Nucleic Acids Research, 2007, Vol. 35, Web Server issue ***W21–W26***
doi:10.1093/nar/gkm298

iHOP web services

José M. Fernández[1,*], Robert Hoffmann[2] and Alfonso Valencia[1]

[1]Structural Biology and Biocomputing Program, CNIO and [2]Decentralized Information Group, Computer Science and Artificial Intelligence Laboratory, MIT.

Received February 5, 2007; Revised April 5, 2007; Accepted April 12, 2007

ABSTRACT

iHOP provides fast, accurate, comprehensive, and up-to-date summary information on more than 80 000 biological molecules by automatically extracting key sentences from millions of PubMed documents. Its intuitive user interface and navigation scheme have made iHOP extremely successful among biologists, counting more than 500 000 visits per month (iHOP access statistics: http://www.ihop-net.org/UniPub/iHOP/info/logs/). Here we describe a public programmatic API that enables the integration of main iHOP functionalities in bioinformatic programs and workflows.

INTRODUCTION

iHOP (1) (iHOP literature server, http://www.ihop.net.org) allows researchers to explore a network of gene and protein interactions by directly navigating the pool of published scientific literature. Rather than providing long lists of entire abstracts upon keyword searches, iHOP selectively retrieves information that is specific to genes and proteins and summarizes their interactions and functions. The system adds value by filtering and ranking extracted sentences according to significance, impact factor, date of publication and syntax.

iHOP web content is pre-compiled and generated in a multi-step process to annotate biomedical texts with gene and protein names, chemical compounds and MeSH terms. This annotation task is computationally expensive because of the sheer number of entities, but more importantly, hindered by a high semantic overloading of abbreviations and synonyms in biomedicine. The continuous development and optimization of heuristics and machine learning algorithms to improve entity detection and synonym disambiguation is therefore a central effort in the maintenance of iHOP.

Given the complexity and effort that goes into the development and maintenance of a text-mining pipeline, it makes sense to build upon the existing infrastructure of iHOP rather than reinventing the wheel. Already numerous online resources are linking to iHOP and novel tools are emerging which are based on the iHOP resource, e.g. iHOPerator (2). The iHOP web service API has already been tested in selected projects over the last 2 years and is made publicly available now. Although on any biocomputing facility APIs are not as visible to the end user, they are very important for the different *omics*, which usually depend on powerful data set analysis. Those powerful analysis run distributed workflows, which have to semantically integrate the results from diverse biocomputing facilities and data sources. Other large-scale biocomputing facilities provide environments such as NCBI Entrez (3) (Entrez CGI services, http://www.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.html; Entrez SOAP services, http://www.ncbi.nlm.nih.gov/entrez/query/static/esoap_help.html) or EBI WS (4) (EBI SOAP services, http://www.ebi.ac.uk/Tools/webservices/).

METHODS

To make the iHOP programmatic interface remotely accessible and integrable on workflows in a way that is neutral to programming languages and vendor independent, we decided to implement the public API in the form of web services (5). Three popular web service API models have been implemented for iHOP: the REST model (6) (Wikipedia description of REST, http://en.wikipedia.org/wiki/Representational_State_Transfer), which the DAS (7) protocol follows; SOAP + WSDL, which is based on WSDL document description and uses SOAP messages and XML for messaging; BioMOBY (8), which is focused on bioinformatic workflow building (9). Table 1 contains a brief description of these API models. All three API implementations are based on a common internal library and common XML schemas to facilitate maintenance and future developments. MOBY implementation required additional efforts to integrate iHOP web services into the MOBY ontology.

The schema design was driven by the iHOP functionalities that are directly useful for bioinformatical workflows (Figures 1 and 2). Table 2 contains a brief description of these functionalities, with their inputs and outputs.

