Applied Ontology 6 (2011) 247–261
DOI 10.3233/AO-2011-0097
IOS Press

Flexibility and utility of the cell cycle ontology

Vladimir Mironov [a], Erick Antezana [a], Mikel Egaña [b], Ward Blondé [c], Bernard De Baets [c], Martin Kuiper [a] and Robert Stevens [b,*]

[a] *Department of Biology, Norwegian University of Science and Technology, Trondheim, Norway*
E-mails: {vladimir.mironov, erick.antezana, martin.kuiper}@bio.ntnu.no
[b] *School of Computer Science, The University of Manchester, Manchester, UK*
E-mails: mikel.egana.aranguren@gmail.com, robert.stevens@manchester.ac.uk
[c] *Department of Mathematical Modelling, Statistics and Bioinformatics, Ghent University, Ghent, Belgium*
E-mails: {ward.blonde, bernard.debaets}@ugent.be

Abstract. The Cell Cycle Ontology (CCO) has the aim to provide a 'one stop shop' for scientists interested in the biology of the cell cycle that would like to ask questions from a molecular and/or systems perspective: what are the genes, proteins, and so on involved in the regulation of cell division? How do they interact to produce the effects observed in the regulation of the cell cycle? To answer these questions, the CCO must integrate a large amount of knowledge from diverse sources; the irregularity and incompleteness of this information suggests an ontology can act as the means of this integration. The volatility and continued expansion of biological knowledge means the content and modelling of the CCO will have to be frequently changed and updated. The CCO is generated from the input data automatically once every two months. This makes it easy to change the representation to enable certain queries; incorporate new knowledge; and consistently apply design patterns across the CCO. The automatic process also allows the CCO to be delivered in a variety of representations that suit the needs of various CCO customers and the abilities of existing toolsets. In this paper we present the CCO and its characteristics of *utility* and *flexibility*, that, from our perspective, make it a *beautiful ontology*.

Keywords: Bio-ontology, knowledge management, cell cycle, utility, flexibility, OWL, OBO, standardisation

1. Introduction

The Cell Cycle Ontology (CCO) is an application ontology, developed to facilitate scientific discovery in the area of cell cycle research (Antezana et al., 2009). The CCO is used to integrate knowledge across a broad, complex and volatile domain where the consensus of terminology and conceptualisation is at best patchy. The CCO exploits activities within bio-ontology development to deliver a flexible, up-to-date, integrated knowledge resource for the study of the cell cycle.

The eukaryotic cell cycle or cell division cycle, is the series of events that happen between two consecutive cell divisions, leading to cell multiplication. The molecular events that control the cell cycle are ordered and directional, i.e., each process occurs in a sequential fashion and it is impossible to reverse the cycle. The typical cycle is composed of four consecutive phases. The first phase to follow cell di-

*Corresponding author: Robert Stevens, School of Computer Science, The University of Manchester, Oxford Road, Manchester, M13 9PL UK. E-mail: robert.stevens@manchester.ac.uk.

vision is called G1 (Gap1), during which the cell grows and prepares to duplicate its genetic material – the DNA, with all the individual genes of an organism. In the next phase, called S (Synthesis of DNA), duplication of the genetic material takes place so that the cell then contains a double complement of genes. In the subsequent phase G2 (Gap2) the cell checks the accuracy of the duplication process and further grows to make sure that the two daughter cells will have sufficient mass after cell division. In the most conspicuous phase of the cycle (M, short for Mitosis) the genetic material is distributed in such a way that the daughter cells will receive equal sets of genes. After the M phase, cellular division ensues and a new cycle commences.

The cell cycle control network is complex and includes hundreds of proteins (de Lichtenberg et al., 2007; Jensen et al., 2006; Van Leene et al., 2010). Each protein can have many relationships with other biological entities (see Fig. 1). Biologists working in this area need access to this knowledge in order to understand their domain and drive further enquiries.

Although the basic principles of cell cycle regulation are now well documented (Alberts et al., 2002), biologists are far from a complete understanding of all the intricacies of the underlying control system, and this understanding frequently changes. A deeper knowledge of the cell cycle is essential to gain insight into the growth and development of eukaryotic organisms. Increased knowledge may lead to the

