What BioPAX communicates and how to extend OWL to help it

Alan Ruttenberg[1], Jonathan Rees[2] and Jeremy Zucker[3]

[1] Millennium Pharmaceuticals, Cambridge, MA, USA
[2] Science Commons, Cambridge, MA, USA
[3] Dana Farber Cancer Institute and Harvard Medical School, Boston, MA, USA

Abstract. BioPAX[1] is a collaborative effort to develop a data exchange format to facilitate integration of biological pathway knowledge. The BioPAX exchange format is currently specified in a combination of English and OWL. In this paper we explore a tension, present in the current BioPAX specification, between representation of records in biological databases and realist representations of biological entities and processes. We propose to reorganize the BioPAX ontology to represent both of these points of view and in the process correct shortcomings in its use of OWL. In doing so we note some issues in using OWL for data exchange. We propose extensions which would both serve accurately model such exchange and help users avoid some common errors when using OWL.

1 Introduction

Most current sources of biological information aren't simply collections of biological truths. Rather their form and content is influenced by a number of factors, among them how well understood the domain is, curation policy, information technology standards and the initial purpose for which the information was collected.

While there is general agreement within the BioPAX community about the subject matter of interest—information about biomolecular interactions and pathways—and the need to share it, it has recently become clear that there are different points of view about what is being shared, and consequently how to best accomplish that goal.

Difficulties in articulating these perspectives have impeded progress of the BioPAX specification, because different approaches are combined in ways that aren't always coherent. We suggest that these differences in epistemological attitude represent essential tensions in the application of semantic web technologies to the life sciences. By representing each attitude explicitly in the specification, much of the confusion can be effectively resolved.

2 Three attitides toward the contents of biological databases

The first attitude is that existing relational databases contain data for which the specification of the semantics are only loosely specified in english. There is value in the contents of these databases regardless of whether the semantics are precisely specified. For example, collecting, within a set of records describing metabolic reactions, those that don't have a value for the field recording the enzyme is a useful first step in helping decide where curation or scientific effort needs to be allocated to filling in these blanks [2].

From this attitude, the primary concern is to identify similarities in the fields in these databases, design a schema for those common elements, and design a file format for packaging up these records so that they can be incorporated into other databases after being checked for well-formedness. Importantly, this point of view does not consider the denotation of much of the contents of these databases, instead assuming that each receiver of information in this form has its own way of interpreting the contents of the database in some manner useful for its audience. We will refer to this attitude as the *record level*.

The second attitude is that what biological databases do, or ought to do, is traffic in representations of *statements* that scientists make. That is, rather than directly asserting statements about biology, qualify each all such statements as being proffered by the researcher who makes it.

It would then be the responsibility of the consumer of this information to decide what to believe about the world. That one represents statements rather than facts is motivated by the recognition that scientists often make contradictory statements, such as reporting differing outcomes of what should be equivalent experiments. Representing the outcomes of such experiments as fact would lead to troublesome inconsistencies in the database. Rather, one might like to accept such situations and recognize them as opportunities to do further experiments to clarify the situation. We will refer to this attitude as the *statement level*.

The third attitide is a desire to make and exchange statements about biological reality. From this point of view, reality is the ultimate subject matter of both the database and statement representations, and in fact grounds these so that, if that aspect of reality is understood, statements and database records might be understood to be correct or incorrect. We will refer to this attitude as the *domain level*.

Our thesis is that each of these attitudes represents valid use cases and that each has existing constituencies. Database providers are understandably interested in the database layer, since it provides a short path toward augmenting their databases and therefore enhancing the services that they supply their users. For example, one subgroup of the BioPAX working group explicitly makes their goal "exchange between database providers".

The producers of information mined from text, such as the Neurocommons project of Science Commons,[3] might choose the statement level as primary as it is a conservative (read epistemologically justified) way of representing the results of their work.

The producers of the OBO Foundry[4] ontologies are primarily aiming at the domain level. They aspire to represent reality, an effort guided by the principles that Barry Smith et al. set forth in the BFO and relational ontology developed in [5], [6], and elsewhere. We would also argue that many biologists, as consumers, are ultimately interested in this level—no matter what the attitude of the information they receive.