A key issue in the development was the design of an XML schema rich enough to describe and integrate

*To whom correspondence should be addressed. Tel: +34 917 328 000; Fax: +34 912 246 976; Email: Jmfernandez@cnio.es

Table 1. Brief description of web services API models, used on iHOP web services

Web services API models	Description
REST	The Representational State Transfer paradigm is based on the HTTP protocol. It is an improvement over CGI sub-protocol, used by web browsers to send the data in the HTML formularies. The main difference is on the provided results: a REST service is usually restricted to answer with an XML document with all the information; a CGI service can return any document type (HTML, GIF, PDF, etc.). There are also differences between CGI and REST is the way to send the queries: a key-value model is used to send the query data to the CGI service; REST implementations send an XML document with the whole information. Some implementations following REST paradigm (e.g. DAS and iHOP CGI-XML), follow the CGI sub-protocol for the queries, and answer an XML document as the result.
SOAP+WSDL	The Simple Object Access Protocol web services paradigm tries to isolate the used communications protocol from the message representation by using XML formatted messages and different message exchange patterns. SOAP messages have an envelope and a body, so complex features and message exchange patterns can be implemented, nevertheless the used communications protocol. Although there are SOAP client and server libraries for FTP, SMTP, POP3 and other protocols, the most used communication protocol is HTTP. Web Services Description Language is a companion technology of SOAP services. WSDL documents are usually used to describe SOAP services: their inputs and outputs, their XML data types, the data encoding and the message patterns to use. Although any SOAP service can be used without the aid of a WSDL document which describes the service, they are a standard way to distribute that information.
BioMOBY	BioMOBY web services architecture has three main roles: MOBY Central, MOBY clients and MOBY services. MOBY Central is the repository of the ontologies used by the architecture: namespace ontology, which is used to label biological references so they can be unambiguously identified; object ontology, which defines the object types which can be consumed and produced by the services; service type ontology, which contains the classification of service types used to label the different services; and the service ontology, which contains the description of all the registered MOBY services. One of the features which distinguishes BioMOBY architecture from SOAP Protocol is that many queries can be clustered on a single message when they are sent to a service. Although BioMOBY architecture has its own message formats and query protocols, SOAP infrastructure is used to wrap MOBY messages. BioMOBY community is now discussing the need to drop SOAP (due its overhead) in favour of lighter REST paradigm.

the valuable information that is already accessible through the iHOP user interface. For instance, annotated sentences are generated by `getSymbolDefinitions`, `getSymbolInteractions` and `getPubMed functionalities`. Each sentence also provides information about the abstract, journal and the journal impact factor. Basic symbol information is provided by `getSymbolInfo`, and it can also be found on `getSymbolDefinitions` and `getSymbolInteractions` results. The designed XML Schema, along with its documentation, is available at the iHOP web services site.

Usually, gene symbol disambiguation is a hard task, made in the last term by the user, and its automation is an essential part in a useful workflow. Using specific heuristics for these web services, we have created an additional functionality called `guessSymbolIdFromSymbolText`, which guesses the nearest unambiguous iHOP gene symbol id from free text input and an optional target organism. This concept is very similar to 'I'm feeling lucky' Google functionality, and the functionality speeds up workflow building. Workflow writers are not tied to this service and its heuristics, because anyone can create their own heuristics about symbol selection using `getRelatedSymbols` output.

Under the REST (Representational State Transfer) paradigm there is a CGI-XML service available for all functionalities described earlier. Special return cases have been modelled using standard HTTP codes: when there is no answer for a query in a CGI-XML service, a 404 Not Found error is returned; if an internal error happens, a 500 Internal Server Error is used; if no input parameter is specified, a 400 Bad Request error is returned.

For SOAP (Simple Object Access Protocol), we created for each functionality variations of the same web service, to simplify workflow building. SOAP services use the RPC/encoded WSDL style, so they can be used from Perl programs with any SOAP::Lite version. Critical errors (no input parameter, internal server error) are reported by the iHOP SOAP services using the standard SOAP fault mechanism. When there is no answer to return, the services return a specific XML structure (*iHOPSOAPNotFound*) designed for these SOAP services, instead of using SOAP fault mechanism. This is important, because some workflow enactment tools (like Taverna) stop the whole workflow when a SOAP service returns a SOAP fault, an undesirable effect when a service invocation has not failed.

