

BIOMANTA – A SCALABLE, SEMANTIC WEB APPROACH TO REASONING ACROSS LARGE-SCALE DISTRIBUTED BIOMOLECULAR PATHWAY DATASETS

Andrew Newman¹
anewman@itee.uq.edu.au

Yuan-Fang Li¹
liyf@itee.uq.edu.au

Jane Hunter¹
j.hunter@uq.edu.au

¹ School of Information Technology and Electrical Engineering,
University of Queensland, St Lucia, Australia 4072

Current protein-protein interaction data is distributed across a wide range of disparate, large-scale, publicly-available databases and repositories. Semantic Web technologies such as RDF, OWL ontologies and the SPARQL query language appear to provide solutions to the data integration challenge. However existing RDF triple stores suffer from limited scalability and poor querying performance. In this paper we present a novel approach that combines Google's distributed processing MapReduce architecture with Semantic Web technologies to enable high-speed querying and reasoning across large-scale protein-protein interaction datasets. We describe the system architecture, implementation and the results of performance evaluations based on queries across integrated PPI data specified by molecular biologists.

Keywords: scalability, MapReduce, RDF, ontology, semantic reasoning

1. Introduction and Background

1.1. The BioMANTA Project

BioMANTA is a collaborative project between Pfizer Research and the University of Queensland that is applying Semantic Web technologies to the modeling of biological pathways and protein-protein interaction data. It aims to enable *in silico* drug discovery and development by identifying candidate therapeutic targets through the analysis of integrated datasets that relate molecular interactions and biochemical pathways to physiological effects such as toxicology and gene-disease associations. As such, BioMANTA is integrating data from protein datasets such as MPact [16], DIP [37], IntAct [23] and MINT [8] via a common model/ontology.

The common model is the BioMANTA OWL-DL [5] ontology¹ that was developed [11, 29] by reusing vocabularies from well-established ontologies such as the Gene Ontology [2], the Cell Type ontology [4], BioPAX [6], PSI-MI [19], and the NCBI taxonomy. Using the BioMANTA ontology, protein datasets are converted to RDF instances and stored in a distributed RDF triple store where they are available for subsequent analysis and querying.

Figure 1 below shows a set of RDF triples describing a yeast protein with UniProt ID "Q12522", together with other information such as host species, genomic sequence, external references, etc., that are compliant with our BioMANTA ontology.

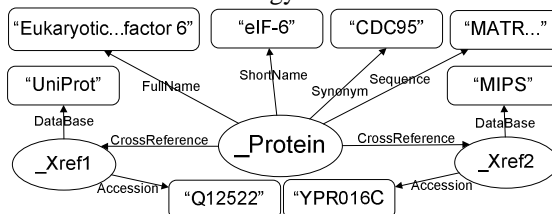


Figure 1. RDF triples about a yeast protein – created by merging data from UniProt and MIPS.

¹ http://biomanta.sourceforge.net/2007/07/biomanta_extension_02.owl

The molecular biologists with whom we are collaborating want rapid responses to queries such as “Show me all the human kinases expressed in the liver that are strongly inhibited by at least two compounds and are localized to the nucleus”.

In order to answer such queries, we need to firstly find solutions to two major barriers to the seamless, semantic integration of large-scale distributed datasets:

1. Poor performance of RDF querying and reasoning;
2. Object co-identification – identifying when two objects are the same.

These issues are particularly problematic within the life sciences domain and were recently identified by the W3C’s Semantic Web Health Care and Life Sciences Interest Group (HCLSIG)² as two of the greatest challenges to the adoption of Semantic Web technologies in the life sciences [35].

1.2. Distributed, Real-time Processing of Large-scale RDF Data

The challenge of large-scale data integration places high demands on processing, storage and querying speed. Distributed processing in a clustered environment offers a low cost, high performance approach to processing massive amounts of RDF instance data (billions or trillions of triples).