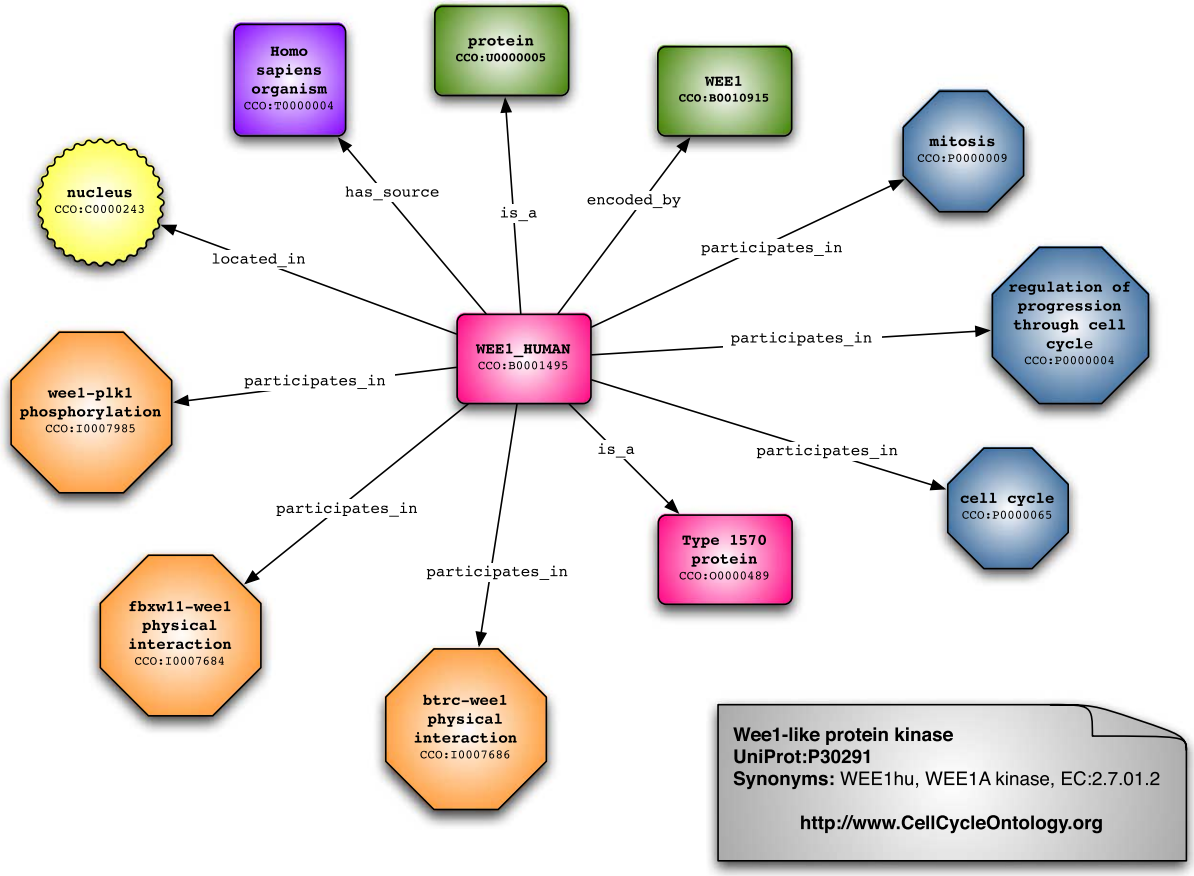

Fig. 1. Part of the local neighbourhood of human protein WEE1. The figure illustrates the types of entities (colour coded) associated with one arbitrarily chosen human protein and the corresponding relationship it has with other biological entities. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/AO-2011-0097.)

development of approaches to combat numerous diseases in which cell cycle aberrations are involved, such as cancer.

Ontologies are now commonly used within the molecular biology sub-discipline of the life sciences (Stevens & Lord, 2008). As briefly outlined above, even a small aspect of molecular biology, such as the system of cell cycle control, is complex, with many genes, proteins and small molecules interacting to give the phenomena we observe. When cross-species features are taken into account, the scene becomes even more diverse.

The inherent dynamics of the biological sciences makes the nature of the knowledge even more complex – continuously, experiments are undertaken that produce new facts that may or may not contradict previous facts. Such experiments are undertaken by many individual scientific groups, often working autonomously. Thus published facts and data are often described with different terminologies that reflect differing underlying conceptualisations of the domain (Attwood et al., 2009). As biologists moved to broader views of their science, comparing biological species and wanting to integrate facts from many areas of biology, the differences in how the individual sub-disciplines conceptualised their understanding of the domain cause myriad problems.

Ontologies are seen as a means by which knowledge could be captured according to a common conceptualisation of the domain (Antezana et al., 2009). There has been a large effort over the past twenty years to develop ontologies that can provide such a common view across the domain. This is a standard ontology based integration paradigm; biological entities are identified as belonging to a particular class of entity; these classes have unique identifiers. Thus, entities with the same identifier in different resources are assumed to be the same entity. Naturally, relationships held between those entities in the ontology may also be transferred to the entity described in some bioinformatics resource (where the problems of identity are rife (Good & Wilkinson, 2006; Zhao et al., 2006).

Even though there is a broad understanding of the cell cycle within a number of model organisms, the frequent generation of new facts means that the knowledge of the cell cycle domain is volatile. Databases are updated as frequently as every month; the ontologies describing these entities likewise change as often. Any system, therefore, that brings together a variety of knowledge about a domain must accommodate the volatility of the domain's knowledge. The CCO fits directly into this standard ontology-based integration paradigm. Its aim is to provide an integration of knowledge about the cell cycle of eukaryotes. From this general aim, a range of requirements for the CCO emerge:

(1) To integrate cell cycle knowledge across several model organisms.
(2) To present a common conceptual view of this knowledge that permits sophisticated querying across cell cycle knowledge.
(3) To allow querying in a variety of forms by a variety of users.
(4) To accommodate frequent changes in data.
(5) To overcome problems in identity of biological entities.

It is the way that the CCO meets these requirements that, from our perspective, makes it an example of a *beautiful ontology*. The combination of *flexibility* and *utility*, together with the process by which these are achieved, is the main argument for the CCO's 'beauty'.

2. The form and content of the Cell Cycle Ontology

To meet the requirements listed above, the ontology needs to cover the following topics:

(1) The cell cycle proteins themselves.

(2) The genes coding for cell cycle proteins.
(3) The molecular function of cell cycle proteins.
(4) Cell cycle proteins' cellular localisation.
(5) Cellular processes associated with cell cycle proteins.
(6) Post-translational modifications of cell cycle proteins.
(7) Protein-protein interactions among cell cycle proteins.
(8) Phylogenetic information for the supported organisms through orthology 'evolutionary relatedness' relationships among cell cycle proteins.[1]

The CCO holds knowledge on these topics for four organisms that are particularly important for cell cycle research: *H. sapiens* (human), *S. cerevisiae* (budding yeast), *S. pombe* (fission yeast) and *A. thaliana* (a model plant). For each of these organisms there is a separate Web Ontology Language (OWL) (Horrocks et al., 2003) ontology. In addition, there is also an integrated OWL ontology covering all four organisms. The complete CCO has the proteins that are central to the cell cycle interlinked by orthology relationships. It is this larger ontology that is described in this section.

The CCO's coverage provides the sort of information shown in Fig. 1. The content of the CCO was determined by interactions with cell biologists, from whom competency questions were gathered. The coverage could extend outside this list, for instance, to the regulatory elements of the genome. This was not a priority for our users, but the CCO does include 'stubs' for such extensions. To achieve this coverage, the CCO currently integrates the following resources:

(1) Genes, proteins and their post-translational modifications from the UniProt database (Consortium, 2008).
(2) The cell cycle, cell division, cell proliferation, DNA replication branches of the Gene Ontology's (Ashburner et al., 2000) Biological Process branch.
(3) The complete Molecular Function branch from the Gene Ontology.
(4) The complete Cellular Component branch from the Gene Ontology.
(5) The Relation Ontology provided by the Open Biomedical Ontologies consortium (OBO) (Smith et al., 2005, 2007).
(6) The Gene Ontology Annotations (Camon et al., 2004) of UniProt proteins for the cell cycle (these are the attachments of GO Cellular Component concepts to proteins to describe their location); GO Biological Process and Molecular Function concepts to describe their functionality.
(7) Biological species relationships as described by the NCBI taxonomy (Sayers et al., 2009).
(8) Protein-protein interactions involving cell cycle proteins from the IntACT database (Kerrien et al., 2007).

The core concepts within the CCO are therefore: *Protein*, *Modified protein*, *Gene*, *Molecular interaction*, *Molecular function*, *Biological process*, *Cellular component* and *Organism*. All these entities in the CCO (proteins – including their modified forms, genes, interactions, etc.) are modelled as classes since they gather shared commonalities that are present in all the individuals (instances) they represent. Table 1 shows some basic metrics for the CCO that give an overview of the CCO.

