Semantic Web for Chemical Genomics – need, how to, and hurdles

Talapady N Bhat
Biochemical Sciences Division, CSTL, NIST, 100 Bureau Drive, Gaithersburg MD20899, USA

ABSTRACT

Semantic Web has been often suggested as the information technology solution to the growing problem in managing the millions of data points generated by modern science such as nanotechnology and high through-put screening for drugs. However, the progress towards this vision envisaged by the W3C has been very limited. Here we discuss –some of the obstacles to the realization of this vision and we make some suggestions as to how one may overcome some of these hurdles? Here we discuss some of these issues and present thoughts on an alternative method to Semantic Web that is less drastic in requirements. This method does not require the use of RDF and Protege, and it works in an environment currently used by the chemical and biological database providers. In our method one attempts to use as many components as possible from the tools already used by the database providers and one brings in far fewer new tools and techniques compared to the method that use RDF or Protégé. Our method uses a standard database environment and web tools rather than the RDF and Protégé to manage user interface and the data is held in a database rather than using RDF. This method shifts the task of building Semantic knowledge-base and ontology from RDF and Protégé to a SQL based database environment.

Keywords: HIV, AIDS, Semantic Web, OWL, RDF, Healthcare, Chem-BLAST

Introduction: There is considerable interest on Semantic Web as a possible solution to the growing information technology needs of biological[1] data and drug discovery[2-4]. Semantic Web is proposed by W3C (http://www.w3.org/) as a 'vision for the future of the Web in which information is given explicit meaning, making it easier for machines to automatically process and integrate information available on the Web'. In spite of all these emphasis, there has been little or no measurable progress in building a successful public chemical or biological semantic web as evidenced from the fact that so far no major database providers (PDB[5],SWISS-PROT[6], GeneBank[7]) have adopted this technology in spite of major recent re-design of some of these Web sites. Also, several new Web resources (PubCheM[8], HIVSD[9, 10]) focused on chemicals and drug-design have come online during the last two years and they did not use Semantic Web search engines. In view of this, a question that comes to ones mind is – is this a slow start of Semantic web- just the usual teething problem of a new technology, or is it a pre-view of more serious difficulties faced by the Semantic Web technology using the guidelines outlined by the W3C? Before we try to address this question, it is worthwhile to compare features of the traditional and very 'contagious' and successful Google type search engines with the proposed Semantic Search engines that is failing to 'infect' any one.

Semantic Web vs. Google type Web: What is the difference between a Semantic Web and other technologies such as the traditional internet that did not face such startup problem? It is difficult to give a precise answer to this question in the absence of a recognized and certified Semantic web resource to compare against the abundant Google type search engines. As per the vision of the W3C, Semantic Web[1, 4, 11] is 'super' Web resource were the search engines are smart and 'human like', and they are fully data aware, and are capable of de-ciphering complicated questions to give precise answers just like an expert human being. In the words of W3C, 'The Semantic Web is a vision for the future of the Web in which information is given explicit meaning, making it easier for machines to automatically process and integrate information available on the Web'.

Semantic web search engines are expected to produce precise hits for complicated questions. To achieve this vision, the Semantic search engines are expected to be fully 'data aware' and they are expected to know the context of each question in light of previous questions, and also possible new questions that may arise from a given question and their answers as well. For this reason, a Semantic Web search engines not only need to know the meaning of a data in a general context but also need to know the meaning of a data in the context of the previous questions and their answers too. For instance, in a Google type web environment, if a user queries for David Brown, the Web tool is expected to produce a list

of all the occurrences of the word -David Brown and display the data associated with each one of them. In a Semantic world, the search tool is expected to be able to make much more meaning out of the key word -David Brown. The web tool is expected to know that the key word defines a specific person; if the preceding question was for Chris Brown, then the Semantic Search engine is expected to know that the present question is likely for data on someone related to Chris Brown (father, son, uncle, et c.) and not for data on any David Brown in the database. Querying a Semantic Web is like interviewing an expert who has in-depth knowledge of each and every data item in the database and who presents the result of a query in a context dependent manner. Semantic search engines are expected to maintain stream of thought between successive query results and they are not expected to shower a user with many irrelevant answers. Imagine the person you are interviewing keeps changing topic all the time! It will be frustrating, obviously!