In particular, Google’s MapReduce architecture [12] provides a software framework to support distributed computation over extremely large datasets using clusters of commodity-grade hardware. Each compute node processes its local copy of data and returns the results, which are subsequently combined to obtain the complete answer. MapReduce has been successfully deployed within Google on a number of large-scale tasks including the indexing of web pages and has been shown to be a highly reliable, scalable and economical architecture. **Our aim is to investigate methods by which MapReduce could be used to expedite querying and reasoning over large-scale RDF triple stores.**

1.3. The Life Sciences Identifier Problem and Blank Nodes

In life sciences, large ontologies such as the UniProt [44], OBO ontologies [40], the Gene Ontology [2], KEGG [21] and NCBI [42] are being actively employed by communities to markup data and integrate datasets – and as a result are undergoing constant refinement and maintenance. The UniProt protein database, for example, contains approximately 400 million base RDF statements based on its ontology. In addition, these ontologies are being merged to facilitate the integration of compliant datasets to enable the discovery of new knowledge – such as new protein interaction pathways that highlight potential targets, biomarkers and drug side-effects.

The difficulty is that many of the most significant large-scale protein databases such as UniProt, DIP [37], IntAct [23] and MPact [16] each employ different naming conventions – both for proteins and their various attributes. A single protein may be annotated with a variety of properties including different accession IDs, labels, its genomic sequence, the host organism, publication information, etc. A protein may participate in interactions with another protein in observed experiments, documented in a variety of databases. For example, the protein identified as “27628” in DIP is the same protein as the one identified as “115 dax human” in IntAct. It is important to be able to refer to the real protein by either identifier and to be able to query and retrieve its properties from both databases.

The harmonization of such databases and their respective ontologies is a significant research challenge for the Semantic Web community and has been the focus of a number of research projects [10, 38, 41]. Previous attempts to standardize naming and identification (e.g., LSIDs

² <http://www.w3.org/2001/sw/hcls/>

[36]) have had limited beneficial impact [15]. We believe that inventing another naming convention or trying to reach a consensus will not solve the identification problem [17]. We also reject the idea of creating yet another URI to create a co-reference bundle [20], instead we propose an identity reconciliation process for the “life sciences identifier problem” based on RDF *blank nodes*.

RDF blank nodes are nodes with no globally addressable names and are used to represent real-world proteins in our solution. A blank node is used to represent a specific protein and to provide the hub that links to the relevant entries in different (translated) datasets to create a single representation encompassing all information about a particular protein. The properties of this protein, including various identifiers from different databases, are modeled as triples with this blank node as the subject. Although RDF blank nodes have previously been demonstrated to provide a useful approach to the object co-identification problem [9], they also introduce a number of associated problems that arise during distribution of a large RDF graph over a distributed MapReduce architecture.

In a scale-out MapReduce architecture, large RDF documents (graphs) need to be sub-divided into smaller ones for distributed processing. During querying, processing results must be merged in order to eliminate duplicate protein representations. This requires the disambiguation and identification of blank nodes. This is complicated by the fact that blank nodes are not *globally* addressable. Hence, a consistent way of uniquely referring to them is required. The concept of RDF molecules³ [13] was proposed to tackle the problem of addressing blank nodes by decomposing an RDF graph losslessly into a set of RDF molecules which distributes graph and query processing. In the original definition, an RDF molecule is essentially a set of RDF triples interconnected by blank nodes. In the work described here, we extend the definition of RDF molecules with hierarchy and ordering to make the storage, retrieval and querying more efficient in the distributed environment.

1.4. Objectives

The high-level objectives of the work described in this paper are to investigate solutions to the two problems of: inefficient semantic querying and reasoning across large-scale triple stores; and co-identification – within the context of the BioMANTA project. These two highly critical issues hinder the adoption of Semantic Web technologies within many disciplines and applications. The more specific objectives are to investigate and evaluate:

- Methods by which the MapReduce scale-out architecture can be used to improve the performance of semantic querying and inferencing over large-scale RDF triples;
- The adoption of RDF molecules for decomposing and distributing RDF graphs across computational nodes in the MapReduce architecture;
- The use of blank nodes to resolve the co-identification problem;
- Extensions to RDF molecules to overcome problems of ambiguity, data loss and inefficiency introduced by blank nodes.

In addition, the aim is to evaluate our proposed scale-out RDF Molecule Store using biomolecular pathway datasets that have been integrated for the purposes of the BioMANTA project.

