Semantic browsing of pathway ontologies and biological networks with RDFScape (working paper*)

Andrea Splendiani**

Unité de Biologie Systémique, Institut Pasteur, Paris and DISCO, Università di Milano-Bicocca.

Abstract. Studying biological organisms at the systems level is a complex task. Computational approaches require structured representations of existing biological knowledge. This necessity has prompted the development of formal representations of specific areas of knowledge, resulting in ontologies such as Gene Ontology and BioPAX. However, only part of this formalized knowledge is exploited for the interpretation of experimental data. Specifically, it is common to use the association between entities and annotations, like genes and functions, while the structure of the annotation is not considered beyond some common features as inheritance. This is partly due to a lack of tools and methods that bridge resources related to ontologies to the ones related to data analysis.
Here we present a platform that merges a semantic web toolkit with a widely adopted modular tool for systems biology investigation. We demonstrate how in this environment it is possible to query ontologies not only as a list of annotations but as a knowledge base from which new information can be derived. We also show how this knowledge can be integrated with biological data.

Introduction

Formal classification schemes have historically been important tools for information-rich areas of biology, such as systematic biology or anatomy. In recent years, the development of new experimental technologies allows to generate information on living systems at the molecular level, on a large, genome-wide scale. This generates a need for formal classification in this area, for two important reasons.

1. The *representation* of this type of large-scale data and knowledge in a form comprehensible by humans requires a formal framework.
2. The *interpretation* of the experimental data on this scale requires computational approaches, for which, in turn, a formal framework is mandatory.

* This paper is addressed to a public with a basic knowledge of ontologies in the context of functional genomics. It is intended as a draft presenting current work on ontology based-data analysis. Feedback is encouraged from the ontology and bioinformatics communities.

** andrea@pasteur.fr

Dagstuhl Seminar Proceedings 05441
Managing and Mining Genome Information: Frontiers in Bioinformatics
http://drops.dagstuhl.de/opus/volltexte/2006/474

To demonstrate our ideas, we focus on functional evaluation of high-throughput data. In this context, measures relative to many genes in an organism are related to what is known about their function in order to characterize the observed condition. This usually involves the analysis of the correlation of attributes associated to genes (for instance, their function) with experimental measurement of some feature of the genes in a specific observed case (for instance, their change in expression after a stimulus).

A review on functional evaluation methods and resources can be found in [8][1]. As the review points out, current approaches lack the ability to consider properly the relations among known functions of genes. That is, they consider functions as defined by attributes, not properly taking into account the relations between these attributes.

One of the most important resources for functional evaluation is Gene Ontology[1]. Gene Ontology defines a set of terms describing some biological properties of genes, and explicitly provides two kind of relations: part-of and is-a. It provides this kind of knowledge representation for biological function, biological process and cellular compartment.

Some other resources go deeper in terms of the level of detail with which they describe relations among properties of genes. This is the case for pathway ontologies, where genes interact in a biological process, and their function and their role are explicitly stated.

It is for this class of ontologies that the ability to view the available knowledge as a network of related concepts, instead of a list of annotations, will have a bigger impact. Thus, this is the focus of the present work.

In particular we focus on the BioPAX [2] ontology, that is a common way to represent information of biological pathways available in public databases such as KEGG[6], BioCyc[7], Reactome[5][3].

A discussion on resources for functional evaluation of gene activity specifically for pathway ontologies can be found in [2]. Detailed representations of the mechanisms in biological pathways are usually approximated such that the description of interactions and roles between genes[4] in a pathway is lost.

[1] This review deal with functional evaluation of data from gene expression experiments. This is one of the settings in which functional evaluation procedures are more important, and the techniques applied are generally used also for other types of data.

[2] For more information we refer to: http://www.biopax.org

[3] This is just a subset of resources that provide a BioPAX export of their information. BioPAX is being defined a standard to represent pathway and interaction information and the amount of information available in its format is expected to grow.

[4] We refer to the term "gene" here. But a pathway generally represent a set of entities, some of which, the proteins, can be considered as related to genes. The current technologies are in general able only to measure properties of genes. These can be used as an indirect measure of properties of proteins, which are key players in the biological processes. For this reason it is common to approximate a pathway, where proteins and molecules interacts in a network, to a list of genes corresponding to proteins.

There are projects specifically dealing with pathways that make use of this richer level of detail, we just cite one of them [9], these are in general able to derive properties as, for instance, possible causal chains between genes. However these tend to be designed for specific pathway databases of representation styles. Thus there is a gap between the ability to use detailed information by these resources, and the more approximate usage of the same information for functional evaluation of high-throughput data.

We believe that one of the reasons causing this gap is the lack of tools and methods, based on open standards, that can bridge the analysis process with the ability to exploit the fine level of detail present in ontologies. Therefore we propose a platform that creates this bridge, by unifying one framework to handle ontologies represented in a standard way with a tool for biological network visualization and analysis.

In this paper we show how in this framework ontologies are not only a set of attributes, but a knowledge base from which specific knowledge can be derived regarding the biological system under investigation. The ultimate goal of this work is to enable the use of semantics in high-throughput data analysis, this means that functional evaluation will not just relate data to set of attributes, but to a network of related concepts.

The rest of the paper is organized as follows: first, a standard way to represent relations among biological elements will be briefly introduced (the Semantic Web). Then it will be shown that our tool can be used to query and browse the information encoded in ontologies, and successively it will be shown how the semantics of relations among elements can be used to derive new information. We will show a brief example of integration of data and ontologies, some implementation details, and we will discuss the presented approach, its limitations and planned future work.

Semantic Web and pathway ontologies

The semantic web, originally envisioned by the Tim Berners-Lee, aims at enriching the current information available on the web with machine readable meta-information. In this vision the web will not provide only generic links between elements of information, but a way to represent the meaning of information resources and the links between them.

The usefulness of the Semantic Web in the Life Science community has been discussed in a number of papers (A generic discussion can be found in [11] while a use case is in [13]).

For the scope of this paper, we point out that the Semantic Web provides a platform, with standard languages, and available tools, to represent semantically rich information and ontologies in particular. A number of ontologies like Gene Ontology[1] and BioPAX have been translated to, or defined with, Semantic Web languages.

A description of Semantic Web is outside of the scope of this paper. We will present only some key concepts here.

URIs and RDF

URIs, Unique Resource Identifiers, are one of the key concepts of the semantic web. A URI is an identifier unambiguously (and stably) associated to a piece of information.

For instance, HTTP://WWW.BIOPAX.ORG/RELEASE/BIOPAX-LEVEL1.OWL is uniquely associated to the biopax ontology level1 itself [5].

Another key component is RDF, the Resource Description Framework. This is a simple language to describe relations among resources, where a resource is defined as something being identified by a URI. RDF expresses these relations[6] as a list of statements having the form SUBJECT PREDICATE OBJECT, where a subject can be a resource, an object can be a subject or a value, and both resources and predicates are identified by URIs[7].