In the design of BioMOBY services it was necessary to comply with the common object ontology on MOBY Central and the portfolio of services that are using this ontology. Although the main iHOP services take the same parameters as input and use the same XML schema as CGI-XML and SOAP for their outputs, the true power of iHOP MOBY service are the additional translation services. These services take as input iHOP XML structures generated by the iHOP services, and translate the content into a collection of usable MOBY objects. This way, other MOBY services which use the same ontology can be chained to this output.

CGI-XML services were tested using both web browsers and command-line HTTP retrieval tools (like wget). We tested and cross validated the functionality of iHOP SOAP web services with unit tests based on the Perl SOAP::Lite library and in the context of Taverna (10,11), a workflow enactment tool extensively used by the bioinformatics community. We found that SOAP::Lite 0.60 had a better behaviour than former versions and some new intermediate ones (last version is 0.69).

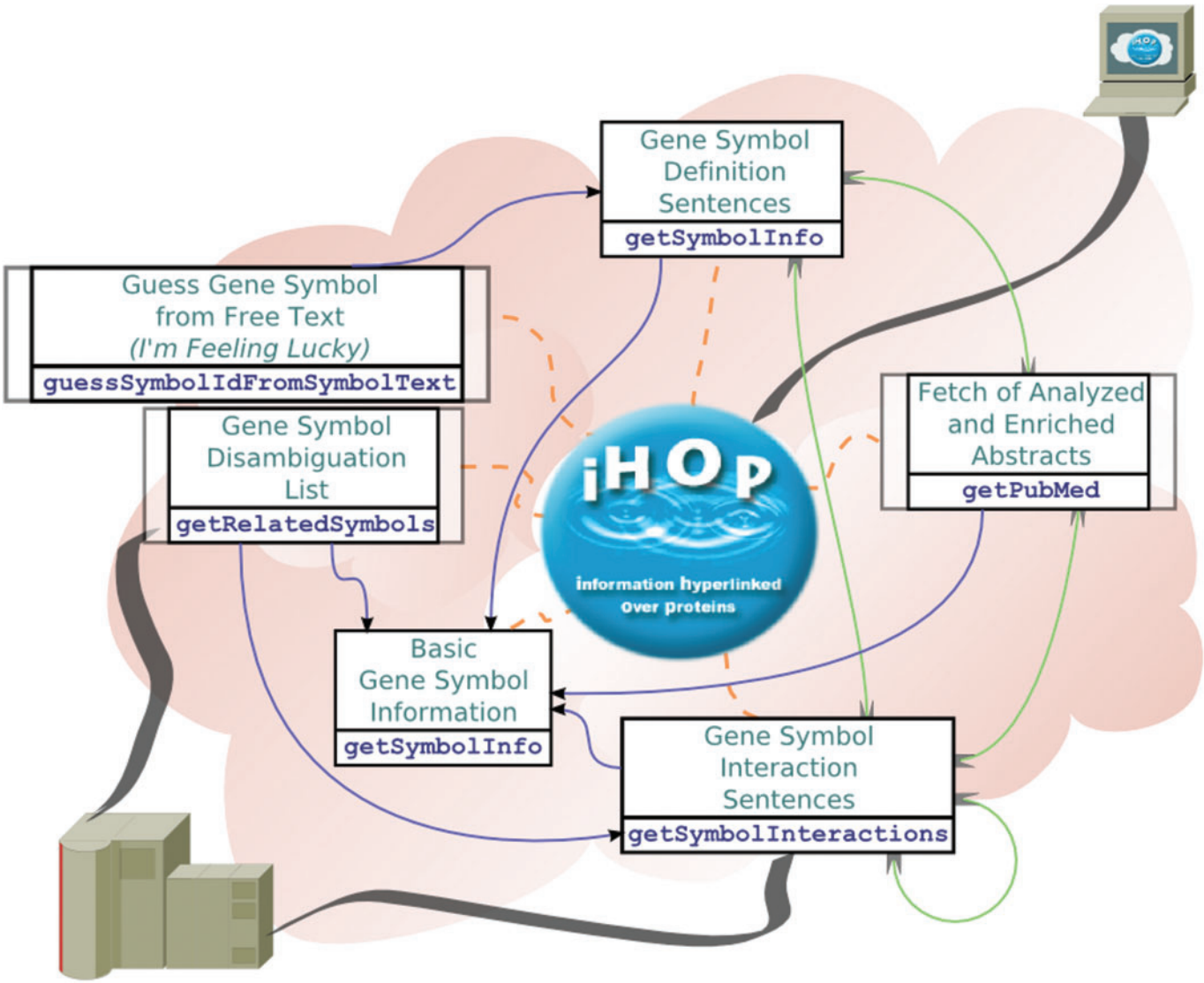

Figure 1. Schema of operations of iHOP web services. Each box is a web service, and double boxes are the recommended starting points for workflows. Links show some suggested flows between services that are useful for workflow building. Green links represent bi-directional flow, and black arrowed blue lines mean uni-directional flow. Grey thick lines illustrate bidirectional external access and orange dashed lines are accesses to the iHOP core (http://www.ihop-net.org/).

Taverna 1.4 and 1.5 are discouraged, because SOAP services results are pruned. Taverna 1.5.1 solves these and other issues, and it is recommended. Older versions, like Taverna 1.3.1, also work, but they have many limitations related to BioMOBY services.

RESULTS AND DISCUSSION

A proof for the functionality, completeness and usefulness of the iHOP web service APIs are a number of collaborative projects that make programmatic use of iHOP content. Table 3 contains a brief list of the projects where iHOP web services have been used, and Figure 3 shows a CARGO (Cases *et al.*, submitted to NAR-WEB 2007) widget using information provided by iHOP CGI-XML services.