We have not chosen to be overly ontologically formal in our description of classes: For example, we do not model dispositions and propensities of proteins to participate in processes and to be found in

[1](Sequence) Orthology is a kind of homology, where two sequences arise from a common ancestor by inheritance, rather than through gene duplication events. Orthology is normally evidenced by a high degree of sequence similarity. Orthologs often share a common structure and function.

Table 1
Metrics giving numbers of classes; breadth and depth of the CCO's hierarchy and the cohesion of classes

Metric	Asserted CCO
No. of classes	89,532
No. of leaf classes	85,235
Max depth	33
Mean depth	15.35
Mean number of subclasses per class	4.61
Max number of subclasses	24,021
No. of properties	52
Property with maximum usage	has_source used 65,110 times
Mean usage per property	6673.69
Mean property usage per class	3.87

a location; instead we make statements such as 'each and every protein x participates in process p'. Modelling with dispositions and propensities may afford a more 'true' representation, but at the cost of increased complexity in comprehension and querying.

For the purpose of CCO engineering an Upper Level Ontology was developed to hold all the integrated resources. An Upper Level Ontology (ULO) is an ontology that structures general types of concepts (such as a process) in generic as well as specific domains to provide an integration scaffold for including other ontologies (Guarino, 1998). A ULO connects a relatively small number of concepts by meaningful and strictly defined relationships. We view the ULO as a framework for deploying the properties that link entities described within the ontology and it is used in this way in the CCO.

To accommodate interlinkage of concepts and relationships for cell cycle knowledge, we developed a ULO for the CCO (see Fig. 2). As many of the ontologies used in the CCO come from OBO that aims to use the Basic Formal Ontology (BFO) (Grenon et al., 2004), the CCO ULO is based on the BFO to aid interoperability of the CCO with ontologies from OBO. Our ULO has been customised for the CCO by inclusion of a few high-level concepts, such as '*cell cycle protein*'.

The ULO acts as a scaffold for a set of *de facto* standard relationships that are used in the ontologies being integrated in the CCO. Most of the ontologies re-used within the CCO originate from the OBO consortium, where use of these relationships is recommended. Figure 3 shows one protein class from the CCO, giving its relationships to the individuals from other classes within the CCO via these relationships.

The CCO uses an ontology design pattern (ODP) (Aranguren et al., 2008; Clark et al., 2003; Gangemi, 2005) to represent the sequence of cell cycle phases. The Sequence ODP[2] was applied (Aranguren et al., 2008), in order to add sequentiality to the phases of the cell cycle described in GO (i.e., certain events are sequentially preceded by other events). The sequence ODP allows the CCO to capture a representation of the cell cycle stages in a way that permits useful queries. Each stage of the cycle is represented as a class. Each stage is part of a kind of cell cycle, so *M phase* is part of the *Mitotic cell cycle*. Each stage can be *preceded_ by* another stage or *precedes* another stage; each of these properties is transitive. These two properties each have an intransitive sub-property *immediately_preceded_by* and *immediately_precedes* respectively. Each 'stage' class in the cell cycle has restrictions to the prior and next stages using these two property pairs, except the penultimate stages in the sequence that only have the 'immediately' sub-

[2]http://www.gong.manchester.ac.uk/odp/html/Sequence.html.

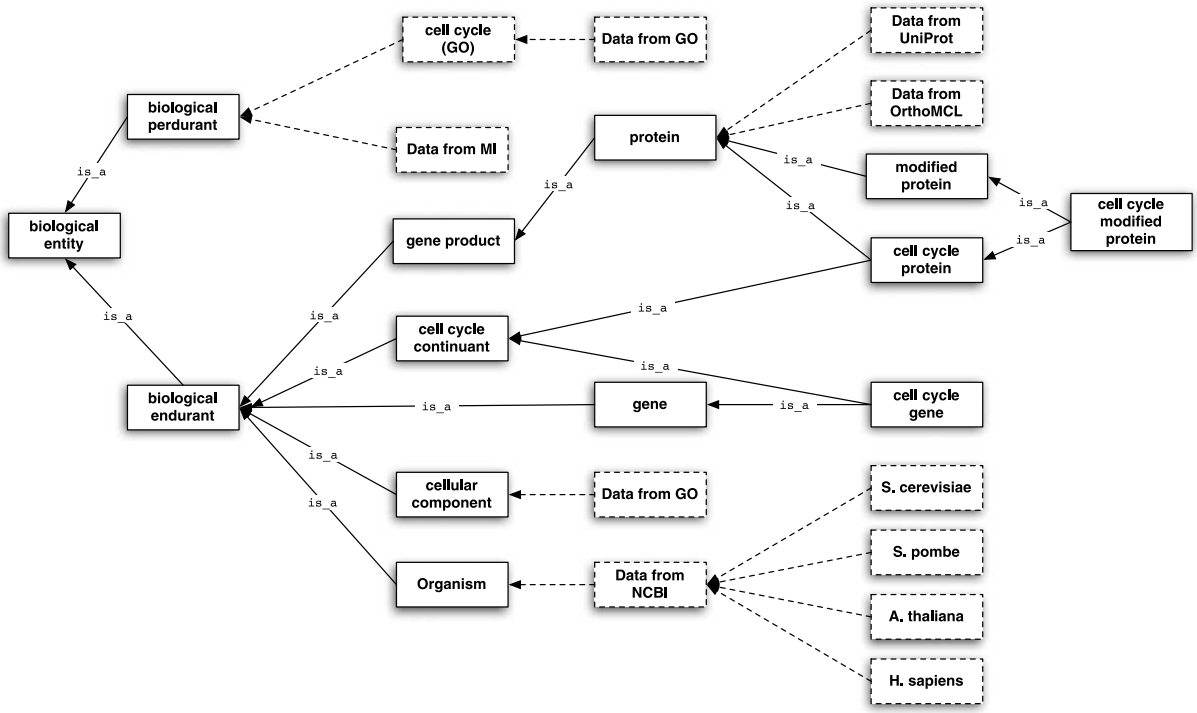

Fig. 2. Upper Level Ontology (ULO) for the CCO. It provides a hierarchical scaffold including generic concepts (e.g., *cell cycle gene*) which serve as "hooks" upon which to hang the integrated resources. The dashed rectangles represent the type of data residing below the parental concepts: the node labelled "Data from GO" shows where the concepts from GO's cellular component ontology reside under the concept *cellular component* (e.g., *nucleus*), the node "Data from UniProt" under the concept *protein* shows the placeholder where protein data from UniProt resides (e.g., *p53*) and so forth.

```
Class: WEE1_HUMAN
SubClassOf: 'protein'
and has_source some 'Homo sapiens organism'
and encoded_by some 'WEE1'
and participates_in some 'cell cycle'
and participates_in some 'mitosis'
and participates_in some 'regulation of progression through cell cycle'
and located_in some 'nucleus'
and participates_in some 'WEE1--PIK1 phosphorylation'
and participates_in some 'FBXW11--WEE1 physical interaction'
and participates_in some 'BTRC--WEE1 physical interaction'
```

Fig. 3. The protein class WEE1 representing the individual WEE1 proteins. Relationships to individuals of other classes within the CCO are shown (see also Fig. 1).

properties as restrictions. This pattern of intransitive sub-property and transitive super-property means that, for instance, asking a query for proteins that participate in the stages preceding M-phase will find

proteins in the G2, S and G1 phases. Using the intransitive sub-property allows queries to be made for only the prior and next steps.