The data model that results is a typed binary graph. It can be represented though several syntaxes among which N3, XML, and graphs have been standardized.

RDFS and OWL

RDF provides a definition of a common model to represent relations among information, but does not address the meaning of these relations. If, for instance, we want to represent a pathway, we can use RDF to represent it (and maybe XML or N3 to write it), but we need a way to define how a pathway can be represented. Not all networks of elements and relations are pathways. RDFS (RDF Schema) addresses this problem defining a set of URIs and a semantics associated to them, that allows the definition of valid models. We refer to www.w3.org/RDF/ for a formal definition of the RDFS semantics[8], and we show here an intuitive example[9]:

```
http://www.kegg.org/owl#proteinx rdf:type biopax:protein
biopax:protein RDFS:subclassOf biopax:elements
entails
http://www.kegg.org/owl#proteinx rdf:type biopax:entity[10]
```

[5] Currently both BioPAX level1 and BioPAX level2 are available. They differ in the extent of the information they can represent, where all that can be represented in level1 can be represented in level2.

[6] since we are going to use RDF to represent our knowledge, we are going to refer from now on to the set of elements and relations represented as knowledge or know-how

[7] We will refer to these statements also as "facts".

[8] A semantic, or meaning of a proposition (for instance an RDF statements) can be denotationally defined as a function on the set of possible propositions.

[9] note that RDFS is still represented as RDF. It is a particular set of resources with an asociated meaning that allow to interpret whether a given RDF model is valid or not

[10] This example illustrates intuitively the concept of entailment. Note that, however, it is not a real-world example and is not accurate, in that it is not stated that protein and elements are classes. Also proteinx and its namespace are invented.

The meaning of entailment is that, whenever the premises are true, the entailed sentence is also true.

OWL, the ontology web language, extends RDFS providing a richer vocabulary of terms (URIs) to define ontologies[11]. In particular OWL introduces URIs to define concepts such as classes (a set of resources).

OWL is defined in three levels (OWL-Lite, OWL-DL, OWL-Full). OWL-DL strictly requires that no resources can be denoted both as elements of classes (instances) and classes. This and other restrictions allow OWL to be decidable. This means that the function used to denote the semantics of any possible OWL-DL description will always yield a valid or non-valid result[12][13].

In summary, an OWL ontology is an RDF list of statements (we will call it a file, taking a pragmatic approach), whose semantics present constraints that define the set of valid ontologies. The stronger the constraints, the smaller is the set of valid RDF models, where a generic RDF file will be valid relatively to the OWL-full semantics, and will be a superset of all possible OWL-DL valid RDF files that would be a superset of all possible OWL-Lite files.

Note on terminology

We refer to the term ontology with several meanings. We define an ontology as formal way to represent concepts and relations among concepts. Sometimes ontologies constitute of classes (sets of individuals) and the term ontology defines the set of classes and relations between them. While the term knowledge-base is used to refer to all the information provided.

We also use ontology as a word to designate information provided in a formal way. Such usage is common when dealing with resources as Gene Ontology. We talk about ontologies also to designate pieces of formalized information provided by specific information provider.

Annotation in this context is intended to be a more general term than ontologies.

We use the term data only to indicate information that is relative to an experimental condition.

BioPAX

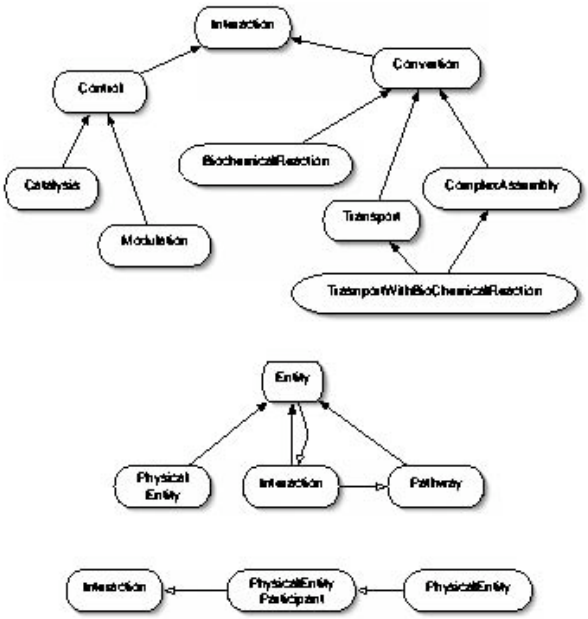
We are using BioPAX as our target ontology and for examples throughout this paper. An explanation of its definition and scope is out of this context. We report

[11] It should be clear why an ontology is often defined as "a formalization of a conceptualization".

[12] This is in general not true for more complex ontologies. By Godel's theorem, any formalism that has a universal quantifier contains at least an undecidable proposition.

[13] To be precise, OWL-DL guarantees the the satisfiability problem (whether classes can have elements) and more generally the subsumption problem (whether two classes are equivalent) are decidable.

here a synthetic graphic overview of its classes, in order to understand the following references to this ontology. Is-a relations are indicated with black arrows, part-of relations with white arrows. We refer to the BioPAX level1 release[14] and to exports from KEGG[15] for the following discussion.



RDFScape

In this paper we propose an environment, RDFScape, that aims at bridging the life science and ontology research methods and instruments. It is realized as a plugin for Cytoscape[12] and makes use of the Jena[16] semantic web libraries.

It inherits all the visualization features of Cytoscape and its plugins to analyze biological networks and data, it enhances it with the ability to visualize, query and reason on ontologies.

Cytoscape is primarily dedicated to the analysis of biological networks, but its features are almost domain independent. RDFScape also is primarly targeted at the analysis of bio-ontologies, but can be effectively used as an interactive visualization and query system for genereal ontologies represented in RDF.

[14] This release is intended to cover metabolic pathway information. http://www.biopax.org/release/biopax-level1.owl

[15] The export is still in beta testing and available at: http://www.norhbears.org/biopax/ in particular the sce (yeast) files are used.

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

RDFScape provides a notion of "analysis context". This defines a set of concepts, instructions and settings relative to a specific area of interest (for instance, BioPAX based pathway analysis context, GO analysis context...).

In general, there are two levels of usage of RDFScape. A novice user, who is not expected to have a deep knowledge of the ontologies he is using, will most likely use pre defined contexts.

An advanced user, on the other other hand, will be able to define it own analysis settings. In its advanced usage, RDFScape can be complemented by other tools like Protégé[17] to understand the structure of the ontologies considered.

At the time of this draft RDFScape is in alpha testing. It will be released under the LGPL license.

Query and visualization of ontologies

RDFScape provides several ways to query ontologies. It provides low level text-based query facilities, where query languages defined in the semantic web can be used to query a set of ontologies[18], it provides type-based queries (for instance, returns all the elements of type proteins), or string queries (for instance: returns everything that has name matching *P53*).

