

[2] Sean Bechhofer, Ian Horrocks, Carole Goble, and Robert Stevens. OilEd: a reason-able ontology editor for the Semantic Web. In *14th International Workshop on Description Logics*, Stanford, CA, 2001.

[3] Tim Berners-Lee, James Hendler, and Ora Lassila. The Semantic Web. *Scientific American*, 284:34–43, 2001.

[4] John H. Gennari, Mark A. Musen, Ray W. Fergerson, William E. Grosso, Monica Crubézy, Henrik Eriksson, Natalya F. Noy, and Samson W. Tu. The evolution of Protégé-2000: An environment for knowledge-based systems development. *International Journal of Human-Computer Studies*, 58(1):89–123, 2003.

[5] Jennifer Golbeck, Gilberto Fragoso, Frank Hartel, James Hendler, Bijan Parsia, and Jim Oberthaler. The national cancer institute's thesaurus and ontology. *Journal of Web Semantics*, 1(1), 12 2003.

[6] Volker Haarslev and Ralf Moeller. RACER user's guider and reference manual. `http://www.cs.concordia.ca/˜faculty/haarslev/racer`, 2003.

[7] Holger Knublauch. An AI tool for the real world: Knowledge modeling with Protégé. JavaWorld, June 20, 2003.

[8] Holger Knublauch, Mark A. Musen, and Alan L. Rector. Editing description logics ontologies with the Protégé OWL plugin. In *International Workshop on Description Logics*, Whistler, BC, Canada, 2004.

[9] Natalya F. Noy, Ray W. Fergerson, and Mark A. Musen. The knowledge model of Protégé-2000: Combining interoperability and flexibility. In *2nd International Conference on Knowledge Engineering and Knowledge Management (EKAW'2000)*, Juan-les-Pins, France, 2000.

[10] Natalya F. Noy, Michael Sintek, Stefan Decker, Monica Crubézy, Ray W. Fergerson, and Mark A. Musen. Creating Semantic Web contents with Protégé-2000. *IEEE Intelligent Systems*, 2(16):60–71, 2001.

[11] Alan Rector. Modularisation of domain ontologies implemented in description logics and related formalisms including OWL. In *2nd International Conference on Knowledge Capture (K-CAP)*, Sanibel Island, FL, 2003.

[12] Alan L. Rector. Clinical terminology: Why is it so hard? *Methods Inf. Med.*, 4–5(38):239–52, 1999.

[13] Michael K. Smith, Chris Welty, and Deborah L. McGuinness. OWL Web Ontology Language Guide. `http://www.w3.org/TR/owl-guide/`, 2003.

SECTION IV: INTERACTIVE SESSION: DEVELOPING A SMALL ONTOLOGY

Goal: *This session will consist of an audience-driven discussion with the goal of developing an ontology to represent DNA sequences. We will focus on the task of representing relationships between introns, exons, start sites, promoters, genes and chromosomes. This domain is accessible to participants with a wide background and has enough complexity that a large fraction of the common mistakes do happen. The mistakes made will be utilized to reinforce the best practices that follow.*

Session Notes

SECTION V: THE DO'S AND DON'TS OF ONTOLOGY CREATION

Goal: *In this section, we will describe the key principles of ontology design that address the most common sources of mistakes in biomedical ontologies. We will then discuss rules of thumb for good ontology design, along with examples of how they will enhance the quality of the resulting ontologies. We will also outline the guiding principles of the OBO Foundry, and describe how they might enable cross-ontology reasoning. (The text in this section is based on work of Andrew Spear and Barry Smith)*

Once the domain information has been assembled in the form of recognizing the important domain entities and relations, and the appropriate scope, granularity and the level of semantic rigor required for the ontology have been determined, the next step is to organize the domain information in a systematic and coherent fashion. ***The goal of this organization is to develop a representational artifact that is as logically coherent, unambiguous and true to the facts of reality as possible.***

ORGANIZING THE ENTITIES

There are three major and interrelated facets of organizing the entities for domain ontologies. These are 1) terminological regimentation, 2) definition of terminology, and 3) construction of taxonomic hierarchies of terms based on *is_a* relations.

TERMINOLOGY

The process of gathering domain-specific information results in the construction of a lexicon or terminology. However, if the goal is to use this domain specific information for purposes of representation in a computer-based ontology, then a more rigorous formalization – regimentation – of this terminology is needed. Regimenting a terminology thus involves both the explicit statement of (as well as ruthlessly consistent adherence to) syntactic conventions for the writing of terms and explicit consideration of the intended audience or user-base for the ontology. Specific principles along these lines include the following.

TERMINOLOGICAL CONSERVATIVISM

Don't reinvent the 'wheel': There are already a sufficient number of words in the world and in the biological and medical communities to ensure that the creation of new highly **specialized terms for purposes of inclusion in a domain ontology will rarely, if ever, be necessary**. The terminological choices of domain ontology builder(s) should be as respectful as possible of the current terminology, usage and practice of contemporary domain experts and potential users of the ontology.

A simple principle to follow, in selecting terms for a domain ontology, is to stay as close as possible to the actual use of people working in the field the domain ontology is about. Terms that are widely used and well-known by domain experts should be given preference over highly specialized and little used terms, and given the same meaning that they currently have in their use by domain experts. Creating new terms to represent things that a community is already familiar with or using a familiar term with a new and different meaning, are both likely to lead to confusion – both in the encoding of information into the ontology, and in its interpretation by end-users.

SINGULAR NOUNS

For the sake of intelligibility: **the general terms in an ontology should be formulated in the singular**, and the ontology's documentation should pay careful attention to the distinction between singular and plural nouns and to the requirement of noun-verb agreement. Thus ‘cat’, not ‘cats’, and ‘eukaryotic cell’, not ‘eukaryotic cells’.

There are a number of reasons why this convention should be adopted. First, it is crucial that some syntactical standard or other be adopted and rigorously adhered to for the encoding of common nouns, in order to ensure that they always appear similar to human users. In this respect rendering all such terms in the singular is as good a decision as any. Additionally, ensuring grammatical intersubstitutability of terms with their corresponding definitions (something that will be further discussed below) will be much easier if all terms have a standard grammatical format. Second, a more principled reason for representing the common nouns or universal terms in an ontology in the singular, is that the common nouns in an ontology always refer either to universals or to defined classes. In either case, however, the reference of these terms is singular. There is only one universal “feline”, even if it has many instances, and there is only one defined class “all the debutants in Texas in 1984”, even if it has many members. Thus it makes sense to use singular rather than plural terms to refer to entities such as universals and classes, and to do this consistently when constructing a domain ontology.[3]

COMMON NOUNS IN LOWER CASE:

For the sake of intelligibility: **represent terms referring to universals or classes in all lowercase letters**. Thus ‘cat’, not ‘Cat’ or ‘CAT’, and ‘eukaryotic cell’, not ‘Eukaryotic Cell’ or ‘EUKARYOTIC CELL’. As with the convention regarding use of singular nouns, this convention is proposed largely because some convention or other must be adopted and rigorously consistently adhered to. However, in English capital letters are normally used to indicate either a proper name (Tom, Seattle, Jupiter) or an acronym (the U.N., the E.U., the U.K.), whereas common nouns normally do not involve capital letters of any sort. It is thus more consistent with English usage to use all lower case letters for the encoding of general terms.[4]

AVOID ACRONYMS

For the sake of intelligibility: **don’t use acronyms as part or all of any term**. Thus, instead of ‘ATP’ or ‘atp’ write ‘adenosine triphosphate’, and instead of ‘dna’ or ‘DNA’ write ‘deoxyribonucleic acid’**.** Using an acronym rather than the term for a universal or class increases the chances of confusion on the part of domain expert-users, while rendering use of the ontology by non-domain experts nearly impossible. In the worst case, an ontology whose terminology is filed with acronyms will be equivalent to an ontology intended to be used by speakers of French that is written in Russian. The ontology itself will not be understandable or usable without some further interpretative or translational guide.

UNIVOCITY

For the sake of intelligibility: **Terms should have the same meaning on every occasion of their use**. In an ontology, ‘cell’ should mean cell, ‘cancer’ should mean cancer, and similarly in all other cases. The principle of univocity in ontology terminology development is difficult to maintain because ordinary

[3] Barry Smith. “Against Idiosyncrasy in Ontology Development.” Forthcoming in B. Bennett and C. Fellbaum (Eds.), *Formal Ontology and Information Systems,* (FOIS 2006), Baltimore November 9—11, 2006.

[4] There are of course other languages with other grammatical rules for capitalization of nouns. Should it prove more intelligible to use capital letters in an ontology the natural language of which is, for example, German, then by all means this convention can be altered. Again, what is crucial is that the convention used by an ontology be explicitly stated and consistently adhered to throughout.

language regularly violates it. For example, the English word ‘bank’ can mean both “a financial institution” and “a stretch of earth directly connected and running parallel to a river”.

The reason for avoiding such ambiguity in the context of ontology design is quite straightforward. If a single term is used in more than one way in a given context, human participants in discourse regarding that context as well as computer applications working under different contexts are likely to become confused; leading to both computational errors and user confusion.[5]

Universal/Instance Univocity:

For the sake of intelligibility: **Terms/expressions referring to Universals, and terms/expressions referring to instances should be clearly demarcated**. For example, the common noun ‘dog’ can be plausibly understood as referring to a type or universal “dog”. The term ‘dog’ which occurs in the sentence ‘The accident was caused by a dog unintentionally ejected from a motor vehicle due to failure to use restraining harness’ can be plausibly understood as referring to a single particular dog. These two uses of the term ‘dog’ should be kept clearly separate in an ontology. There are a number of different ways to do this. One simple way would be to abide by the conventions we have already put forward for representing common nouns, using ‘dog’ to refer to the universal dog, while using a capital letter, proper names or alphanumeric strings to refer to particular dogs, as in ‘Dog’, ‘Fido’ or ‘#d437’ (importantly, by using one of these conventions consistently, not by mixing all three together!).[6]

DEFINITION OF TERMS

Regimenting the definitions of terms in an ontology is a semantic task, one that has to do with providing a definitive statement of the nature of the things that the terms refer to. In a scientific ontology we are not interested in the *lexical* or *conventional* definition of a term, a definition that reports the meaning that all or most members of a given language community attribute to a term (as can be found in a dictionary), but rather in the *real* definition of a term, that is, in a scientific statement of the basic nature of the kind of thing that that term refers to. It is important that the definitions be as clear, consistent and accurate as possible, while also being organized in terms of a coherent and consistently applied set of conventions. In the following we put forward a number of principles for the formulation of domain-ontology definitions.

ESSENTIAL FEATURES

Use essential features in defining terms: **The definition of a term referring to a universal or kind should be stated in terms of the essential features of the entities that are instances of that kind**. The essential features of a thing are those features without which the thing would not be the kind of thing that it is. For example, it is arguably not essential to a thing’s being an instance of the universal human that it have precisely two legs, ten fingers, an appendix or blond hair. On the other hand, if an entity is unable to engage in any kind of communication, to think in a (somewhat) rational way, or to have certain kinds of self-reflective or self-aware thoughts, then this might be grounds for maintaining that this thing is indeed

[5] For a discussion of the violation of the principle of univocity regarding the relations *part_of* and *is_a* in the Gene Ontology, see Smith, B., Köhler, J., & Kumar, A. On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology; see also Barry Smith. “Against Idiosyncrasy in Ontology Development.” Forthcoming in B. Bennett and C. Fellbaum (Eds.), *Formal Ontology and Information Systems,* (FOIS 2006), Baltimore November 9—11, 2006.

[6] Barry Smith. “Against Idiosyncrasy in Ontology Development.” Forthcoming in B. Bennett and C. Fellbaum (Eds.), *Formal Ontology and Information Systems,* (FOIS 2006), Baltimore November 9—11, 2006.

not an instance of the universal human. Thus Aristotle defined ‘human’ as “an animal that is rational”. Taking rationality and being an animal to be essential features of human beings means saying that while many other features of a thing may change (such as hair color, skin color, body parts, height, weight, strength, taste in food, etc.), develop or be eliminated altogether, a thing cannot lack either the feature of being an animal or the feature of being rational and still be an instance of the universal human being.

For natural objects, such as those studied by chemistry, biology and physics, the essential features of a thing are usually the features of that thing that play a prominent role in scientific explanation of its existence and behavior. Thus a good Aristotelian definition of ‘water’ would be water is a molecular compound consisting of hydrogen and oxygen, or water is the molecular compound H20. *For artifacts*, objects created by humans to be used in various contexts, the essential features usually have to do with the purpose or use for which the artifact was created. Thus a knife is a tool for cutting things, while a chair is furniture that can accommodate a normal sitting human being.

Definitions in a domain ontology should always be in terms of the essential features of the entities under consideration. What is essential to the domain as a whole will, as a general rule, be determined by the statement of the intended scope of the ontology. A final point is that with regard to defined classes that do not refer to universals on the side of reality, the essential features to be used in the definition just are the features mentioned in the arbitrary designation of the class. Thus the ‘essential’ features of the class of all people suffering from HIV on the African continent just are “to be suffering from HIV” and to be “on the African continent”.

Some examples of definitions that fail to utilize essential features of the things being defined include the definition of ‘water’ from the International Classification of Nursing Procedures as “a type of Nursing Phenomenon of Physical Environment with the specific characteristics: clear liquid compound of hydrogen and oxygen that is essential for most plant and animal life influencing life and development of human beings”,[7] the definition of ‘living subject’ as “a subtype of Entity representing an organism or complex animal, alive or not” from the HL7 RIM, [8] and the definition of ‘person’ as “A living subject representing a single human being [sic] who is uniquely identifiable through one or more legal documents”, from the HL7 Glossary.[9]

ARISTOTELIAN STRUCTURE

Use Aristotelian structure when formulating definitions: Consider again Aristotle’s definition of ‘human’: a human is an animal that is rational. This definition **has the basic form *An A is a B that Cs/is (a) C***; an A (human) is a B (animal) that Cs (is rational). This basic format should be used to structure the definitions that are provided for terms anywhere in a domain ontology. The advantages of using this structure are that A, B, and C will always occupy the same places in the definition, and they can always be interpreted in similar ways, regardless of the specific domain in which terms are being defined.

[7] International Classification of Nursing Procedures (ICNP), http://www.icn.ch/icnp.htm, accessed May, 2006.

[8] HL7 Version 3.0 accessed via Knowledge Source Server Version 2006AC, Thursday, September 28, 2006.

[9] Various Contributors eds., HL7 Publishing Technical Committee. Last Published 11/22/2005 8:05 PM. HL7® Version 3 Standard, © 2005 Health Level Seven®, Inc.

