



University of Pennsylvania
ScholarlyCommons

Departmental Papers (CIS)

Department of Computer & Information Science

July 2005

Selecting biomedical data sources according to user preferences

Sarah Cohen-Boulakia

University of Pennsylvania, sarahcb@seas.upenn.edu

Severine Lair

Institut Curie

Nicolas Stransky

Institut Curie

Stephane Graziani

Isoft

Francois Radvanyi

Institut Curie

See next page for additional authors

Follow this and additional works at: http://repository.upenn.edu/cis_papers

Recommended Citation

Sarah Cohen-Boulakia, Severine Lair, Nicolas Stransky, Stephane Graziani, Francois Radvanyi, Emmanuel Barillot, and Christine Froidevaux, "Selecting biomedical data sources according to user preferences", . July 2005.

Postprint version. Published in *Bioinformatics*, Volume 20 Supplement 1, August 2004, pages i86-93.

Publisher URL: <http://dx.doi.org/10.1093/bioinformatics/bth949>

NOTE: At the time of publication, author Sarah Cohen-Boulakia was affiliated with the Université Paris-Sud. Currently February 2007, she is a faculty member in the Department of Bioengineering at the University of Pennsylvania.

This paper is posted at ScholarlyCommons. http://repository.upenn.edu/cis_papers/298

For more information, please contact libraryrepository@pobox.upenn.edu.

Selecting biomedical data sources according to user preferences

Abstract

Motivation: Biologists are now faced with the problem of integrating information from multiple heterogeneous public sources with their own experimental data contained in individual sources. The selection of the sources to be considered is thus critically important.

Results: Our aim is to support biologists by developing a module based on an algorithm that presents a selection of sources relevant to their query and matched to their own preferences. We approached this task by investigating the characteristics of biomedical data and introducing several preference criteria useful for bioinformaticians. This work was carried out in the framework of a project which aims to develop an integrative platform for the multiple parametric analysis of cancer. We illustrate our study through an elementary biomedical query occurring in a CGH analysis scenario.

Keywords

bioinformatics, biological databases, metadata, user preferences

Comments

Postprint version. Published in *Bioinformatics*, Volume 20 Supplement 1, August 2004, pages i86-93.

Publisher URL: <http://dx.doi.org/10.1093/bioinformatics/bth949>

NOTE: At the time of publication, author Sarah Cohen-Boulakia was affiliated with the Université Paris-Sud. Currently February 2007, she is a faculty member in the Department of Bioengineering at the University of Pennsylvania.

Author(s)

Sarah Cohen-Boulakia, Severine Lair, Nicolas Stransky, Stephane Graziani, Francois Radvanyi, Emmanuel Barillot, and Christine Froidevaux

Selecting Biomedical Data Sources according to User Preferences

Sarah Cohen Boulakia¹, Séverine Lair^{2,3}, Nicolas Stransky²,
Stéphane Graziani⁴, François Radvanyi², Emmanuel Barillot³ and
Christine Froidevaux^{1,*}

¹Laboratoire de Recherche en Informatique (LRI), CNRS UMR 8623, Université Paris-Sud, F-91405 Orsay Cedex, ²CNRS UMR 144, ³Service de Bioinformatique, Institut Curie, 26 rue d'Ulm, F-75248 Paris Cedex 05, and ⁴Isoft, Chemin de Moulon, F-91190 Gif-sur-Yvette, France

ABSTRACT

Motivation: Biologists are now faced with the problem of integrating information from multiple heterogeneous public sources with their own experimental data contained in individual sources. The selection of the sources to be considered is thus critically important.

Results: Our aim is to support biologists by developing a module based on an algorithm that presents a selection of sources relevant to their query and matched to their own preferences. We approached this task by investigating the characteristics of biomedical data and introducing several preference criteria useful for bioinformaticians. This work was carried out in the framework of a project which aims to develop an integrative platform for the multiple parametric analysis of cancer. We illustrate our study through an elementary biomedical query occurring in a CGH analysis scenario.

