Bottom-up ontology development reusing semi-structured life sciences diagrams

C. Maria Keet
School of Computer Science
University of KwaZulu-Natal
Durban, South Africa
Email: keet@ukzn.ac.za

***Abstract*—Bio-ontology development is a resource-consuming task despite the many open source ontologies available for reuse. Various strategies and tools for bottom-up ontology development have been proposed from a computing angle, yet the most obvious one from a domain expert perspective is unexplored: the abundant diagrams. To speed up and simplify bio-ontology development, we propose guidelines to formalise such diagrams in several logic languages availing also of a foundational ontology to achieve a more precise representation of the subject domain semantics and increase interoperability. The guidelines are demonstrated with a transformation of the Pathway Studio diagrams into an OWLized bio-ontology with BFO and RO.**

I. Introduction

The use of ontologies is a well-known approach to aid integration of data, databases, and conceptual data models, because they provide a logic-based representation of the domain of interest that is independent of specific applications. Integration can be done by means of annotation of instances across databases with a domain ontology like the widely used Gene Ontology [1] and similar OBO ontologies, linking ontologies to conceptual data models or directly to data in different sources through a mapping layer (e.g., [2]). Increasingly, an ontology is also seen as an end in itself, whereby it is deployed as a way to represent the knowledge of a particular subject domain [3], [4] and may be used for hypothesis elimination by reducing the theoretical options to those that are logically consistent with the theory before commencement of laboratory experiments [5], [6]. Inherent in such endeavours is the sharing of the knowledge, which, at least in the life sciences, proceeds in an environment committed to open source, such as the OBO Foundry [7] and the 255 ontologies available through the BioPortal at http://bioportal.bioontology.org. This is partly because life scientists have realised the need for ontologies to avoid duplication of costly research and to manage the exponentially growing amount of data to push science forward (e.g., [1], [8]). Development of ontologies is a resource-intensive task. Most bio-ontologies are developed by consortia across institutional boundaries, where such efforts avail of non-ontological resources—'legacy' representations of the scientific knowledge—that are, often manually, consulted to ensure adequate coverage and to ease the ontology development process. Many efforts have gone into automating this bottom-up development process, principally reverse engineering of databases [9] and natural language processing [10], which are relatively generic in their algorithms and heuristics and therefore noisy. Given that the life sciences are very diagram-oriented, it is surprising that such semi-structured diagrams, like depicted in Fig. 1, have not been used to find (candidate) classes and relationships. Also, those diagrams can provide a sought-after *intermediate representation* that domain experts are familiar with and computer scientists also can handle [11], [12] and, once formalised, consistency of the biological theories can be checked with automated reasoners.

The aim is to fill this hiatus in bottom-up ontology development. This requires two principal components in order to lay solid foundations: a formalisation step and adequate treatment of the subject domain semantics. In this paper, we propose guidelines to formalise such semi-structured biological diagrams. The formalisation aims at two common usage scenarios: (*i*) the so-called 'low-hanging fruit' with OBO or SKOS and its use for data annotation and computational linguistics, and (*ii*) to capture the necessary details for theory analysis by formalising it in a very expressive (Semantic Web) ontology language, with OWL 2 DL as a minimum. The formalization procedure will be expounded using the biochemical pathway modelling tool Pathway Studio (PS)[1] as example. Its vocabulary is analysed and then categorised with a foundational ontology so that the icons are given both a formal semantics and a precise subject domain semantics.

[1]http://www.ariadnegenomics.com/products/pathway-studio/

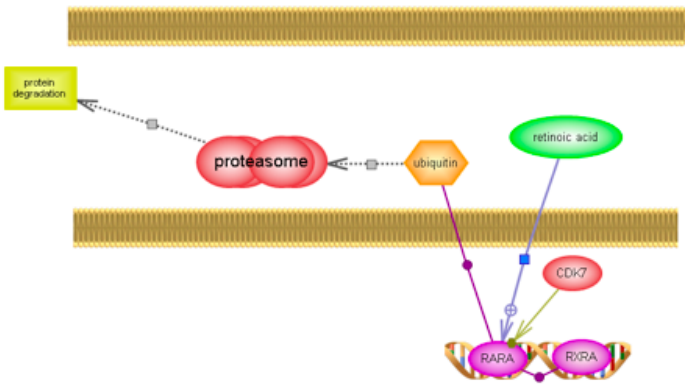

Fig. 1. A diagram representing the interactions between the various molecules in a pathway: the NHR pathway about acetaminophen activity and toxicity.