3 Exchange of information at each level operates with distinct semantics

Exchange at each level operates by a its own semantics, by which we mean that each has different constraints on well formedness, and each can and should generate different sorts of inferences and inconsistencies.

The underlying guiding priciple for the record level is that there should be a transparent way for the exchanged contents to be added to a relational database, datatypes should be compatible with relational fields, and objects should be consistent with representation in a relational table, with ObjectProperty values represented as foreign keys. From an exchange point of view the database level is thus primarily concerned with ensuring that records are complete, and that field cardinalities are known (since they impact the choice of whether to use mapping tables, or checking whether foreign keys refer to the correct tables).

The statement level differs in several respects. Although its requirements for well-formedness are similar to those of the database level, the objects related at the statement level can be elements of reality, and so there is a need to be able to refer to those statements without the commitment to consistency that a statement at the domain level would entail. Examples of statements at this level are "Dr. X concludes that protein P is phosphorylated at position 110 in its peptide sequence" and "Dr. Y concludes that protein P is phosphorylated only at positions 220 and 387 in its peptide sequence." Even though these statements conflict[1] , there should be no inconsistency at the statement level, as both are true renditions of what scientists say.

The domain level, or what might be called the consensus level, represents information that should be, at a minimum, consistent, and is meant to comprise true statements about the world. The truth of these statements can in principle be checked by doing experiments. Inconsistencies should indicate that consensus has not been reached or that there are errors in encoding. It is at this level that meaningful automated inference can be the most valuable. Checks of knowledge consistency avoid having scientists waste time looking for them, or, worse, acting on incorrect information. So advances in this area can lead to more effective scientific discovery. An example of a statement at this level is "In *E. coli* K-12, the protein encoded by the gene ECK0647 when in inner membrane, facilitates the transport of glutamate from the periplasm to the cytoplasm." A

[1] $\mathsf{p} \sqsubseteq \mathsf{protein}, \mathsf{p} \sqsubseteq \exists \mathsf{pSite}.\{"110"\}, \mathsf{p} \sqsubseteq \exists \mathsf{pSite}.\{"220"\} \sqcap \exists \mathsf{pSite}.\{"387"\} \sqcap = 2\ \mathsf{pSite}.\top$

useful consistency check might be to detect when elsewhere it is stated that protein encoded by ECK0647 localizes exclusively in the cytoplasm, e.g. by checking for the presence of unsatisfiable classes in the ontology[2].

Observe that there is no committment here to specificity. For example it would be fine to say that glutamate is transported from periplasm to cytoplasm without reference to the enzyme, if it wasn't known whether the transport was active or passive. In fact, it is characteristic of the current state of biology that much information is only known to a certain degree of specificity—and it is a challenge to organize these facts with different degrees of specificity and at different granularities in a coherent way.

Note also that there are legitimate issues from the philosophy of science that might argue for one or another form of the facts encoded at this level. For example, one might choose to report either that a certain protein causes a certain effect, or take the point of view that all knowledge is the result of experiments so that we should say that if one sets up two experiments in which the only difference is that protein is present or absent, then one will find that the measure a certain quantity is different in these cases. The point here is that at the domain level inconsistency matters and the results of inference are interesting.

4 BioPAX doesn't choose a single level, and definitions mix levels

In this section we review some definitions from the BioPAX level 2 specification to illustrate our claim that it does not consistently choose a single point of view, and comment on where and how this causes problems for different providers and consumers. We show that the BioPAX specification is closest to the record level.

We start with the definition of an entity in BioPAX taken from the vBioPAX level 2 ontology [7, 8].

> Entity: A discrete biological unit used when describing pathways. Comment: This is the root class for all biological concepts in the ontology, which include pathways, interactions and physical entities. As the most abstract class in the ontology, instances of the entity class should never be created. Instead, more specific classes should be used. Synonyms: thing, object, bioentity

The class has two restrictions. There must be at most one NAME and SHORT-NAME. In addition, entity is given as the domain of properties such as DATA-SOURCE, SYNONYMS, and the range of others, such as PARTICIPANTS.