The traditional Aristotelian definition structure should be understood in the following way: A is the term that is being defined ('human', 'chair', 'cell', etc.), B refers to the genus of the original term, the next highest class/universal in the hierarchy of classes/universals in which the term is located, and C refers to the differentia of the universal designated by A. The differentiae of A are the essential features of A, those features that any entity must possess in order to be an instance of A, and those features that distinguish entities of kind A from all other entities. Thus in the Aristotelian definition of human, a human (A) is an animal (B: genus) that is rational (C: differentia). The structure of Aristotelian definitions can be understood against the background of species-genus hierarchies, the taxonomies that universals naturally form in which the higher levels of the hierarchy represent universals of greater generality (genus) relative to the lower levels (species) in the hierarchy.

One advantage of consistently using the Aristotelian definitional structure is that it can be used to explicitly locate the place in *is_a* hierarchies of the universals referred to by the terms being defined. The Aristotelian definitional structure thus represents a consistent format for the representation of definitions that can be used regardless of ontological domain, and that is inherently directed at explicitly representing the location of the term defined in an *is_a* hierarchy based on the informational content and structure of the definition alone.

Aristotelian approach

Define the terms in an ontology from the top down: **Terms in an ontology should be defined by beginning with the most general universals, and then by systematically working 'downwards' towards the least general**. This procedure is highly consistent with the principle requiring the use of Aristotelian structure in definitions. Beginning with an undefined or primitive top node or root term, terms on the next level down will be defined by saying that an A is a B (top level node and genus) that Cs/ is (a) C (differentia). This procedure can be reiterated as many times, and at as many different levels as necessary, but starting from the most general level keeps things simple at the beginning, and gives the ontology developer a better perspective from which to assess the comprehensiveness of the ontology that she is building.

A more general consideration in favor of the top down approach comes from the point, already discussed earlier, that an ontology should have a well defined and delimited domain to which it is intended to apply, one that is determined, as much as possible, by the actual unity of scientific and practical domains of research in reality. Thus beginning with the more general entities and relations in an ontology and working downwards ensures ruling out, form the beginning, consideration of entities that are not relevant to the domain one's ontology is intended to represent.

Positivity

Don't use negatives: Definitions in a scientific domain ontology are intended to convey the essential information about their subject-matter to a user. Utilizing negative predicates (non-physical, non-environmental, non-cellular), or negative characterizations (not a part of the heart, not a breathing thing) involves providing much less information about the entities referred to by the term being defined than would be provided if only positive characterizations were given.

Compare, for example, a definition of heart as "an organ that is *not* part of the nervous system", with the definition from the FMA, the heart is an "Organ with cavitated organ parts, which is continuous with the systemic and pulmonary arterial and venous trees". The first, negative, definition of the heart ensures only that the heart is not the brain, while leaving entirely open the possibility that it is the lungs, the kidneys or any number of other organs that are "not part of the nervous system". Thus, while negative definitions do provide *some* information about the entities being defined, positive definitions are much more exact and provide much more information, and they should be preferred and formulated whenever possible in the construction of domain ontologies.

INTELLIGIBILITY

Keep it simple: **The terms used in a definition should be simpler (more intelligible, more scientifically, logically or ontologically basic) than the term to be defined**. Definitions of terms are given in order to explain to people who do not know the meaning of the term what that meaning is. It is generally the case that a person who does not know the meaning of a term, especially a technical term, also does not know the meaning(s) of terms more abstract or complex than the original term. Thus a definition that uses such abstract or complex terms in defining the original term is unlikely to serve its purpose.

In scientific contexts it is inevitable that definitions will involve a certain degree of complexity and specialized terminology, however this should be kept to an absolute minimum in ontology design. Further, when specialized and potentially obscure terminology is used in the definition of a given term, the ontology should either itself include or at the very least include references to clear definitions of this terminology itself.[10] Some examples of definitions that violate the principle of intelligibility in actual ontologies include the old BIRNLex definition of 'mouse' as a "common name for the species *mus musculus",*[11] and the old 'GO:0007512: adult heart development', which was defined as "generation and development of the heart of a fully developed and mature organism".[12]

NON-CIRCULARITY

Avoid circularity in the definition of terms: **A definition is circular if the term to be defined, or a near synonym of that term, occurs in the definition itself**. For example, defining 'plant cell' as "a cell that is found in plants" or 'surgical tool' as a device that is used in surgical procedures. These definitions are circular because they provide no more information about the nature of the things the terms refer to than the terms themselves provide. Since definitions are intended to explain the meaning of a term to someone who does not already understand that term's meaning, using the term itself or some very similar expression in its own definition defeats the purpose of providing a definition in the first place.[13]

[10] Smith, B., Köhler, J., & Kumar, A. On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology."

[11] BIRNLex, http://137.110.143.4:8080/BIRNLex/. Note: this definition has now been fixed.

[12] Smith, B., Köhler, J., & Kumar, A. On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology. Note: this definition has now been fixed.

[13] See Kohler J, Munn K, Ruegg A, Skusa A, Smith B. "Quality Control for Terms and Definitions in Ontologies and Taxonomies." *BMC Bioinformatics*. 2006 Apr 19;7(1):212.

Werner Ceusters and Barry Smith. "A Realism-Based Approach to the Evolution of Biomedical Ontologies." forthcoming in *Proceedings of AMIA Symposium. 2006.* http://ontology.buffalo.edu/bfo/Versioning.pdf, and also Barry Smith. "Against Idiosyncrasy in Ontology Development." Forthcoming in B. Bennett and C. Fellbaum (Eds.), *Formal Ontology and Information Systems,* (FOIS 2006), Baltimore November 9—11, 2006.

Some (now repaired) examples include the old GO definition of 'hemolysis' as "the causes of hemolysis",[14] and the old BIRNLex definition of 'eyeball' as "the eyeball and its constituent parts".[15]

Term-Definition Intersubstitutability

In all extensional contexts **a defined term should be intersubstitutable with its definition in such a way that the result is both grammatically correct and truth-preserving**. The basic idea behind this principle is that wherever a term refers to a thing the definition of that term should also successfully refer to that thing. The intersubstitutability of a term and its definition with regard to the truth-value of sentences in which they occur is important both for preserving truth across inference in automated reasoning contexts and for ensuring intelligibility for human users of ontologies. If replacing a term with its definition results in a grammatically incorrect expression, this will substantially impede the human usability of an ontology.[16]

Context-independence

Don't leave the definition of a term open to interpretation: It should not be up to the end user of a domain ontology to decide or interpret whether or not the term 'heart' in the ontology means "human heart" or "canine heart", nor should it be up to the user to decide whether 'cell' means "animal cell", "plant cell" or "cell in general". This information should be explicitly included, either in the term itself (say 'plant cell' rather than just 'cell'), or in the definition of the term.

Scientific theories are intended to express the truth about reality in their respective domains, full stop. Thus a scientific definition, ideally, is not just sometimes or partially true, but is true, period. Definitions of scientific terms should attempt to capture this fact. Conversely, rendering the definitions of many scientific terms context independent will involve including more information about context in these terms themselves and/or in their definitions. For example, anatomy is the study of the physical structures present in organisms in general, whereas human anatomy, mouse anatomy, etc. are particular sub-fields of anatomy in general. Thus terms and definitions within these sub-fields, in order to be as context free as possible, should include the fact that they are definitions within a sub-field. The following definition of 'cell' as "structural and physicological unit of a living organism; it (i.e., plant cell) consists of protoplast and cell wall" from the Plant Ontology, violates this principle because it implicitly characterizes the term 'cell', which one would normally expect to refer to the general universal "cell", as applying only to cells within a specific domain, namely "plant cells".[17] Here it would be much better if, instead of 'cell', the term 'plant cell' was used.

Modularity

A set of definitions should be modular: Modularity is not a feature of a single definition, but rather a property that a set of definitions has if it has been structured in a certain way. A set of definitions

[14] Gene Ontology, http://www.geneontology.org/. Note: this definition has now been fixed.

[15] BIRNLex, http://137.110.143.4:8080/BIRNLex/. Note: this definition has now been fixed.

[16] For an example from GO, see Barry Smith, Jacob Köhler and Anand Kumar. "On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology" in *Data Integration in the Life Sciences, First International Workshop*, DILS 2004, Leipzig, Germany, March 25-26, 2004, Proceedings, Editor: Erhard Rahm.

[17] Plant Ontology, http://www.plantontology.org/. Note: this definition has now been fixed.

satisfies the requirement of modularity if they are organized into levels, with level 0 (root or most general) terms picked out as undefined primitives, and terms on levels n+1 for every n greater than 0 being defined by referring exclusively to logical and ontological constants together with already defined terms taken from levels less than n+1. In practice what this rather complicated principle means is just that the terms in an ontology should be defined in Aristotelian fashion, from the most general to the most particular, while making use of the Aristotelian definitional structure, and abiding by the other principles that have already been discussed.

The principle of modularity is thus closely related to the fact that a system of well-defined terms regarding a specific domain should in normal cases form a hierarchically structured taxonomy. More specifically, if all or most of the terms being defined refer to universals on the side of reality, then the hierarchy amongst universals from more specific (cat, fern, human) to more general (mammal, plant, organism) should be reflected in the definitions of the terms that refer to these universals. The principle of modularity is explicitly intended to ensure that terms lower down in a taxonomic hierarchy inherit all properties and characteristics from their parents, and following this principle helps to ensure logical consistency in the definition of terms, clear demarcations amongst levels of abstractness within the ontology, and the possibility of automated reasoning.[18]

Term-definition/taxonomic-location transparency

is_a should be built in: **Ideally, each term's definition will represent the location in a term hierarchy to which that term belongs**. The principle of term-definition/taxonomic-location transparency essentially summarizes and requires that ontology construction abide by the principles that have already been discussed. If the principle of modularity and the Aristotelian top-down approach to term definition have been adhered to strictly, then the principle of term-definition/taxonomic location transparency will also be satisfied. Alternatively, violation of the principle of term-definition/taxonomic location transparency by a system of defined terms suggests that the principle of modularity or the Aristotelian approach or both have been violated. An ontology that adheres to this principle will be humanly intelligibly and, as a general rule, computationally tractable insofar as all of the terms defined in the ontology will also stand in clear relationships to one another.[19]

Construction of a Taxonomic hierarchy

Considered literally, a taxonomy is a tree-like structure consisting of nodes and branches, usually with a root node, leaf nodes, and intermediate nodes connected to each other, and to the root and leaf nodes by branches. Taxonomies are normally used to represent the hierarchical relationships amongst defined classes or universals in terms of the *is_a* relationship.

Importantly, taxonomic structures can be generated amongst universals and defined classes in terms of a number of different relationships. For example, the *part_of* relationship can be used to generate a taxonomic structure amongst universals. In such a case, the taxonomy generated might better be

[18] Barry Smith, Jacob Köhler and Anand Kumar. "On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology" in *Data Integration in the Life Sciences, First International Workshop*, DILS 2004, Leipzig, Germany, March 25-26, 2004, Proceedings, Editor: Erhard Rahm.

[19] Barry Smith, Jacob Köhler and Anand Kumar. "On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology" in *Data Integration in the Life Sciences, First International Workshop*, DILS 2004, Leipzig, Germany, March 25-26, 2004, Proceedings, Editor: Erhard Rahm.

referred to as a partonomy. However, a partonomy is not a taxonomy based on the *is_a* relation. *Is_a* and *part_of* have different meanings, and therefore must be kept strictly separate during the process of ontology design. Similarly, the relationship *descended_from* can be used to generate taxonomic structures amongst biological species universals, as in phylogenetic trees.[20] Once again, such a *descends_from* relationship is different from either *is_a* or *part_of,* and taxonomies based on these various relationships should be kept rigorously separate.

In what follows we will focus exclusively on taxonomies amongst universals structured by the *is_a* relation.[21] This discussion thus focuses on taxonomies understood as consisting of "finitely many universals arranged in a tree-like hierarchy."[22]

Classification

Taxonomy is closely related to the issue of classifying entities. Indeed, a taxonomy just is one of the most common kinds of classifications of entities. But what is a classification? Relative to ontology, there are two major senses of the term 'classification'.

The first has to do with identifying entities as instances of a given kind. This can happen in two ways. Particulars can be identified as instances of universals that they instantiate (as in "this cat is an instance of the kind cat"), and universals themselves can be identified as belonging to formal ontological categories (as in "the universal cat belongs in the formal ontological category of object or substance"). In this sense of 'classification', classifying an entity just involves recognizing what type of entity it is, either at the domain level, or at the level of formal ontological categories.

The second sense of 'classification' is the one that is directly related to taxonomies, though it always presupposes that some amount of classification in the first sense of the term has already taken place. In this second sense, a classification is a systematic organization of entities belonging to a given ontological category based on the relationships that these entities stand in both within and across ontological categories. To make this characterization concrete, consider the universals "eukaryotic cell" and "cell". It is clear based on current biological knowledge that a eukaryotic cell *is_a* cell. However, fully understanding the import of this *is_a* relation rests on two things. First, the universals eukaryotic cell and cell are both substantial universals; instances of both universals are entities that persist as identical through time, gain and lose qualities and parts, and are wholly present at any time at which they exist at all. However, the universal eukaryotic cell is differentiated from the universal cell in virtue of its having as part a cell nucleus, such that eukaryotic cell *has_part* cell nucleus is true.

[20] We did not define the *descends_from* relation or anything like it in the earlier section on ontological categories and relations; however this could be done using the same basic strategy of starting with a primitive instance-level relation, and then defining the relationship amongst universals in terms of it.

[21] Much of the following discussion can be usefully compared with the more technically oriented proposals of Rector, A. L. "Modularisation of Domain Ontologies Implemented in Description Logics and related formalisms including OWL." http://www.w3.org/TR/webont-req.

[22] Neuhaus, F., Grenon, P. & Smith, B. "A Formal Theory of Substances, Qualities and Universals", in Achille Varzi and Laure Vieu (eds.), *Proceedings of FOIS 2004. International Conference on Formal Ontology and Information Systems*, Turin, 4—6 November 2004.

Similarly, plant cell *is_a* eukaryotic cell. Both plant cell and eukaryotic cell are substantial universals, but plant cell is differentiated from eukaryotic cell by the possession of a cell wall as part, thus plant cell *has_part* cell wall is true. So, what is going on here is that substantial universals (cell, eukaryotic cell, plant cell) are being classified in terms of their standing in a specific ontological relationship (*has_part*) to other substantial universals (cell nucleus, cell wall). Notice that this also fits the Aristotelian definitional structure perfectly, "a eukaryotic cell (A) is a cell (B) that has a nucleus (C)".