Keywords: biomedical data source, databases integration, interoperation, metadata, user preferences.

Availability: <http://www.lri.fr/~cohen/dss/dss.html>

Contact: cohen@lri.fr, chris@lri.fr

1 INTRODUCTION

With the increasing amount of disparate biomedical data, there is now a clear need for interoperability between sources in bioinformatics. Biologists are now faced with the problem of integrating relevant information from multiple heterogeneous public sources (e.g. changes in genomic DNA, presence of various protein modifications etc.) with their own experimental data (e.g. mRNA and protein levels etc.) contained in individual sources. The main goal of an integration system is to offer transparent access to data held in multiple disparate sources via a single interface. Biological integration systems should not try to replace human experts, but should instead facilitate data interpretation, and increase efficiency making it possible to interact with the sources, resulting in cooperative integration. An automatic module, guiding the user in the choice of the sources to be accessed, would be very useful in this respect.

The module described here was designed in the framework of the European HKIS project¹, which aims to set up an integrative platform supporting biomedical experts in their data-driven experiments and involving biomedical data (especially data used in cancer studies). The global approach of an HKIS user is based on a set of analysis scenarios describing different analysis methodologies and reflecting the expertise of the biologists and health professional partners involved in the project. At each step of a scenario, the user may have to ask questions necessitating the consultation of various sources. The selection of the sources to be considered is thus critically important.

We describe here a module to help the user to choose the sources to be consulted during the querying process. We have designed a data sources selection algorithm (DSS) that takes into account both the query and the user's preferences. The DSS algorithm is related neither to the specific architecture underlying the platform nor to the format of the sources consulted and could therefore be used in other contexts. We demonstrate the utility of DSS by introducing the *BAC augmentation scenario*, which is part of a more general scenario - the CGH scenario - and assessing the biological relevance of the results DSS yields.

We will begin by specifying the biological entities and biomedical sources considered (section 2). We will then present the BAC augmentation scenario, used to illustrate our approach (section 3). The data sources selection algorithm (DSS) is described in section 4, which also contains definitions of several preference criteria. In section 5, we describe an example of how DSS generated paths can be implemented in the HKIS platform. Finally, we compare the module described here with previous work and draw our conclusions (section 6).

¹ <http://www.hkis-project.com/>

2 BIOLOGICAL ENTITIES AND DATA SOURCES

2.1 Biological entities

We present here the unifying model used by HKIS. We do not aim to propose a new complete conceptual model for biological and biomedical data (see Cornell *et al.*, 2003 and Davidson *et al.*, 2000) or a new ontology (see Ben Miled *et al.*, 2003 or Backer *et al.*, 1999), but instead to provide the main biological entities that would be addressed in our application domain, the study of cancer. The biologists involved in the project identified the entities considered to be important. The list of these entities was compiled from a thorough study of the HKIS analysis scenarios. This list includes the main entities of the various sources used in the project. It should be noted that this unifying model differs from a global, complete model in that only the shared biological entities are considered (no exhaustiveness is sought).

We provide a graphical representation of the data model, which may be viewed as a classical semantic network (Hendrix, 1979) in the same spirit as that in the GeneSeek project (Mork *et al.*, 2001). Each node represents an entity in the domain (biological conceptual object). The edges connecting these nodes represent biological relationships between the corresponding entities. If desired, each user can adapt the model according to his or her own needs. Our data model is thus very flexible. Part of the HKIS conceptual model is shown in Figure 1.

In section 4, we will show how this graph of entities together with the graph of data sources presented in the next subsection can be used to guide the querying process.