By defining entity as something used to describe pathways, this definition starts off by suggesting that it is a computational thing. However, the comment suggests that it represents biological entities (root class of both processes/occurants

[2] $\mathsf{inMembrane} \sqcap \mathsf{inCytoplasm} \sqsubseteq \bot, \mathsf{ECK0647\_Protein} \sqsubseteq \mathsf{inCytoplasm}$
$\mathsf{glutamateTransport} \sqsubseteq \exists \mathsf{participant}.\mathsf{ECK0647\_Protein} \sqcap \mathsf{inMembrane}$

and objects/continuants). How are we to interpret the comment that "instances of the entity class should never be created. Instead, more specific classes should be used"? First, "instance" should read "direct instance". This constraint makes sense in a situation where no database has a table for entities at this level of generalization. On the other hand, at the domain level it is perfectly reasonable to posit that there is some biological entity that is the infectious agent responsible for a communicable disease, without knowing exactly what that entity is.

The restriction NAME maxCardinality(1) is a further indication that this class is meant to represent records. Proteins, of course, typically have many names and the domain level would not define the biological entities by properties of their names. But the choice of a single name as primary does differ from database to database, and most databases choose only one primary name for user interface purposes. Complicating matters, the restriction doesn't express what is desired in that case. It says that all entities have at most one NAME. Thus asserting two values for NAME would lead the reasoner to conclude that the object is not an entity, perhaps it is some other kind of thing, instead of detecting an inconsistency [9]. As we discuss below, such expressivity is not provided by OWL, but probably should be.

As another example, consider the disjoint subclasses of physicalEntity. Let's focus on smallMolecule and protein.

> Protein: A physical entity consisting of a sequence of amino acids; a protein monomer; a single polypeptide chain. Examples: The epidermal growth factor receptor (EGFR) protein.

A protein has two restrictions: That it have exactly at most one SEQUENCE, and that it have at most one ORGANISM. Here is the definition of smallMolecule

> Definition: Any bioactive molecule that is not a peptide, DNA, or RNA. Generally these are non-polymeric, but complex carbohydrates are not explicitly modeled as classes in this version of the ontology, thus are forced into this class. Comment: Recently, a number of small molecule databases have become available to cross-reference from this class. Examples: glucose, penicillin, phosphatidylinositol

Among other axioms related to small molecules, we find that the property CHEMICAL-FORMULA has smallMolecule as its domain. According to the OWL semantics, this means that anything that has a value for CHEMICAL-FORMULA must be a small molecule.

Once again there is the mixture of features that suggest database records and those that suggest reality. The definitions suggest things in the world: "A physical entity consisting of a sequence of amino acids", "A bioactive molecule". However, if we consider the rest of the ontology to be about the world, we can certainly give more explicit, computational definitions. For instance we could define the class of amino acids, and represent the constraint that a protein at least has parts which are amino acids. Or we could define bioactive as being

a participant in some reaction or part of some complex that participates in a reaction and add that constraint to smallMolecule.

However, the actual restrictions imply a representation of records. *Every* protein can be represented as a molecule that has a sequence, an abbreviation for the sequence of amino acids that form the primary structure. That the restriction is maxCardinality instead of Cardinality is an attempt to represent that this field is optional—not all databases are expected to provide it.

That only small molecules have chemical formulae doesn't make sense if we are talking about things in the world. A chemical formula is one level of description of any molecule, including proteins. On the other hand, most extant pathway databases do choose to represent proteins in sequence-centric manner. So once again we come to the conclusion that the proteins and small molecules in BioPAX are better understood as representations of protein database entries and chemical database entries, than things in the world.

5 A way forward

Rather than debating which single attitude is the best one, we propose that the three different approaches be recognized as valid, and that the ontology be reorganized and definitions amended to make clear which attitude/approach is being used to represent different kinds of content. There should be a clear and effective way to verify well-formedness and exchange database records, to model attributed statements, and to make statements about the domain that can be checked for consistency and from which inferences can be computationally drawn.