The remainder of the paper is organized as follows. In Section 2 we discuss related work. Section 3 provides a high-level description of system architecture and the BioMANTA testbed used in evaluation. In Section 4 we present the initial results of the system’s performance of critical steps

³ Note that the word “molecule” here and throughout the paper is not about molecules in biological sense, but a definition in RDF.

of the system: graph decomposition/merging and SPARQL [33] querying. Finally in Section 5 we present our conclusions and discuss future work.

2. Previous Related Work

A number of projects have focused on the application of Semantic Web technologies to the integration of data associated with protein-protein interactions [39, 43]. In [39], a protein ontology was developed to define common vocabularies in the ontology language OWL [5] for the description of proteins of all species. In [43], the authors presented a case study that involved developing an OWL ontology for a protein family in OWL DL and reasoning over a genome. In [22], a biological pathways tools ontology/schema based on Minsky-style frame system (a precursor of Semantic Web ontology languages) was developed. In this work, entities such as chemicals, organisms, enzymatic-reactions are modeled as classes (frames) and annotated with values (in slots). GORouter [45] is an RDF model that defines vocabularies to describe GO terms, creates additional mappings between GO terms and introduces a set of inference rules. RDF as a standard data model has been proposed in [24] to facilitate the representation and integration of neuroscience datasets BrainPharm⁴ and SWAN [14]/Alzform⁵. The integration process in this work is performed in an ad-hoc way based on queries. In [25], a number of case studies of semantic integration, aggregation and inferencing for pathways using the BioPax ontology [3] as a common representation have been presented. The integration strategy is based on external references (publication, unification, DB reference, etc.).

However, as far as we are aware, none of these previous efforts has resolved the problems of poor querying performance or the life sciences identifier problem.

Apart from our own previous work [31], there have also been a number of similar or related approaches to support scalable semantic querying across large RDF triple stores. Abadi et al and Muster [1, 28] investigated improving RDF query performance through the use of column databases that vertically partition the data. This approach improves query performance for certain types of data and uses a very similar indexing approach to our proposal but does not take advantage of multiple compute nodes in a cluster. The YARS2 and SWSE use a similar “shared nothing” scale-out approach to achieve scalability [18] but it is not based MapReduce processing. There have also been a couple of discussions on the Web proposing MapReduce for improved RDF query performance. Hadoop’s HRDF project⁶ and OpenLink Virtuoso’s clustering technology⁷ use the MapReduce scale-out architecture. However, as far as we are aware, there have been no publications describing actual implementations or results.

Other work has suggested ways to increase the utility of MapReduce by adding a Merge stage to provide a relationally complete scale-out system [46]. A similar, but alternative idea is found in Yahoo’s Pig Latin [32]. Both of these systems could be used to store and process RDF by treating RDF as tuples. The drawback with both of these approaches is that they are “batch” oriented and not real-time.

3. The BioMANTA System

In this section, we briefly present the BioMANTA system architecture and the biological data that we use as a testbed for the integration and querying of PPI data. A brief discussion on the implementation of SPARQL querying over RDF molecule store is also presented.

⁴ <http://senselab.med.yale.edu/BrainPharm>

⁵ <http://www.alzforum.org>

⁶ <http://wiki.apache.org/hadoop/HRDF>

⁷ <http://www.openlinksw.com/weblog/oerling/?id=1270>

3.1. System Architecture

The five-step approach that we have adopted consists of:

1. The PSI-MI XML files are converted to RDF molecules using UniProt identifiers and genomic sequences as primary criteria to identify same proteins.
2. The RDF molecules are added to the Hadoop Cluster and merged with existing information on a per node basis.
3. The RDF molecules form one conceptual RDF graph, the RDF molecule store.
4. Queries issued by the query engine are made across the RDF molecule store stored across the many nodes in the cluster. RDF molecules, across multiple nodes, are merged together in order to return all the data required for query results.
5. Queries are issued by the client to the query engine, using SPARQL and results are returned.

Figure 2 below illustrates this approach.