This simple ODP allows queries such as:

```
participates_in some
(precedes some 'M phase of mitotic cell cycle' or
part_of some (precedes some 'M phase of mitotic cell cycle'))
```

This query means 'give me any protein that participates in any phase before M-phase in a mitotic cell cycle'. The answer is:

- BRE1A_ARATH (participates in 'regulation of G2/M transition of mitotic cell cycle').
- Q9SGD5_ARATH (participates in 'positive regulation of S phase of mitotic cell cycle').

Using the non-transitive relationship, we can ask the query:

```
participates_in some
(immediately_preceded_by some
'G2 phase of mitotic cell cycle' or part_of some
(immediately_preceded_by some 'G2 phase of mitotic cell cycle'))
```

This query means 'give me any protein that participates in a phase immediately after G2, and only that phase'.

With the 'protein hubs' and the sequence ODP for the stages of the cell cycle, the CCO has localisation and participation of proteins in parts of the processes of the cell cycle. As described, proteins are the implementation of most of the cell's machinery. The cell cycle stages are the temporal organisation of these cell's deployment. Thus, these hubs and stages allow much of the cell cycle specific querying of the CCO.

3. Main characteristics of the Cell Cycle Ontology

The characteristics of the CCO naturally reflect the requirements set out in Section 1. In summary, the characteristics are as follows:

Criterion I *Answers appropriate queries*. Biologists can visit one knowledge resource over which a broad range of queries can be asked.

Criterion II *Knowledge integration*. By integrating appropriate information sources for cell cycle biologists, using data from existing ontologies, databases and tools, cell cycle knowledge is integrated. The following features of this integrated knowledge enable biologists to ask queries that are otherwise difficult:

(a) The CCO has a unified view over four model organisms.
(b) The CCO has a protein centric view exposing important proteins as information 'hubs'.
(c) The use of ontology design patterns to capture the right information for queries.

Criterion III *Flexibility of representation*. Exposure of the CCO as either OWL, RDF or XML, etc., means it can be viewed and used in many ways.

Criterion IV *Flexibility as to modelling*. Scripting of ontology generation means changes are easier to impose. The CCO is not hand-built, but generated by a process captured in scripts. This means that changes to modelling are readily made and can be rapidly and consistently applied.

Criterion V *Responsive to change*. Automatic generation of the CCO means changes in knowledge are more straightforwardly accommodated. Again, this makes it easier to offer biologists appropriate query facilities.

One of the major developments in biology in the past two decades has been the determination of the complete genome code (the DNA sequence) of many organisms. As soon as this happened, biologists wished to compare across species. This 'turning outwards' of the different 'model' organism communities afforded both opportunities and challenges. Organisms that appear vastly different may have many similarities at the molecular level and information from one species can inform biologists about potential mechanisms in another. Beyond this, of course, the variations in mechanism are also biologically interesting. The challenge is how to align the knowledge about the biological entities being compared; hence the use of ontologies (Smith et al., 2007) to achieve an integration of the discipline's knowledge.

The inclusion of four model organisms makes the integration broad. We use ODP (Aranguren et al., 2008; Clark et al., 2003; Gangemi, 2005) to expose domain knowledge appropriately and to enable useful inferences. The ODP provides solutions to common modelling situations in a way that makes the knowledge explicit and amenable to automated reasoning. In addition, a common model has obvious virtues for integration.

Being responsive to changing domain knowledge and having to consistently apply this across a large ontology implies that the manual authoring and maintenance of the CCO is not sustainable. We apply ODPs to capture some aspects of the conceptualisation of the cell cycle; the manual consistent application of these and other patterns across the CCO is problematic in the same way as the manual authoring of any large ontology. Also, any change can be applied straightforwardly in an automated process. Consequently, the enriched version of the CCO is generated programmatically via an OWL scripting language called the Ontology Preprocessor Language (OPPL) (Breis et al., 2010; Egaña et al., 2008).

From our perspective, the CCO can be seen as a beautiful ontology for the reason that it elegantly combines two seemingly contradicting requirements; being both useful and flexible. These main characteristics are intimately linked. As an application ontology integrating a broad variety of volatile resources, the CCO is large, complex and subject to frequent change. Thus the utility can only be delivered by a flexible process that avoids the hand-crafting of axioms that would make it *inflexible* to change and update.

3.1. The Cell Cycle Ontology as a useful ontology

The CCO is knowledge-rich, offering many advantages for cell cycle researchers. Through its integration of knowledge from a broad range of resources, the CCO offers a resource for biologists wishing to explore the current understanding of cell cycle knowledge.

The CCO is protein centric, meaning that proteins are used as 'hubs' to integrate and connect knowledge (Fig. 1). A gene can encode multiple proteins with distinct functions, and as the proteins actually perform the functions in the cell, they are the focus of the ontology. This organisation affords the types of questions that biologists need to ask, as gathered from our users.

The CCO supports complex and biologically relevant queries not possible with the disparate original data sources. Examples of such queries gathered from our users include:

(1) Return the orthologs (proteins with the same biological function) of the protein X and include all the biological processes and molecular functions in which these orthologs participate.
(2) Find all the proteins involved in the cell cycle and located in the mitochondrion.
(3) Identify all cell cycle proteins implicated in a particular disease.
(4) Find all the proteins involved in the regulation of anaphase in *H. sapiens* with orthologs in *S. cerevisiae*.

Of course, many of these queries can be combined to give more complex queries. Many more examples of such queries can be found on the CCO web site,[3] that provides a library of biologically relevant queries.

Ontologies act as a suitable means for this kind of integration in the life sciences as they are suited to capturing the incomplete and irregular information (Horrocks, 2008) that is typical of biology's data (Attwood et al., 2009). The open world assumption of OWL's representation and the ease of 'under-specifying' information is a great asset in biology where much is either unknown or only known at some high-level of understanding.