II. How to formalise it?

To the best of our knowledge, only one study is similar in spirit to the one proposed here [3], which is, however, with a much simpler graphical modelling tool and it does not provide a structured approach toward formalisation. At the other end of the spectrum, there are methodologies for ontology development, such as METHONTOLOGY [11], the NeON methodology [13], and OntoSpec [14], where the former two mention non-ontological resource reuse, but they do not elaborate on how exactly this is to be carried out. OntoSpec focuses on formalising subject domain knowledge in detail and is informed by DOLCE and OntoClean, but does not include non-ontological resources. The OBO Foundry [7] has a set of resources and principles, but no method other than the cumbersome manual examination of scientific literature [4]. The extensions required to cater for bottom-up development with bio-diagrams are the selection process of non-ontological resources, and, moreover, how to formalize it whilst being faithful to the subject domain semantics. The latter comprises choosing a language, which depends mainly on the purpose of the ontology and what reasoning services are required [15], and choosing a foundational ontology. We will assess the formalization step first, touch upon ontology reuse, and subsequently structure them into guidelines.

A. Formalization in different languages

The first aspect is to choose a suitable logic-based language to formalise the icons in the biological diagrams. Current usage of bio-ontologies fall broadly into two categories: annotation of resources, such as data in databases and text in scientific literature, and scientific ontologies representing the knowledge of a subject domain. The former requires support for navigation, queries to retrieve a simple class in the hierarchy, and scalability; hence, a language with low expressiveness suffices, such as the Open Biological and biomedical Ontologies' `obo`-format (a directed acyclic graph), the W3C standardised SKOS language [16] (essentially RDF), or perhaps the W3C standardised OWL 2 EL profile [17]. Scientific ontologies require very expressive languages to represent fine-grained distinctions between the entities and reasoning services such as satisfiability of the ontology, classification of classes in the hierarchy, and complex class queries. One can choose any language, be it full fist order predicate logic (FOL), an extension (e.g., temporal), or a standardised decidable fragment of FOL to guarantee termination of the reasoning services and foster interoperability and reuse with other ontologies. The second option indicates that the most expressive language OWL 2 DL [18] may be suitable. This information is summarised in Fig. 2.

For both scenarios, the first step of the formalization is to assess the "icon vocabulary" for unary 'object-like' entities and n-ary ($n \geq 2$) 'relationship-like' entities. Among the n-aries, one then distinguishes between generic relationships, such as parthood, and other recurring relationships (in the subject domain of the life sciences, they are relations such as development, regulation, and transformation). From here on,

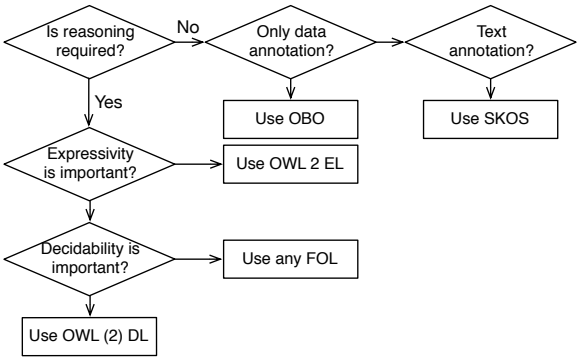

Fig. 2. Simple decision diagram to choose a suitable language.

the formalisation steps differ for the chosen languages. The straight-forward procedure for OBO and SKOS is included in the guidelines in Section II-C only, whereas the expressive languages require more explanation about the choices. For OWL 2 DL, one has to analyse the vocabulary and assess its use in sample diagrams to check for cardinality restrictions and to check for sequences of the same or different n-aries, which indicate possible transitivity or property chaining. n-aries where $n \geq 3$ can be formalised as reified relations, but this makes the overall ontology logically complicated and difficult to understand for the domain expert, and therefore should be used sparingly. For an arbitrary expressive logic language, there are more options to consider, such as spatiality and temporality, which both feature in many diagrams implicitly. Spatiality is often represented with sections of different background colour, lipid bi-layers, or the name of the (type of) cell, tissue, or organ, therefore requiring inclusion of both spatial relations as well as spatial entities at the appropriate level of granularity. Temporal aspects are normally represented as chains of unaries and n-aries with indicative labels like transports, transcribes, or flows.

An additional decision point in the formalization concerns foundational ontologies, which we address in the next section.