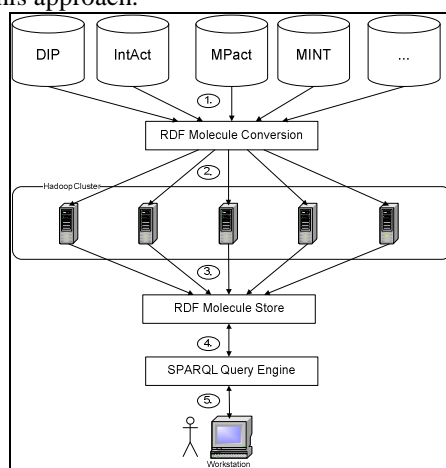


Figure 2. Architecture of the BioMANTA RDF Molecule Store.

3.2. The BioMANTA Testbed

For the purpose of the BioMANTA project, we initially selected datasets from DIP, IntAct and MINT. In our previous work [30], we have developed an integration process to (a) represent them as RDF instances compliant with the BioMANTA ontology and (b) integrate the PPI RDF instances to form new RDF graphs based on UniProt IDs and genomic sequences of proteins, which are represented as RDF blank nodes. The integrated RDF graphs were subsequently decomposed into molecules, distributed into the molecule store and queried.

In protein-protein interaction (PPI) networks, a protein may be annotated with properties such as identifiers, external references, a genomic sequence string, a host organism, etc. The protein may also participate in interactions with other proteins. As discussed in Section 1, RDF blank nodes are used to represent proteins; they are also used to represent interactions, external references, etc. Hence, each proteins and all its associated information will belong to a single RDF molecule, as shown in the example below.

Figure 3 below shows the corresponding RDF molecule of the triples in Figure 1. Note the different indentation levels representing different hierarchies within the molecule; the lexicographical and “groundedness” ordering of the triples and the differentiation of blank nodes `_Xref1` and `_Xref2` based on hierarchy.

_Protein	FullName	"Eukaryotic...factor 6"
_Protein	Sequence	"MATR..."
_Protein	ShortName	"eLF-6"
_Protein	Synonym	"CDC95"
_Protein	CrossReference	_Xref1
_Xref1	Accession	"Q12522"
_Xref1	Database	"UniProt"
_Protein	CrossReference	_Xref2
_Xref2	Accession	"YPR016C"
_Xref2	Database	"MIPS"

Figure 3. The RDF molecule corresponding to a simplified yeast protein.

Such molecules will be stored in a distributed RDF molecule store, which when queried, will return partial answers based on local molecules. The partial answers will be combined to generate the complete result. Our molecular biologist collaborators identified a set of queries that would be likely to reveal previously unrecognized protein-protein-interactions. For instance, the query "Find all yeast protein-protein interactions that are known to be localized to the endosomal system" helps biologists to filter protein-protein interactions (PPIs) more efficiently and requires the integration of the Gene Ontology, the NCBI taxonomy and PPI datasets. Given the size of the PPI data and associated datasets (well over 1 billion triples), only a distributed processing environment would be capable of handling the integration and querying on such scale.

3.3. Semantic Querying

We have implemented a memory-based SPARQL query engine over the RDF molecule store based on the JRDF⁸ project. An indexing structure for storing the molecules was designed with four indices. Three of the indices use the different possible combinations of subject, predicate and object (spo, pos, and osp) together with an additional molecule ID pointing to the molecule that contains this triple. The fourth index holds the parent molecule ID (0 if there is none) and the molecule ID together with the triples. This supports efficient addition, retrieval and removal of RDF molecules in the molecule store. An adaptor that wraps around the molecule store provides an RDF API and SPARQL query functionality. Future development of additional index adaptors would allow query engines from other RDF triple stores such as Jena and Sesame to be reused.

4. Evaluation Results

In this section, we provide initial performance evaluation results for the critical steps in our methodology: RDF graph decomposition, RDF molecule merging and SPARQL querying.

4.1. Graph Decomposition & RDF Molecule Merging

We have developed the graph decomposition and merging algorithms to decompose an RDF graph into a set of RDF molecules, and then merge them back to form an equivalent graph. RDF graph equivalence testing is a critical step for data integration and analysis as equivalence relationships between corresponding nodes in two graphs need to be established in order to integrate them. It has been shown in [7] that the problem of determining RDF graph equivalence can be reduced to the problem of general graph isomorphism [34], which aims at determining equivalence between graphs. In order to evaluate the performance of the molecule store, we conducted a comparison against Jena [26], which is, to the best of our knowledge, the only RDF triple store that provides the functionality to perform graph isomorphism testing.