Similarly, the Aristotelian definition of human, "a human is an animal that is rational" classifies substantial universals into an *is_a* hierarchy based on their standing in the *inheres_in* relation to universals belonging to the ontological category of qualities. Thus rationality *inheres_in* human, and differentiates the universal human from other substantial universals for kinds of animals.

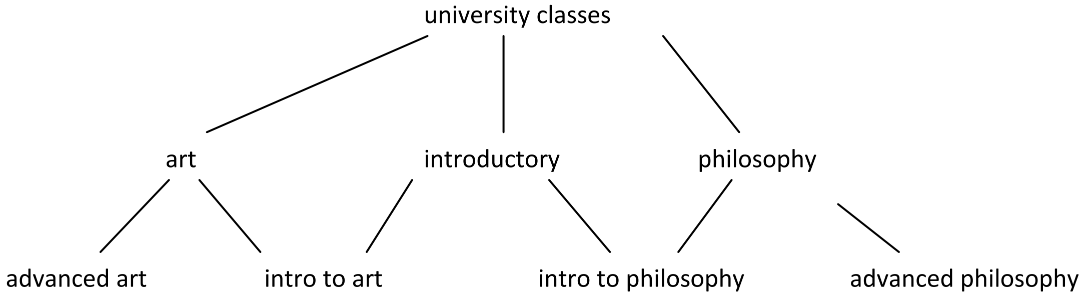
So, in this second (primary) sense, a classification is a systematic organization of entities belonging to a given ontological category based on the relationships that these entities stand in both within and across ontological categories. It is taxonomies in this sense that the universals in a good ontology should be organized in terms of. However, there is one more important feature of such classifications, and this is the principle in terms of which the classification has been generated.

The Principle of Classification

Every classification should be structured according to the semantics for the *is_a* relation and should identify distinct universals or classes only when these can be distinguished both from the universals or classes at the next level up in the hierarchy, and from other universals or classes at the same level, by some ontological feature or relation (such as possession of a characteristic part or quality). However, in addition to these features, classifications that form *is_a* hierarchies are also normally structured in terms of a guiding principle or criterion of classification, one that further determines the specific meaning of the location of universals in the hierarchy, and determines, in part, which universals are relevant for inclusion in the hierarchy.

For example, it is possible to classify the classes in a university in terms of their subject matter, i.e. history, chemistry, biology, philosophy, etc. Alternatively, it is possible to classify the classes in a university in terms of their difficulty level, i.e. introductory, intermediate, advanced, graduate, etc. Each of these 'ways of classifying' university classes amounts to adopting a principle for the classification of university classes. **What is important is that, for any given classification, the principle that is being used be specified as clearly as possible at the outset, and then consistently adhered to throughout**. Further, two different principles should not be applied at the same level in the same hierarchy. Thus, a classification that attempted to simultaneously classify university classes both by subject matter and by difficulty level would end up looking something like the following:

Diagram 1.



The problem with this hierarchy is that saying A *is_a* B is ambiguous. For example, 'art *is_a* university class' can mean either that art is a university class of a particular kind *or* that art is a university class of a specific difficulty level. In other words, the relation *'is_a'* in this hieararchy is ambiguous. And while it may be clear from context in this particular case which meaning the '*is_a'* relation should be given in each particular context, in a more complicated case such as biological or medical science, such ambiguities, especially if they are perpetrated through an entire hierarchy of *'is_a'* relations, are likely to lead to a great deal of confusion, at least for human users, and often also for automated reasoning.

More sophisticated principles of classification include similarity and difference of anatomical structure (one principle) or similarity and difference of genetic code (a second distinct principle) for organisms, atomic number for the elements in the periodic table, and kind of patient treated by (one principle) or kind of procedure performed by (a second distinct principle) for the classification of doctors in a hospital. Once more, what is crucial is that the principle being appealed to in classifying the entities in an *is_a* hierarchy be both explicitly identified and consistently adhered to from the beginning.

Up until this point the principles for best ontology practice that have been discussed have been intended to apply to all kinds of entities and domains whatsoever. The following principles for best ontology practice, though arguably having wider application as well, are primarily offered in the spirit of these traditional treatments of classification of substantial or objectual entities in terms of their characteristic qualities and or parts.[23]

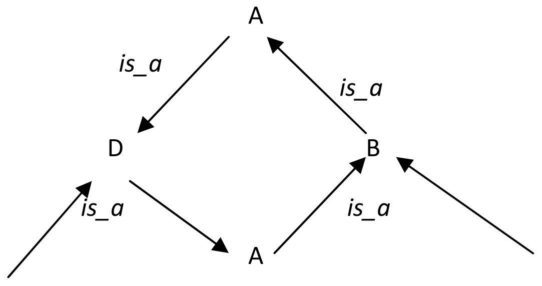
Use Single Inheritance

No diamonds: Based on the forgoing discussion, **an important principle for the classification of universals and classes in an ontology is to use single, not double- or multiple-, inheritance**. In a classification the relationship of inheritance is the relationship that a less abstract class stands in to the more abstract class that is directly above it in a classificatory hierarchy. Thus cat stands in the inheritance relationship to mammal, plant cell to eukaryotic cell, and eukaryotic cell to cell. Saying that a classification should use single inheritance means saying that every universal or class included in the classification should stand in

[23] See Barry Smith. "The Logic of Biological Classification and the Foundations of Biomedical Ontology." In Dag Westerstahl (ed.), *Invited Papers from the 10th International Conference in Logic Methodology and Philosophy of Science,* Oviedo, Spain, 2003. Elsevier-North-Holland, 2004; Neuhaus, F., Grenon, P. & Smith, B. "A Formal Theory of Substances, Qualities and Universals", in Achille Varzi and Laure Vieu (eds.), *Proceedings of FOIS 2004. International Conference on Formal Ontology and Information Systems*, Turin, 4—6 November 2004.

an inheritance relationship to exactly one universal or class at the next highest level. When this principle is violated, taxonomies take on a diamond-like structure.

Such diamonds or 'multiple inheritance' should be avoided within a single classification in an ontology for a number of reasons. First, as they lead to an ambiguity in how the '*is_a'* relation is to be interpreted in such classifications. Second, allowing such diamonds into a classificatory scheme can lead to the existence of loops within the classification, such that A *is_a* B, B *is_a* C, C *is_a* D, and D *is_a* A, as illustrated in diagram below. Allowing the existence of such loops in a classification amounts to adopting circular definitions for all or most of the terms located in the loop, and can lead to both human confusion and computational errors such as infinite loops.



The following is an example of a loop that can be found in the UMLS:

"Topographic regions: General terms

Physical anatomical entity

Anatomical spatial entity

Anatomical surface

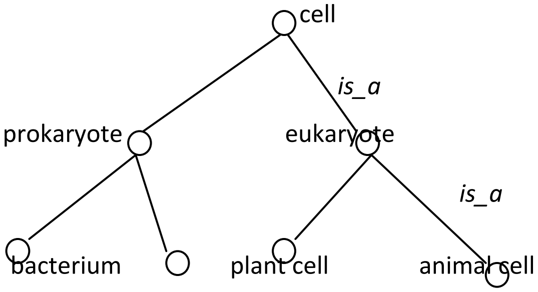
Body regions

Topographic regions"[24]

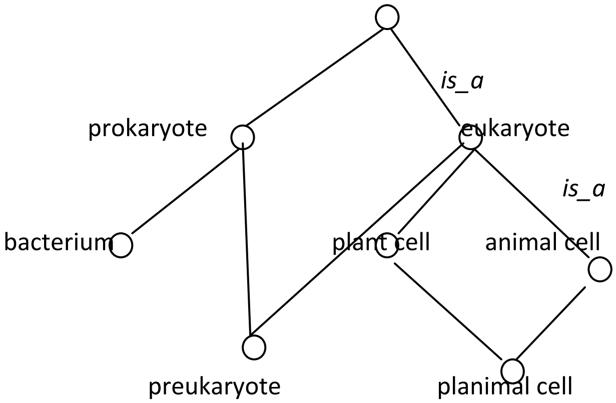
A third problem with multiple inheritance is that it can lead to double-counting and hence to double-naming of entities in a classification. For example, the class "Intro to Art" from diagrams 7 inherits its properties from both the class "Introductory classes" and the class "Art classes"; if care is not taken, it can easily happen that a class inheriting from multiple super-classes will be taken as itself identifying multiple classes (double or multiple counting), each of which requires a separate name (hence double or multiple naming).

[24] See Bodenreider O. Circular hierarchical relationships in the UMLS: etiology, diagnosis, treatment, complications and prevention. Proc AMIA Symp 2001:57-61.

A fourth and final problem with allowing multiple inheritance into an ontology is the following: Each classification should involve one and only one principle of classification in order to avoid the possibility of ambiguity in its interpretation, as has already been discussed. Ideally, the principles used for organizing universals in a scientific classification will be as closely related as possible to the essential features of the entities that instantiate those universals. Thus good candidate principles for the classification of biological species include similarity and difference of anatomical structure, on the one hand, and similarity and difference of genetic material on the other. It is largely for biological scientists to decide which of these features is more essential in the identification and classification of biological species. However, what is important here is that if the classification of universals in a domain is carried out by specifying essential features and, if the Aristotelian procedure and definitional structure (an A (term) is a B (genus) that Cs (differentia)) is adhered to, then universals at each level should be identified specifically in terms of features that *distinguish* them from all other universals or classes at that level. This means that no two universals at the same level should have any instances in common, and in turn means that no sub-universals or sub-classes of these universals should have any instances in common. Consider the classification of kinds of cells in **Diagram 2**.



The classification in Diagram 2 is consistent with the following Aristotelian structured definitions: “Prokaryote is a cell that lacks a nucleus”, “Eukaryote is a cell in which the genetic material is organized into a membrane-bound nucleus”, “Plant cell is a eukaryote cell that has a large central vacuole and a cell wall”, and “Animal cell is a eukaryote cell that has a small central vacuole and lacks a cell wall.” Now, given these definitions, suppose we were to allow the two instances of multiple inheritance from **Diagram 3** to occur.



The Aristotelian definition of ‘preukaryote’ would be “a prokaryote and eukaryote that has genetic material that is organized into a membrane-bound nucleus and lacks a nucleus”, while a ‘planimal cell’ would be 'a eukaryote cell that has a large central vacuole and a cell wall, and has a small central vacuole and lacks a cell wall". Now, the classes 'preukaryote' and 'planimal cell' should strike us as absurd. Their definitions contain manifest contradictions, and they seem like totally arbitrary and contrived denizens of the domain of cells. Note, however, that these classes were derived simply by allowing multiple inheritance into a classification that was well structured and organized according to a single principle along the lines of essential features of the universals in its domain, namely the kinds of cells and their essential differences in terms of possessing or not possessing certain kinds of parts (cell walls, a nucleus, etc.). Allowing multiple inheritance into a good classification leads to manifest absurdity, while a classification that straightforwardly yields multiple inheritance is probably a bad classification to begin with, either using a non-essential principle of classification for its classes and universals, or ambiguously attempting to apply multiple principles at the same time. While there are certainly cases of classification that are more complicated than cell biology, and where the choice of a principle of classification is not entirely clear, these cases are not different in kind, but only in degree of complexity from the case here discussed.[25]

Note: Importantly, it is often possible to classify a single universal in more than one way. For example, doctors can be classified in terms of the kinds of patients that they treat on the one hand, or in terms of the kinds of procedures that they perform on the other, and a term such as 'pediatric surgeon' could be classified in both of these ways (and would probably occupy a rather different place in the two classifications). However, in such cases, the answer is not to allow one taxonomy with multiple inheritance, but rather to construct two separate classifications, and use the definitions of the terms that appear in them, as well as the formal ontology (categories and relations) serving as the background of the domain ontology, to spell out the important relations between these two (or more) separate and diamondless classifications.[26]

Joint Exhaustiveness

Don't leave relevant universals out: **When classifying kinds of entities in a given domain, as much care as possible should be taken to ensure that all relevant universals are included at each level in a taxonomy**. An ideal classification would include all existing domain universals along with identifying and differentiating information for each, at each level in the hierarchy of organization. This does not mean that the designers of an ontology should sit around waiting for new scientific information (of which there is always more) before completing their ontology and making it available for use, but it does mean that all relevant domain universals that are discussed in contemporary domain literature and by contemporary domain experts should be included.

[25] Barry Smith, Jacob Köhler and Anand Kumar. "On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology" in *Data Integration in the Life Sciences, First International Workshop*, DILS 2004, Leipzig, Germany, March 25-26, 2004, Proceedings, Editor: Erhard Rahm; Barry Smith. "The Logic of Biological Classification and the Foundations of Biomedical Ontology." In Dag Westerstahl (ed.), *Invited Papers from the 10th International Conference in Logic Methodology and Philosophy of Science,* Oviedo, Spain, 2003. Elsevier-North-Holland, 2004.

[26] Bittner, Smith, "Normalizing Medical Ontologies using Basic Formal Ontology", in *Kooperative Versorgung, Vernetzte Forschung, Ubiquitäre Information* (Proceedings of GMDS Innsbruck, 26—30 September 2004), Niebüll: Videel OHG, 199—201. http://ontology.buffalo.edu/medo/gmds2004Norm.pdf#search=%22Bittner%2C%20Smith%2C%20%E2%80%9CNormalizing%20Medical%20Ontologies%20using%20Basic%20Formal%20Ontology%22; see also Rector, A. L. "Modularisation of Domain Ontologies Implemented in Description Logics and related formalisms including OWL." http://www.w3.org/TR/webont-req.

Mutual Exclusivity

No shared sub-classes: No two universals or classes in a classification should have any sub-classes in common. In keeping with the above example, in a classification in which eukaryotic cell and prokaryotic cell occur at the same level as separate universals, they should have no sub-classes or sub-universals (no multiple inheritance) in common. The requirement that the universals or classes at each level of a classification be mutually exclusive is a straightforward consequence of the prohibition of multiple inheritance, and conversely.

Class Positivity

Complements of classes are not themselves classes: The complement of a class is the class containing all of the entities that do not belong in that class. Thus the complement of the class "dog" is the class "non-dog". As a general rule of thumb, class-compliments should be avoided when selecting the classes for and constructing the classification hierarchies in an ontology. The only thing that all of the members of a class-complement are guaranteed to have in common is the fact that there is some other class to which they all do not belong. Thus saying, of a given sub-class that it is subsumed by, for example, the class of all "non-conifer trees" is providing very little information about that sub-class. Further, class-complements rarely pick out genuine universals on the side of reality. Thus including many class complements in one's ontology is likely to render it less accurate to the facts of reality, and so less useful for both scientific and practical purposes.