Approach: The vision of the Semantic Web is very exciting from the view point of users. Certainly, it is expected to make users life much simpler and to provide superior on-line experience. However, developing such Web pages requires not only smart search engines to handle data, but also it requires smart data annotation techniques to correctly cross-index all related relevant data. Computers, unlike humans, are 'not self learning', annotation and data indexing is a proven way to establish a knowledgebase for search engines. This step is commonly known as data annotation and it establishes all relevant indices for data to convert them to knowledge. For instance, in the above example, data annotation step establishes relationship between searchable and related elements such as David Brown, his son's name, ex-wife's name and so on. The search engines use these indices to generate precise, and context dependent answers. Therefore, the development of a true Semantic web requires both IT and database developers to work in symbiosis- both working together to complement each others work.

The development of a semantic Web has two major closely knitted components, IT (search engines, and web input and display tools) and database development (data acquisition, annotation and data rendering to the search engines). These two components are expected to dynamically blend during a query from a user, namely the search engines are expected to become 'data-aware' using the indices found in the database.

The data-awareness of the search engines may be achieved either by developing a custom made search engine for each problem or by training a tunable search engine such as Protégé using RDF. Both of these type of search engines have their own limitations. The development of Semantic Web is like building a house to satisfy a customer specification- that has two components; a) building of blocks such as windows and doors, b) and the actual construction of the house from these blocks. The database development is like the design and development of windows and doors that may be assembled to produce a user defined house and the development of search engines is like building the technology needed to build houses from these blocks. Thus it is obvious that a high degree of co-ordination is required between these two independently performed tasks of database and search engine development. Lack of co-ordination between the IT and database development may be one of the reasons to the observed slow progress in developing Semantic Web pages. Apart from this reason, we also believe that the recommendation by the W3C to use Protégé as the search engine is also partly responsible for this slow progress.

W3C recommends the use Protégé as the software front-end for semantic web and this concept may be summarized in fig 1. In this approach, on one-hand, the availability of public domain software may look like a welcome situation for some users, on the other-hand, the complications of using a new software may discourage other users – a challenge is often more welcome than a solution that is difficult to understand. Giving up the familiar database environments and home-made software tools may be a difficult choice for many of the database providers. Many of the current biological web resources (PDB, SwisProt, HIVSDB) use databases such as ORACLE or MySQL to build knowledgebase[12, 13] and the use of a different setup such as those of RDF is a big quantum jump for these database providers.

To reduce the effect of this big quantum jump needed to develop a semantic web using OWL, we propose to take (at least initially) a different

approach (Fig 2). In this approach, the knowledgebase and ontology are built during annotation step using a database (such as ORACLE or MySQL) of a user's choice, and search engines are built using SQL and a program language such as Perl or Java. The intermediate step involving RDF is made as a .part of the database development. Database providers are the best experts in the 'Semantics' of the data they provide to users and for that reason, our methods assign the responsibility of establishing knowledgebase to the database providers. Further, our method allows one to take advantage of the convenience of the modern database software while developing Semantic Web like data resources without abandoning ones favorite database environment. Further, in this method one may also gradually shift an existing web to a truly Semantic Web as data annotation step continues and better search engines are built.

Implementation: Here we illustrate our implementation of our method of building a Semantic Web like resource in the context of AIDS structural database – HIVSDB (http://xpdb.nist.gov/hivsdb/hivsdb.html). A full description of the chemical ontology, database structure and software features will be published elsewhere. One of the goals of the HIVSDB is to facilitate query and comparison of inhibitors of an AIDS target enzyme -HIV-1 protease. About half of the clinically used AIDS drugs target this enzyme. Cure for AIDS is still a work in progress and thus resources such as the HIVSDB play a critical role in fighting this global epidemic. HIVSDB has over 1000 compounds and this database has the largest collection of 3-D structures of HIV-1 protease complexes available in the public domain. HIVSDB also has information on biological and antiviral data related to these drugs. During the annotation part of the work of developing Semantic Web using our method, structural ontology is built to enumerate the mode of interaction of these inhibitors with the active site residues of the HIV-1 protease. This information is organized into a chemical data-tree (Fig 2) using database tables stored in ORACLE. The different layers of the data-tree may be imagined to represent the different echelons of an RDF. Search engines are developed using Perl to present the data-tree and related structural and biological data in a series of steps. In each of these steps a user has the option of choosing structural features of his interest from the many possibilities. Different data relationships established by the data-tree are used by the search engines to produce succinct answers to complicated questions related to enzyme drug interactions. Fig 3 is a snap-shot of the enzyme-drug interaction resulting from a query on of one of the drugs clinically prescribed during combination drug-therapy for AIDS. Unlike many other search engines (e.g. PDB) our search engines dynamically establish and display structural similarity using a chemical ontology. Search engines use this ontology to become 'data aware' and using this ontology they provide context dependent answers that are unique to a query and its pre-curser queries. The search engines also use the ontology to guide the users for possible new queries using context dependent hyperlinks. Thus the chemical ontology permits the search engines to minimize missed hits without increasing the number of irrelevant hits.