A set of RDF graphs was created for the comparison. The graph contains triples that have chaining blank nodes, e.g., `_:1 p1 _:2, _:2 p2 _:3, _:3 p3 _:4`. Different number and depth of chains are generated for comparison. For example, in the left table about Jena, the table cell with number 0.05 means that given that chain depth is 3 and chain number is 10 (total graph size is

⁸ <http://jrdf.sourceforge.net/>

30), the time taken to perform isomorphism test is 0.05 seconds. Below is a summarization of the statistics of the result of Jena and RDF molecule. Note that DNF stands for “Did Not Finish” (> 900 seconds).

Table 1. Time measurement of Jena and molecule on graph isomorphism (in secs).

Jena	Depth=3	5	10	20	Molecule	Depth=3	5	10	20
Chain=10	0.05	0.07	0.1	0.3	Chain=10	0.06	0.09	0.1	0.2
100	0.2	0.4	1.8	9.2	100	0.2	0.3	0.4	0.7
1000	13.1	37.7	197.7	DNF	1000	0.9	1.3	2.5	5.0
10000	DNF	DNF	DNF	DNF	10000	7.7	13.0	26.4	57.4

From the above two tables, we can see that the RDF molecule approach is faster as the number of chains becomes 100. Moreover, the RDF molecule implementation gives consistently superior performance as both the number of chains and chain depth increase. As can be seen from the tables, when chain depth is at least 10 and graph size (#chains times chain depth) is at least 1,000 triples (i.e., 100 chains and chain depth of 10), the molecule implementation performs orders of magnitude better than Jena, with Jena not being able to determine isomorphism for graph sizes over 20,000 triples. Also note that with the increase of chain size and depth, the performance of molecule implementation exhibits linear degradation.

4.2. SPARQL Query Responses

As mentioned in Section 3.3, our SPARQL query engine has been developed by adapting the indexing structure of our RDF molecule store so that it is compatible to the indexing structure of the JRDF RDF triple store. Hence, comparable query performance and memory usage is expected. We ran the same SPARQL query (see table below) over the RDF molecule store and the JRDF triple store using an RDF graph describing yeast PPIs obtained and translated from the IntAct dataset. The RDF graph contains 70,613 triples.

The SPARQL query (below) tries to find all instances of the class “physicalEntity” that has NCBI taxonomy ID 4932 and has a primary reference from the UniProt database with ID “o13516”.

```

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX biopax: <http://www.biopax.org/release/biopax-level2.owl#>
PREFIX biomanta: <http://biomanta.sourceforge.net/2007/07/biomanta_extension_02.owl#>
PREFIX ncbi: <http://biomanta.sourceforge.net/2007/10/ncbi_taxo.owl#>
SELECT ?x
WHERE {
  ?x rdf:type biopax:physicalEntity .
  ?x biomanta:hasNCBI ncbi:ncbi_taxo_4932_ind .
  ?x biomanta:hasPrimaryRef ?y .
  ?y biomanta:fromKB biomanta:uniprotkb .
  ?y biomanta:hasID biomanta:o13516
}

```

At this scale level, the query time is almost the same for the two stores (~18 seconds) with the RDF molecule store taking up more memory heap space (87MB versus 55MB) which translates to ~1 million triples per gigabyte. This is due to the fact that greater indexing information is maintained for RDF molecules and no compression or other space-saving optimizations have been applied at this time.

As shown in previous modeling [27], the response time of Nutch (a clustered search engine using Hadoop⁹ – an open-source MapReduce implementation) is essentially constant as the number of servers reaches 2000 nodes with up to 40 GB of data per node. We expect that our implementation of the on-disk, distributed RDF molecule store will conservatively reach 800 billion

⁹ <http://hadoop.apache.org/core/>

triples with a similar setup. Improving index efficiency will easily boost the capacity. Note also that given a certain overhead in communication and node balancing, the performance gain will depend very much on the number of compute nodes in the cluster; the larger the cluster, the greater the benefits.