B. Foundational ontology commitments

Most bio-ontologies do not exist in isolation, but are linked to other ontologies, be they other domain ontologies or foundational ontologies, which affect the formalization. Using a foundational ontology with its generic categories of entity types and relationships across subject domains can facilitate ontology interoperation [7] and have been shown to improve quality and it speeds up ontology development [19]. Some of such ontologies are DOLCE, BFO(+RO), and SUMO.

The principal problem they introduce, is that it forces one to choose between n-aries-as-unaries (classes in OWL) or n-aries-as-n-aries (object properties in OWL). An intuitive formalization of the n-aries is to keep them as such, so that there is also a close correspondence with the original diagram; this easily can be done also in OWL and any arbitrary FOL language. Foundational ontologies, however, have a separate branch for 'processes' (Perdurant in DOLCE

[20] and Occurrent in BFO) and relate this with a *new* relation to 'objects' (Endurant in DOLCE, Continuant in BFO), such that an endurant is a *participant in* a perdurant; e.g., the person Mary is a participant in a running instance that, in turn, is part of a marathon, but not that there is a 1-to-1 formalisation of "Mary runs a marathon" where "runs" is the label for a binary relation. Thus, a biological diagram icon may be an arrow denoting regulation, which can be formalised as an OWL object property (binary relationship) `regulates` or `regulatedBy`, or as an OWL class (unary predicate) `Regulation` as subtype of DOLCE's Process or BFO's ProcessualEntity. The former results in a more compact representation closer to the domain expert's understanding, and therefore is likely to be more useful in praxis. The latter is more generic, and thereby likely to increase reusability of the ontology. At the time of writing, it has not been determined experimentally which option is better for domain ontologies.

C. Guidelines for formalizing a diagrammatic vocabulary

The considerations and decision points described in the previous two sections can be summarised as follows.

1) Basic assessment of the icons in the diagram's "legend":
   a) Divide between unaries and n-aries;
   b) Divide n-aries by parthood, participation, dependency, and other relationships;
2) Use OBO? If no: go to Item 3; If yes, do:
   a) Represent each unary as an OBO Class, and its usage in a diagram is a new child of the respective main class;
   b) Parthood as part-of that is transitive, remainder of the binaries as user-defined OBO Relation;
3) SKOS? If no: go to Item 4; If yes, do:
   a) Declare unaries to be of `rdf:type skos:Concept`, and their usage in the diagrams as `skos:broader` their respective core entities;
   b) Binaries as `skos:related` and/or extend SKOS RDF Schema accordingly;
4) Choose a foundational ontology.
5) N-aries as classes? If no: go to Item 6; If yes, do:
   a) Declare *part-of* and *participates* as binary relationships;
   b) Declare `participates`' domain as `Endurant` (continuant) and range as `Perdurant` (occurrent);
   c) Declare all other n-aries as classes, suitably positioned under `Perdurant`;
6) N-aries as relationships. Do:
   a) Consider also sample diagrams;
   b) An n-ary has relations to > 1 unary? If yes: note cardinality;
   c) Chaining of n-aries? If yes: note concatenation;
   d) Use OWL 2 DL? If no: go to Item 6(e); If yes, do:
      i) Declare unaries as subclasses of `Endurant`;
      ii) Declare binaries as object properties;
      iii) Add *part-of* and its relational properties;
      iv) Add *participates* and declare its domain as `Endurant` and range as `Perdurant`;
      v) Sequences of the same binary? If yes: declare transitivity for that binary;
      vi) n-aries with $n > 2$? If yes: if used often, drop it, if used sparingly, reify it;
   e) FOL or more. Do:
      i) Examine at least the spatial and temporal dimension;
      ii) Are there any "system" icons? If yes: consider granularity;
      iii) Declare unaries as unary predicates subsumed by `Endurant`;
      iv) Declare n-aries;
      v) Declare same as in items 6(d)iii-6(d)v.
7) Ontology population by transforming the diagrams.

The guidelines can be extended and enforced by a formalisation workflow (not pursued here due to space limitations).

III. Transformation of the Pathway Studio Graphical Vocabulary

In order to demonstrate the summarised guidelines, we take Pathway Studio (PS) as an example. PS lets one "Build, expand and analyze pathways" and "Find relationships among genes, proteins, cell processes and diseases", among other things. The source data for the diagrams originate from NLP of scientific literature and manual curation. Some specific user requirements can be abstracted from the biological examples as follows: Compound X (e.g., a potential drug) binds and activates Y, which is a main switch in pathway Z that should be interrupted to cure the disease. Sample queries are:

Q1: Is Y involved in some other pathway?