Once results are provided by one of these queries, these can be visualized in a graph via Cytoscape. This graph can be interactively expanded to browse the information (and meta-information) associated to each element in the ontologies. It can be edited and used to define visual queries. These aspects will now be explained in detail.

Querying ontologies through RDQL

RDFScape provides an interface to perform RDQL queries on ontologies. RQDL is one of the languages provided in the semantic web framework to perform queries on Semantic Web knowledge bases. During the development of this project, a new language, SPARQL, has been defined and proposed as standard. RDFS is planned to adopt SPARQL instead of RDQL soon[19]. This type of query is targeted at users that have a moderate knowledge of the ontology. It allows quick inspection and debugging. Novice users are provided with other text queries that avoid this complexity, or with pre-defined queries and inference rules that avoid non-intuitive relations stated in ontologies.

RDQL queries are equivalent to graph pattern matching operations performed on the graph of concepts represented in RDF. For each possible selection of tuples matching this pattern, a projection of the required variables is returned. Here is an example, taken from BioPAX:

[17] http://protege.Stanford.edu

[18] These must be indicated by the user. They can be both present on the local system or available on the network.

[19] RDQL is a subset of SPARQL, with only minor syntactic differences.

```
    Select ?interactionName ?proteinName
    Where (?interaction bp:PARTICIPANTS ?physEntity) (?interaction bp:NAME
?interactionName)
    (?physEntity bp:PHYS-ENTITY-PARTICIPANT ?myProtein) (?myProtein bp:NAME
?proteinName)
    (?protein rdf:type bp:protein)
    USING bp FOR <http://www.biopax.org/release/biopax-level1.owl#>
    rdf for <www.w3c.org/1999/02/22-rdf-syntax-ns#>
```

An example of a possible result[20] is shown in table .

```
Prostaglandin and leukotriene metabolism delta 1-pyrroline-5-carboxylate reductase
Prostaglandin and leukotriene metabolism Acetylornithine aminotransferase
Prostaglandin and leukotriene metabolism argininosuccinate lyase
Prostaglandin and leukotriene metabolism seccnd enzyme in prcline biosynthesis
Prostaglandin and leukotriene metabolism responsive to contrcl of general amino acid biosynthesis
Prostaglandin and leukotriene metabolism Sixth step in arginine biosynthesis
Prostaglandin and leukotriene metabolism Urea amidolyase (contains urea carboxylase and allophanate hydrolase)
Prostaglandin and leukotriene metabolism biosynthesis of spermidine
Prostaglandin and leukotriene metabolism ornithine aminotransferase
Prostaglandin and leukotriene metabolism ExtraCellular Mutant; cik1 suppressor
Prostaglandin and leukotriene metabolism arginosuccinate synthetase
Prostaglandin and leukotriene metabolism N-acetyl-gamma-glutamyl-phosphate reductase and acetylglutamate kinase
Prostaglandin and leukotriene metabolism arginase
Prostaglandin and leukotriene metabolism Spermine Synthase
Prostaglandin and leukotriene metabolism catalyzes first step in proline biosynthesis
Prostaglandin and leukotriene metabolism Similar to human LTA4 hydrolase but in vivo substrates not yet defined.
Prostaglandin and leukotriene metabolism ExtraCellular Mutant
Prostaglandin and leukotriene metabolism Rate limiting step cf polyamine biosynthesis pathway
Urea cycle and metabolism of amino groups delta 1-pyrroline-5-carboxylate reductase
Urea cycle and metabolism of amino groups Acetylornithine aminotransferase
Urea cycle and metabolism of amino groups argininosuccinate lyase
Urea cycle and metabolism of amino groups second enzyme in proline biosynthesis
Urea cycle and metabolism of amino groups responsive to control of general amino acid biosynthesis
Urea cycle and metabolism of amino groups Sixth step in arginine biosynthesis
Urea cycle and metabolism of amino groups Urea amidolyase (ccntains urea carboxylase and allophanate hydrolase)
Urea cycle and metabolism of amino groups bicsynthesis of spermidine
Urea cycle and metabolism of amino groups ornithine aminotransferase
Urea cycle and metabolism of amino groups ExtraCellular Mutant; cik1 suppressor
Urea cycle and metabolism of amino groups arginosuccinate synthetase
Urea cycle and metabolism of amino groups N-acetyl-gamma-glutamyl-phosphate reductase and acetylglutamate kinase
Urea cycle and metabolism of amino groups arginase
Urea cycle and metabolism of amino groups Spermine Synthase
Urea cycle and metabolism of amino groups catalyzes first step in proline bicsynthesis
Urea cycle and metabolism of amino groups Similar to human LTA4 hydrolase but in vivo substrates not yet defined.
Urea cycle and metabolism of amino groups ExtraCellular Mutant
Urea cycle and metabolism of amino groups Rate limiting step of polyamine bicsynthesis pathway
Novobiocin biosynthesis delta 1-pyrroline-5-carboxylate reductase
Novobiocin biosynthesis Acetylornithine aminotransferase
Novobiocin biosynthesis argininosuccinate lyase
Novobiocin biosynthesis seccnd enzyme in prcline biosynthesis
Novobiocin biosynthesis responsive to contrcl of general amino acid biosynthesis
Novobiocin biosynthesis Sixth step in arginine biosynthesis
Novobiocin biosynthesis Urea amidolyase (contains urea carboxylase and allophanate hydrolase)
Novobiocin biosynthesis biosynthesis of spermidine
Novobiocin biosynthesis ornithine aminotransferase
Novobiocin biosynthesis ExtraCellular Mutant; cik1 suppressor
Novobiocin biosynthesis arginosuccinate synthetase
Novobiocin biosynthesis N-acetyl-gamma-glutamyl-phosphate reductase and acetylglutamate kinase
Novobiocin biosynthesis arginase
Novobiocin biosynthesis Spermine Synthase
Novobiocin biosynthesis catalyzes first step in proline biosynthesis
Novobiocin biosynthesis Similar to human LTA4 hydrolase but in vivo substrates not yet defined.
Novobiocin biosynthesis ExtraCellular Mutant

Novobiocin biosynthesis Rate limiting step cf polyamine biosynthesis pathway
```

The query intends to associate interactions to their protein participants. It requires knowledge of the meta-information represented in the biopax ontology

[20] These are the actual results from the application of this query on files sce00040.owl, sce00053.owl and sce00061.owl from the BioPAX Kegg export available at http://www.northbears.org/biopax/

and its semantics. For instance, it requires the knowledge of PARTICIPANTS and PHYS-ENTITY-PARTICIPANTS terms.

BioPAX specifies each interaction as having one or more participants. These participants are defined as physical entity participants, that is, entities (proteins, small molecules...) enriched by additional information relative to the they way they participate in the interaction (stoichiometry).

For further detail on the BioPAX ontology we refer to the BioPAX documentation.