The CCO has the utility typical of many ontologies; as the knowledge about a class of entities is made explicit, it is also queryable.

Further utility is added to the CCO by offering it in a range of representations, each with its particular advantages for users. First of all, the CCO is available in two ontology languages – The Open Biomedical Ontologies Format (OBOF)[4] and OWL. We have presented the OWL version of the CCO in order to describe its content and main features. The OWL version of the CCO is the most expressive and exceeds the other versions in information content as it has new axioms and Ontology Design Patterns that enrich its representation.

On the other hand, OBOF is the *de facto* standard for knowledge representation in the bio-ontology community (Antezana et al., 2008). Many tools have been built to use OBOF (e.g., OBO-Edit (Day-Richter et al., 2007) and OBO Explorer (Aitken et al., 2008)), and they are widely used by biologists. Much of the biological knowledge already captured in ontologies is represented in OBOF. Having the CCO in OBOF makes it available to a broader community of users than does the OWL representation on its own.

OBOF lacks some of the expressivity of OWL and while it has an import mechanism, it lacks the URI based integration mechanism of the Semantic Web languages RDF and OWL. Thus OBOF has less use as the basic means of integration, but is good as a delivery mechanism within the target community for the CCO. OBOF queries are limited to simple exploration of the ontology structure. OWL's query capabilities are greater, though it can suffer with issues of scalability when using automated reasoners. The performance limitations encountered with OWL mean that it is prohibitive for specific tools, such as Protégé, when launching complex queries over a large ontology such as the CCO. With these differing and sometimes contradictory capabilities, we develop the CCO in OWL, but deliver the 'complete' in other representations such as OBOF for users that wish to use those particular representations.

In contrast to OBOF, OWL allows integration of other ontologies within the CCO by using the URI-based importing mechanism, meaning that extant encoded knowledge can be effectively added and exploited. Additionally, the OWLDoc server[5] allows on-line queries over the CCO.[6]

[3] http://www.semantic-systems-biology.org/cco/queryingcco/sparql.
[4] http://www.geneontology.org/GO.format.obo-1_4.shtml.
[5] http://code.google.com/p/ontology-browser/.
[6] http://www.semantic-systems-biology.org/cco/queryingcco/owl-dl.

More efficient querying than that seen in OWL can be achieved with RDF. We have loaded the RDF version of the CCO into Open Virtuoso[7] triple store to enable complex queries via SPARQL. A SPARQL query form[8] and a SPARQL query service[9] are available to exploit the CCO. The CCO RDF allows a move toward exploiting Semantic Web technologies (Good & Wilkinson, 2006), as it offers the possibility to integrate knowledge from external resources (Ruttenberg et al., 2007). Tools like RDFScape (Splendiani, 2008) (a plug-in for Cytoscape, Shannon et al., 2003) can also be used to explore this representation of the CCO.

GML (Graph Modelling Language),[10] VisANT/XML (Hu et al., 2007) and DOT[11] allow visual exploration of the CCO by tools such as Cytoscape, VisANT and GraphViz.[12] VisANT, in particular, provides a user friendly interface to examine the CCO network of concepts and relationships.

In summary, the above representations provide three ways of interaction with the CCO: a basic exploration of the structure (OBOF); expressive queries including the possibility to combine the CCO with other resources (RDF and OWL); and visual exploration (GML, VisANT/XML and DOT). The CCO itself has the utility of drawing domain knowledge together. This utility is, however, not apparent if it cannot be effectively delivered to users. No one representation alone currently affords all the utility necessary; therefore the CCO is delivered in a variety of representations.

3.2. Flexibility of the Cell Cycle Ontology through its means of production

Here we describe the process by which the CCO is built and maintained that give it flexibility. The CCO is built from scratch by an automatic pipeline every two months, and only the identifiers for the entities in the ontology are kept for consistency between releases. The CCO is sufficiently large, complex and rapidly changing that a manual authoring process is not feasible; only automation can match the requirements set out in Section 1. The automatic pipeline encompasses the typical life cycle of an integrated system: set-up, data integration, and system maintenance (Fig. 4). All the integrated information is cross-referenced to the original sources to ensure data provenance. The integration pipeline relies on the ability to programmatically manipulate ontologies, concepts and relations, a functionality offered by ONTO-PERL (Antezana et al., 2008).

The generation of the CCO capitalises on the common naming scheme provided by the development of the OBO library of ontologies and their use in describing biological data. The UniProt database of protein sequences describes proteins, their genes, biological species and many other features. Each entry in UniProt has its attributes described according to the molecular function, biological processes and cellular components of the Gene Ontology. Each protein has an identifier; each Gene Ontology concept has an identifier; and so on. The same works for UniProt proteins and the interactions described in IntACT, as UniProt identifiers are also used in IntACT. Thus it is relatively straightforward to produce an ontology programmatically that enables an integration exploiting the naming schemes provided by UniProt and GO.

The pipeline produces the four species-specific ontologies plus a composite ontology that integrates the species-specific ontologies via orthology relationships. The major steps in the CCO workflow are:

[7] http://virtuoso.openlinksw.com/.

[8] http://www.semantic-systems-biology.org/cco/queryingcco/sparql.

[9] http://genetools.ntnu.no:8892/sparql.

[10] http://www.infosun.fim.uni-passau.de/Graphlet/GML/.

[11] http://www.graphviz.org/doc/info/lang.html.

[12] http://www.graphviz.org/.