In the context of the use of iHOP as a web service it is necessary to be aware of the current limitations of biological text mining. BioCreAtIvE (12) and other blind community assessments (13) have clearly shown that name identification and in particular matching gene/name in the literature with the corresponding database entries is a hard problem and the best systems are still far from perfect (14). Our own evaluation of iHOP in 2005 (15) shows that in model organisms the average precision is around 94% and the recall around 87%. Even if the inclusion of additional refinements and dictionaries is producing continuous progress the poor adhesion of the community to naming standards (16) will continue creating problems in this area.

Other obvious limitations of iHOP and all other current text mining systems are imposed by the limited availability of full text sources [main reason for the common use of

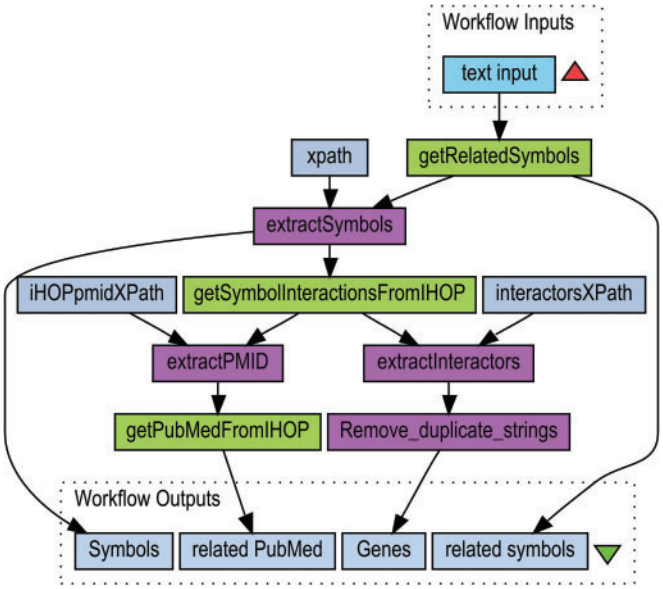

Figure 2. This is a Taverna workflow diagram which takes as input some free text (e.g. 'breast cancer'). The workflow fetches the gene and protein symbols related to the input free text, and it returns those symbols, their synonyms and all the abstracts with sentences where the symbols are showing interaction evidences with other protein or gene symbols.

abstract collections (17)] and the still limited possibilities to incorporate effective Natural Processing Techniques for the extraction of additional features from biomedical text. A more detailed description of the status of this fast developing field can be found in (18–20).

Despite these general limitations in the field, the iHOP web interface has become popular among biologists searching for information about the function and relation of the genes and proteins of their interest. To our knowledge, iHOP is the only large-scale text mining resource in biology that is offered as an open web service, we therefore, expect that the novel possibilities described in this work will contribute to the use of iHOP as part of numerous high-throughput analysis environments.