Q2: What are the characteristics of the other pathways that Y is involved in? E.g., are they spatially separated (e.g., in different tissues)?

Q3: Is there an activation of some Y' by X that is also a signaling molecule in pathway Z'?

Q1 can be a simple class-query, but the others show that neither OBO nor SKOS is sufficient to meet all the desired inferences, hence OWL or arbitrary FOL should be chosen to formalise the Pathway Studio Vocabulary (PSV). To foster interoperability with other ontologies, OWL 2 DL is chosen. Note that this still permits simplification to an ontology for NLP that can aid text mining to find data for the diagrams.

Assuming foundational ontology use, then BFO is a strategic option, because many bio-ontologies align themselves with it. The main question is to choose to formalise PS's binaries as classes or as object properties. The limited set of arrows (Fig. 3, bottom) suggests formalising them as object properties, which also concurs with the relations of the Relation Ontology (RO) [21] integrated with the Basic Formal Ontology (BFO, http://www.ifomis.org/bfo/) and its extensions under consideration. Domain ontologies may be useful for the individual diagrams. Given the named categories (Fig. 3, top section), BioPax will be useful to consult, which covers metabolic pathways and molecular interactions. The OWLized Gene Ontology, Cell Cycle Ontology, and Protein Ontology may also be useful.

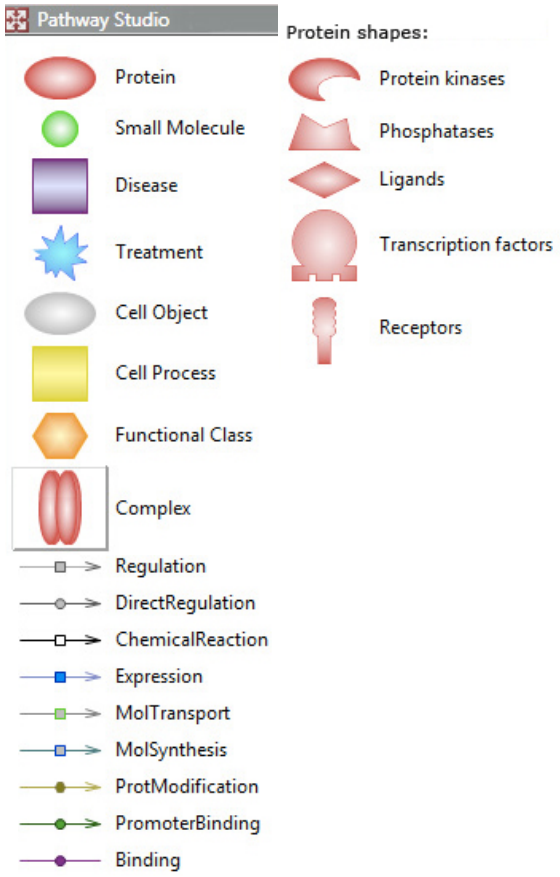

Fig. 3. Pathway Studio's icons.

A. Pathway Studio Vocabulary

The following high-level informal descriptions are intended to give a non-biochemist an indication of the kind of things in the PSV. Subject domain semantics especially useful in the formalization is italicized when they first appear.

- Protein: biopolymer consisting of many linked α-amino acids. Shapes denote subclasses:
  - Kinases are enzymes, i.e., proteins with a *function/role*;
  - Phosphatases are also enzymes;
  - Ligand is a molecule that *binds* to a receptor;
  - Transcription factor is a molecule that binds to a binding site and thereby *regulates* expression of a *nearby* gene;
  - Receptor: molecule with a role in a receptor-ligand binding; it can be a protein when [*in/on*] a membrane or a Nuclear receptor (purple oval) bound to DNA.
- Small Molecule: refers to, e.g., glucose, nitric oxide, (not 'macromolecule', such as protein, starch);
- Treatment represent a '*system*', here: a *cascade* of processes in which molecules *participate*;
- Cell Object represents a combination of a structural entity at a certain *location* in the cell, includes organelles;
- Cell Process represents a (combination of) process(es) located in the cell;
- Functional Class: molecule with a particular function;
- Complex *consists of* at least one protein and at least one other molecule that may be bound to it.
- The arrows are binary *relations*, with Mol an abbreviation of molecule, and Prot of protein modification. A "$\oplus$" in the arrow means *positive* effect, a line negative effect of the type of *interaction* indicated by the arrow's colour.