There are exceptions to this rule, including some of the examples given above. For example, prokaryotic cells are distinguished form eukaryotic and from all other cells precisely by the fact that they lack a cell nucleus. This is, in effect, negative information used to define a class. However, in this particular case there is overwhelming scientific evidence to the effect that, at this level of generality, dividing up cell universals in this way does lead to a principled and exhaustive classification of all kinds of cells. In such cases including negative information, or even featuring it in the definition of a term and the demarcation of a class may be unavoidable. However, even in these cases every effort should be made to include some positive information about the kinds of entities being defined and classified as well.

Class Objectivity

Which classes exist is not a function of the current state of biological knowledge. Genuine classes, that is, the universals treated by natural science in any given domain, are discovered, not invented or created. This fact suggests a certain kind of general attitude or mind-set that should be taken towards the identification of classes in an ontology, namely, one that seriously takes into account the best available scientific information about reality in any given domain, and attempts to systematically organize that information according to its most essential characteristics.[27]

Class-Univocity

As with terms, so with classes: No distinctions without differences. Every class should be clearly distinct from every other class in the ontology with respect to at least one property or characteristic. The best that having two classes characterized by exactly the same set of properties can achieve is redundancy.

[27] Barry Smith, Jacob Köhler and Anand Kumar. "On the Application of Formal Principles to Life Science Data: A Case Study in the Gene Ontology" in *Data Integration in the Life Sciences, First International Workshop*, DILS 2004, Leipzig, Germany, March 25-26, 2004, Proceedings, Editor: Erhard Rahm.

Failure to adhere to the principle of class-univocity can also be problematic insofar as assigning different names to classes that are the same will be likely to lead to human and computational confusion.

Hierarchical Structure

Classifications have a hierarchical structure: Given the discussion that has gone before, the point should already be quite clear, that the terms in a classificatory hierarchy should be divided into predetermined levels. A classification hierarchy should include terms both for more specific and for more general universals. Thus besides including “red”, “human” or “H20” a good classification should also include, as separate entries, “color”, “animal” or “compound”. Explicitly including this information will generate a division of terms into levels of generality, as can be seen in the examples discussed above. Hierarchically structured classifications are also a straightforward result of strictly adhering to the Aristotelian structure and method for good definitions, of identifying entities in terms of essential characteristics, and of classifying entities consistently in terms of a single principle of classification.

Summary

The process of regimenting the domain information for an ontology thus includes the following steps, all to be carried out in terms of the principles that have been put forward above. First, select the exact terms and the format of these terms that are to be included in the ontology, based on domain information that has already been gathered. Second, provide clear, scientifically accurate and logically coherent definitions for each of these terms. Third, explicitly recognize the place or places of each of these terms in a hierarchical classification of the domain information. When this task has been consistently carried out, the domain information should be ready for the last great step in ontology development: formalization and computer implementation.

Organizing the relationships

Relations in
biomedical ontologies

Participants are requested to read the included published paper, *Relations in Biomedical Ontologies*, for details on this topic.

Method

Open Access

Relations in biomedical ontologies

Barry Smith[*†], Werner Ceusters[‡], Bert Klagges[§], Jacob Köhler[¶], Anand Kumar[*], Jane Lomax[¥], Chris Mungall[#], Fabian Neuhaus[*], Alan L Rector[**] and Cornelius Rosse[††]

Addresses: [*]Institute for Formal Ontology and Medical Information Science, Saarland University, D-66041 Saarbrücken, Germany. [†]Department of Philosophy, University at Buffalo, Buffalo, NY 14260, USA. [‡]European Centre for Ontological Research, Saarland University, D-66041 Saarbrücken, Germany. [§]Department of Genetics, University of Leipzig, D-04103 Leipzig, Germany. [¶]Rothamsted Research, Harpenden, AL5 2JQ, UK. [¥]European Bioinformatics Institute, Hinxton, CB10 1SD, UK. [#]HHMI, Department of Molecular and Cellular Biology, University of California, Berkeley, CA 94729, USA. [**]Department of Computer Science, University of Manchester, M13 9PL, UK. [††]Department of Biological Structure, University of Washington, Seattle, WA 98195, USA.

Correspondence: Barry Smith. E-mail: phismith@buffalo.edu

Published: 28 April 2005

Genome ***Biology*** 2005, **6:**R46 (doi:10.1186/gb-2005-6-5-r46)

The electronic version of this article is the complete one and can be found online at http://genomebiology.com/2005/6/5/R46

Received: 28 October 2004
Revised: 3 February 2005
Accepted: 31 March 2005

Abstract

To enhance the treatment of relations in biomedical ontologies we advance a methodology for providing consistent and unambiguous formal definitions of the relational expressions used in such ontologies in a way designed to assist developers and users in avoiding errors in coding and annotation. The resulting Relation Ontology can promote interoperability of ontologies and support new types of automated reasoning about the spatial and temporal dimensions of biological and medical phenomena.

Background

Controlled vocabularies in bioinformatics

The background to this paper is the now widespread recognition that many existing biological and medical ontologies (or 'controlled vocabularies') can be improved by adopting tools and methods that bring a greater degree of logical and ontological rigor. We describe one endeavor along these lines, which is part of the current reform efforts of the Open Biomedical Ontologies (OBO) consortium [1,2] and which has implications for ontology construction in the life sciences generally.

The OBO ontology library [1] is a repository of controlled vocabularies developed for shared use across different biological and medical domains. Thus the Gene Ontology (GO) [3,4] consists of three controlled vocabularies (for cellular components, molecular functions, and biological processes) designed to be used in annotations of genes or gene products. Some ontologies in the library - for example the Cell and Sequence Ontologies, as well as the GO itself - contain terms which can be used in annotations applying to all organisms. Others, especially OBO's range of anatomy ontologies, contain terms applying to specific taxonomic groups such as fly, fungus, yeast, or zebrafish.

Controlled vocabularies can be conceived as graph-theoretical structures consisting on the one hand of *terms* (which form the nodes of each corresponding graph) linked together by means of edges called *relations*. The ontologies in the OBO library are organized in this way by means of different types of relations. OBO's Mouse Anatomy ontology, for example, uses just one type of edge, labeled *part_of*. The GO currently uses two, labeled *is_a* and *part_of*. The *Drosophila* Anatomy ontology includes also a *develops_from* link. Other OBO ontologies include further links, for example (in the Sequence Ontology) *position_of* and *disjoint_from*. The National Cancer Institute (NCI) Thesaurus adds many additional links, including *has_location* for anatomical structures and different *part_of* relations for structures and for processes.

The problem is that when OBO and similar ontologies incorporate such relations they typically do so in informal ways, often providing no definitions at all, so that the logical interconnections between the various relations employed are unclear, and even the relations *is_a* and *part_of* are not always used in consistent fashion both within and between ontologies. Our task in what follows is to rectify these defects, drawing on the requirements analysis presented in [5].

Of the criteria that ontologies must currently satisfy if they are to be included in the OBO library, the most important for our purposes are: first, inclusion of textual definitions or descriptions designed to ensure that the precise meanings of terms as used within particular ontologies will be clear to a human reader; second, employment of a standard syntax, such as the OWL or OBO flatfile syntax; third, orthogonality to the other ontologies already included in the library. These criteria are designed to support the integration of OBO ontologies, above all by ensuring the compatibility of ontologies pertaining to an identical subject matter. OBO has now added a fourth criterion to assist in achieving such compatibility, namely that the relations (edges) used to connect terms in OBO ontologies should be applied in ways consistent with their definitions as set forth in this paper.

The Relation Ontology offered here is designed to put flesh on this criterion. How, exactly, should *part_of* or *located_in* be defined in order to ensure maximally reliable curation of each single ontology while at the same time guaranteeing maximal leverage in building a solid base for life-science knowledge integration in general? We describe a rigorous methodology for providing an answer to this question and illustrate its use in the construction of an easily extendible list of ten relations of a type familiar to those working in the bio-ontological field. This list forms the core of the new OBO Relation Ontology. What is distinctive about our methodology is that, while the relations are each provided with rigorous formal definitions, these definitions can at the same time be formulated in such a way that the underlying technical details remain invisible to ontology authors and curators.

Shortcomings of biomedical ontologies

While considerable effort has been invested in the formulation and definition of terms in biomedical ontologies, too little attention has been paid in the ontological literature to the associated relations. A number of characteristic types of shortcomings of controlled vocabularies can be traced back especially to the neglect of issues of formal structure in the treatment of relations [5-10]. To take just one example, the pre-2004 versions of GO allowed at least three different readings of the expression 'part of' as representing simultaneously: inclusion relations between vocabularies; a relation of possible parthood between biological entities; a relation of necessary parthood between biological entities. As was shown in [6], this coexistence of conflicting readings meant that three of the four rules given in the then effective documentation for reasoning with GO's hierarchies were logically incorrect.

Another characteristic family of problems turns on the paucity of resources for expressing relations in ontologies like GO. For example, because GO has no direct means of asserting location relations, it must capture such relations indirectly by constructing new terms involving syntactic operators such as 'site of', 'within', 'extrinsic to', 'space', 'region', and so on. It then simulates assertions of location by means of '*is_a*' and '*part_of*' statements involving such composites, for example in:

extracellular region is_a cellular component

extrinsic to membrane part_of membrane

both of which are erroneous. Additional problems arise from the fact that GO's *extracellular region* and *extracellular space* are both specified in their definitions as referring to the space (how large a space?) external to the outermost structure of a cell.

Another type of problem turns on the failure to distinguish relational expressions which, though closely related in meaning, are revealed to be crucially distinct when explicated in the formally precise way that is demanded by computer implementations. An example is provided by the simultaneous use in OBO's Cell Ontology of both *derives_from* and *develops_from* while no clear distinction is drawn between the two [11]. This problem is resolved in the treatment of derivation and transformation below, and has been correspondingly corrected in versions 1.14 and later of the Cell Ontology.

Efforts to improve GO from the standpoint of increased formal rigor have thus far been concentrated on re-expressing the existing GO schema in a description logic (DL) framework. This has allowed the use of a DL-reasoner that can identify certain kinds of errors and omissions, which have been corrected in later versions of GO [12]. DLs, however, can do no more than guarantee consistent reasoning according to the definitions provided to them. If the latter are themselves problematic, then a DL can do very little to identify or resolve the problems which result. Here, accordingly, we take a more radical approach, which consists in re-examining the basic definitions of the relations used in GO and in related ontologies in an attempt to arrive at a methodology which will lead to the construction of ontologies which are more fundamentally sound and thus more secure against errors and more amenable to the use of powerful reasoning tools.

This approach is designed also to be maximally helpful to biologists by avoiding the problems which arise by virtue of the fact that the syntax favored in the DL-community is of a type which can normally be understood only by DL-specialists.

A theory of classes and instances

The relations in biological ontologies connect classes as their relata. The term 'class' here is used to refer to what is general in reality, or in other words to what, in the knowledge-representation literature, is typically (and often somewhat confusingly [13]) referred to under the heading 'concept' and in the literature of philosophical ontology under the headings 'universal', 'type' or 'kind'. Biological classes are in first approximation those classes which have been implicitly sanctioned through usage of the corresponding general terms in the biological literature, for example *cell* or *fat body development*.

Our task is to develop a suite of coherently defined bio-ontological relations that is sufficiently compact to be easily learned and applied, yet sufficiently broad in scope to capture a wide range of the relations currently coded in standard biomedical ontologies. Unfortunately the realization of this task is not a trivial matter. This is because, while the terms in biomedical ontologies refer exclusively to classes - to what is *general* in reality - we cannot define what it means for one class to stand to another, for example in the *part_of* relation, without taking the corresponding instances into account [6]. Here the term 'instance' refers to what is *particular* in reality, to what are otherwise called 'tokens' or 'individuals' - entities (including processes) which exist in space and time and stand to each other in a variety of instance-level relations. Thus we cannot make sense of what it means to say *cell nucleus part_of cell* unless we realize that this is a statement to the effect that each instance of the class *cell nucleus* stands in an instance-level part relation to some corresponding instance of the class *cell*.

This dependence of class-relations on relations among corresponding instances has long been recognized by logicians, including those working in the field of description logics, where the (*all* - *some*) form of definition we utilize below has been basic to the formalism from the start [14]. Definitions of this type were incorporated also into the DL-based GALEN medical ontology [15], though the significance of such definitions, and more generally of the role of instances in defining class relations, has still not been appreciated in many user communities.

It is also characteristically not realized that talk of classes involves in every case a more-or-less explicit reference to corresponding instances. When we assert that one class stands in an *is_a* relation to another (that is, that the first is a subtype of the second), for example, that *glucose metabolism is_a carbohydrate metabolism*, then we are stating that instances of the first class are *ipso facto* instances of the second. When we are dealing exclusively with *is_a* relations there is little reason to take explicit notice of this two-sided nature of ontological relations. When, however, we move to ontological relations of other types, then it becomes indispensable, if many characteristic families of errors are to be avoided, that the implicit reference to instances be taken carefully into account.

Types of relations

We focus here exclusively on genuinely ontological relations, which we take to mean relations that obtain between entities in reality, independently of our ways of gaining knowledge about such entities (and thus of our experimental methods) and independently of our ways of representing or processing such knowledge in computers. A relation like *annotates* is not ontological in this sense, as it links classes not to other classes in nature but rather to terms in a vocabulary that we ourselves have constructed. We focus also on general-purpose relations - relations which can be employed, in principle, in all biological ontologies - rather than on those specific relations (such as *genome_of* or *sequence_of* employed by OBO's Sequence Ontology) which apply only to biological entities of certain kinds. The latter will, however, need to be defined in due course in accordance with the methodology advanced here.

The ontologies in OBO are designed to serve as controlled vocabularies for expressing the results of biological science. Sentences of the form '*A relation B*' (where '*A*' and '*B*' are terms in a biological ontology and '*relation*' stands in for '*part_of*' or some similar expression) can thus be conceived as expressing general statements about the corresponding biological classes or types. Assertions about corresponding instances or tokens (for example about the mass of this particular specimen in this particular Petri dish), while indispensable to biological research, do not belong to the general statements of biological science and thus they fall outside the scope of OBO and similar ontologies as these are presented to the user as finished products.

Yet such assertions are still relevant to ontologies. For it turns out that it is only by means of a detour through instances that the definitions and rules for coding relations between classes can be formulated in an intuitive and unambiguous - and thus reliably applicable - way.