AVAILABILITY

Information relevant to developers, like detailed documentation of the iHOP web service XML file format, the URLs required to invoke the REST API, the WSDL document describing SOAP services and usage examples in Perl and Taverna are available at http://www.ihop-net.org/UniPub/iHOP/webservices/.

Table 2. Logical iHOP web services functionalities. These functionalities and the web services which implement them are focused on automation, so almost all functionalities have more than one input type. So, depending on the API model, some of these functionalities have been implemented more than once, based on each one of the possible input types.

Functionality	Inputs	Results
`getRelatedSymbols`	Free text (e.g. P53, breast cancer or BRCA2), and an optional NCBI TaxID.	A list of the possible iHOP identified gene or protein symbols, related to the input.
`getSymbolsFromReference`	A biological database reference (e.g. UniProt accession, NCBI GENE).	The iHOP identified gene or protein symbol related to the input.
`guessSymbolIdFromSymbolText`	Free text, and an optional NCBI TaxID.	The iHOP identified gene or protein symbol related to the input, chosen by naïve heuristics.
`guessSymbolIdFromReference`	A biological database reference	The iHOP identified gene or protein symbol related to the input.
`getSymbolInfo`	Free text, and an optional NCBI TaxID,or an iHOP gene or protein symbol ID,or a biological database reference.	The information available at the iHOP server about the iHOP identified gene or protein related to the input (name, organism, database references, synonyms, etc.). When free text is used, naïve heuristics have been used to choose the iHOP identified gene.
`getSymbolDefinitions`	Free text, and an optional NCBI TaxID,or an iHOP gene or protein symbol ID,or a biological database reference.	The abstract sentences available at the iHOP server which uniquely define the iHOP identified gene or protein related to the input, along with their score, iHOP abstract ID, journal impact, etc. When free text is used, naïve heuristics have been used to choose the iHOP symbol.
`getSymbolInteractions`	Free text, and an optional NCBI TaxID,or an iHOP gene or protein symbol ID,or a biological database reference.	The abstract sentences available at the iHOP server which show evidences about interactions between the iHOP identified gene or protein symbol related to the input and other iHOP symbols, along with their score, iHOP abstract ID, journal impact, etc. When free text is used, naïve heuristics have been used to choose the iHOP symbol.
`getPubMed`	A PubMed PMID,or and iHOP abstract ID.	iHOP analysed and enriched PubMed abstract associated to the input. All the abstract sentences are annotated, focusing on remarkable sentence elements (verbs, nouns, adjectives, gene or protein symbols, etc.).

Table 3. Projects where iHOP web services have been (or are being) used

Project	Description	URL/Reference
ORIEL	The CGI-XML API was developed in the context of this project, and it was integrated with other biological information resources.	http://www.oriel.org/
DIAMONDS	The iHOP SOAP interface was developed and applied to the dynamic extraction of proteins related with cell cycle in various genomes with special emphasis in Arabidopsis proteins. This information was used as input for the modelling approaches developed by the partners of this project.	http://www.sbcellcycle.org/
ECID COMBIO ENFIN (Enabling Systems Biology) Network of Excellence CARGO	These projects and their tools (e.g. ENFIN Spindle proteins tool, CARGO framework) are currently using these iHOP web service APIs in different biological and technical contexts.	http://www.pdg.cnb.uam.es/ecid/ http://somosierra.cnb.uam.es/Servers/COMBIO/ http://www.enfin.org/ http://www.pdg.cnb.uam.es/ENFIN/index_spindle.php http://cargo.bioinfo.cnio.es/
INB	BioMOBY iHOP web services were funded by the Spanish National Bioinformatics Institute (INB), and published on the INB specific MOBY repository, integrated in the INB curated bioinformatics object ontology. The services will also be made available in the central MOBY repository of Canada.	http://www.inab.org/