B. Formalisation

Having committed to OWL 2 DL, BFO+RO, and n-aries as relationships, we need to assess sample diagrams; a small one is depicted in Fig. 1 and many larger ones can be consulted online (URL: see footnote 1). This revealed that regulation is transitive, each pathway has at least three molecules, and nuclear receptors are bound to exactly one DNA molecule.

1) The PS vocabulary: Due to space limitations as well as the principle of minimum necessary commitment, only conservative axioms are described here. Given that OWL 2 DL is based on the Description Logics (DL) language $\mathcal{SROIQ}$ [22], we use the more concise DL notation. OWL 2 DL's model-theoretic semantics can be consulted online [18] and the essentials of DLs are described in [23]; e.g., `Protein` $\sqsubseteq$ `Molecule` can be represented equivalently in FOL as $\forall x(Protein(x) \rightarrow Molecule(x))$ and in OWL 2 DL functional syntax as `SubClassOf(Protein Molecule)`.

The following classes, object properties, and axioms are declared for the formalization of the PSV, where entities in *italic courier* are BFO or RO entities:

`Protein` $\sqsubseteq$ `Molecule`, `SmallMolecule` $\sqsubseteq$ `Molecule`,
`ProteinComplex` $\sqsubseteq$ `Molecule`, `Molecule` $\sqsubseteq$ *`Object`*,
`TranscriptionFactor` $\sqsubseteq$ $\exists$`inheres_in.Protein`,
`Ligand` $\sqsubseteq$ $\exists$`inheres_in.Protein`,
`Receptor` $\sqsubseteq$ $\exists$`inheres_in.Protein`,
`Enzyme` $\sqsubseteq$ $\exists$`inheres_in.Protein`,
`Kinase` $\sqsubseteq$ `Enzyme`, `Phosphatase` $\sqsubseteq$ `Enzyme`,
`Kinase` $\sqsubseteq \neg$`Phosphatase`,
`CellProcess` $\doteq$ *`Process`* $\sqcap$ $\exists$*`located_in`*`.Cell`,
`CellObject` $\doteq$ *`Object`* $\sqcap$ $\exists$*`contained_in`*`.Cell`,
`ExtracellularProtein` $\doteq$ `Protein` $\sqcap$
    $\exists$*`located_in`*`.`$\neg$`Cell`,
`NuclearReceptor` $\doteq$ `Receptor` $\sqcap$ `=1 binds`$^-$`.DNA`,
`ProteinComplex` $\doteq$ `Complex` $\sqcap$
    $\exists$*`has_part`*`.Protein` $\sqcap$ $\exists$*`has_part`*`.Molecule` $\sqcap$
    $\forall$*`has_part`*`.(Protein` $\sqcup$ `Molecule)`,
$\exists$`binds.TranscriptionFactorBindingSite` $\sqsubseteq$
    `TranscriptionFactor`,
`Pathway` $\sqsubseteq$ `System` $\sqcap \geq$`3` *`has_part`*`.Molecule` $\sqcap$
    $\forall$*`has_part`*`.Molecule`,
`Treatment` $\sqsubseteq$ $\exists$*`has_participant`*`.Molecule`,
`System` $\sqsubseteq$ *`GenericallyDependentContinuant`*,
`binds_promoter` $\sqsubseteq$ `binds`,
`binds` $\sqsubseteq$ `reacts_chemically`,
`up_regulates` $\sqsubseteq$ `regulates`,
`regulates_directly` $\sqsubseteq$ `regulates`,
`modifies` $\sqsubseteq$ `reacts_chemically`,
`modifies_protein` $\sqsubseteq$ `modifies`,
$\exists$`modifies_protein`$^-$ $\sqsubseteq$ `Protein`,

synthesis ⊑ reacts_chemically,
molecular_synthesis ⊑ synthesis,
molecular_transport ⊑ transports,
∃molecular_transport⁻ ⊑ Molecule.
The declared subclasses of Molecule are disjoint.
FunctionalClass, Receptor, TranscriptionFactor, Enzyme, and Ligand are disjoint and subclasses of *SpecificallyDependentContinuant*.
regulates, expresses, transports, and reacts_chemically are sub-properties of topObjectProperty.

Practically, BFO and RO were imported into a new ontology with owl:import and the above-listed statements were added to create the combined ontology OWLPathS.owl, which is available online at http://www.meteck.org/files/ontologies/OWLPathS.owl. Although BFO needs only $\mathcal{ALC}$ features, the DL language used for OWLPathS is $\mathcal{SHIQ}$, i.e., indeed requiring OWL 2 DL expressivity.