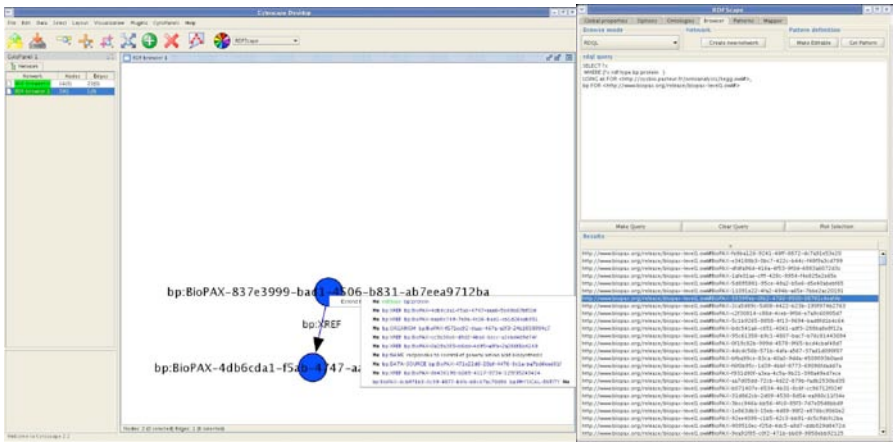
We note that we must specifically state in the query that we want the name property listed. Otherwise URIs for those elements will be returned. These are not required to have a meaning associated, and the same query would yield results of the form:

> http://www.biopax.org/release/biopax-level1#07f1e5f7-5fb1-470a-ad3b-61ea687b399e http://www.biopax.org/biopax-level1#BioPAX-5ea089ce-6900-46d8-a514-183395f75ff6[21]

Elements resulted from a query can be represented on a Cytoscape graph. A context menu associated to each element allows the expansion the network of annotations interactively.

Browsing ontologies

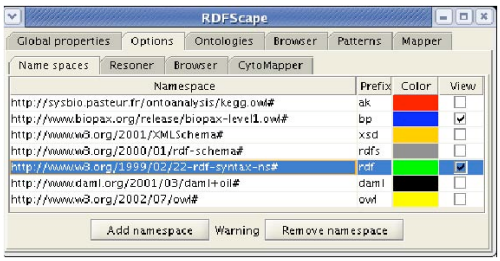
Once a query is performed, the results can be mapped on a Cytoscape graph. It is then possible to interactively browse this graphs through a common right-selection expand user interface mechanism.



One characteristic feature RDFScape provides is the ability to filter and color the relations to be navigated based on namespaces[22]. An example of this can be seen above and the configuration is reported here.

[21] Note that actual files used in examples have an error in putting entities' URIs under the biopax namespace.

[22] A namespace is a logical grouping of terms, usually identified by a common prefix.



Since namespaces tend to be associated to levels of information, this is a practical way to navigate the desired information. For instance disabling owl, rdf and all the meta information vocabulary will let the user navigate only the more informative content of an ontology, while associating different colors to a selection of namespaces can be useful to study the relation between data and meta-data or between different sources (and possibly kinds) of data.

In the following example we show one graph from the BioPAX ontology.

On the left, a set of resources and relations among resources describe a piece of information about a pathway. These are defined in their proper namespace associated to the blue color[23], some of them are related to a string through the property NAME[24]. These are then related to another set of resources, on the right, associated to classes[25]. Classes are then related among themselves through different relations, among which owl:differentFrom and rdfs:subclassOf are shown. Note how the color allow to easily distinguish the different levels of meta information involved[26].

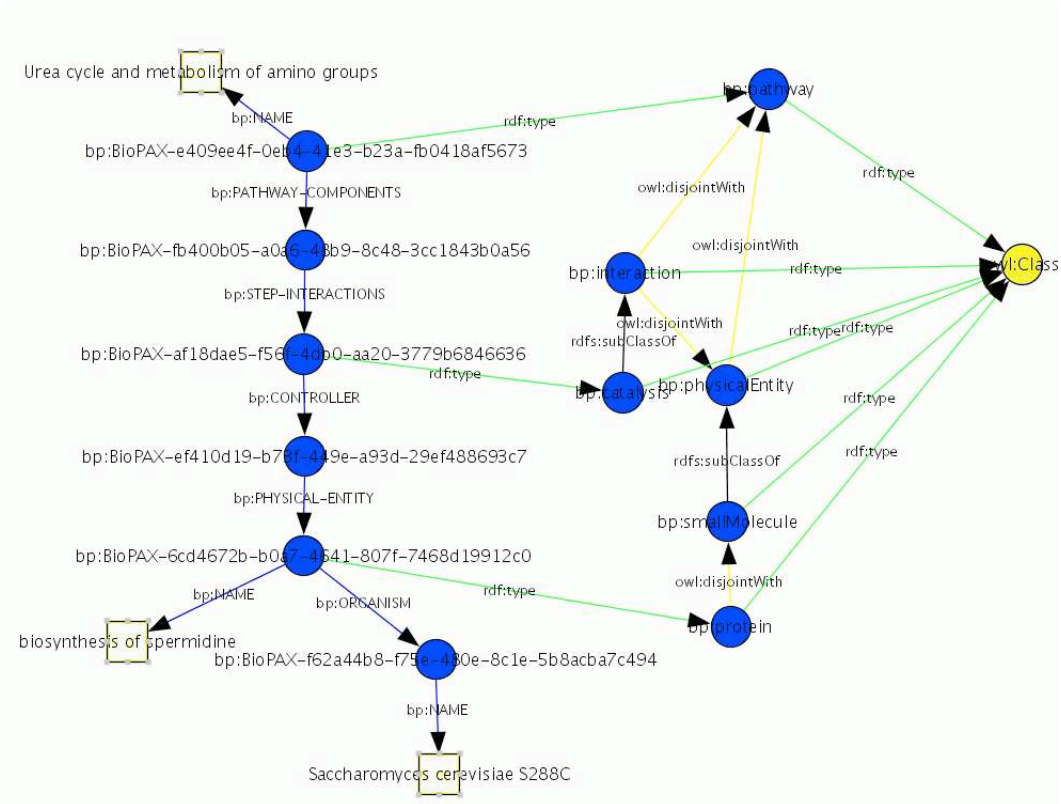
While this picture can be too complex and far from the biological content, it is only a demo to illustrate the meta-information richness of ontologies. As will be shown later, the user doesn't need to understand the standard meta-information proposed, because this can be used by a reasoner to provide a richer model of the system under study. Anyway we note how this approach allows the user to understand the meaning of concepts. In this case, we see that an element of type protein is considered different from an interaction in our model, while both proteins and smallMolecules are of type entity.

[23] These resources are defined in the BioPAX (bp) namespace. This is an error present in current version of the ontologies used for these examples at the time of this paper. They should be associated with a distinct namespace. If namespaces were correct, resources on the left (related to individuals) would be represent with a different color from the ones on the right (related to classes).

[24] Without going in details, literals have a representation than resources.