In this framework, a primary concern, whether to use instances or classes as the basis of representation - repeatedly an matter of consternation when using OWL - becomes much easier to deal with. At the record level, it is clear that instances are appropriate—each record is a single individual. At the domain level, it is equally clear that many facts about biology are constructed from classes—the set of reactions that convert a to a', the class of proteins that have a certain primary structure and some known (and some unknown) post translational modifications.

It is clear that there are relationships between the different levels. For example, consider, as we pointed out earlier, that most biologists will make some judgement about the real-world meaning of the information that they find in pathway databases. We suggest that we can model some aspects of these as a set of transformations or *lifting productions* that take representations in one level and generate representations in a different level. The primary motivation for such transformations would be to either take advantage of the inferences that are possible in the domain level or to translate a representation of a database that has a clear attitude to the appropriate level.

As an example of the latter, a database of what are clearly statements could be both represented as database records (initially) and later translated into the statement level.

There are many cases where there is utility in lifting information from the database level to the domain level. First of all, because of the specific lack of checkable semantics at the record level, any such checks need to happen at a different level. Second, the increased expressiveness at the domain level can aid scientific inquiry.

An example is a project to integrate two databases of metabolic reactions in *E. coli*, EcoCyc[10], and iJR904[11]. Both databases are highly curated, but use different conventions for naming metabolites and reactions. At the record level, an entry describing glucose in EcoCyc is different from an entry describing glucose in iJR904. However, at the domain level, they should be equivalent. Our representation at the domain level was a hybrid of statement level and domain level assertions. We took into account the semantics of cross-references to external databases - each database has a single identifier for a thing and the identifier is meant to be inverseFunctional. In order to have the reasoner compute relationships between reactions, we represented the number and kind of participants (including generic classes of compounds) in a reaction as qualified cardinality restrictions and used an analysis similar to [12] to analyze the computed cross-database relationships. The effort, ongoing, has already surfaced quite a few errors in the source databases, and has contributed to the reaction mapping effort.

The details of how to encode the different levels, what forms of lifting productions might be used etc, are subjects for future development. However articulating that these levels exist, and clearly making and communicating the choice can only help make development of ontologies like BioPAX more effective.

6 Extending OWL to support database record exchange

We now take up the issue of the representation of database records in OWL, because they currently present a roadblock for the segment of the community concerned with communicating at the record level.

Adding a few easily implemented additions to the OWL specification would accomplish two things. First, they would make OWL able to express and validate some checks on the well-formedness of database records, and second they would discourage the incorrect use of existing OWL constructs[13]. The most important proposals are three integrity constraints. While integrity constraints have been proposed as either part of epistemic additions[14] to OWL or as part of a union of OWL and logic-based programming[15], those proposals are substantially more far-reaching and require a more substantial implementation burden than the more limited proposals here.

We propose adequate but not obligatory implementations of the proposals in order to demonstrate that addition of these constructs is feasible. The general framework is that after a reasoning phase, a check is made, and if the check fails, an inconsistency is signalled.

The specific suggestions are:

In order to capture the idea of a required field, a new restriction mustBeAtLeastOneKnown would replace those cases where a minCardinality(1) restriction might be incorrectly used. So instead of the current

```
Class(sequenceFeature partial ...
    restriction(FEATURE-TYPE minCardinality(1)))
```

we would write:

```
Class(sequenceFeature partial ...
    restriction(FEATURE-TYPE mustBeAtLeastOneKnown))
```

The proposed semantics are that the restriction operates as a minCardinality(1) restriction during reasoning, followed by a query for known individuals of the class. Should there be no known individuals, an error is signalled. For BioPAX's purposes, the meaning of the restriction in contexts other than at the top level of the class definition are not needed. However we recognize that in full generality there are some confusing constructions, e.g.

```
Class(sequenceFeature partial ...
    complementOf(restriction(FEATURE-TYPE mustBeAtLeastOneKnown)))
```

and that the behavior of such a constraint when not at top level in a class definition would need to be specified or disallowed.