2) The diagrams: Given this core formalization of the PSV, let us consider Fig. 1 again. The label proteasome is associated with a Complex-icon, hence an assertion Proteasome ⊑ Complex is added to the ontology, the label RARA has a purple protein-shape, hence a nuclear receptor, so that the assertion RARA ⊑ NuclearReceptor can be added, and so forth for the other elements. The procedure is similar for all polygons, such that the class is added under one of the main categories described earlier. Thus, a set of rules can be devised to automate the ontology populations procedure for the classes. The procedure for the relationships is longer because (*i*) it depends on the colour, adornments, and direction of the arrow, and (*ii*) the assertion for the direction has to be added in the inverse. The reason for the latter is that while retinoic acid *expresses* RARA (see Fig. 1), this is not the case for *all* retinoic acid molecules; it holds that RARA is *expressed by some* retinoic acid, hence RARA ⊑ ∃expressed_by.RetinoicAcid is the appropriate axiom to add to the ontology. A (partial) design-level procedure for the arrows can be as follows:

$x \leftarrow$ getArrow() // select an arrow and obtain information
$colour \leftarrow$ getArrowColour(x)
$y \leftarrow$ getArrowBase(x)
$z \leftarrow$ getArrowHead(x)
select case
  $colour$ = blue:
    add z ⊑ ∃expressed_by.y to the ontology
  $colour$ = grey:
    $shape \leftarrow$ getArrowShapeMiddle(x)
    if $shape$ = square **then**
      add z ⊑ ∃regulated_by.y to the ontology
    else // i.e., it is a cricle
      add z ⊑ ∃regulated_directly_by.y to the ontology
    end if
  $colour$ = purple: // and so forth for the other arrow colours
end select case

Optionally, the procedures can be extended by incorporating the BFO+RO-based BioTop [24], BioPax etc. and to include a search step "check if y (resp. z) is already in the ontology".

IV. Discussion

As demonstrated with the PS transformation, one clearly can obtain a lot of information from the diagrams once the icon vocabulary is formalized and related to categories of a foundational ontology. One may question nevertheless why specifically the guidelines proposed in Section II-C are any good. First, it has to be noted that there are no extant AI techniques that can handle the bio-diagrams in a structured fashion. Second, it provides a methodological approach toward formalising informal 'legacy' resources to standardised knowledge representation languages that is sufficiently generic to work with any graphical language of bio-diagrams, yet not too generic to render it of little use for biological resources (as is the case of database reverse engineering and mining conceptual data models). Moreover, it incorporates foundational ontology use to also handle subject domain semantics as opposed to a mere formalization into an arbitrary logical theory. Consequences of this aspect are the inclusion of the hitherto neglected representation decision to represent n-aries as classes vs n-aries as relationships and the reuse of classes and relationships that are also used in other bio-ontologies so as to foster interoperability upfront. Last, it acknowledges that different migration paths may be viable. The guidelines do not help with the transformation of implicit information, which requires subject domain knowledge, such as knowing that a kinase is an enzyme, which is a hurdle that other bottom-up approaches face as well (NLP for science uses bio-adjusted heuristics [10], e.g., the suffix "-ase" denotes the name of an enzyme). Hence, knowledge in both computation and the subject domain is necessary for devising effective automated bottom-up ontology development procedures leveraging legacy resources (which might be the prohibitive step why there are few tools to this end in the life sciences [25]).

Considering possible extensions to the guidelines, then being able to handle time and location in the formalization will be useful, which is challenging because they are deeply embedded in the diagrams. Time is implicit with the very notion of pathway—i.e., some specific sequence of interactions—that is approximated with the arrows. Efforts to try to capture this with "precedes" and "immediately precedes" relations, as proposed informally by BFO+RO (http://www.obofoundry.org/ro/, announcement still active on 22-3-'11), bears no formal semantics, hence cannot be used in automated reasoning: neither OBO nor OWL is expressive enough to assert that "a immediately precedes b" means that we have not only $(\mathtt{a}, t)$ and $(\mathtt{b}, t')$ but also $\neg\exists t''.t < t'' < t'$. Moreover, all Allen temporal relations can be useful. Unlike BFO, the Time Ontology [26] provides those basics, which suffices if one only wants to annotate resources with a time component. Few temporal DLs exist and no usable technological solution for temporal ontologies exist yet that lets one use it with automated reasoning. Regarding location, compartmentalization is represented generally with lines, different shaded areas, or both; e.g., Fig. 1's two thick lines that represent membranes to