Fig 1 shows the traditional path (left) and proposed method (right) of establishing a Semantic Web

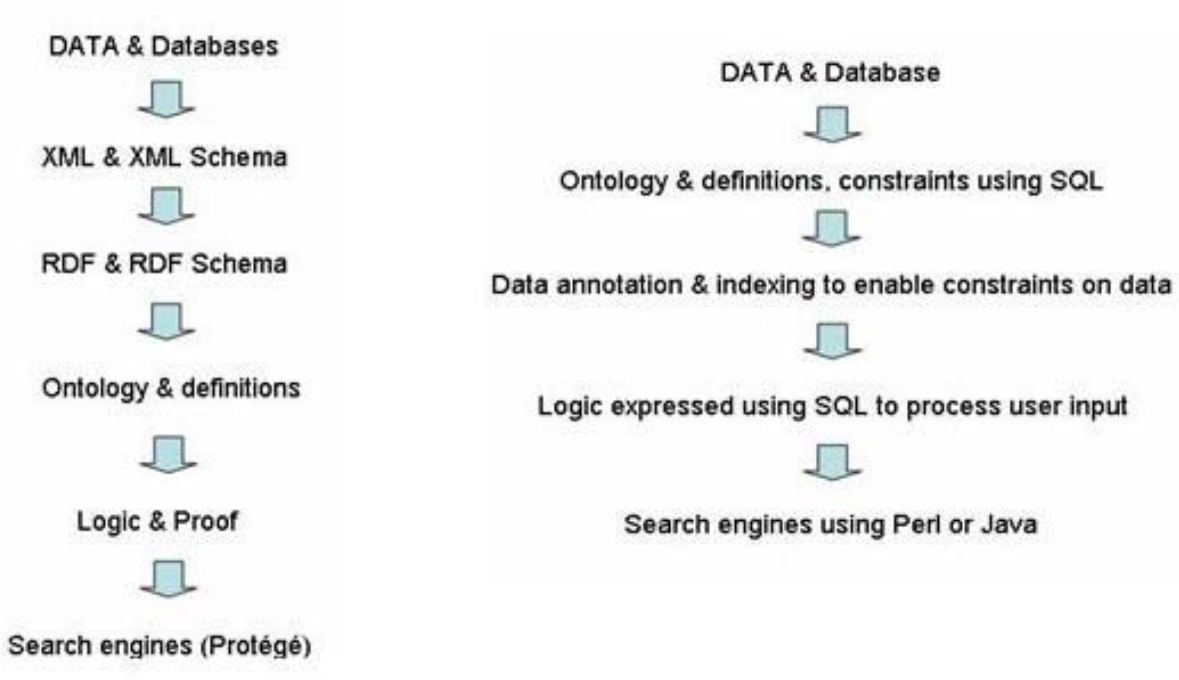


Fig 2 shows a chemical data-tree that defines the data hierarchy for chemical fragments. These layers establish RDF like relationships for chemical structures of AIDS drugs.