5. Conclusions

Efficient querying and inferencing across large-scale integrated datasets drawn from many distributed sources is a challenge facing many communities.

Semantic Web technologies such as RDF, OWL and SPARQL are ideal candidates for the task of data integration as they offer open, unambiguous and extensible solutions. At the same time, distributed processing paradigms such as MapReduce have demonstrated economic and practical ways to index and process massive amounts (petabytes) of data. Hence, the synergistic combination of MapReduce and Semantic Web technologies appears to offer a perfect solution to the problem of large-scale heterogeneous data integration, querying and reasoning.

However the co-identification problem, particularly within disciplines such as life sciences, introduces additional complications. Attempts to standardize naming conventions have had limited impact. RDF blank nodes, on the other hand, provide a novel way of referring to entities of common interest without creating new names or coming up with new naming conventions. But RDF blank nodes introduce complications when attempting to distribute RDF graphs across a MapReduce architecture.

In a MapReduce framework, it is a necessary first step to decompose large datasets into smaller units for processing. With the ubiquitous presence of blank nodes, RDF graphs provide too coarse a granularity for effective processing as the context of an entire graph is needed to disambiguate RDF blank nodes. A finer granularity is required to support the distributed integration and processing of RDF data. We believe that RDF molecules provide a finer grained solution to the semantic integration and distribution/decomposition problem. As such, we have developed optimized algorithms to losslessly decompose an RDF graph into a set of smaller RDF "molecules" and subsequently merge them, enabling MapReduce-style processing of RDF graphs. However this process revealed that the presence of RDF blank nodes can cause problems of data loss, integrity loss, ambiguity and slow performance. Consequently we have had to extend the definition of RDF molecules to include hierarchy and ordering. By incorporating hierarchy, originally flat RDF molecules now contain explicit structural information, beneficial in enabling more intelligent processing. More importantly, a hierarchy makes it possible to disambiguate blank nodes within a molecule. The ordering of triples provides an efficient way of cross-checking data integrity during the processing of molecules.

We have implemented an in-memory, local version of the RDF molecule store and evaluated the performance of critical algorithms such as RDF graph decomposition and RDF molecule merging. The performance of the RDF graph decomposition and merging steps was compared with the graph isomorphism algorithm in Jena and we obtained promising results. We have also run SPARQL queries over the RDF molecule store, and demonstrated performance that is comparable to the JRDF triple store for moderate numbers of RDF triples. As greater numbers of triples are loaded into the scale-out RDF molecule store and as the size of the computational cluster grows, we can expect the performance to increase relative to traditional RDF triple stores.

Finally, an important future research direction is the development of a disk-based, distributed, processing environment for the extended RDF molecules. We believe that such an environment will greatly enhance our ability to query and reason across large amounts of data efficiently. While our current RDF molecule definition decomposes a graph according to blank node connectedness, the definition could be extended to create molecules based on the connectedness of grounded triples (triples without blank nodes) with matching subjects and objects as well.

Acknowledgments

We would like to acknowledge Pfizer for funding the research described in this paper and to thank Abdul Alabri, Melissa Davis, Imran Khan and Muhammad Shoaib B. Sehgal for their contributions and ideas.