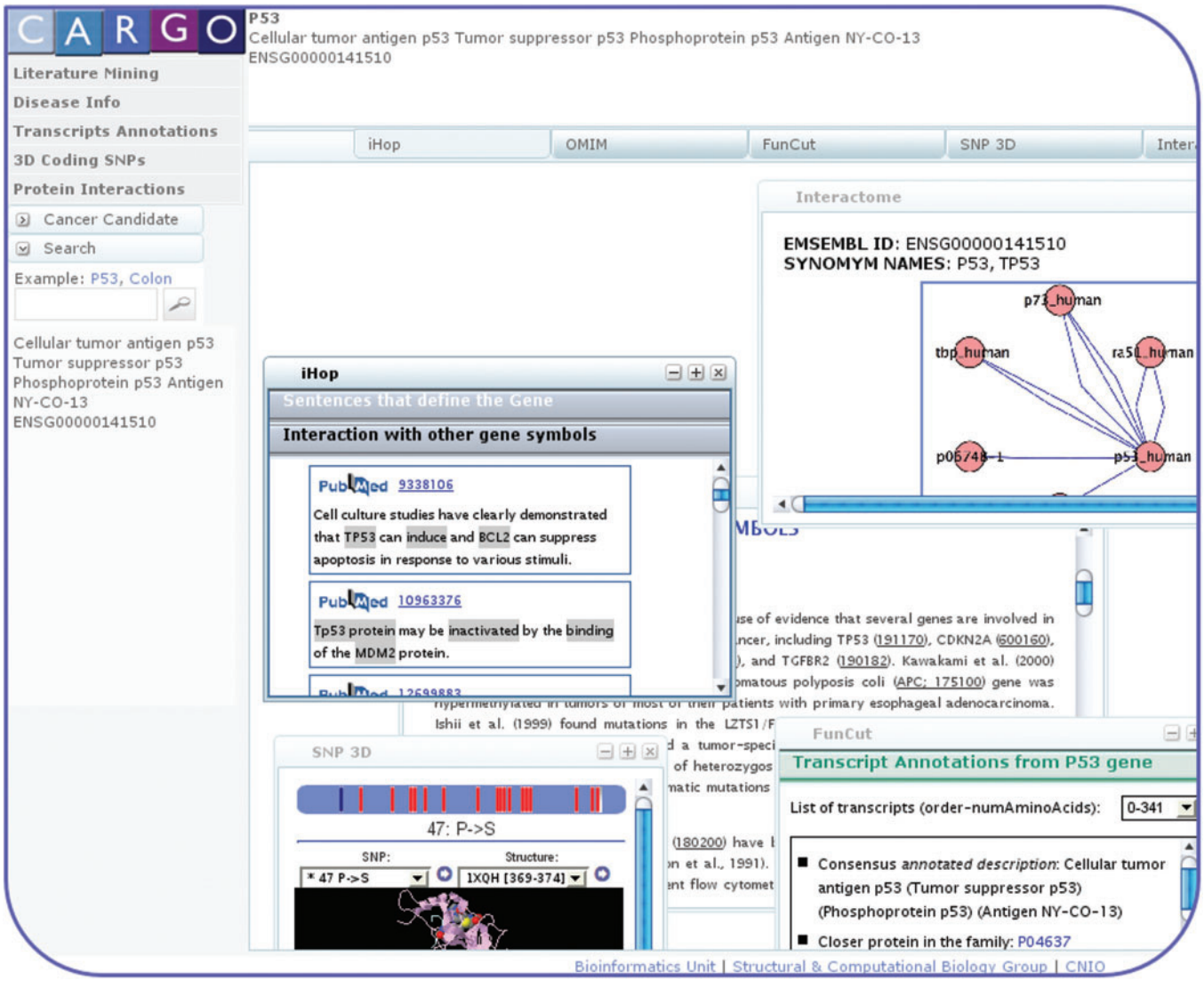

Figure 3. This is a snapshot of the CARGO framework, showing information about P53. The iHOP widget shows sentences with evidences of some relationship between P53 and other genes.

ACKNOWLEDGEMENTS

Funding to pay the Open Access publication charges for this article was provided by ENFIN Network of Excellence (LSHG-CT-2005-518254).

Conflict of interest statement. None declared.