We can distinguish, in fact, the following three kinds of binary relations:

<class, class>: for example, the *is_a* relation obtaining between the class *SWR1 complex* and the class *chromatin remodeling complex*, or between the class *exocytosis* and the class *secretion*;

<instance, class>: for example, the relation **instance_of** obtaining between this particular vesicle membrane and the

class *vesicle membrane*, or between this particular instance of mitosis and the class *mitosis*;

<instance, instance>: for example, the relation of instance-level parthood (called **part_of** in what follows), obtaining between this particular vesicle membrane and the endomembrane system in the corresponding cell, or between this particular M phase of some mitotic cell cycle and the entire cell cycle of the particular cell involved.

Here classes and the relations between them are represented in *italic*; all other relations are picked out in **bold**.

Continuants and processes

The terms 'continuant' and 'process' are generalizations of GO's 'cellular component' and 'biological process' but applied to entities at all levels of granularity, from molecule to whole organism. Continuants are those entities which endure, or continue to exist, through time while undergoing different sorts of changes, including changes of place. Processes are entities that unfold themselves in successive temporal phases [16]. The terms 'continuant' and 'process' thus correspond to what, in the literature of philosophical ontology, are known respectively as 'things' (objects, endurants) and 'occurrents' (activities, events, perdurants) respectively. A continuant is what changes; a process is the change itself. The continuant classes relevant to biological ontologies include *molecule*, *cell*, *membrane*, *organ*; the process classes include *ion transport*, *cell division*, *fat body development*, *breathing*.

To formulate precise definitions of the <class, class> relations which form the target of ontology construction in biology we will need to employ a vocabulary that allows reference both to classes and to instances. For this we take advantage of the machinery of logic, and more specifically of the standard device of variables and quantifiers [17], using different sorts of variables to range across the classes and instances of continuants and processes, spatial regions and temporal instants, respectively. For the sake of intelligibility we use a semi-formal syntax, which can, however, be translated in a simple way into standard logical notation.

We use variables of the following sorts:

C, C_1, ... to range over continuant classes;

P, P_1, ... to range over process classes;

c, c_1, ... to range over continuant instances;

p, p_1, ... to range over process instances;

r, r_1, ... to range over three-dimensional spatial regions;

t, t_1, ... to range over instants of time.

In an expanded version of our formal machinery we will need also to incorporate further variables, ranging for example over temporal intervals, biological functions, attributes and values.

Note that continuants and processes form non-overlapping categories. This means in particular that no subtype or parthood relations cross the continuant-process divide. The tripartite structure of the GO recognizes this categorical exclusivity and extends it to functions also.

Continuants can be *material* (a mitochondrion, a cell, a membrane), or *immaterial* (a cavity, a conduit, an orifice), and this, too, is an exclusive divide. Immaterial continuants have much in common with spatial regions [18]. They are distinguished therefrom, however, in that they are *parts of organisms*, which means that, like material continuants, they move from one spatial region to another with the movements of their hosts.

The three-dimensional continuants that are our primary focus here typically have a top and a bottom, an anterior and a posterior, an interior and an exterior. Processes, in contrast, have a beginning, a middle and an end. Processes, but not continuants, can thus be partitioned along the time axis, so that, for example, your youth and your adulthood are temporal parts of that biological process which is your life.

As child and adult are continuants, so youth and adulthood are processes. We are thus clearly dealing here with two complementary - space-focused and time-focused - views of the same underlying subject matter, with determinate logical and ontological connections between them [16]. The framework advanced below allows us to capture these connections by incorporating reference to spatial regions and to temporal instants, both of which can be thought of as special kinds of instances.

We shall also need to distinguish two kinds of instance-level relations: those (applying to continuants) whose representations must involve a temporal index, and those (applying to processes) which do not. Note that the drawing of this distinction is still perfectly consistent with the fact that processes themselves occur in time, and that processes may be built out of successive subprocesses instantiating distinct classes.

Primitive instance-level relations

We cannot, on pain of infinite regress, define all relations, and this means that some relations must be accepted as primitive. The relations selected for this purpose should be self-explanatory and they should as far as possible be domain-neutral, which means that they should apply to entities in all regions of being and not just to those in the domain of biology.

Our choice of primitive relations is as follows:

c **instance_of** *C* **at** *t* - a primitive relation between a continuant instance and a class which it instantiates at a specific time

p **instance_of** *P* - a primitive relation between a process instance and a class which it instantiates holding independently of time

c **part_of** c_1 **at** *t* - a primitive relation between two continuant instances and a time at which the one is part of the other

p **part_of** p_1, *r* **part_of** r_1 - a primitive relation of parthood, holding independently of time, either between process instances (one a subprocess of the other), or between spatial regions (one a subregion of the other)

c **located_in** *r* **at** *t* - a primitive relation between a continuant instance, a spatial region which it occupies, and a time

r **adjacent_to** r_1 - a primitive relation of proximity between two disjoint continuants

t **earlier** t_1 - a primitive relation between two times

c **derives_from** c_1 - a primitive relation involving two distinct material continuants *c* and c_1

p **has_participant** *c* **at** *t* - a primitive relation between a process, a continuant, and a time

p **has_agent** *c* **at** *t* - a primitive relation between a process, a continuant and a time at which the continuant is causally active in the process

This list includes only those <instance-instance> relations, together with one <instance-class> relation, which are needed for defining the <class, class> relations which are our principal target in this paper. The items on the list have been selected because they enjoy a high degree of intelligibility to the human authors and curators of biological ontologies. For purposes of supporting computer applications, however, the meanings of the corresponding relational expressions must be specified formally via axioms, for example in the case of '**part_of**' by axioms of mereology (the theory of part and whole: see below), and in the case of '**earlier**' by axioms governing a linear order [17]. The relation **located_in** will satisfy axioms to the effect that for every continuant there is some region in which it is located; **instance_of** will satisfy axioms to the effect that all classes have (at some stage in their existence) instances, and that all instances are instances of some class.

The formal machinery for reasoning with such axioms is in place, and a comprehensive set of axioms is being compiled. For the typical human user of biological ontologies, however, the listed primitive relations and associated axioms are designed to work invisibly behind the scenes. That is, they serve as part of the background framework that guides the construction and maintenance of such ontologies.

Results

Methodology

We employed a multi-stage methodology for the selection of the relations to be included in this ontology and for the formulation of corresponding definitions. First, a sample of researchers involved in ontology construction in the life sciences, representing different groups and including the co-authors of this paper, was asked to prepare lists of principal relations in light of their own specific experience but focusing on relations which would be: 'ontological' in the sense introduced above; 'general-purpose' in the sense that they apply across all biological domains; and also such as to manifest a high degree of universality (in the sense explained in the section 'Types of relational assertions' below). The submitted lists manifested a significant degree of overlap, which allowed us to prepare a core list in whose terms a large number of the remaining relations on the list could be simply defined.

A further constraint on the process was the goal of providing a simple formal definition for each included <class-class> relation. Those relations for which an appropriate simple definition could not be agreed upon were not included in this interim list. This includes most conspicuously relations involving analogs of the GO notion of molecular function. The relation *has_agent* was, however, included in light of a common understanding that the notion of agency would be involved in whatever candidate definition of function in biology is eventually accepted for use in OBO. This further constraint was chosen in light of the fact that our capacity to provide simple formal definitions - definitions which will at one and the same time be intelligible to ontology authors and curators and also able to support logic-based tools for automatic reasoning and consistency-checking - is the primary rationale for the methodology here advanced.

The two relations *is_a* and *part_of* were unproblematic candidates for inclusion in the resulting list (though providing simple definitions even for these relations was not, as we shall see, a simple matter). *Is_a* and *part_of* have established themselves as foundational to current ontologies. They have a central role in almost all domain ontologies, including the Foundational Model of Anatomy (FMA) [19,20], GO and other ontologies in OBO, as well as in influential top-level ontologies such as DOLCE [21] and in digitalized lexical resources such as WordNet [22].

In preparing our sample lists we drew on representatives not only of the OBO consortium but also of GALEN and the FMA (itself a candidate for inclusion in OBO). Our temporal relations draw on existing OBO practice (where *transformation_of* is a generalization of the *develops_from*

Table 1

First version of the OBO Relation Ontology
Foundational relations
is_a
part_of
Spatial relations (connecting one entity to another in terms of relations between the spatial regions they occupy)
located_in
contained_in
adjacent_to
Temporal relations (connecting entities existing at different times)
transformation_of
derives_from
preceded_by
Participation relations (connecting processes to their bearers)
has_participant
has_agent

relation used in OBO's cell and anatomy ontologies) and our participation relations draw on current work addressing the need to provide relations that link entities in different ontologies (for example entities in GO's process, function and component ontologies) and on an evolving Physiology Reference Ontology that is being developed in conjunction with the FMA [23], from which our spatial relations were extracted.

The OBO Relation Ontology

The first proposed version of the OBO Relation Ontology is shown in Table 1. We shall deal here with each of the ten relations listed in Table 1 in turn, providing rigorous yet easily understandable definitions.

Is_a

It is commonly assumed in the literature of knowledge representation that the relation *is_a* (meaning 'is a subtype of') can be identified with the subset or set inclusion relation with which we are familiar from mathematical set theory [17]. **Instance_of** functions on this reading as a counterpart of the usual set-theoretic membership relation, yielding a definition of *A is_a B* along the lines of: for all *x*, if *x* **instance_of** *A*, then *x* **instance_of** *B*. Unfortunately, this reading provides at best a necessary condition for the truth of *A is_a B*. It falls short of providing a sufficient condition for two reasons. The first is because it admits cases of contingent inclusion such as: *bacterium in 90 mm × 18 mm glass Petri dish is_a bacterium*, and the second is because it fails to take account of time, so that when applied to classes of continuants it yields false positives such as *adult is_a child* (because every instance of *adult* was at some time an instance of *child*).

We resolve the first problem by admitting as *is_a* links only assertions that reflect truths of biological science - assertions involving genuine biological class names (such as 'enzyme' or 'apoptosis') rather than, for example, commercial or indexical names (such as 'bacterium in this Petri dish'). The second problem we resolve by exploiting our machinery for taking account of time in the assertion of *is_a* relations involving continuants.

We can then define:

C is_a C_1 = [definition] for all *c*, *t*, if *c* **instance_of** *C* **at** *t* then *c* **instance_of** C_1 **at** *t*.

P is_a P_1 = [definition] for all *p*, if *p* **instance_of** *P* then *p* **instance_of** P_1.

Note how the device of logical quantifiers (for all ..., for some ...) allows us to refer to instances 'in general' - which means without the need to call on the proper names or indexical expressions (such as 'this' or 'here') which we use when referring to instances 'in specific'. Note also how instantiation for continuants involves a temporal argument. This reflects the fact that continuants, but not processes, can instantiate different classes in the course of their existence and yet preserve their identity.

For simplicity of expression we shall henceforth write '*Cct*' and '*Pp*', as abbreviations for: '*c* **instance_of C at *t*** ' and '*p* **instance_of** *P* ', respectively.

Part_of

Parthood as a relation between instances. The primitive instance-level relation *p* **part_of** p_1 is illustrated in assertions such as: this instance of *rhodopsin mediated phototransduction* **part_of** this instance of *visual perception*.

This relation satisfies at least the following standard axioms of mereology: reflexivity (for all *p*, *p* **part_of** *p*); anti-symmetry (for all *p*, p_1, if *p* **part_of** p_1 and p_1 **part_of** *p* then *p* and p_1 are identical); and transitivity (for all *p*, p_1, p_2, if *p* **part_of** p_1 and p_1 **part_of** p_2, then *p* **part_of** p_2). Analogous axioms hold also for parthood as a relation between spatial regions.

For parthood as a relation between continuants, these axioms need to be modified to take account of the incorporation of a temporal argument. Thus for example the axiom of transitivity for continuants will assert that if *c* **part_of** c_1 **at** *t* and c_1 **part_of** c_2 **at** *t*, then also *c* **part_of** c_2 at *t*.

Parthood as a relation between classes. To define *part_of* as a relation between classes we again need to distinguish the two cases of continuants and processes, even though the explicit reference to instants of time now falls away. For continuants, we have *C part_of* C_1 if and only if any instance of *C* at any time is an instance-level part of some instance of C_1 at that time, as for example in: *cell nucleus part_ of cell*.

Formally:

C part_of C_1 = [definition] for all *c*, *t*, if *Cct* then there is some c_1 such that C_1c_1t and *c* **part_of** c_1 **at** *t*.

Note the 'all-some' structure of this definition, a structure which will recur in almost all the relations treated here.

C part_of C_1 defines a relational property of permanent parthood for *C*s. It tells us that *C*s, whenever they exist, exist as parts of C_1s. We can also define in the obvious way *C temporary_part_of* C_1 (every *C* exists at some time in its existence as part of some C_1) and also *C initial_part_of* C_1 (every *C* is such that it begins to exist as part of some instance of C_1).

For processes, we have by analogy, *P part_of* P_1 if and only if any instance of *P* is an instance-level part of some instance of P_1, as for example in: *M phase part_of cell cycle* or *neuroblast cell fate determination part_of neurogenesis*. Formally:

P part_of P_1 = [definition] for all *p*, if *Pp* then there is some p_1 such that: P_1p_1 and *p* **part_of** p_1.

An assertion to the effect that *P part_of* P_1 thus tells us that *P*s in general are in every case such as to exist as parts of P_1s. P_1s themselves, however, may exist without having *P*s as parts (consider: *menopause part_of aging*).

Note that *part_of* is in fact two relations, one linking classes of continuants, the other linking classes of processes. While both of the mentioned relations are transitive, this does not mean that *part_of* relations could be inferred which would cross the continuant-process divide.

Located_in

Location as a relation between instances. The primitive instance-level relation *c* **located_in** *r* **at** *t* reflects the fact that each continuant is at any given time associated with exactly one spatial region, namely its exact location [24]. Following [25] we can use this relation to define a further instance-level location relation - not between a continuant and the region which it exactly occupies, but rather between one continuant and another. *c* is located in c_1, in this sense, whenever the spatial region occupied by *c* is **part_of** the spatial region occupied by c_1. Formally:

c **located_in** c_1 at *t* = [definition] for some *r*, r_1, *c* **located_in** *r* **at** *t* and c_1 **located_in** r_1 **at** *t* and *r* **part_of** r_1.

Note that this relation comprehends both the relation of exact location between one continuant and another which obtains when *r* and r_1 are identical (for example, when a portion of fluid exactly fills a cavity), as well as those sorts of inexact location relations which obtain, for example, between brain and head or between ovum and uterus.