delineate the cell's nucleus from cytoplasma from the exterior. Inferring the implicit location is not easy due to both how it is represented in the diagrams and (mereo)topological representation and reasoning is not solved for decidable languages and scalability of reasoning [27], [28]. In addition, recollect sample question Q2 in Section III, which moves easily between the molecule-level in a cell to its location in some tissue, thus indicating the need to take into account granularity. What may be feasible to handle are the Cell Process icons, such as Protein Degradation (Fig. 1), that involves several reactions, hence is a common 'folding' operation [29]. One also may want to modularize the knowledge along those lines, using an arbitrary-logic or OWL-based technique [30], [31].

Nevertheless, the rich formalization of the icon vocabulary already provides a solid basis to simplify and speed up ontology development compared to manual efforts or NLP. In addition, using a more expressive language invites the domain expert to be more precise so as to resolve ambiguities, a benefit which was already observed in [3] for eco-ontologies. Only then can it be checked computationally if the many diagrammatic pathways are consistent together and gaps can be found easily [6], which motivates further modelling or can serve as impetus for laboratory experimentation.

V. Conclusion

To speed up and simplify bio-ontology development, we proposed a method to formalise semi-structured life science diagrams, which is aimed at extracting explicit and implicit knowledge from such 'legacy' resources. Four trajectories for formalization were identified—OBO, SKOS, OWL 2 DL, and arbitrary FOL—with the option to integrate it with a foundational ontology so that both a formal and precise subject domain semantics is generated. The approach was demonstrated with OWL 2 DL, BFO+RO, and n-aries as relationships applied to the extensive icon vocabulary of Pathway Studio. Added benefits of the approach are that such diagrams also can be deployed as intermediate representation of the knowledge so as to facilitate understanding and communication between logicians and the content providers. Also, it can bring the information modelled in such diagrams—often hidden or locked in, e.g., expensive hardcopy textbooks—into the open access domain for free use and reuse. We are looking into implementing it and validating the effectiveness experimentally with domain experts.

Acknowledgment

The author gratefully acknowledges Kristina Hettne for providing several models of NHR pathways and sample queries.

References

[1] Gene Ontology Consortium, "Gene Ontology: tool for the unification of biology," *Nature Genetics*, vol. 25, pp. 25–29, 2000.

[2] A. Poggi, D. Lembo, D. Calvanese *et al.*, "Linking data to ontologies," *J. on Data Semantics*, vol. X, pp. 133–173, 2008.

[3] C. M. Keet, "Factors affecting ontology development in ecology," in *Proc. of DILS2005*, ser. LNBI, Springer Verlag, 2005, pp. 46–62.

[4] B. Smith and W. Ceusters, "Ontological realism: A methodology for coordinated evolution of scientific ontologies," *Applied Ontology*, vol. 5, pp. 79–108, 2010.

[5] S. Bandini and A. Mosca, "Mereological knowledge representation for the chemical formulation," in *2nd Workshop on Formal Ontologies Meets Industry 2006 (FOMI2006)*, Trento, Italy, December 2006, pp. 55–69.

[6] C. M. Keet, M. Roos, and M. S. Marshall, "A survey of requirements for automated reasoning services for bio-ontologies in OWL," in *Proc. of OWLED 2007*, ser. CEUR-WS, vol. 258, 2007. Austria.

[7] B. Smith, *et al.*, "The OBO Foundry: Coordinated evolution of ontologies to support biomedical data integration," *Nature Biotechnology*, vol. 25, no. 11, pp. 1251–1255, 2007.

[8] J. S. Madin, S. Bowers, M. P. Schildhauer, and M. B. Jones, "Advancing ecological research with ontologies," *Trends in Ecology & Evolution*, vol. 23, no. 3, pp. 159–168, 2008.

[9] L. Lubyte and S. Tessaris, "Automated extraction of ontologies wrapping relational data sources," in *Proc. of DEXA'09*, 2009.

[10] D. Alexopoulou, T. Wächter, L. Pickersgill, C. Eyre, and M. Schroeder, "Terminologies for text-mining; an experiment in the lipoprotein metabolism domain," *BMC Bioinform.*, vol. 9, no. Suppl 4, p. S2, 2008.