[25] These classes are defined properly in the BioPAX namespace. See previous note.

[26] In particular, note as resources denoting elements are related by edf:type to classes and how these classes are related by rdf:type to the class of classes, in owl namespace.



Still, it is clear that this level of understanding of ontologies is far from the user interested in data analysis. Thus RDFScape exploit other features of the Cytoscape platform and of the semantic web environment to bring these two world closer.

Defining visual queries

Since there is an equivalence between RDQL queries and graph patters, RDFScape allows the user to define queries editing a graph and declaring variable nodes or edges. These queries can then be used to search matching subgraphs in a graph[27] or to generate new derived ontologies. Examples of how a visual query looks like in RDFScape are shown later, where queries are used in conjunction with inference.

[27] Derived from data or representing ontologies

Inference support

The knowledge on which queries are performed is not the union of the statements in the ontology files proposed by the user. It is this knowledge augmented by the set of true facts that must hold given the stated premises.

New facts are entailed by RDFS or OWL semantics, as well as by additional rules defined by the user (or provided in a specific analysis context).

RDFScape offers support for inference based on OWL/RDFS semantics and allows the user to define custom rules for application-specific reasoning.

This is a key feature of RDFScape not present in other systems and that has impact on queries, visualization and analysis of ontologies and data.

Implementation details of the reasoning mechanisms are discussed later. Here we present a few examples that demonstrates what can be achieved through the use of inference.

For a given analysis context, a set of reasoning settings (as well as a set of rules) is pre-defined, and can be altered or edited by the user.

In particular the user has the choice of the standard entailments to be applied (RDFS, OWL) as well as custom ones (rules).

Since there is a trade-off between the extent to which entailments are computed and efficiency, for standard entailments three levels of complexity are provided (Low, Medium, High), were Low means very efficient but possibly incomplete.

RDFScape provides an editor to define rules, that must be written with an had-hoc language

User defined rules are defined through an ad-hoc language[28]. Note that misuse of rules could make the ontology undecidable or make the system loop. Therefore rule design is intended for advanced users. Novice users should rely on pre-defined rules[29].

We will now show two examples of how inference can be applied to improve the usage of ontologies in biological investigation.

Abstraction level The way the BioPAX ontology defines the relations between interaction, control, catalysis and conversion was briefly illustrated previously.

Note that depending on the type of information represented in biopax, a specific kind of interaction is stated. For instance, the BioPAX representation of KEGG files explicitly states that enzymes have a control function in a catalysis reaction and that metabolites are participants in a biochemical reaction.

If this query is applied to the BioPAX representation of KEGG yeast files without inference support, it would yield no results:

[28] The language in which rules are serialized by the internal engine, Jena, is used. This is a straightforward language to learn. There is not a clearly defined rule language to use in combination with OWL yet. But it is expected to adopt it when a standard will be defined.

[29] An improper definition of rules could result in the system aborting for memory overflow.

```
SELECT ?interactor1 ?interactor2
WHERE (?x rdf:type bp:interaction) (?x bp:PARTICIPANTS ?px1)
(?x bp:PARTICIPANTS ?px2) (?px1 bp:PHYSICAL-ENTITY ?interactor1)
(?px2 bp:PHYSICAL-ENTITY ?interactor2)
USING bp FOR <http://www.biopax.org/release/biopax-level1.owl#>
rdf for <www.w3c.org/1999/02/22-rdf-syntax-ns#>
```

However if the inference engine is activated with entailments for RDFS (or OWL)[30], it will deduce that biochemicalReaction is a specific kind of interaction, therefore it will return all elements for which biochemicalReaction is explicitly stated[31]. It will not, however return catalysis reactions, since these don't need a physical-entity-participant (they have not stoichiometry associated) and match a different pattern. This is a limit derived from the way BioPAX is designed now. Again, we can overcome this inconvenience with custom designed inference, as shown in next paragraph.

It can be argued that a user knowing the ontology in use could define a query giving equivalent results without the need for automated reasoning.

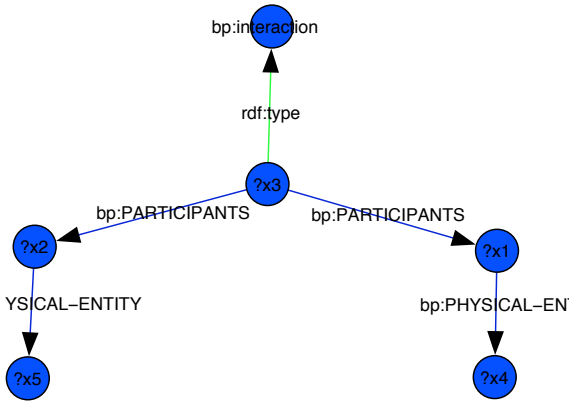
However, here the user may only ask for a well defined bp:PARTICIPANTS of a bp:interaction pattern, with no need to know more about the ontology or the specific usage of the ontologies in a context. Moreover, the query could be performed on a mix of different resources whose explicitly stated information is unknown to the user. Finally, not all inference procedures involves only subclassing, and they may not be trivial to implement in a query.

We note also that while the user should indicate the correct level of inference desired, this is only for performance issues (the set of deductions performed with OWL, High settings will include all other possible deductions with other settings[32]).

It is easy to experiment reasoning and queries, and default selections are provided in pre-defined analysis contexts.

Derived relations for readability The query in the previous example has the following visual equivalent[33]:

[30] Any level of reasoning will be sufficient in this case.
[31] More precisely this query will return all pairwise associations between elements associated through the PARTICIPANT property to the same interaction.
[32] We assume the our ontology is in OWL-DL.
[33] Variable are given a consecutive variable identifiers by RDFScape.

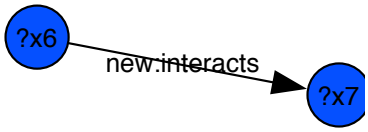


However, from a user point of view, it would be more intuitive to be able to request all interacting elements in a simpler way.

If we define a custom rule that directly asserts interactions between two molecules ignoring information on the modality:

```
   [Skip-Context: (?interactor1 new:interacts ?interactor2) <- (?x rdf:type
bp:interaction) (?x bp:PARTICIPANTS ?px1)
   (?x bp:PARTICIPANTS ?px2) (?px1 bp:PHYSICAL-ENTITY ?interactor1)
(?px2 bp:PHYSICAL-ENTITY ?interactor2)]
```

We could express the above query as:



This is a simple example of the use of inference to enhance readability of an ontology.

If we add a second rule as:

```
   [Define-interction: (?interactor1 new:interacts ?interactor2) <-
(?x rdf:type bp:interaction)
   (?x bp:PARTICIPANTS ?interactor1) (?x bp:PARTICIPANTS ?interactor2)]
```

We have obtained a complete "operative" generalization of the notion of interaction. We can now apply the above query to the whole set of KEGG yeast files, and we would get the abstract view of interactions between elements in pathways in yest.