Location as a relation between classes. To define location as a relation between classes - represented by sentences such as *ribosome located_in cytoplasm*, *intracellular located_in cell* - we now set:

C located_in C_1 = [definition] for all *c*, *t*, if *Cct* then there is some c_1 such that C_1c_1t and *c* **located_in** c_1 **at** *t*.

Note that *C located_in* C_1 is an assertion about *C*s in general, which does not tell us anything about C_1s in general (for example, that they have *C*s located in them).

Contained_in

If *c* **part_of** c_1 **at** *t* then we have also, by our definition and by the axioms of mereology applied to spatial regions, *c* **located_in** c_1 **at** *t*. Thus, many examples of instance-level location relations for continuants are in fact cases of instance-level parthood. For material continuants location and parthood coincide. Containment is location not involving parthood, and arises only where some immaterial continuant is involved. To understand this relation, we first define overlap for continuants as follows:

C_1 **overlap** c_2 **at** *t* = [definition] for some *c*, *c* **part_of** c_1 **at** *t* and *c* **part_of** c_2 **at** *t*.

The containment relation on the instance level can then be defined as follows:

c **contained_in** c_1 **at** *t* = [definition] *c* **located_in** c_1 **at** *t* and not *c* **overlap** c_1 **at** *t*.

On the class level this yields:

C contained_in C_1 = [definition] for all *c*, *t*, if *Cct* then there is some c_1 such that: C_1c_1t and *c* **contained_in** c_1 **at** *t*.

Containment obtains in each case between material and immaterial continuants, for instance: *lung contained_in thoracic cavity*; *bladder contained_in pelvic cavity*. Hence containment is not a transitive relation.

Adjacent_to

We can define additional spatial relations by appealing to the primitive **adjacent_to**, a relation of proximity between disjoint continuants. **Adjacent_to** satisfies some of the axioms

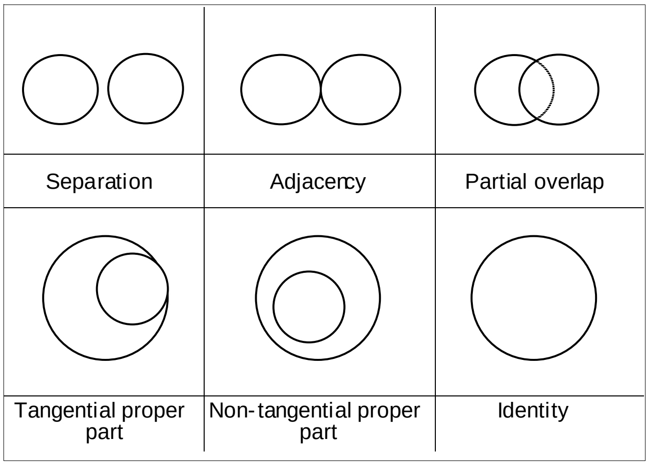


Figure 1
Standard mereotopological relations between spatial regions.

governing the relation referred to in the literature of qualitative topology as 'external connectedness' [26]. Analogs of other mereotopological relations (qualitative relations between spatial regions involving parthood, boundary and connectedness) (Figure 1) can also be defined, and these too can be applied to the material and immaterial continuants which occupy such regions on the instance level.

We define overlap for spatial regions as follows:

r_1 **overlap** r_2 = [definition] for some r, r **part_of** r_1 and r **part_of** r_2.

We then assert axiomatically that r_1 **adjacent_to** r_2 implies not r_1 **overlap** r_2

We can then define the counterpart relation of adjacency between classes as follows:

C adjacent_to C_1 = [definition] for all *c*, *t*, if *Cct*, there is some c_1 such that: C_1c_1t and *c* **adjacent_to** c_1 **at** *t*.

Note that *adjacent_to* as thus defined is not a symmetric relation, in contrast to its instance-level counterpart. For it can be the case that *C*s are in general such as to be adjacent to instances of C_1 while no analogous statement holds for C_1s in general in relation to instances of *C*. Examples are:

nuclear membrane adjacent_to cytoplasm

seminal vesicle adjacent_to urinary bladder

ovary adjacent_to parietal pelvic peritoneum.

We can, however, very simply define a symmetric relation of co-adjacency on the class level as follows:

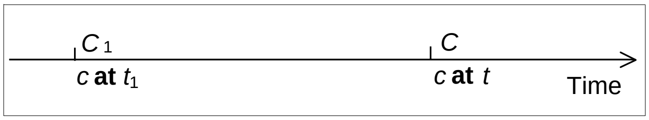


Figure 2
Transformation.

C_1 *co-adjacent_to* C_2 = [definition] C_1 *adjacent_to* C_2 and C_2 *adjacent_to* C_1.

Examples are:

inner layer of plasma membrane co-adjacent_to outer layer of plasma membrane

right pulmonary artery co-adjacent_to right principal bronchus

urinary bladder of female co-adjacent_to parietal peritoneum of female pelvis.

Transformation_of

When an embryonic oenocyte (a type of insect cell) is transformed into a larval oenocyte, one and the same continuant entity preserves its identity while instantiating distinct classes at distinct times. The class-level relation *transformation_of* obtains between continuant classes *C* and C_1 wherever each instance of the class *C* is such as to have existed at some earlier time as an instance of the distinct class C_1 (see Figure 2). This relation is illustrated first of all at the molecular level of granularity by the relation between *mature RNA* and the *pre-RNA* from which it is processed, or between (*UV-induced*) *thymine-dimer* and *thymine dinucleotide*. At coarser levels of granularity it is illustrated by the transformations involved in the creation of red blood cells, for example, from *reticulocyte* to *erythrocyte*, and by processes of development, for example, from *larva* to *pupa*, or from (post-gastrular) *embryo* to *fetus* [27] or from *child* to *adult*. It is also manifest in pathological transformations, for example, of *normal colon* into *carcinomatous colon*. In each such case, one and the same continuant entity instantiates distinct classes at different times in virtue of phenotypic changes.

As definition for this relation we offer:

C transformation_of C_1 = [definition] *C* and C_1 for all *c*, *t*, if *Cct*, then there is some t_1 such that C_1ct_1, and t_1 **earlier** *t*, and there is no t_2 such that Cct_2 and C_1ct_2.

That is to say, the class *C* is a transformation of the class C_1 if and only if every instance *c* of *C* is at some earlier time an instance of C_1, and there is no time at which it is an instance of both *C* and C_1. (The final clause, which asserts that *C* and C_1

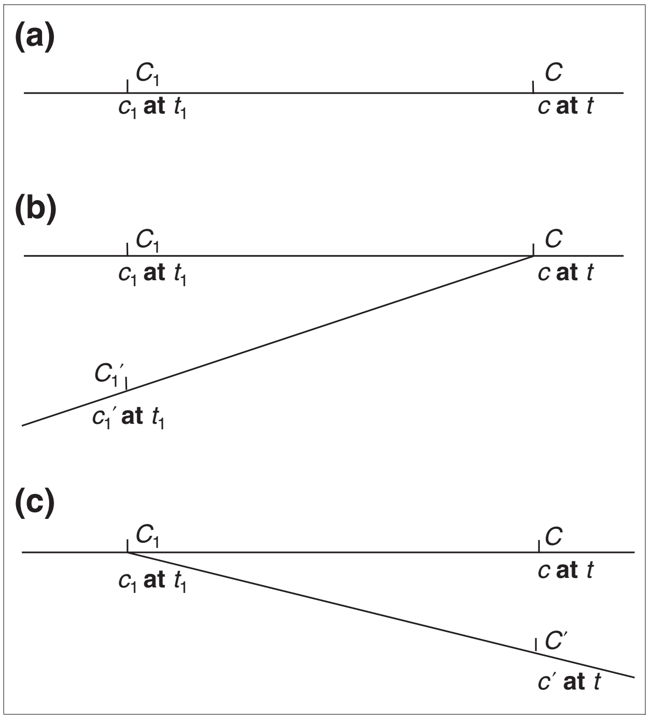


Figure 3
Three simple cases of derivation. **(a)** Continuation; **(b)** fusion; **(c)** fission.

do not share instances at a time, is inserted in order to rule out, for example, *adult human transformation_of human*.)

Note that *C transformation_of* C_1 is a statement about *C*s in general. It does not tell us of C_1s in general that each gives rise to some *C* which stands to it in a *transformation_of* relation.

Derives_from

Derivation as a relation between instances. The temporal relation of derivation is more complex. Transformation, on the instance level, is just the relation of identity: each adult is identical to some child existing at some earlier time. Derivation on the instance-level is a relation holding between non-identicals. More precisely, it holds between distinct material continuants when one succeeds the other across a temporal divide in such a way that at least a biologically significant portion of the matter of the earlier continuant is inherited by the later. Thus we will have axioms to the effect that from *c* **derives_from** c_1 we can infer that *c* and c_1 are not identical and that there is some instant of time *t* such that c_1 exists only prior to and *c* only subsequent to *t*. We will also be able to infer that the spatial region occupied by *c* as it begins to exist at *t* overlaps with the spatial region occupied by c_1 as it ceases to exist in the same instant.

Three simple kinds of instance-level derivation can then be distinguished (Figure 3): first, the succession of one single continuant by another single continuant across a temporal threshold (for example, this blastocyst derives from this zygote); second, the fusion of two or more continuants into one continuant (for example, this zygote derives from this sperm and from this ovum); and third, the fission of an earlier single continuant to create a plurality of later continuants (for example, these promyelocytes derive from this myeoloblast). In all cases we have two continuants *c* and c_1 which are such that *c* begins to exist at the same instant of time at which c_1 ceases to exist, and at least a significant portion of the matter of c_1 is inherited by its successor *c*.

Derivation of the first type is still essentially weaker than transformation, for the latter involves the identity of the continuant instances existing on either side of the relevant temporal divide. In derivation of the second type, the successor continuant takes the bulk of its matter from a plurality of precursors, where in cases of the third type, the bulk of the matter of a single precursor continuant is shared among a plurality of successors. We can also represent more complex cases where transformation and an analog of derivation are combined, for example in the case of *budding* in yeast [27], where one continuant continues to exist identically through a process wherein a second continuant floats free from its host; or in *absorption*, where one continuant continues to exist identically through a process wherein it absorbs another continuant, for example through digestion.

Derivation as a relation between classes. To avoid troubling counter-examples, the relation of derivation we are seeking on the class level must be defined in two steps. First, the class-level counterpart of the relation of derivation on the instance level is identified as a relation of immediate derivation:

C derives_immediately_from C_1 = [definition] for all *c*, *t*, if *Cct*, then there is some c_1,t_1, such that: t_1 **earlier** *t* and $C_1c_1t_1$ and *c* **derives_from** c_1.

The more general class level derivation relation must then be defined in terms of chains of immediate derivation relations, as follows:

C derives_from C_1 = [definition] there is some sequence $C = C_k, C_{k-1}, ..., C_2, C_1$, such that for each C_i ($1 \leq i < k$), C_{i+1} *derives_immediately_from* C_i.

In this way we can represent cases of derivation involved in the formation of lineages where there occurs a sequence of cell divisions or speciation events.

Preceded_by

With the primitive relations **has_participant** and **earlier** at our disposal we can define the instance-level relation *p* **occurring_at** *t* as follows:

p **occurring_at** *t* = [definition] for some *c*, *p* **has_participant** *c* **at** *t*.

comment
reviews
reports
deposited research
refereed research
interactions
information

We can then define:

c **exists_at** *t* = [definition] for some *p, p* **has_participant** *c* **at** *t*

p **preceded_by** p_1 = [definition] for all *t,* t_1, if *p* **occurring_at** *t* and p_1 **occurring_at** t_1, then t_1 **earlier** *t*

t **first_instant** *p* = [definition] *p* **occurring_at** *t* and for all t_1, if t_1 **earlier** *t*, then not *p* **occurring_at** t_1

t **last_instant** *p* = [definition] *p* **occurring_at** *t* and for all t_1, if *t* **earlier** t_1, then not *p* occurring_at t_1

p **immediately_preceded_by** p_1 = [definition] for some *t, t* **first_instant** *p* and *t* **last_instant** p_1.

At the class level we have:

P preceded_by P_1 = [definition] for all *p*, if *Pp* then there is some p_1 such that $P_1p_1$and *p* **preceded_by** p_1.

An example is: *translation preceded_by transcription*; *aging preceded_by development* (not however *death preceded_by aging*). Where *derives_from* links classes of continuants, *preceded_by* links classes of processes. Clearly, however, these two relations are not independent of each other. Thus if cells of type C_1 *derive_from* cells of type *C*, then any cell division involving an instance of C_1 in a given lineage is *preceded_by* cellular processes involving an instance of *C*.

The assertion *P preceded_by* P_1 tells us something about *P*s in general: that is, it tells us something about what happened earlier, given what we know about what happened later. Thus it does not provide information pointing in the opposite direction, concerning instances of P_1 in general; that is, that each is such as to be succeeded by some instance of *P*. Note that an assertion to the effect that *P preceded_by* P_1 is rather weak; it tells us little about the relations between the underlying instances in virtue of which the *preceded_by* relation obtains. Typically we will be interested in stronger relations, for example in the relation *immediately_preceded_by*, or in relations which combine *preceded_by* with a condition to the effect that the corresponding instances of *P* and P_1 share participants, or that their participants are connected by relations of derivation, or (as a first step along the road to a treatment of causality) that the one process in some way affects (for example, initiates or regulates) the other.

Has_participant

Has_participant is a primitive instance-level relation between a process, a continuant, and a time at which the continuant participates in some way in the process. The relation obtains, for example, when this particular process of oxygen exchange across this particular alveolar membrane **has_participant** this particular sample of hemoglobin at this particular time.

To define the class-level counterpart of the participation relation we set:

P has_participant C = [definition] for all *p*, if *Pp* then there is some *c*, *t* such that *Cct* and *p* **has_participant** *c* **at** *t*.

Examples are:

cell transport has_participant cell

death has_participant organism

breathing has_participant thorax.

Once again, *P has_participant C* provides information only about *P*s in general (that is, that they require instances of *C* as bearers).

Has_agent

Special types of participation can be distinguished according to whether a continuant is agent or patient in a process (for a survey see [28].) Here we focus on the factor of agency, which is involved, for example, when an adult engages in adult walking behavior. It is not involved when the same adult is the victim of an infection. Synonyms of 'is agent in' include: 'actively participates in', 'does', 'executes', 'performs', and so forth.

We introduce the primitive instance-level relation **has_agent**, which obtains between a process, a continuant and a time whenever the continuant is a participant in the process and is at the same time directly causally responsible for its occurrence. Thus we have an axiom to the effect that agency implies participation: for all *p*, *c*, *t*, if *p* **has_agent** *c* **at** *t*, then *p* **has_participant** *c* **at** *t*. In addition we will have axioms to the effect that only material continuants can fill the agent role, that if *c* fills the agent role at *t*, then *c* must have existed at times earlier than *t,* that it must exercise its agent role for an interval of time including *t*, and so on.