In order to state that a field can only be part of a certain record type, in place of currently incorrectly used domain declarations, we suggest an axiom in the same position as domain called, say, makeSenseForTypeKnownToBe. So instead of writing

```
ObjectProperty(COFACTOR domain(catalysis) ...)
```

we would write

```
ObjectProperty(COFACTOR makeSenseForTypeKnownToBe(catalysis) ...)
```

It is proposed that there be no reasoning implications for this keyword. For the post reasoning check known individuals of the class restriction(COFACTOR minCardinality(1)) are retrieved and checked to see whether they are instances of the class catalysis. If not an inconsistency is signalled.

In order to state that a field's value can only be filled with values of an appropriate type, in place of currently incorrectly used range declarations we suggest an axiom in an analogous position as range called, for example, canOnlyBeKnownType.

```
ObjectProperty(COFACTOR ... range(physicalEntityParticipant))
```

we would write

```
ObjectProperty(COFACTOR ... canOnlyBeKnownType(physicalEntityParticipant))
```

It is proposed that there be no reasoning implications for this keyword. For the post reasoning check known individuals of the class restriction(inverseOf(COFACTOR) minCardinality(1)) are retrieved and checked that they are instances of the class physicalEntityParticipant.

In order to represent abstract classes—classes used to consolidate common aspects of definitions, but which are not intended to be directly instantiated—we propose a class keyword noKnownDirectInstances. So instead of writing

> Class(externalReferenceUtilityClass partial Annotation(rdfs:comment "... This class is for organizational purposes only; direct instances of this class should not be created."))

we would write

> Class(externalReferenceUtilityClass partial noKnownDirectInstances
> Annotation(rdfs:comment "... This class is for organizational purposes only"))

It is proposed that there be no reasoning implications for this keyword. For the post reasoning check known individuals of the the class are checked to see whether they are known to be direct instances of any subclass of the class. If so, an inconsistency is signalled.

Two further suggestions address issues that have been raised, but we are not convinced that they are necessary. However, since simple implementations are possible, we raise them for discussion.

To aid in a common case where in a database the stated field values are defined to be the only ones, introduce a closed keyword. For example, instead of incorrectly assuming that whenever are three stated PARTICIPANTS[3] in a BioPAX reaction, that these are the only three, declare ObjectProperty(PARTICIPANTS closed)

The implementation is as follows. Following reasoning, each known instance of restriction(PARTICIPANTS minCardinality(1)) is queried for the known values of its PARTICIPANTS property, which are counted. To count, each known individual is checked to see whether it is known to be sameAs any of the other values. If not, the count is incremented. If the count is N, a type axiom type(restriction PARTICIPANTS cardinality(N)) is added to each instance, and once all of these types are added, the resulting ontology is checked a second time.

In BioPAX, there is ordinarily no need to infer sameness of database records (say a copy coming from a different machine), and since it can be burdonsome to assert the commonly assumed case that all database records are distinct individuals, we propose an extension to enable the unique name assumption in a localized way. The keyword on individuals uniqueName would indicate that there is no other differently named individual which is sameAs this one. The implementation (which is only needed if at least one individual has the keyword)

[3] At the Japan BioPAX workshop in November 2007, some attendees reacted in horror when they realized that the reactions they were encoding could legitimately be considered to have unstated additional reactants

is a transformation on the input ontology. We add two disjoint classes and one datatype property, the names of which are guaranteed to not already be in the ontology (indicated here with a * in their names) so that we can be sure that they can't be explicitly asserted by the writer of the ontology. The definitions are

DatatypeProperty(hasName*)
Class(UniquelyNamed* partial restriction(hasName* cardinality(1)))
Class(NotUniquelyNamed12345 partial complementOf(UniquelyNamed*))

Individuals with the uniqueName keyword have their type asserted as UniquelyNamed*. Individuals without the uniqueName keyword have their type asserted as NotUniquelyNamed. All individuals have the property value hasName* set to the string value of their URI.

7 Conclusion

We have reviewed the current state of the BioPAX ontology and identified several distinct approaches to representation that are consistent with the stated goals of the project. These are the record level, the statement level, and the domain level. We observe that the current BioPAX specification suggest a database record semantics, and that OWL is not expressive enough to model this style of representation in a straightforward way, so we propose extensions to extend OWL so that it would be.