Derived relations for analysis We now show how inference can be used to provide an application specific transformation of the original ontology.

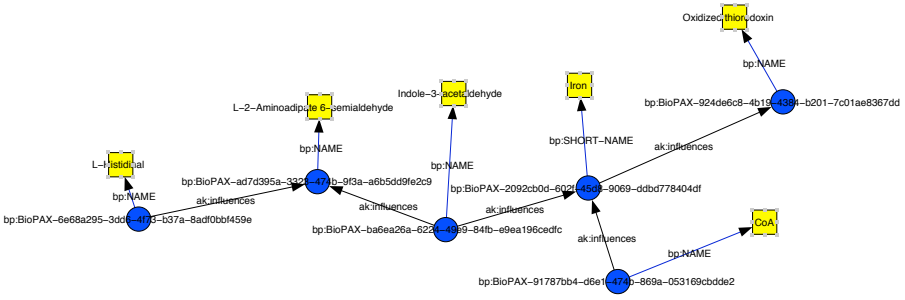
Without investigating deeply its biological relevance, we define a custom function "affects" with the following semantics: a affects b if there is a reaction that requires a to produce a third element c, and c is needed by another reaction to generate b.

This is almost a demo rule but it's easy to think how with some investigation biologically meaningful relations can be defined.

This rules can be represented as follows:

```
   [influence: (?x ak:influences ?y) <- (?p1 rdf:type bp:biochemicalReaction)
(?p2 rdf:type bp:biochemicalReaction)
   (?p1 bp:RIGHT ?k1)(?k1 bp:PHYSICAL-ENTITY ?k)
   (?p2 bp:LEFT ?k2)(?k2 bp:PHYSICAL-ENTITY ?k)
   (?p1 bp:LEFT ?k3)(?k3 bp:PHYSICAL-ENTITY ?x)
   (?p2 bp:RIGHT ?k4)(?k4 bp:PHYSICAL-ENTITY ?y) ]
```

This derived information is available both while browsing the graph and while querying it. Here is an example of some relations obtained while browsing it, starting from "Iron".



It could also be possible to visualize the graph of all possible influences between elements. This would be a transformation of the original graph and it would be obtained by applying a ?X NEW:INTERACTS ?Y pattern (this is the equivalent query) to the graph enriched by the above custom inference rule.

This query would perform a filter operation (affine to a projection) to the entailment closure [34] of truth conditions (this could be thought slightly related to a join).

Note that this new relation will be available (if the proper namespace is selected) also while browsing the ontology.

Of particular interest in our approach is that ontologies are represented only through standard languages, and reasoning uses standard, almost interchangeable, engines.

Integration of ontologies and data in the same environment

Another features of RDFScape is the possibility to link graphs representing ontologies with other graphs present in cytoscape. It is assumed that knowledge about the correspondence between identifiers is present in the ontology. This assumption is reasonable since it's easy to convert a file containing a mapping of identifiers in rdf, and the sameAs relation can be used to asses their correspondence, a proper reasoner should in theory handle this consistently. Another reason for this choice is that the mapping problem may be non trivial: having information about the mapping represented in RDF (or OWL) it will be possible to build complex solutions on top of it.

In any case, RDFScape must be instructed as to which attribute is a valid identifier in a graph, and to which class in the ontology it maps. In this version we assume that the graph will always use the name of the node as the identifier.

As an example we consider BioPAX KEGG yeast files and some yeast datasets available in the Cytoscape distribution[35] (BindYeast and GalFiltered).

We indicate the correspondence between identifiers in this way[36]:

```
(?x <bp:XREF> ?y) (?y <bp:DB> ?z) (?y <bp:ID> ?w) AND ?z eq <SGD>
```

We obtain the following figures:

Dataset	nodes/edges	URI matching	ID resolved	Conflicts	Multi	Ont. Cov.	Graph Cov.
BindYeast	5800/27943	510	413	79	0	93%	7%
GalFiltered	331/362	510	413	79	0	9%	11%

We can add microarray data to this graph. We choose a yeast dataset gal5936x20 from the the Cytoscape distribution. We obtain the following additional figures:

[34] It is not really the closure of entailed relations given the modularization of the reasoning steps described in the implementation section.

[35] test data

[36] This can be defined in the proper analysis context

Dataset	Ontology coverage in data	Data coverage in ontology
BindYeast	100%	7%
GalFiltered	100%	7%

Just a few notes. "URI matching" and "ID resolved" are relative to the correspondence between data identifiers and URIs that can be found in the ontology. "Conflicts" are errors due to the lack of RDF-identifiers (and of a measures to compensate this lack).

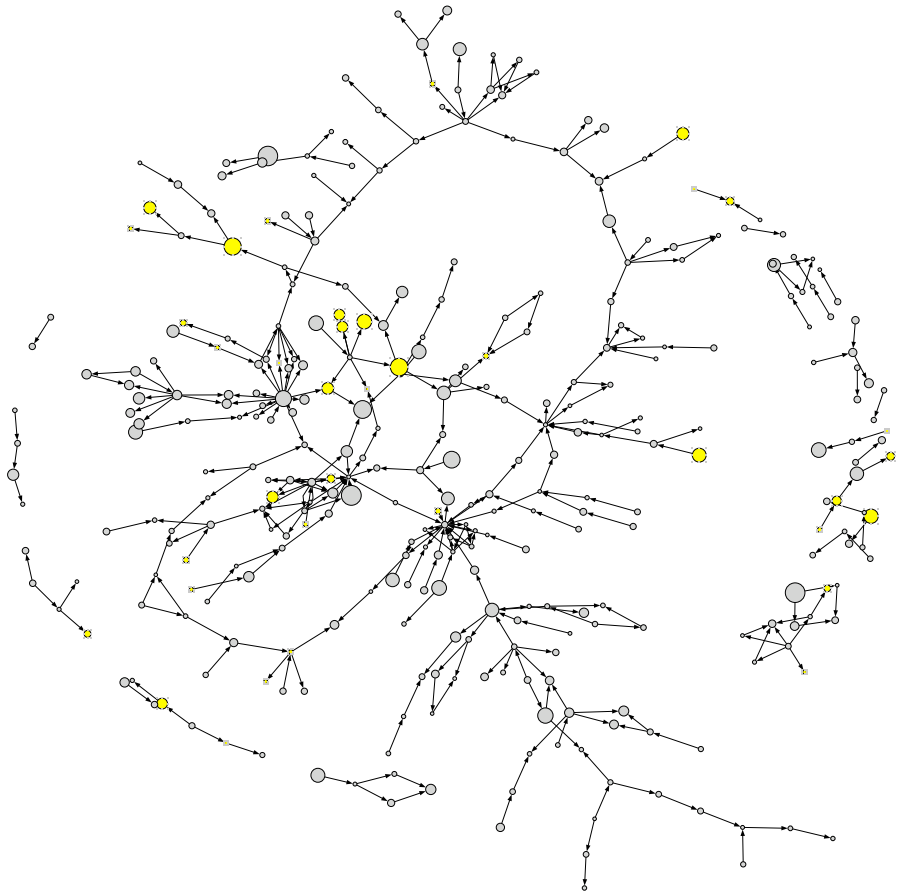
"Ontology coverage" figures indicate how many elements in the ontology have associated elements in the graph or data values. Vice-versa, "Graph coverage" and "Data coverage" indicate how many elements in the data graph or in the values can be related to ontology entries.