We can then define the class-level relation *has_agent* by stipulating:

P has_agent C = [definition] for all *p*, if *Pp* then there is some *c*, *t* such that *Cct* and *p* **has_agent** *c* **at** *t*

This relation gives us the means to capture the directionality (the from-to) nature of biological processes such as signaling, transcription, and expression, via assertions, for example, to the effect that in an interaction between molecules of types m_1 and m_2 it is molecules of the first type that play the role of agent.

One privileged type of agency consists in the realization of a biological function. To say that a continuant has a function is to assert, in first approximation, that it is predisposed (has the potential, the casual power) to cause (to realize as agent) a process of a certain type. Thus to say that your heart has the function: *to pump blood* is to assert that your heart is predisposed to realize as agent a process of the type *pumping blood* [29]. Regulation, promotion, inhibition, suppression, activation, and so forth, are among the varieties of agency that fall under this heading.

On the other hand, many processes - such as metabolic reactions involving enzymes, cofactors, and metabolites - involve no clear factor of agent participation, but rather require more nuanced classifications of the roles of participants - as acceptors or donors, for example. Hence the *has_agent* relation should be used in curation with special care. It should be borne in mind in this connection that agency is in every case a matter of the imposition of direct causal influence of a continuant in a process (a constraint that is designed to rule out inheritance of agency along causal chains), and also that (by our definition) only continuants can be agents. Where biologists describe processes as agents, for example, in talking about the effects of diffusion in development and differentiation, such phenomena are of a type that call for an expansion of our proposed Relation Ontology in the direction, again, of a treatment of the factor of causality.

Discussion

The logic of biological relations

Inverse and reciprocal relations

The inverse of a relation R is defined as that relation which obtains between each pair of relata of R when taken in reverse order. Inverses can be unproblematically defined for all instance-level relations. What, then, of inverses for class-level relations? The inverse relation for *is_a* can be defined trivially as follows:

A has_subclass B = [definition] *B is_a A*.

For the remaining class-level relations on our list, in contrast, the issue of corresponding inverses is more problematic [7]. Thus, while we have the true relational assertion *human testis part_of human* - which means that all instances of *human testis* are part of instances of some *human* - there is no corresponding true relational assertion linking instances of *human* to instances of *human testis* as their parts. For these remaining relations we need to work not with inverses but rather with what, following GALEN, we can call reciprocal relations. These are defined using the same family of instance-level primitives we introduced earlier. As reciprocal relations for the two varieties of *part_of* we have:

C has_part C_1 = [definition] for all *c*, *t*, if *Cct* then there is some c_1 such that C_1c_1t and c_1 **part_of** *c* **at** *t*

P has_part P_1 = [definition] for all *p*, if *Pp* then there is some p_1 such that P_1p_1 and p_1 **part_of** *p*

Note that from *A part_of B* we cannot infer that *B has_ part A*; similarly, from *A has_ part B* we cannot infer that *B part_of A*. Thus *cell nucleus part_of cell*, but not *cell has_part cell nucleus*; *running has_ part breathing*, but not *breathing part_of running*. A third significant relation conjoining *part_of* and *has_part* can be defined as [6,30]:

C integral_part_of C_1 = [definition] *C part_of* C_1 and C_1 *has_part C*.

For *contained_in* we have similarly the reciprocal relation:

C contains C_1 = [definition] for all *C*, *t*, if *Cct* then there is some c_1 such that: C_1c_1t and *c* **located_in** *c* **at** *t*

For participation we can usefully define two alternative reciprocal relations:

C sometimes_ participates_in P = [definition] for all *c* there is some *t* and some *p* such that *Cct* and *Pp* and *p* **has_participant** *c* **at** *t*

C always_participates_in P = [definition] for all *c*, *t*, if *Cct* then there is some *p* such that *Pp* and *p* **has_participant** *c* **at** *t*

We can also define, for example, what it is for continuants of a given type to participate at every stage in a process of a given type. Thus if a sperm participates in the penetration of an ovum, then it does so throughout the penetration.

Types of relational assertions

In light of the above, we can now observe certain differences in what we might call the relative universality of class-level relational assertions. There are many cases, above all involving *is_a* relations, where relational assertions hold with a maximal degree of universality, which means that they hold for every instance of the classes in question because they are a matter of analytic connections, that is, connections resting on the compositional nature of the class terms involved [10], as, for example, in: *eukaryotic cell is_a cell*, or *adult walking behavior has_participant adult*. (Contrast, *adult participates_in adult walking behavior*.)

There are also other kinds of statements enjoying a high degree of universality, for example: *penetration of ovum has_participant sperm*. The first of our two corresponding reciprocal statements - *sperm participates_in penetration of ovum* - is in contrast true only in relation to certain isolated instances of *sperm*, and the second of our reciprocal statements - *sperm always_participates_in penetration of ovum* - is true in relation to no instances at all.

comment
reviews
reports
deposited research
refereed research
interactions
information

Table 2

Definitions and examples of class-level relations

Relations and relata	Definitions	Examples
C *is_a* C_1; Cs and C_1s are continuants	Every C at any time is at the same time a C_1	*myelin is_a lipoprotein* *serotonin is_a biogenic amine* *mitochondrion is_a membranous cytoplasmic organelle* *protein kinase is_a kinase* *DNA is_a nucleic acid*
P *is_a* P_1; Ps and P_1s are processes	Every P is a P_1	*endomitosos is_a DNA replication* *catabolic process is_a metabolic process* *photosynthesis is_a physiological process* *gonad development is_a organogenesis* *intracellular signaling cascade is_a signal transduction*
C *part_of* C_1; Cs and C_1s are continuants	Every C at any time is part of some C_1 at the same time	*mitochondrial matrix part_of mitochondrion* *microtubule part_of cytoskeleton* *nuclear pore complex part_of nuclear membrane* *nucleoplasm part_of nucleus* *promotor part_of gene*
P *part_of* P_1; Ps and P_1s are processes	Every P is part of some P_1	*gastrulation part_of embryonic development* *cystoblast cell division part_of germ cell development* *cytokinesis part_of cell proliferation* *transcription part_of gene expression* *neurotransmitter release part_of synaptic transmission*
C *located_in* C_1; Cs and C_1s are continuants	Every C at any given time occupies a spatial region which is part of the region occupied by some C_1 at the same time	*66s pre-ribosome located_in nucleolus* *intron located_in gene* *nucleolus located_in nucleus* *membrane receptor located_in cell membrane* *chlorophyll located_in thylakoid*
C *contained_in* C_1; Cs are material continuants, C_1s are immaterial continuants (holes, cavities)	Every C at any given time is located in but shares no parts in common with some C_1 at the same time	*thoracic aorta contained_in posterior mediastinal cavity* *cytosol contained_in cell compartment space* *thylakoid contained_in chloroplast membrane* *synaptic vesicle contained_in neuron*
C *adjacent_to* C_1; Cs and C_1s are continuants	Every C at any time is proximate to some C_1 at the same time	*Golgi apparatus adjacent_to endoplasmic reticulum* *intron adjacent_to exon* *cell wall adjacent_to cytoplasm* *periplasm adjacent_to plasma membrane* *presynaptic membrane adjacent_to synaptic cleft*
C *transformation_of* C_1; Cs and C_1s are material continuants	Every C at any time is identical with some C_1 at some earlier time	*facultative heterochromatin transformation_of euchromatin* *mature mRNA transformation_of pre-mRNA* *hemosiderin transformation_of hemoglobin* *red blood cell transformation_of reticulocyte* *fetus transformation_of embryo*

Table 2 *(Continued)*

Definitions and examples of class-level relations

C derives_from C_1; *C*s and C_1s are material continuants	Every *C* is such that in the first moment of its existence it occupies a spatial region which overlaps the spatial region occupied by some C_1 in the last moment of its existence	*plasma cell derives_from B lymphocyte* *fatty acid derives_from triglyceride* *triple oxygen molecule derives_from oxygen molecule* *Barr body derives_from X-chromosome* *mammal derives_from gamete*
P preceded_by P_1; *P*s and P_1s are processes	Every *P* is such that there is some earlier P_1	*translation preceded_by transcription* *meiosis preceded_by chromosome duplication* *cytokinesis preceded_by DNA replication* *apoptotic cell death preceded_by nuclear chromatin degradation* *digestion preceded_by ingestion*
P has_participant C; *P*s are processes, *C*s are continuants	Every *P* involves some *C* as participant	*mitochondrial acetylCoA formation has_participant pyruvate dehydrogenase complex* *translation has_participant amino acid* *photosynthesis has_participant chlorophyll* *apoptosis has_participant cell* *cell division has_participant chromosome*
P has_agent C; *P*s are processes, *C*s are material continuants	Every *P* involves some *C* as agent (the *C* is involved in and is causally responsible for the *P*)	*gene expression has_agent RNA polymerase* *signal transduction has_agent receptor* *pathogenesis has_agent pathogen* *transcription has_agent RNA polymerase* *translation has_agent ribosome*

It then seems reasonable to insist that biomedical ontologies should reflect those sorts of biological assertions that enjoy a high degree of universality (typically assertions involving just one of each pair of reciprocal relations).

Tools for ontology curation

We hope that, by providing clear and unambiguous specifications of what the class-level relational expressions used in biological ontologies mean, our formal definitions will assist curators engaged in ontology creation and maintenance. The corresponding definitions are summarized in Table 2, which also contains representative examples for each of the relations distinguished.

Our definitions are designed to ensure that the corresponding general-purpose relational expressions are used in a uniform way in all biological ontologies. In this way we shall be in a position to contribute to the realization of the goal of bringing about a high degree of interoperability even where ontologies are produced by different groups and for different purposes. These definitions are designed also to enable the automatic detection of errors in biomedical ontologies, for example by allowing the construction of extensions of OBO-Edit and similar tools with the facility to test whether given relations are employed in an ontology in such a way as to involve relata of the appropriate types [31] or in such a way as to have the formal characteristics, such as transitivity or reflexivity, dictated by the definitions (Table 3). The framework can also support reasoning applications designed to enable the automated derivation of information from existing bodies of knowledge - for example to infer the parts of a given cell continuant via the traversal of a *part_of* hierarchy - including instance-based knowledge derived from the clinical record.

Conclusion

The Relation Ontology outlined above arose through collaboration between formal ontologists and biologists in the OBO, FMA and GALEN research groups and also incorporates suggestions from a number of other authors and curators of biomedical ontologies. It is designed to be large enough to overcome some of the problems arising in GO and similar systems as a result of the paucity of resources available hitherto for expressing relations between the classes in such ontolo-

Table 3

Some properties of the relations in the OBO Relation Ontology

Relation	Transitive	Symmetric	Reflexive	Antisymmetric
is_a	+	-	+	+
part_of	+	-	+	+
located_in	+	-	+	-
contained_in	-	-	-	-
adjacent_to	-	-	-	-
transformation_of	+	-	-	-
derives_ from	+	-	-	-
preceded_by	+	-	-	-
has_participant	-	-	-	-
has_agent	-	-	-	-

gies [32]. It is this paucity of resources, above all, which gives rise to cases of multiple inheritance in GO as presently constructed, and we note here that multiple inheritance often goes hand in hand with errors in ontology construction not least because it encourages a relaxed reading of *is_a* (often a reading which involves the assertion of *is_a* relations which erroneously cross the divide between different ontological categories) [5,33]. Our present framework can contribute to error resolution not only by dictating a common interpretation of *is_a* which can serve as orientation for ontology authors and curators in their future work, but also by providing richer resources for the assertion of class-class relations within and between ontologies in such a way that the appeal to contrived and error-prone *is_a* relations can be more easily avoided.

At the same time our suite of relations has been designed to be sufficiently small to attract wide acceptance in a range of different types of life-science communities. Where the latter use further, general-purpose or domain-specific relations of their own, we plan in due course to subject such relations to the same kind of analysis as presented here in order to preserve interoperability. The Relation Ontology has been incorporated into the OBO ontology library [34] and curators of the GO and FMA ontologies and also of the ChEBI chemical entities vocabulary [35] are already applying the relevant parts of the ontology in their work. The ontology has already been used to find errors not only in GO but also in SNOMED [36]. It is also being applied systematically in evaluations of the NCI Thesaurus [37] and the UMLS (Unified Medical Language System) Semantic Network of the National Library of Medicine. We are currently testing methodologies to obtain reliable quantitative evaluations of the utility of the proposed framework for purposes of ontology authoring and also for use in annotation and reasoning. We are also testing ways in which the framework can be expanded through the admission of pre-coordinated disjunctions (for example: *either derivation or transformation*), which can allow the coding of information in those cases where the precise nature of the relations involved is insufficiently clear to allow unique assignment.

The Relation Ontology will be evaluated on two levels. First, on whether it succeeds in preventing those characteristic kinds of errors which have been associated with a poor treatment of relations in biomedical ontologies in the past. Second, and more important, on whether it helps to achieve greater interoperability of biomedical ontologies and thus to improve reasoning about biological phenomena.

Acknowledgements

Work on this paper was carried out under the auspices of the Wolfgang Paul Program of the Alexander von Humboldt Foundation, the EU Network of Excellence in Medical Informatics and Semantic Data Mining, the Project 'Forms of Life' sponsored by the Volkswagen Foundation, and the DARPA Virtual Soldier Project. Thanks go to Michael Ashburner, Fabrice Correia, Maureen Donnelly, Kai Hauser, Win Hyde, Ingvar Johansson, Janet Kelso, Suzanna Lewis, Katherine Munn, Maria Reicher, Alan Ruttenberg, Mark Scala, Stefan Schulz, Neil Williams, Lina Yip, Sumi Yoshikawa, and anonymous referees for valuable comments.