[11] M. Fernandez, A. Gomez-Perez, A. Pazos, and J. Pazos, "Building a chemical ontology using METHONTOLOGY and the ontology design environment," *IEEE Expert*, vol. January/February, pp. 37–46, 1999.

[12] C. Ghidini, B. Kump, S. Lindstaedt, N. Mabhub, V. Pammer, M. Rospocher, and L. Serafini, "Moki: The enterprise modelling wiki," in *Proc. of ESWC2009*, 2009, Heraklion, Greece, 2009 (demo).

[13] M. C. Suarez-Figueroa *et al.*, "NeOn methodology for building contextualized ontology networks," NeOn Project, Deliverable D5.4.1, 2008.

[14] G. Kassel, "Integration of the DOLCE top-level ontology into the OntoSpec methodology," Laboratoire de Recherche en Informatique d'Amiens (LaRIA), Technical Report HAL : hal-00012203, October 2005, http://hal.archives-ouvertes.fr/ccsd-00012203.

[15] C. M. Keet, "Dependencies between ontology design parameters," *Int. J. of Metadata, Semantics and Ontologies*, vol. 5, no. 4, pp. 265–284, 2010.

[16] *SKOS Core Vocabulary Specification*, W3C Std. [Online]. Available: http://www.w3.org/TR/swbp-skos-core-spec

[17] B. Motik, B. C. Grau, I. Horrocks, Z. Wu, A. Fokoue, and C. Lutz, "OWL 2 Web Ontology Language Profiles,", W3C Recommendation, 27 Oct. 2009, http://www.w3.org/TR/owl2-profiles/.

[18] B. Motik, P. F. Patel-Schneider, and B. C. Grau, "OWL 2 web ontology language: Direct semantics,", W3C Recommendation, 27 Oct. 2009. [Online]. Available: http://www.w3.org/TR/owl2-direct-semantics/

[19] C. M. Keet, "The use of foundational ontologies in ontology development: an empirical assessment," in *Proc. of ESWC'11*, ser. LNCS. Springer, 2011, Heraklion, Crete, Greece, 29 May-2 June, 2011.

[20] C. Masolo, S. Borgo, A. Gangemi, N. Guarino, and A. Oltramari, "Ontology library," WonderWeb Deliverable D18 (ver. 1.0, 31-12-2003)., 2003, http://wonderweb.semanticweb.org.

[21] B. Smith, *et al.*, "Relations in biomedical ontologies," *Genome Biology*, vol. 6, p. R46, 2005.

[22] I. Horrocks, O. Kutz, and U. Sattler, "The even more irresistible $\mathcal{SROIQ}$," *Proc. of KR'06*, pp. 452–457, 2006.

[23] F. Baader, D. Calvanese, D. L. McGuinness, D. Nardi, and P. F. Patel-Schneider, Eds., *The Description Logics Handbook – Theory and Applications*, 2nd ed. Cambridge University Press, 2008.

[24] E. Beisswanger, S. Schulz, H. Stenzhorn, and U. Hahn, "BioTop: An upper domain ontology for the life sciences," *Applied Ontology*, vol. 3, no. 4, pp. 205–212, 2008.

[25] A. Ruttenberg, *et al.*, "Advancing translational research with the semantic web," *BMC Bioinformatics*, vol. 8, no. Suppl 3, p. S2, 2007.

[26] J. R. Hobbs and F. Pan, "An ontology of time for the semantic web," *ACM Trans. on Asian Language Processing (TALIP)*, vol. 3, no. 1, pp. 66–85, 2004.

[27] M. Wessel, "Obstacles on the way to qualitative spatial reasoning with description logics: some undecidability results," in *Proc. of DL'01*, ser. CEUR WS, vol. 49, 2001, stanford, CA, USA, August 1-3, 2001.

[28] M. Aiello, I. Pratt-Hartmann, and J. van Benthem, Eds., *Handbook of Spatial Logics*. Springer, 2007.

[29] C. M. Keet, "A formal theory of granularity," PhD Thesis, KRDB Research Centre, Faculty of Computer Science, Free University of Bozen-Bolzano, Italy, April 2008.

[30] O. Kutz, C. Lutz, F. Wolter, and M. Zakharyaschev, "E-connections of abstract description systems," *Artificial Intelligence*, vol. 156, no. 1, pp. 1–73, 2004.

[31] B. Cuenca Grau, B. Parsia, and E. Sirin, "Combining OWL ontologies using ε-connections," *J. Web Semantics*, vol. 4, no. 1, pp. 40–59, 2006.