References

- [1] Abadi, D.J., et al. Scalable Semantic Web Data Management Using Vertical Partitioning. in *VLDB 2007*. 2007. University of Vienna, Austria.
- [2] Ashburner, M., et al., Gene Ontology: tool for the unification of biology. *Nature Genetics*, 2000. **25**: p. 25-29.
- [3] Bader, G.D. and M.P. Cary, *BioPAX – Biological Pathways Exchange Language Level 2, Version 1.0 Documentation*. 2005, BioPAX.
- [4] Bard, J., S.Y. Rhee, and M. Ashburner, An Ontology for Cell Types. *Genome Biology*, 2005. **6**(2).
- [5] Bechhofer, S., et al., *OWL Web Ontology Language Reference*, in *W3C Recommendations*, M. Dean and G. Schreiber, Editors. 2004, World Wide Web Consortium.
- [6] BioPAX Workgroup, *BioPAX – Biological Pathways Exchange Language Level 2, Version 1.0 Documentation*. 2005, BioPAX.
- [7] Carroll, J.J., *Matching RDF Graphs*. 2001, HP Laboratories Bristol.
- [8] Chatr-aryamontri, A., et al., MINT: the Molecular INTERaction database. *Nucleic Acids Res*, 2007. **35**(Database issue): p. 572-574.
- [9] Chen, H., Z. Wu, and Y. Mao. RDF-Based Ontology View for Relational Schema Mediation in Semantic Web. in *9th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2005)*. 2005. Melbourne, Australia.
- [10] Cheung, K.-H., et al., YeastHub: a semantic web use case for integrating data in the life sciences domain. *Bioinformatics*, 2005. **21**(Supp. 1): p. 85-96.
- [11] Davis, M., et al. Integrating Hierarchical Controlled Vocabularies with OWL Ontology: A Case Study from the Domain of Molecule Interactions. in *6th Asia Pacific Bioinformatics Conference (APBC08)*. 2008. Kyoto, Japan.
- [12] Dean, J. and S. Ghemawat. MapReduce: Simplified Data Processing on Large Clusters. in *Proceedings of the 6th Conference on Symposium on Operating Systems Design & Implementation*. 2004. San Francisco, CA: USENIX Association.
- [13] Ding, L., et al., Tracking RDF Graph Provenance using RDF Molecules. *Proc. of the 4th International Semantic Web Conference (Poster)*, 2005.
- [14] Gao, Y., et al., SWAN: A distributed knowledge infrastructure for Alzheimer disease research *Journal of Web Semantics*, 2006. **4**(3): p. 222-228.
- [15] Good, B.M. and M.D. Wilkinson, The Life Sciences Semantic Web is full of creeps! *Briefings in Bioinformatics*, 2006. **7**(3): p. 275-286.
- [16] Güldener, U., et al., MPact: the MIPS protein interaction resource on yeast. *Nucleic Acids Res*, 2006. **34**(Database issue): p. 436-441.
- [17] Halpin, H., Identity, Reference, and Meaning on the Web. *Proceedings of the Workshop on Identity, Meaning and the Web (IMW06) at WWW2006, Edinburgh, Scotland, 2006*.
- [18] Harth, A., et al., *YARS2: A Federated Repository for Searching and Querying Graph Structured Data*. 2007, DERI Galway, Ireland.
- [19] Hermjakob, H., et al., The HUPO PSI's Molecular Interaction format—a community standard for the representation of protein interaction data. *Nat Biotechnol*, 2004. **22**(2): p. 177-83.
- [20] Jaffri, A., H. Glaser, and I.C. Millard. Managing URI Synonymity to Enable Consistent Reference on the Semantic Web. in *1st International Workshop on Identity and Reference on the Semantic Web (IRSW2008)* 2008. Tenerife, Spain.
- [21] Kanehisa, M., et al., From genomics to chemical genomics: new developments in KEGG. *Nucleic Acids Research*, 2006. **34**(Database Issue): p. 354-357.
- [22] Karp, P. Pathway Tools and BioCyc: Progress and Future Directions. in *Pathway Tools Workshop*. 2006. Menlo Park, CA, USA.