References

1. **OBO: Open Biomedical Ontologies** [http://obo.sourceforge.net]
2. Mungall C: **OBOL: integrating language and meaning in bio-ontologies.** *Comp Funct Genomics* 2004, **5:**509-520.
3. Gene Ontology Consortium: **Creating the Gene Ontology resource: design and implementation.** *Genome Res* 2001, **11:**1425-1433.
4. Bada M, Stevens R, Goble C, Gil Y, Ashburner M, Blake JA, Cherry JM, Harris M, Lewis S: **A short study on the success of the GeneOntology.** *J Web Semantics* 2004, **1:**235-240.
5. Smith B, Köhler J, Kumar A: **On the application of formal principles to life science data: a case study in the Gene Ontology.** *DILS 2004: Data Integration in the Life Sciences. Lecture Notes in Computer Science 2994* 2004:124-139.
6. Smith B, Rosse C: **The role of foundational relations in the alignment of biomedical ontologies.** In *Proceedings Medinf 2004* Amsterdam: IOS Press; 2004:444-448.
7. Smith B, Kumar A: **On controlled vocabularies in bioinformatics: a case study in the Gene Ontology.** *BioSilico: Inform Technol Drug Discovery* 2004, **2:**246-252.
8. Smith B, Williams J, Schulze-Kremer S: **The ontology of the Gene Ontology.** *Proc AMIA Symp* 2003:609-13.
9. Ogren PV, Cohen KB, Acquaah-Mensah GK, Eberlein J, Hunter L: **The compositional structure of Gene Ontology terms.** *Pac Symp Biocomput* 2004:214-225.
10. Ogren P, Bretonnel Cohen K, Hunter L: **Implications of compositionality in the Gene Ontology for its curation and usage.** *Pac Symp Biocomput* 2005:174-185.
11. Bard J, Rhee SY, Ashburner M: **An ontology for cell types.** *Genome Biol* 2005, **6:**R21.
12. Wroe C, Stevens R, Goble CA, Ashburner M: **An evolutionary methodology to migrate the Gene Ontology to a Description Logic environment using DAML+OIL.** *Pac Symp Biocomput* 2003:624-635.
13. Smith B: **Beyond concepts: ontology as reality representation.** In *Formal Ontology and Information Systems 2004* Amsterdam: IOS Press; 2004:73-84.
14. Levesque HJ, Brachman RJ: **A fundamental tradeoff in knowledge representation and reasoning.** In *Readings in Knowledge Representation* San Francisco: Morgan Kaufman; 1985:41-70.
15. Rogers J, Rector AL: **The GALEN ontology.** In *Medical Informatics Europe 1996* Amsterdam: IOS Press; 1996:174-178.

16. Grenon P, Smith B, Goldberg L: **Biodynamic ontology: applying BFO in the biomedical domain.** In *Ontologies in Medicine* Amsterdam: IOS Press; 2004:20-38.
17. Stoll R: *Set Theory and Logic* New York: Dover Publications; 1979.
18. Casati R, Varzi AC: *Holes and Other Superficialities* Cambridge, MA: MIT Press; 1994.
19. Rosse C, Mejino JLV Jr: **A reference ontology for bioinformatics: the Foundational Model of Anatomy.** *J Biomed Inform* 2003, **36:**478-500.
20. Rogers J, Rector AL: **GALEN's model of parts and wholes: experience and comparisons.** In *Proceedings AMIA Symposium 2000* Bethesda, MD: American Medical Informatics Association; 2000:819-823.
21. Gangemi A, Guarino N, Masolo C, Oltramari A: **Sweetening WordNet with DOLCE.** *AI Magazine* 2003, **24:**13-24.
22. Fellbaum C, Ed: *Wordnet. An Electronic Lexical Database* Cambridge, MA: MIT Press; 1998.
23. Cook DL, Mejino JLV Jr, Rosse C: **Evolution of a Foundational Model of Physiology: symbolic representation for functional bioinformatics.** In *Proceedings MedInfo 2004* Amsterdam: IOS Press; 2004:336-340.
24. Bittner T: **Axioms for parthood and containment relations in bio-ontologies.** In *KR-MED 2004: Workshop on Formal Biomedical Knowledge Representation* Aachen: University of Aachen; 2004:4-11.
25. Donnelly M: **Layered mereotopology.** In *Proceedings 18th Joint International Conference on Artificial Intelligence* San Francisco: Morgan Kaufman; 2003:1269-1274.
26. Smith B: **Mereotopology: a theory of parts and boundaries.** *Data Knowledge Eng* 1996, **20:**287-303.
27. Smith B, Brogaard B: **Sixteen days.** *J Med Philos* 2003, **28:**45-78.
28. Smith B, Grenon P: **The cornucopia of formal-ontological relations.** *Dialectica* 2004, **58:**279-296.
29. Johansson I, Smith B, Munn K, Tsikolia N, Elsner K, Ernst D, Siebert D: **Functional anatomy: a taxonomic proposal.** *Acta Biotheoret* 2005 in press.
30. Schulz S, Hahn U: **Towards a computational paradigm for biomedical structure.** In *KR-MED 2004: Workshop on Formal Biomedical Knowledge Representation* Aachen: University of Aachen; 2004:63-71.
31. dos Santos MC, Dhaen C, Fielding M, Ceusters W: **Philosophical scrutiny for run-time support of application ontology development.** In *Formal Ontology and Information Systems* Amsterdam: IOS Press; 2004:342-352.
32. Kumar A, Smith B, Borgelt C: **Dependence relationships between Gene Ontology terms based on TIGR gene product annotations.** In *Proceedings CompuTerm 2004* Geneva: COLING; 2004:31-38.
33. Bouaud J, Bachimont B, Charlet J, Zweigenbaum P: **Acquisition and structuring of an ontology within conceptual graphs.** *Proceedings 2nd International Conference on Conceptual Structures: Workshop on Knowledge Acquisition using Conceptual Graph Theory. Lecture Notes Computer Sci* 1994, **835:**1-25.
34. **OBO Relationship Ontology** [http://obo.sourceforge.net/relationship]
35. **ChEBI: Chemical Entities of Biological Interest** [http://www.ebi.ac.uk/chebi]
36. Ceusters W, Smith B, Kumar A, Dhaen C: **Ontology-based error detection in SNOMED-CT.** In *Proceedings Medinfo 2004* Amsterdam: IOS Press; 2004:482-486.
37. Ceusters W, Smith B, Goldberg L: **A terminological and ontological analysis of the NCI Thesaurus.** *Meth Inform Medicine.* 2005, in press.

WRAP UP QUESTIONS AND FURTHER READING

Goal: *The final section will be devoted to clarifying doubts, providing suggestions on particular use-cases the participants might have as well as identifying potential collaborators amongst the participants themselves.*

REFERENCES

1. Hey, T. and A.E. Trefethen, *Cyberinfrastructure for e-Science.* Science, 2005. **308**(5723): p. 817-21.
2. Fedoroff, N., S.A. Racunas, and J. Shrager, *Making Biological Computing Smarter*, in *The Scientist*. 2005. p. 20-21.
3. Cimino, J.J. and X. Zhu, *The practical impact of ontologies on biomedical informatics.* Methods Inf Med, 2006. **45 Suppl 1**: p. 124-35.
4. Yu, A.C., *Methods in biomedical ontology.* J Biomed Inform, 2006. **39**(3): p. 252-66.
5. Bodenreider, O. and R. Stevens, *Bio-ontologies: current trends and future directions.* Brief Bioinform, 2006. **7**(3): p. 256-74.
6. Ashburner, M., et al., *Gene ontology: tool for the unification of biology. The Gene Ontology Consortium.* Nat Genet, 2000. **25**(1): p. 25-9.
7. Khatri, P. and S. Draghici, *Ontological analysis of gene expression data: current tools, limitations, and open problems.* Bioinformatics, 2005. **21**(18): p. 3587-95.
8. *GO Slim*. [Web page] 2003, [cited 2003; Available from: http://www.geneontology.org/GO.slims.shtml.
9. Alterovitz, G., et al., *GO PaD: the Gene Ontology Partition Database.* Nucleic Acids Res, 2007. **35**(Database issue): p. D322-7.
10. Lowe, H.J. and G.O. Barnett, *Understanding and using the medical subject headings (MeSH) vocabulary to perform literature searches.* Jama, 1994. **271**(14): p. 1103-8.
11. Bodenreider, O., *Using UMLS semantics for classification purposes.* Proc AMIA Symp, 2000: p. 86-90.
12. Hersh, W., et al., *Evaluation of SAPHIRE: an automated approach to indexing and retrieving medical literature.* Proc Annu Symp Comput Appl Med Care, 1991: p. 808-12.
13. Rubin, D.L., et al., *A statistical approach to scanning the biomedical literature for pharmacogenetics knowledge.* J Am Med Inform Assoc, 2005. **12**(2): p. 121-9.
14. Suomela, B.P. and M.A. Andrade, *Ranking the whole MEDLINE database according to a large training set using text indexing.* BMC Bioinformatics, 2005. **6**: p. 75.
15. Hartel, F.W., et al., *Modeling a description logic vocabulary for cancer research.* J Biomed Inform, 2005. **38**(2): p. 114-29.
16. Sioutos, N., et al., *NCI Thesaurus: a semantic model integrating cancer-related clinical and molecular information.* J Biomed Inform, 2007. **40**(1): p. 30-43.
17. Shah, N., et al., *Ontology-based Annotation and Query of Tissue Microarray Data.* AMIA Annu Symp Proc, 2006: p. 709-13.
18. Bodenreider, O., *The Unified Medical Language System (UMLS): integrating biomedical terminology.* Nucleic Acids Res, 2004. **32**(Database issue): p. D267-70.
19. Cimino, J.J., *Review paper: coding systems in health care.* Methods Inf Med, 1996. **35**(4-5): p. 273-84.
20. Langlotz, C.P., *RadLex: a new method for indexing online educational materials.* Radiographics, 2006. **26**(6): p. 1595-7.
21. Martone, M.E., A. Gupta, and M.H. Ellisman, *E-neuroscience: challenges and triumphs in integrating distributed data from molecules to brains.* Nat Neurosci, 2004. **7**(5): p. 467-72.

22. Swedlow, J.R., et al., *Informatics and quantitative analysis in biological imaging.* Science, 2003. **300**(5616): p. 100-2.

23. Rosse, C. and J.L. Mejino, Jr., *A reference ontology for biomedical informatics: the Foundational Model of Anatomy.* J Biomed Inform, 2003. **36**(6): p. 478-500.

24. Rubin, D.L., et al., *Using ontologies linked with geometric models to reason about penetrating injuries.* Artif Intell Med, 2006. **37**(3): p. 167-76.

25. Brinkley, J.F., *Structural informatics and its applications in medicine and biology.* Acad Med, 1991. **66**(10): p. 589-91.

26. Rosse, C., et al., *Motivation and organizational principles for anatomical knowledge representation: the digital anatomist symbolic knowledge base.* J Am Med Inform Assoc, 1998. **5**(1): p. 17-40.

27. Dameron, O., et al., *Towards a sharable numeric and symbolic knowledge base on cerebral cortex anatomy: lessons learned from a prototype.* Proc AMIA Symp, 2002: p. 185-9.

28. Kahn, C.E., Jr., D.S. Channin, and D.L. Rubin, *An ontology for PACS integration.* J Digit Imaging, 2006. **19**(4): p. 316-27.

29. Noy, N.F., et al., *Protege-2000: an open-source ontology-development and knowledge-acquisition environment.* AMIA Annu Symp Proc, 2003: p. 953.

30. Horrocks, I., *An ontology language for the semantic Web.* Ieee Intelligent Systems, 2002. **17**(2): p. 74-75.

31. Ball, C.A. and A. Brazma, *MGED standards: work in progress.* Omics, 2006. **10**(2): p. 138-44.

32. BioPax-Consortium. *BioPAX: Biological Pathways Exchange*. 2006 [cited 2006 Dec 2006]; Available from: http://www.biopax.org/.

33. Kanehisa, M., et al., *From genomics to chemical genomics: new developments in KEGG.* Nucleic Acids Res, 2006. **34**(Database issue): p. D354-7.

34. Karp, P.D., et al., *Expansion of the BioCyc collection of pathway/genome databases to 160 genomes.* Nucleic Acids Res, 2005. **33**(19): p. 6083-9.

35. Joshi-Tope, G., et al., *Reactome: a knowledgebase of biological pathways.* Nucleic Acids Res, 2005. **33 Database Issue**: p. D428-32.

36. Demir, E., et al., *PATIKA: an integrated visual environment for collaborative construction and analysis of cellular pathways.* Bioinformatics, 2002. **18**(7): p. 996-1003.

37. Shannon, P., et al., *Cytoscape: a software environment for integrated models of biomolecular interaction networks.* Genome Res, 2003. **13**(11): p. 2498-504.

38. Stevens, R., et al., *TAMBIS: transparent access to multiple bioinformatics information sources.* Bioinformatics, 2000. **16**(2): p. 184-5.

39. Gupta, A., et al., *Towards a formalization of disease-specific ontologies for neuroinformatics.* Neural Netw, 2003. **16**(9): p. 1277-92.

40. Li, C., et al., *OntoQuest: exploring ontological data made easy*, in *Proceedings of the 32nd international conference on Very large data bases - Volume 32*. 2006, VLDB Endowment: Seoul, Korea.

41. Kuchinsky, A., et al. *Biological Storytelling: a software tool for biological information organization based upon narrative structure.* in *Advanced Visual Interfaces*. 2002. Trento, Italy.

42. Karp, P.D., *Pathway databases: a case study in computational symbolic theories.* Science, 2001. **293**(5537): p. 2040-4.

43. Gifford, D.K., *Blazing pathways through genetic mountains.* Science, 2001. **293**(5537): p. 2049-51.

44. Racunas, S.A., et al., *HyBrow: a prototype system for computer-aided hypothesis evaluation.* Bioinformatics, 2004. **20**(suppl_1): p. i257-264.

45. Racunas, S.A., N. Shah, and N.V. Fedoroff. *A Contradiction-Based Framework for Testing Gene Regulation Hypotheses*. in *IEEE Bioinformatics*. 2003. Stanford University, Palo Alto, California: IEEE Computer Society.

46. Proteome. *Yeast Proteome Database*. 2001 [cited 2002 4/14/2002]; Available from: http://www.proteome.com/YPDhome.html.

47. Dameron, O., M.A. Musen, and B. Gibaud, *Using semantic dependencies for consistency management of an ontology of brain-cortex anatomy.* Artif Intell Med, 2007. **39**(3): p. 217-25.

48. Rubin, D.L., O. Dameron, and M.A. Musen, *Use of description logic classification to reason about consequences of penetrating injuries.* AMIA Annu Symp Proc, 2005: p. 649-53.

49. Boustil, A., et al. *Classification des compte-rendus mammographiques a partir d'une ontologie radiologique en OWL*. in *Extraction et gestion de Connaissances (EGC'2006)*. 2006.

50. Rzhetsky, A., et al., *A knowledge model for analysis and simulation of regulatory networks.* Bioinformatics, 2000. **16**(12): p. 1120-8.

51. Rubin, D.L., et al., *National Center for Biomedical Ontology: advancing biomedicine through structured organization of scientific knowledge.* Omics, 2006. **10**(2): p. 185-98.