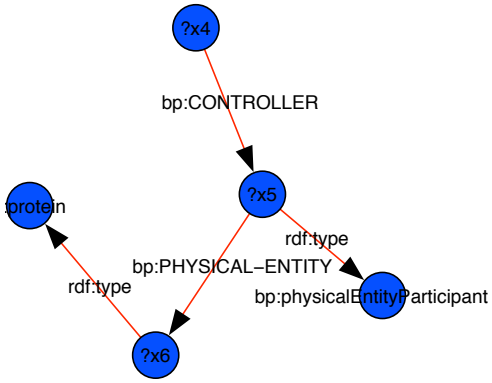
The figures reflect the scarcity of pathway ontological annotation.

At this point we can use search functions based on ontology properties to find elements in a graph including experimental values. Here we provide a simple example. In the following figure a protein interaction graph is represented with the dimension of nodes proportional to their "fold change"[37] values (derived from experimental data). At the same time, all elements that are known to be controllers[38] highlighted. This derives from our formal pathway knowledge. The corresponding query is also shown.

[37] Each node represents a protein. The fold change is a measure of the differential activation of a gene (that encodes a protein) in two conditions.

[38] Controller is an abstract term, these nodes may have be determined through inference





Implementation

RDFScape is realized as a Cytoscape plugin. It makes use of the Jena libraries v. 2.3.

It makes use of the Jena inference engine in a particular way, here described.

A set of ontologies indicated by the user are parsed and stored in a Jena internal representation of a unified ontology. This representation is completely handled in memory for efficiency[39]. This representation is used as a list of known facts, on top of which a first set of user defined rules is applied. This set of rules is intended to provide some support for unification of ontologies (for instance, to resolve unificationXrefs in BioPAX).

Resulting from this inference is a richer knowledge, including both facts stated in the original ontologies, and derived ones. This resulting knowledge is than considered as the input of the inference process that derives additional facts through the application of OWL and RDFS entailments.

This last set of information is again enriched with a second set of user defined rules.

The reason for this layered reasoning process reflects the current level of maturity of ontologies as BioPAX.

In fact, a first level of user defined rules is intended to unify different ontologies. For instance it is expected that equivalence of terms with different URIs in different files will be detected and resolved. Note that this is a transient necessity due to the lack of full support of the RDF specifications by some information providers.

A second level of user defined rules is intended to provide application-oriented inference. Ontologies for pathways now available are rich in instances and relatively poor in classes structure. Moreover, different ontologies[40] will have the

[39] The Jana platform itself provides more options

[40] Here with the meaning of a specific ontology file, including classes and instances.

same classes and different instances. Thus most of the application oriented inference is not expected to alter the OWL/RDFS entailments, but to have knowledge of it. This condition is enforced by this layered architecture to avoid rule design errors complex to debug.

The rules are defined as backward[41], hence they are triggered on demand by the reasoner.

As a last note regarding the inference system, the mapping between the levels of reasoning indicated by the user and the reasoning setting used in the Jena reasoner[42] is realized as follows.

	RDFS	OWL
Low	Transitive reasoner	Transitive reasoner
Medium	Rule reasoner (RDFS_INF)	Rule reasoner (MICRO_RULE)
High	Rule reasoner (RDFS_INF)	Rule reasoner (OWL_RULE)

We remind to the Jena documentation for more details[43][44].

The Jena engine is wrapped by additional data structures that link it to one or more Cytoscape graphs and holds additional information. Query to the model are handled both through RDQL (RDQL queries and visual queries) and through a direct access to the Jena representation (context-menu).

The implementation details of the rest of the features of RDFScape are not discussed here.

Limitations

There are several limitations to the proposed approach to handle ontologies for high-throughput data functional profiling: availability of ontologies, immaturity of semantic web technologies, performance.

Regarding the availability of ontologies, the number of genes for which description in the form of pathway ontologies exists is about one order of magnitude less than the ones for which only a characterization such as a gene-ontology terms is present.

However, the attention about this type of characterization is increasing. We have already cited Reactome as a recent initiative to provide a public pathway repository. The Pathway Resource List lists[45] more than 200 resources providing pathways's information. There is also a gradual shift toward more formal representation of knowledge complementing traditional scientific literature, as happened for microarray experiment descriptions and MAGE[46]. Therefore, the amount of know-how available in a formalized way is increasing.

[41] An backward rule has the form: $Postcondition(vars) \leftarrow Precondition(vars)$

[42] This is really composed of a transitive reasoner and a rule based reasoner, that can produce different entailments depending on the set of rules it operates on.

[43] http://jena.sourceforge.net/inference/index.html

[44] Note that Medium and High makes no difference when RDFS entailments are considered

[45] http://cbio.mskcc.org/prl

[46] Several journals require that submission dealing with microarray data must be accompanied by a defined set of informations on the experiments and link to a possibly

Another limitations comes from the immaturity in the adoption of semantic web languages. Stable RDF-IDs are not always used and some available ontologies are affected by design errors. However the availability of tools that will bring a relatively wide user base with complex problems to this technologies will easily improve this situation[47].

Regarding performance issues, two serious and related constraints are given by reasoner efficiency and memory.

Memory it's a issue since we have decided to adopt in-memory data structures for speed efficiency. Since inferred relations are added to the known facts, the memory footprint even for relatively small ontologies can be too high. As a rule of thumbs it easy to handle a few pathways even on modest hardware, while performing some queries for all the pathways known in a genome may be out of the reach of even an high-end systems.

This depend also on the way inference rules are defined and by the reasoner used, were different reasoners may have different performance for different types of inference. It is relatively easy to delegate the reasoning to an external server[48], and it's expected that the performance of reasoners, being a standard component of the semantic web architecture, will increase. Thus the availability of tools such as RDFScape could generate requirements for specific reasoners (or rules, or even language features) of importance for the life science community.

Other performance issues are due to the current implementation of RDFScape, and are going to be addressed in future releases.

Another class of limitation in the current version regards visualization issues. Representing each attribute as a property linked to an entity that has no more information than its URI it's a clean model, but it's far from the intuitiveness of direct mapping of attributes on the same visual element that represents the entity. This is also going to be resolved in a future release[49].

Finally, the current approach does not allow properties to have attributes. This can be provided in the semantic web context through the use of reification[50], but this is out of the RDFScape implementation.