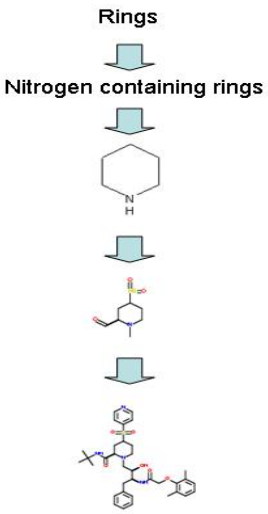
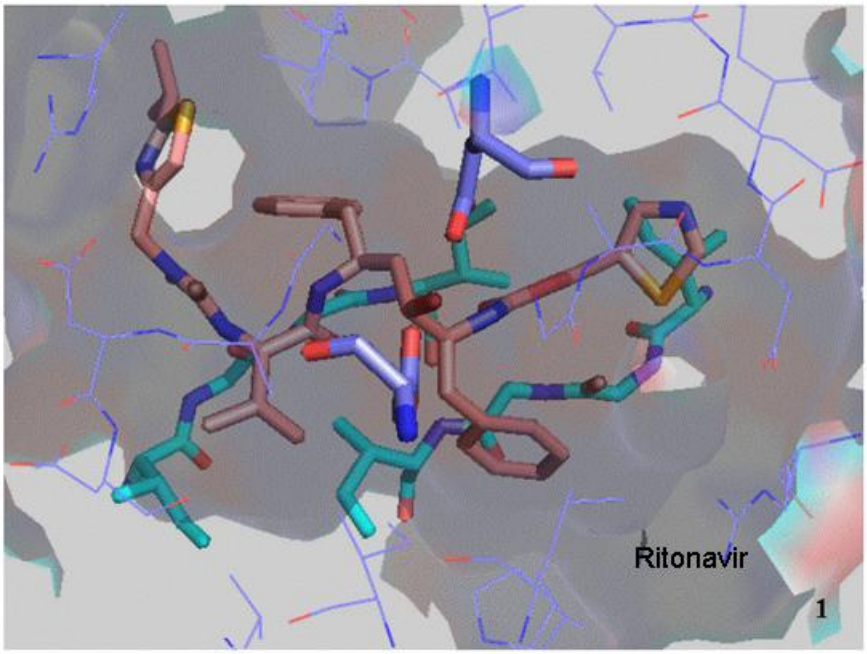


Fig 3 shows the interaction of a clinical drug Ritonavir with its target enzyme. This view may be obtained by querying using the chemical data-tree that establishes RDF like relationships among the chemical components of the
drug.



Disclaimer: Certain trade and company products are identified in this paper to specify adequately the computer products needed to develop this data system. In no case does such identification imply endorsement by the National Institute of Standards and Technology (NIST), or does it imply that the products are necessarily the best available for the purpose.

References:

1. Clark, T. Looking Ahead: The Semantic Web in Life Science. *Genome Technology*, 23 -24 (2005).
2. MacNeil, J.S. Diving Deep Into The Chemical Genome. *Genome Technology*, 26 - 35 (2005).
3. Austin, C.P., Brady, L.S., Insel, T.R. & Collins, F.S. NIH Molecular Libraries Initiative. *Science* **306**, 1138-1139 (2004).
4. Neumann, E., Quan, D. Biodash: A Semantic Web Dashboard for Drug Development. *Pacific Sympoosium on Biocomputing* **11**, 176-187 (2006).
5. Berman, H.M. et al. The Protein Data Bank. *Nucleic Acids Res* **28**, 235-242 (2000).
6. Boeckmann, B. et al. The SWISS-PROT protein knowledgebase and its supplement TrEMBL. *Nucleic Acids Res.* **31**, 365-370 (2003).
7. Benson, D.A., Karsch-Mizrachi, I., Lipman, D.J., Ostell, J. & Wheeler, D.L. GenBank. *Nucleic Acids Res.* **31**, 23-27 (2003).
8. Wheeler, D.L. et al. Database resources of the National Center for Biotechnology Information.

Nucleic Acids Res **34**, D173-180 (2006).
9. Prasanna, M., Vondrasek, J., Wlodawer, A., Bhat, TN. Application of InChI to curate, index and query 3-D structures. *Proteins, Structure, Function, and Bioinformatics* **60**, 1-4 (2005).
10. Prasanna, M.D., Vondrasek, J., Wlodawer, A., Rodriguez, H. & Bhat, T.N. Chemical Compound Navigator: A Web-based Chem-BLAST Search Engine for Browsing Compounds. *PROTEINS: Structure, Function, and Bioinformatics* **63**, 907-917 (2006).
11. Slaughter, L.A., Soergel, D. & Rindflesch, T.C. Semantic representation of consumer questions and physician answers. *Int J Med Inform* (2005).
12. Bhat, T.N. et al. The PDB data uniformity project. *Nucleic Acids Res* **29**, 214-218 (2001).
13. Bhat, T.N. Migration from Static to Dynamic Web Interface - Enzyme Thermodynamics Database as an Example. *Proceedings of The 9th World Multi-Conference on Systemics, Cybernetics and Informatics, Orlando, Florida, USA* **1**, 69-74 (2005).