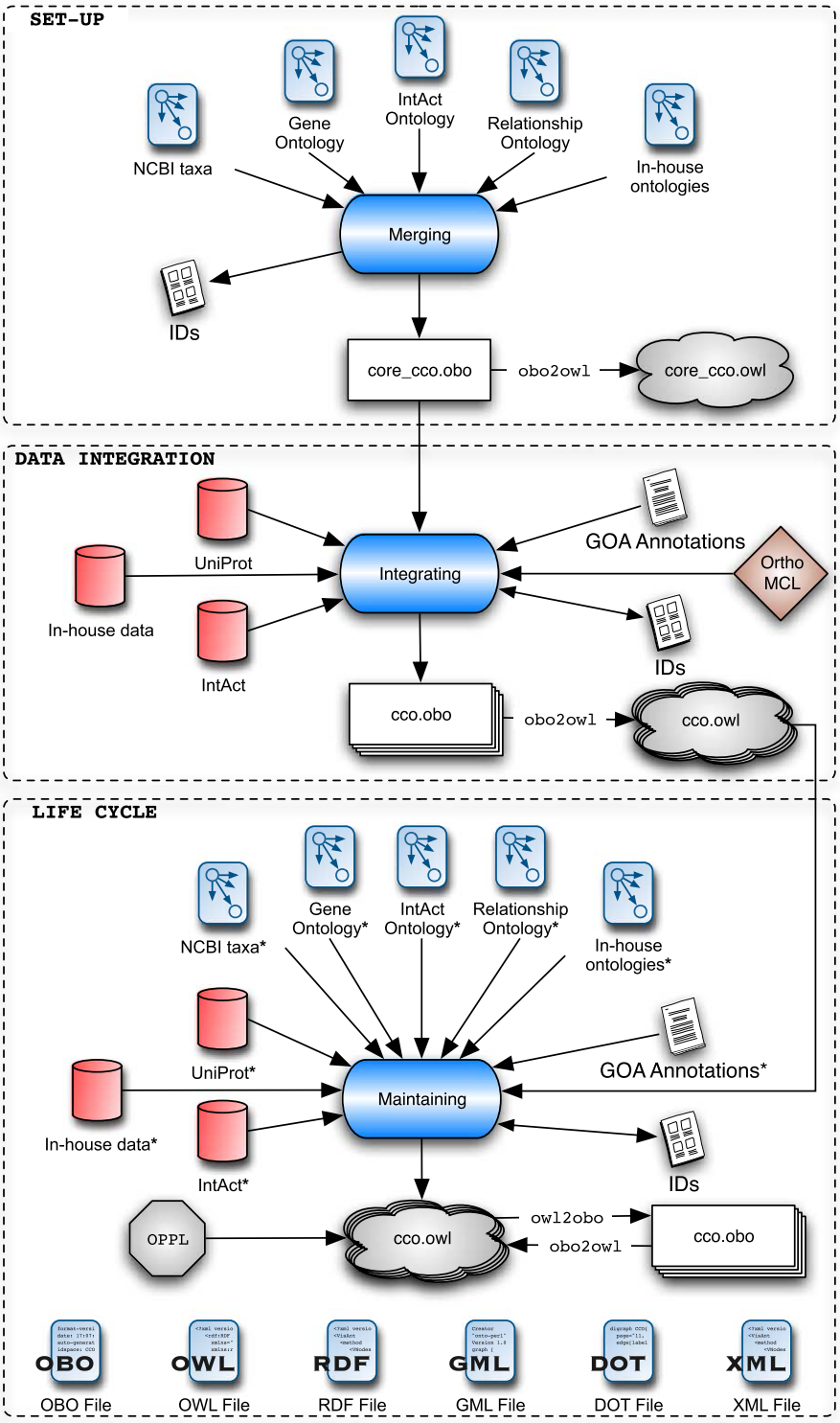

Fig. 4. The fully automated pipeline that assembles, integrates and uploads the CCO. (Colors are visible in the online version of the article; http://dx.doi.org/10.3233/AO-2011-0097.)

(1) Initially, a 'pre-cell cycle ontology' is built that constitutes the backbone for the CCO's sub-ontologies (the four species-specific ontologies and the integrated ontology). This involves selecting portions from various OBO source ontologies.
(2) Next, the OBO Relation Ontology is fully incorporated and the 'interaction type' branch from the Molecular Interaction ontology from OBO is integrated.
(3) Then, a specific taxonomy is built on the basis of the NCBI taxonomy for *H. sapiens*, *S. cerevisiae*, *S. pombe* and *A. thaliana*.
(4) Next, the protein 'hubs' that connect their relevant annotation data (such as the molecular functions in which they participate) are generated. We define the proteins described with cell cycle concepts in Gene Ontology Annotation (GOA) files (Camon et al., 2004) as the 'core cell cycle proteins'. These proteins are added to the CCO as the children of the concept *core cell cycle protein* (CCO:B0000000) and used as the starting point (seed) for the data integration process.
(5) The 'core cell cycle' protein section of the CCO is then expanded to include the proteins known to interact with the core cell cycle proteins, as documented in IntACT (Kerrien et al., 2007). These new protein classes are included as subclasses of the class *cell cycle protein*. Then, protein information (such as synonym names, encoding genes and cross references) is fetched from the UniProt knowledge base. In addition, post-translational modification data, when available in UniProt, are also added by creating new concepts defined by their specific modification.
(6) Next, the OrthoMCL clustering utility (Chen et al., 2006) is used to generate clusters of putative orthologs for the four species included in the CCO. This tool is used to infer evolutionary relationships between classes of proteins. The proteins from the clusters containing at least one core cell cycle protein are added to the CCO as subclasses of the class *cell cycle protein*.
(7) All the imported entries from the sources are cross-referenced so that the data can be traced back to its source.
(8) The four organism-specific ontologies and the composite cell cycle ontology are checked and made available producing the official release of the system.
(9) Now that the CCO is available with all the knowledge in place, semantic enrichment can occur. We use OPPL to further transform the CCO with application of ontology design patterns.
(10) Finally, validation and verification processes are carried out while building the CCO to ensure its soundness. The ontologies generated in the previous phase are manually and automatically checked by ontology editors, validators and reasoners. In addition, the pipeline log execution files are inspected in detail. These files are sufficiently detailed to point out any possible problem. This has allowed us to assemble a fully automated pipeline that uploads the ontologies and their exports in the different formats to the CCO website and to all related and supporting repositories.

OPPL[13] is a scripting language for the automatic manipulation of OWL ontologies (Egaña et al., 2008; Iannone et al., 2009). We use OPPL to provide semantic enrichment of the CCO. OPPL is a declarative language: the ontologist writes in a script the actions to be performed and the script is executed against the ontology by the OPPL interpreter that performs the changes. One of the main uses of OPPL is as a means of applying Ontology Design Patterns (ODPs) in OWL ontologies. An ODP encapsulates a fragment of the expressivity of OWL in a thoroughly documented concrete set of axioms. Therefore, OPPL offers a language for storing each ODP in an executable script (Egaña et al., 2008; Iannone et al., 2009). The sequence ODP described in Section 2 is applied to the '*cell cycle*' subset of processes from GO's biological process ontology (Aranguren et al., 2008).

[13] http://oppl.sourceforge.net/.

Using ODPs in the modelling of ontologies offers several advantages. The main one is that the modelling process becomes an explicit assembly of well-known modules (ODPs), with known features and issues; the modelling process becomes faster, more traceable and more robust. Also, the modelling is more flexible, as complex modelling can be initiated or cancelled by simply executing a script (doing the same process manually will usually involve more steps and yield less consistency). The application of ODPs in the CCO is in its early stages, and it is expected that more ODPs will be developed and other ODPs from other sources will be applied, as the CCO evolves further.