More broadly we suggest that it is already and will continue to be common for there to be confusion and tension in projects of this nature because of a lack of clarity about the different levels and their different semantics. A future OWL specification could help this situation by making sure that the language has sufficient distinctions to be able to represent each level's constraints correctly. Further, by doing so, a large class of common OWL errors could be avoided because the availability of appropriate constructs would remove the temptation to use constructs that only superficially match what is needed to express the desired meaning.

8 Acknowledgments

Conversations with Emek Demir, Matthias Samwald and Andrea Splendiani helped crystalize some of these ideas.

BioPAX is the work of the BioPAX working group: Mirit Aladjem, Gary D. Bader, Erik Brauner, Michael P. Cary, Dan Corwin, Kam Dahlquist, Emek Demir, Peter D'Eustachio, Ken Fukuda, Frank Gibbons, Marc Gillespie, Robert Goldberg, Chris Hogue, Olivier Hubault, Michael Hucka, Geeta Joshi-Tope, David Kane, Peter Karp, Christian Lemer, Joanne Luciano, Natalia Maltsev, Debbie Marks, Eric Neumann, Suzanne Paley, Elgar Pichler, John Pick, Harsha Rajasimha, Jonathan Rees, Aviv Regev, Alan Ruttenberg, Andrey Rzhetsky, Chris Sander, Matthias Samwald, Vincent Schachter, Imran Shah, Andrea

Splendiani, Mustafa Syed, Edgar Wingender, Guanming Wu, Jeremy Zucker. (The working group is a dynamic community. We apologize if we have omitted a member from this list.)

References

1. BioPAX Working Group: Biopax web site. http://biopaxwiki.org/ (2006)
2. Karp, P.D.: Call for an enzyme genomics initiative. Genome Biol **5**(8) (2004)
3. Creative Commons, Inc.: The neurocommons. http://neurocommons.org/ (2006)
4. The OBO Foundry: The obo foundry. http://obofoundry.org/ (2006)
5. Smith, B., Ceusters, W., Klagges, B., Khler, J., Kumar, A., Lomax, J., Mungall, C., Neuhaus, F., Rector, A.L., Rosse, C.: Relations in biomedical ontologies. Genome Biol **6**(5) (2005)
6. Grenon, P., Smith, B.: Snap and span: Towards geospatial dynamics. **4** (2004) 69–103
7. BioPAX Working Group: Biopax level 2 documentation. http://www.biopax.org/release/biopax-level2-documentation.pdf (2005)
8. BioPAX Working Group: Biopax level 2 ontology. http://www.biopax.org/release/biopax-level2.owl (2005)
9. Ruttenberg, A., Rees, J., Luciano, J.: Experience using OWL DL for the exchange of biological pathway information. In Grau, B.C., Horrocks, I., Parsia, B., Patel-Schneider, P., eds.: Proceedings of Workshop on OWL Experiences and Directions, Galway, Ireland (2005)
10. Keseler, I.M., Collado-Vides, J., Gama-Castro, S., Ingraham, J., Paley, S., Paulsen, I.T., Peralta-Gil, M., Karp, P.D.: Ecocyc: a comprehensive database resource for escherichia coli. Nucleic Acids Res **33**(Database issue) (2005)
11. Reed, J.L., Vo, T.D., Schilling, C.H., Palsson, B.O.: An expanded genome-scale model of escherichia coli k-12 (ijr904 gsm/gpr). Genome Biol **4**(9) (2003)
12. Li, L., Horrocks, I.: A software framework for matchmaking based on semantic web technology (2003)
13. Horridge, I.: Common errors in OWL. http://protege.stanford.edu/conference/2004/slides/6.1_Horridge_CommonErrorsInOWL.pdf (2004) (7th International Protégé Conference).
14. Katz, Y., Parsia, B.: Towards a nonmonotonic extension to OWL. In Grau, B.C., Horrocks, I., Parsia, B., Patel-Schneider, P., eds.: Proceedings of Workshop on OWL Experiences and Directions, Galway, Ireland (2005)
15. Motik, B., Horrocks, I., Rosati, R., Sattler, U.: Can OWL and logic programming live together happily ever after? In: Proc. of ISWC-06, Springer-Verlag LNCS (2006) to appear.