REFERENCES

1. Hoffmann,R. and Valencia,A. (2004) A gene network for navigating the literature. *Nat. Genet.*, **36**, 664–664.
2. Good,B.M., Kawas,E.A., Kuo,B.Y. and Wilkinson,M.D. (2006) iHOPerator: User-scripting a personalized bioinformatics Web, starting with the iHOP website. *BMC Bioinformatics*, **7**, 534.
3. Wheeler,D.L., Barrett,T., Benson,D.A., Bryant,S.H., Canese,K., Chetvernin,V., Church,D.M., DiCuccio,M., Edgar,R. *et al.* (2007) Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res.*, **35**, D5–D12.
4. Labarga,A., Pilai,S., Valentin,F., Anderson,M. and Lopez,R. (2005) Web services at EBI. *EMBnet.news*, **11**, 18–23.
5. Kreger,H. (2001) *Web Sevices Conceptual Architecture (WSCA) 1.0*. IBM Software Group.
6. Fielding,R.T. (2000) *Architectural Styles and the Design of Network-based Software Architectures*. Doctoral dissertation, University of California, Irvine.
7. Dowell,R.D., Jokerst,R.M., Day,A., Eddy,S.R. and Stein,L. (2001) The distributed annotation system. *BMC Bioinformatics*, **2**, 7.
8. Wilkinson,M.D. and Links,M. (2002) BioMOBY: an open source biological web services proposal. *Brief. Bioinform.*, **3**, 331–341.
9. Wilkinson,M., Schoof,H., Ernst,R. and Haase,D. (2005) BioMOBY successfully integrates distributed heterogeneous bioinformatics Web Services. The PlaNet exemplar case. *Plant Physiol.*, **138**, 5–17.
10. Oinn,T., Addis,M., Ferris,J., Marvin,D., Senger,M., Greenwood,M., Carver,T., Glover,K. and Pocock,M.R. (2004) Taverna: a tool for the composition and enactment of bioinformatics workflows. *Bioinformatics*, **20**, 3045–3054.
11. Hull,D., Wolstencroft,K., Stevens,R., Goble,C., Pocock,M.R., Li,P. and Oinn,T. (2006) Taverna: a tool for building and running workflows of services. *Nucleic Acids Res.*, **34**, W729–W732.
12. Hirschman,L., Yeh,A., Blaschke,C. and Valencia,A. (2005) Overview of BioCreAtIvE: critical assessment of information extraction for biology. *BMC Bioinformatics*, **6**(Suppl. 1), S1.
13. Jin-Dong,K., Ohta,T., Tsuruoka,Y., Tateisi,Y. and Collier,N. (2004) Introduction to the bio-entity recognition task at JNLPBA. In *Proceedings of the International Workshop on Natural Language Processing in Biomedicine and its Applications (JNLPBA-04)*, 70–75.
14. Hirschman,L., Colosimo,M., Morgan,A. and Yeh,A. (2005) Overview of BioCreAtIvE task 1B: normalized gene lists. *BMC Bioinformatics*, **6**(Suppl. 1), S11.
15. Hoffmann,R. and Valencia,A. (2005) Implementing the iHOP concept for navigation of biomedical literature. *Bioinformatics*, **21**(Suppl. 2), ii252–ii258.
16. Tamames,J. and Valencia,A. (2006) The success (or not) of HUGO nomenclature. *Genome Biol.*, **7**, 402.
17. Schuemie,M.J., Weeber,M., Schijvenaars,B.J., van Mulligen,E.M., van der Eijk,C.C., Jelier,R., Mons,B. and Kors,J.A. (2004) Distribution of information in biomedical abstracts and full-text publications. *Bioinformatics*, **20**, 2597–2604.
18. Krallinger,M., Erhardt,R.A. and Valencia,A. (2005) Text-mining approaches in molecular biology and biomedicine. *Drug Discov. Today*, **10**, 439–445.
19. Hoffmann,R., Krallinger,M., Andres,E., Tamames,J., Blaschke,C. and Valencia,A. (2005) Text mining for metabolic pathways, signaling cascades, and protein networks. *Sci. STKE*, **2005**, 21.
20. Krallinger,M. and Valencia,A. (2005) Text-mining and information-retrieval services for molecular biology. *Genome Biol.*, **6**, 224.