4. Discussion

We have described the two main characteristics of the CCO, and how we believe the combination of these two aspects make it, in its own context, a beautiful ontology: The process by which the CCO is produced and the utility of the artefact itself in answering biologically useful questions. The CCO can be thought to meet broader 'beauty' criteria through these two criteria as it means the CCO has good coverage of the domain, it conforms to some 'style' guidelines to give it structure, meets the requirements given to us from our users and has utility through its ability to answer pertinent questions.

There are many aspects to 'ontological beauty' and the CCO only meets some possible characteristics of a beautiful ontology. For example, the CCO is not formally ontologically beautiful; it does not strictly follow any philosophical principles to make 'proper' ontological distinctions. The delivery of functionality to users may be thought to contradict the perceived complexity of formal ontological principles. Take, for example, the modelling of dispositions of functions that are realised in processes;[14] this may well be more ontologically pleasing, but does not add any utility in query answering. In the various aspects of ontological beauty there will often be some tension between formal beauty and pragmatic utility – beauty is in the eye of the beholder.

Another aspect to ontological beauty comes from the use of languages such as OWL. It is possible to maximise the use of OWL's expressivity in the modelling of a domain to have as precise a model as possible, as well as to maximise inference from automated reasoners. Again, the utility criterion contradicts this stance; the over-riding need is to allow query asking and answering and enough expressivity is used to achieve those questions the CCO needs to answer. The criterion of flexibility and CCO's means of production does mean that as tools improve and needs arise, then the CCO can be re-modelled to accommodate such changes with relative ease.

The two main beauty characteristics of the CCO work together to make the CCO beautiful. The requirements for flexibility and responsiveness to change, together with the size of the CCO, make an automated production of the CCO necessary. Such automation is, however, not something that should be restricted to large ontologies. Hand-crafting of ontologies is not sustainable in a service provision context. Ontologists' thought should go into the development of the 'plan' for the ontology and machines should collect knowledge and generate the knowledge according to that plan. Ontologies are not one-off creations; their need for update, changes in representation, etc., all mean that automation of ontology management is necessary.

One conception of ontologies is as knowledge resources for information systems; they are not, in this context, an end in themselves. The CCO very much takes this approach. Utility is the driving motivation behind its creation. The CCO capitalises upon work undertaken in the wider biological community to

[14] http://precedings.nature.com/documents/1941/version/1.

produce ontologies both describing the domain and describing the data according to those ontologies. This is fundamentally what enables the utility of the CCO; the main work here is the process by which the CCO has been produced, offering both utility and flexibility.

Acknowledgements

We acknowledge Waclaw Kusnierczyk, Bijan Parsia, Alan Ruttenberg and Barry Smith for interesting and motivating discussions; Nirmala Seethappan, Kent Overholdt and Bjorn Lindi for their help in setting-up the CCO at NTNU. This work was funded by the European Union's Sixth Research Framework Programme (LSHG-CT-2004-512143) and the European Science Foundation. ME was funded by the Engineering and Physical Sciences Research Council (EPSRC) of the UK and the University of Manchester. VM was funded by FUGE Mid-Norway.

References

Aitken, S., Chen, Y. & Bard, J. (2008). OBO explorer: an editor for open biomedical ontologies in owl. *Bioinformatics*, *24*(3), 443–444.

Alberts, B., Johnson, A., Lewis, J., Raff, M., Roberts, K. & Walter, P. (2002). In *Molecular Biology of the Cell*, 4th edn (pp. 983–1026). New York: Garland Science.

Antezana, E., Egaña, M., Blonde, W., Illarramendi, A., Bilbao, I., De Baets, B., Stevens, R., Mironov, V. & Kuiper, M. (2009a). The cell cycle ontology: An application ontology for the representation and integrated analysis of the cell cycle process. *Genome Biol.*, *10*(5), R58.

Antezana, E., Egaña, M., De Baets, B., Kuiper, M. & Mironov, V. (2008). ONTO-PERL: an API for supporting the development and analysis of bio-ontologies. *Bioinformatics*, *24*, 885–887.

Antezana, E., Kuiper, M. & Mironov, V. (2009b). Biological knowledge management: the emerging role of the semantic web technologies. *Brief. Bioinform.*, *10*(4), 392–407.

Aranguren, M.E., Antezana, E., Kuiper, M. & Stevens, R. (2008). Ontology design patterns for bio-ontologies: a case study on the cell cycle ontology. *BMC Bioinformatics*, *9*(Suppl. 5), S1.

Ashburner, M., Ball, C.A., Blake, J.A., Botstein, D., Butler, H., Cherry, J.M., Davis, A.P., Dolinski, K., Dwight, S.S., Eppig, J.T., Harris, M.A., Hill, D.P., Issel-Tarver, L., Kasarskis, A., Lewis, S., Matese, J.C., Richardson, J.E., Ringwald, M., Rubin, G.M. & Sherlock, G. (2000). Gene ontology: tool for the unification of biology. The gene ontology consortium. *Nat. Genet.*, *25*(1), 25–29.

Attwood, T.K., Kell, D.B., McDermott, P., Marsh, J., Pettifer, S.R. & Thorne, D. (2009). Calling international rescue: knowledge lost in literature and data landslide! *Biochem. J.*, *424*(3), 317–333.

Camon, E., Magrane, M., Barrell, D., Lee, V., Dimmer, E., Maslen, J., Binns, D., Harte, N., Lopez, R. & Apweiler, R. (2004). The gene ontology annotation (GOA) database: sharing knowledge in uniprot with gene ontology. *Nucleic Acids Res.*, *32*, D262–D266.

Chen, F., Mackey, A.J., Stoeckert, C.J. & Roos, D.S. (2006). Orthomcl-db: querying a comprehensive multi-species collection of ortholog groups. *Nucleic Acids Res.*, *34*(Database issue), D363–D368.

Clark, P., Thompson, J. & Porter, B. (2003). Knowledge patterns. In S. Staab and R. Studer (Eds.), *Handbook on Ontologies*, International Handbooks on Information Systems (pp. 121–134). Berlin: Springer.

Day-Richter, J., Harris, M., Haendel, M. & Lewis, S. (2007). OBO-edit – an ontology editor for biologists. *Bioinformatics*, *23*(16), 2198–2200.

de Lichtenberg, U., Jensen, T.S., Brunak, S., Bork, P. & Jensen, L.J. (2007). Evolution of cell cycle control: same molecular machines, different regulation. *Cell Cycle*, *6*(15), 1819–1825.

Egaña, M., Antezana, E. & Stevens, R. (2008a). Transforming the axiomisation of ontologies: the ontology pre-processor language. In *OWLed DC*, Washington, DC, USA.