- [23] Kerrien, S., et al., IntAct--open source resource for molecular interaction data. *Nucleic Acids Res*, 2007. **35**(Database issue): p. D561-5.
- [24] Lam, H.Y.K., et al. Semantic Web Meets e-Neuroscience: An RDF Use Case. in *International Workshop on Semantic e-Science (co-located with ASWC2006)*. 2006. Beijing, China.
- [25] Luciano, J. and J. Zucker. Semantic Aggregation, Integration, and Inference of Pathway Data. in *Annual Meeting of the International Society for Computational Biology (ISMB2005)*. 2005. Detroit, Michigan, USA.
- [26] McBride, B., Jena: a semantic Web toolkit. *IEEE Internet Computing*, 2002. **6**(6): p. 55-59.
- [27] Moreira, J.E., et al. Scalability of the Nutch search engine. in *Proceedings of the 21st Annual International Conference on Supercomputing*. 2007. Seattle, Washington: ACM Press.
- [28] Muster, P., *Quantitative and Qualitative Evaluation of a SPARQL Front-End for MonetDB*, in *Department of Informatics*. 2007, University of Zurich: Zurich.
- [29] Newman, A., et al. BioMANTA Ontology: The Integration of Protein-Protein Interaction Data. in *Interdisciplinary Ontology Conference (InterOntology08 Tokyo)*. 2008. Tokyo, Japan.
- [30] Newman, A., et al. BioMANTA Ontology: The Integration of Protein-Protein Interaction Data. in *Interdisciplinary Ontology Conference (InterOntology08)*. 2007. Tokyo, Japan.
- [31] Newman, A., et al. A Scale-Out RDF Molecule Store for Distributed Processing of Biomedical Data. in *Semantic Web for Health Care and Life Sciences Workshop (HCLS'08) at the 17th International Conference on World Wide Web (WWW'08)*. 2008. Beijing, China.
- [32] Olston, C., et al. Pig Latin: A Not-So-Foreign Language for Data Processing. in *Proceedings of the 2008 ACM SIGMOD International Conference on Management of Data*. 2008. Vancouver, Canada: ACM.
- [33] Prud'hommeaux, E. and A. Seaborne, *SPARQL Query Language for RDF*. 2008, W3C Recommendation, <http://www.w3.org/TR/rdf-sparql-query/>.
- [34] Ronald Read and D. Corneil, The graph isomorphism disease. *Journal of Graph Theory*, 1977. **1**: p. 339-363.
- [35] Ruttenger, A., et al., Advancing translational research with the Semantic Web. *BMC Bioinformatics*, 2007. **8**(Suppl 3).
- [36] Salamone, S., *LSID: An Informatics Lifesaver*. 2004, Bio-ITWorld, <http://www.bio-itworld.com/archive/011204/lsid.html>.
- [37] Salwinski, L., et al., The Database of Interacting Proteins: 2004 update. *Nucleic Acids Res*, 2004. **32**(Database issue): p. D449-51.
- [38] Schroeter, R. and J. Hunter. Annotating Relationships Between Multiple Mixed-Media Digital Objects by Extending Annotea. in *Proceedings of the 4th European Semantic Web Conference (ESWC 2007)*. 2007. Innsbruck, Austria: Springer.
- [39] Sidhu, A.S., et al. Protein Ontology: Semantic Data Integration in Proteomics. in *4th International Joint Conference of InCoB, AASBi and KSBI 2005 (BIOINFO 2005)*. 2005. Busan, Korea: KAIST Press.
- [40] Smith, B., et al., The OBO Foundry: coordinated evolution of ontologies to support biomedical data integration. *Nature Biotechnology*, 2007. **25**: p. 1251-1255.
- [41] Stephens, S.M., A. Quinlan, M., Applying semantic Web technologies to drug safety determination. *IEEE Intelligent Systems*, 2006. **21**(1): p. 82-88.
- [42] Wheeler DL, C.C., Lash AE, Leipe DD, Madden TL, Schuler GD, Tatusova TA, Rapp BA, Database resources of the National Center for Biotechnology Information. *Nucleic Acids Research*, 2000. **28**(1): p. 10-4.
- [43] Wolstencroft, K., R. Stevens, and V. Haarslev, *Applying OWL Reasoning to Genomic Data*, in *Semantic Web: Revolutionizing Knowledge Discovery in the Life Sciences*, C.J.O. Baker and K.-H. Cheung, Editors. 2007, Springer US: Secaucus, NJ, USA. p. 225-248.
- [44] Wu, C.H., et al., The Universal Protein Resource (UniProt): an expanding universe of protein information. *Nucleic Acids Res*, 2006. **34**(Database issue): p. 187-91.
- [45] Xu, Q., et al., GORouter: an RDF model for providing semantic query and inference services for Gene Ontology and its associations. *BMC Bioinformatics*, 2008. **9**(S6).
- [46] Yang, H.-c., et al. Map-Reduce-Merge: Simplified Relational Data Processing on Large Clusters. in *Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data*. 2007. Beijing, China.