---

formalized representation of experimental setting in a public database. MIAME defines the minimum set of information required. This information is formalized in the MAGE object model and makes use of the MAGE-ontology

[47] As for stable RDF-IDs, there is also a proposal for the adoption of Life Science Identifiers (LSIDs) [3]

[48] There are several reasoners already available, like Racer (http://www.sts.tu-harburg.de/~r.f.moeller/racer/), Pellet (http://www.mindswap.org/2004/pellet/), Fact++ (http://owl.man.ac.uk/factplusplus/). They often adopt an interface (DIG) that exports reasoning services remotely.

[49] It is straightforward to associate tranfrom some properties into attributes, but the are more complex problems in visualization regarding the identity of URIs and visual elements. For instance H2O has a unique URI but several visual elements may be associated to it. Thus the problem of an intuitive visualization of the ontology content is not addressed in the current release.

[50] A resource is used to identify a statement (subject, property, object). Therefore attribute of the property can be expressed as property of this resource.

Discussion

The semantic web provides a standard platform to express and handle ontologies. Here we propose a tool that build on this platform to provide enhanced possibilities to the data analysis community.

Semantic web technologies are still in their infancy, hence RDFScape does not to provide a comprehensive usage of current reasoners and support for several rule languages. It is expected that it will help shaping a user community for semantic web technologies in the Life Sciences that will than indicate directions for targeted future improvements[51].

Some problematics like performance are deliberately overlooked here. Since the platform we are using is designed to be employed on web-wide real problems, performance improvements will be delivered by the improvement of the technology, as it becomes mainstream.

The idea of applying inference to derive user-friendly information from ontologies was already presented in [4]. Here we go one step further by making intuitive to the user to define and utilize these transformation. We also make the process of exploration interactive and provide a link to data analysis through Cytoscape.

We note that the ontology modeling effort present right now in ontologies as PioPAX or gene ontology is relatively immature, and reflects a transition from database based representation and ontology based ones. As this situation improves, and if the semantic web technologies will yield their promises, we may rely upon inference to define biologically relevant information (see [14] for an preliminary experience).

Respect to other works such as the Pathway Query Language [10]or Pathway Tools [9](and other resources) , what we propose differentiates in these ways:

- we rely on a standard ontology representation and not on a custom data structure.
- we don't consider only the relations between elements in pathways, but the semantics of the relations among those elements and the meta relations about abstract entities.
- we propose inference to the user as a way to derive information meaningful for specific analysis.
- we provide an interactive and graphical environment, with a plugin architecture for data analysis.
- we are semantically neutral and can accommodate different information. For instance we could integrate a biopax file and a gene ontology file to allow the exploration of what is known about a set of genes, or integrate user defined abstract information[52].

[51] It provides specific features to overcome some temporary problem (like the layered inference architecture) for unification.

[52] Relations as "maybe related to" could be integrated in a pathway ontology, where they could be compared to inferred relations as "related if sharing a controller"

– We don't provide pathway information, we don't provide a production level environment and we don't provide a performant system.

Therefore our system is suited to experiment new directions more than for ordinary usage.

Future work

The overall goal of this project is to improve the way ontologies are used in data analysis by taking into account the structure and the semantics expressed in ontologies. We have shown a platform to integrate ontologies and reasoning in a user interactive environment where a link between data and ontologies can be established.

Objectives of the future work on RDFScape will be the ability to derive ontological relations and to compute evidence for ontology assertions from experimental data. The current version of RDFScape offers these features as an early prototype.

Minor improvements planned are re engineering for performance improvement, adoption of SPARQL and user friendly visualization features.

References

1. The Gene Ontology (GO) project in 2006. *Nucleic Acids Res*, 34(Database issue):D322–6, 2006.
2. Duccio Cavalieri and Carlotta De Filippo. Bioinformatic methods for integrating whole-genome expression results into cellular networks. *Drug Discov Today*, 10(10):727–34, 2005.
3. Tim Clark, Sean Martin, and Ted Liefeld. Globally distributed object identification for biological knowledgebases. *Brief Bioinform*, 5(1):59–70, 2004.
4. Dennis Quan Eric K. Neumann. BIODASH: A SEMANTIC WEB DASHBOARD FOR DRUG DEVELOPMENT. *Pac. Symp. Biocomput.*, 2006.
5. G Joshi-Tope, M Gillespie, I Vastrik, P D'Eustachio, E Schmidt, B de Bono, B Jassal, G R Gopinath, G R Wu, L Matthews, S Lewis, E Birney, and L Stein. Reactome: a knowledgebase of biological pathways. *Nucleic Acids Res*, 33(Database issue):D428–32, 2005.
6. Minoru Kanehisa, Susumu Goto, Masahiro Hattori, Kiyoko F Aoki-Kinoshita, Masumi Itoh, Shuichi Kawashima, Toshiaki Katayama, Michihiro Araki, and Mika Hirakawa. From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Res*, 34(Database issue):D354–7, 2006.
7. Peter D Karp, Christos A Ouzounis, Caroline Moore-Kochlacs, Leon Goldovsky, Pallavi Kaipa, Dag Ahren, Sophia Tsoka, Nikos Darzentas, Victor Kunin, and Nuria Lopez-Bigas. Expansion of the BioCyc collection of pathway/genome databases to 160 genomes. *Nucleic Acids Res*, 33(19):6083–9, 2005.
8. Purvesh Khatri and Sorin Draghici. Ontological analysis of gene expression data: current tools, limitatuions, and open problems. *Bioinformatics*, 21(18):3587–95, 2005.

9. Markus Krummenacker, Suzanne Paley, Lukas Mueller, Thomas Yan, and Peter D Karp. Querying and computing with BioCyc databases. *Bioinformatics*, 21(16):3454–5, 2005.
10. Ulf Leser. A query language for biological networks. *Bioinformatics*, 21 Suppl 2(NIL):ii33–ii39, 2005.
11. Eric Neumann. A life science Semantic Web: are we there yet? *Sci STKE*, 2005(283):pe22, May 2005.
12. Paul Shannon, Andrew Markiel, Owen Ozier, Nitin S Baliga, Jonathan T Wang, Daniel Ramage, Nada Amin, Benno Schwikowski, and Trey Ideker. Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res*, 13(11):2498–504, 2003.
13. Xiaoshu Wang, Robert Gorlitsky, and Jonas S Almeida. From XML to RDF: how semantic web technologies will change the design of 'omic' standards. *Nat Biotechnol*, 23(9):1099–1103, Sep 2005.
14. K. Wolstencroft, Andy Brass, Ian Horrocks, Phillip W. Lord, Ulrike Sattler, Daniele Turi, and Robert Stevens. A Little Semantic Web Goes a Long Way in Biology. In Yolanda Gil, Enrico Motta, V. Richard Benjamins, and Mark A. Musen, editors, *International Semantic Web Conference*, volume 3729 of *Lecture Notes in Computer Science*, pages 786–800. Springer, 2005.