Egaña, M., Rector, A., Stevens, R. & Antezana, E. (2008b). Applying ontology design patterns in bio-ontologies. In A. Gangemi and J. Euzenat (Eds.), *EKAW*, LNCS (Vol. 5268, pp. 7–16). Berlin: Springer.

Fernandez-Breis, J.T., Iannone, L., Palmisano, I., Rector, A. & Stevens, R. (2010). Enriching the gene ontology via the dissection of labels using the ontology pre-processor language. In *EKAW – Knowledge Engineering and Knowledge Management by the Masses* (pp. 59–73). Berlin: Springer.

Gangemi, A. (2005). Ontology design patterns for semantic web content. In *ISWC*, LNCS (Vol. 1729, pp. 262–276). Berlin: Springer.

Good, B.M. & Wilkinson, M.D. (2006). The life sciences semantic web is full of creeps! *Brief. Bioinform.*, *7*(3), 275–286.

Grenon, P., Smith, B. & Goldberg, L. (2004). Biodynamic ontology: applying BFO in the biomedical domain. In D.M. Pisanelli (Ed.), *Ontologies in Medicine. Proceedings of the Workshop on Medical Ontologies*, Rome (pp. 20–38). Amsterdam: IOS Press.

Guarino, N. (1998). Formal ontology and information systems. In *International Conference on Formal Ontology in Information Systems FOIS'98*, Trento, Italy (pp. 3–15). Amsterdam: IOS Press.

Horrocks, I. (2008). Ontologies and the semantic web. *Commun. ACM*, *51*(12), 58–67.

Horrocks, I., Patel-Schneider, P.F. & van Harmelen, F. (2003). From $\mathcal{SHIQ}$ and RDF to OWL: The making of a web ontology language. *J. Web Semantics*, *1*(1), 7–26.

Hu, Z., Ng, D.M., Yamada, T., Chen, C., Kawashima, S., Mellor, J., Linghu, B., Kanehisa, M., Stuart, J.M. & DeLisi, C. (2007). Visant 3.0: new modules for pathway visualization, editing, prediction and construction. *Nucleic Acids Res.*, *35*(Web Server issue), W625–W632.

Iannone, L., Rector, A. & Stevens, R. (2009). Embedding knowledge patterns into owl. In *ESWC* (pp. 218–232). Berlin: Springer.

Jensen, L.J., Jensen, T.S., de Lichtenberg, U., Brunak, S. & Bork, P. (2006). Co-evolution of transcriptional and post-translational cell-cycle regulation. *Nature*, *443*(7111), 594–597.

Kerrien, S., Alam-Faruque, Y., Aranda, B., Bancarz, I., Bridge, A., Derow, C., Dimmer, E., Feuermann, M., Friedrichsen, A., Huntley, R., Kohler, C., Khadake, J., Leroy, C., Liban, A., Lieftink, C., Montecchi-Palazzi, L., Orchard, S., Risse, J., Robbe, K., Roechert, B., Thorneycroft, D., Zhang, Y., Apweiler, R. & Hermjakob, H. (2007). Intact–open source resource for molecular interaction data. *Nucleic Acids Res.*, *35*(Database issue), D561–D565.

Ruttenberg, A., Clark, T., Bug, W., Samwald, M., Bodenreider, O., Chen, H., Doherty, D., Forsberg, K., Gao, Y., Kashyap, V., Kinoshita, J., Luciano, J., Marshall, S.M., Ogbuji, C., Rees, J., Stephens, S., Wong, G., Wu, E., Zaccagnini, D., Hongsermeier, T., Neumann, E., Herman, I. & Cheung, K.H. (2007). Advancing translational research with the semantic web. *BMC Bioinformatics*, *8*(Suppl. 3), S2.

Sayers, E., Barrett, T., Benson, D., Bryant, S., Canese, K., Chetvernin, V., Church, D., DiCuccio, M., Edgar, R., Federhen, S., Feolo, M., Geer, L., Helmberg, W., Kapustin, Y., Landsman, D., Lipman, D., Madden, T., Maglott, D., Miller, V., Mizrachi, I., Ostell, J., Pruitt, K., Schuler, G., Sequeira, E., Sherry, S., Shumway, M., Sirotkin, K., Souvorov, A., Starchenko, G., Tatusova, T., Wagner, L., Yaschenko, E. & Ye, J. (2009). Database resources of the national center for biotechnology information. *Nucleic Acids Res.*, *37*(Database issue), D5–D15.

Shannon, P., Markiel, A., Ozier, O., Baliga, N., Wang, J., Ramage, D., Amin, N., Schwikowski, B. & Ideker, T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Res.*, *13*(11), 2498–2504.

Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L.J., Eilbeck, K., Ireland, A., Mungall, C.J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S.-A., Scheuermann, R.H., Shah, N., Whetzel, P.L. & Lewis, S. (2007). The obo foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat. Biotechnol.*, *25*, 1251–1255.

Smith, B., Ceusters, W., Klagges, B., Köhler, J., Kumar, A., Lomax, J., Mungall, C., Neuhaus, F., Rector, A.L. & Rosse, C. (2005). Relations in biomedical ontologies. *Genome Biol.*, *6*, R46.

Splendiani, A. (2008). Rdfscape: semantic web meets systems biology. *BMC Bioinformatics*, *9*(Suppl. 4), S6,

Stevens, R. & Lord, P. (2008). Application of ontologies in bioinformatics. In S. Staab and R. Studer (Eds.), *Handbook on Ontologies in Information Systems* (pp. 635–657). Berlin: Springer.

UniProt Consortium (2008). The universal protein resource (uniprot). *Nucleic Acids Res.*, *36*(Database issue), D190–D195,

Van Leene, J., Hollunder, J., Eeckhout, D., Persiau, G., Van De Slijke, E., Stals, H., Van Isterdael, G., Verkest, A., Neirynck, S., Buffel, Y., De Bodt, S., Maere, S., Laukens, K., Pharazyn, A., Ferreira, P., Eloy, N., Renne, C., Meyer, C., Faure, J.-D., Steinbrenner, J., Beynon, J., Larkin, J., Van de Peer, Y., Hilson, P., Kuiper, M., De Veylder, L., Van Onckelen, H., Inzé, D., Witters, E. & De Jaeger, G. (2010). Targeted interactomics reveals a complex core cell cycle machinery in *Arabidopsis thaliana*. *Mol. Syst. Biol.*, *6*, 397.

Zhao, J., Goble, C. & Stevens, R. (2006). An identity crisis in the life sciences. In *Proc. of the 3rd International Provenance and Annotation Workshop*, Chicago, IL, USA.