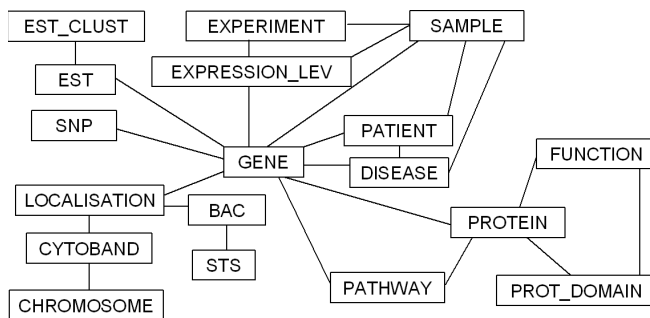


Fig. 1. Graph of entities

2.2 Data sources: content and meta-data

For medical and clinical research, health professionals increasingly rely on correlating their diagnosis with the information available in public-domain or commercial databases (usually accessible via the Internet).

We selected about thirty data banks frequently consulted in studies of cancer, including GenBank², UCSCGenome³ and MapView⁴ for genomic data, GEO⁵ and ArrayExpress⁶ for transcriptomic data and Swiss-Prot and TrEMBL⁷ for proteomic data. For medical research, information is also required concerning diseases, from, for example, OMIM⁸ or GeneCards⁹, and this involves a constant search for the dynamically controlled vocabulary provided by certain biological ontologies, such as Gene Ontology¹⁰.

We carried out a thorough study of the selected data banks. Some of the banks supplied different kinds of information and had to be split into several data sources. For example, GenBank had to be split into three sources: *GenBankS*, corresponding to the Nucleotide section, *GenBankG*, corresponding to the Genome section, and *GenBankP*, corresponding to the Protein section. The MapView bank had to be split into two different sources: *MapViewFish* and *MapView*, corresponding to the two types of clone information provided by MapView (Fish mapping or not).

Each of the selected sources was described at a meta level, based on a framework, the structure of which is described below. We have listed the entities present in each source and indicated the focus of each source. The focus is defined as the entity around which the source is organised. For example, *Swiss-Prot* contains information on the entities PROTEIN, GENE, DISEASE, and FUNCTION and *Swiss-Prot*'s focus is PROTEIN because each *Swiss-Prot* entry corresponds to a protein. The framework also provides information about the quality (degree of reliability) of the entities contained in the source. For example, on a scale of reliability from 1 to 10 (10 being the highest level of reliability), some users may assign a level of 9 and 10 to the GENE and PROTEIN entities of *Swiss-Prot* but levels of only 7 and 8 to these entities in *GenBankP*. Obviously, the quality property is subjective, and its value can be modified by each user.

The metadata of the sources are described in an XML file available from www.lri.fr/~cohen/dss/default.xml. New sources or entities can easily be added and the mapping between the sources and the unifying model easily modified by loading a new XML file.

2.3 Data source links

Although the data banks considered were designed by different research teams in different contexts, and were therefore highly heterogeneous, they are nonetheless related. In

² <http://www.ncbi.nlm.nih.gov/Genbank/index.html>

³ <http://genome.ucsc.edu/cgi-bin/hgGateway>

⁴ http://www.ncbi.nlm.nih.gov/mapview/map_search.cgi

⁵ <http://www.ncbi.nlm.nih.gov/geo/>

⁶ <http://www.ebi.ac.uk/arrayexpress/>

⁷ <http://www.expasy.org/sprot/>

⁸ <http://www.ncbi.nlm.nih.gov/omim/>

⁹ <http://bioinfo.weizmann.ac.il/cards/index.html>

¹⁰ <http://www.geneontology.org/>

particular, banks increasingly frequently refer to each other by means of hypertext links called cross-references. These links may be very useful as they make it possible to obtain additional information concerning a single instance of one entity in a given source by providing access to complementary and more detailed information in other sources. Like entities in data sources, the reliability of cross-references may be variable, depending on whether the cross-references concerned were added manually or generated automatically.

In the HKIS project, we consider that each data source is composed of different parts, one part for each entity contained in the source. We therefore had to introduce another kind of link - *internal links* - used to join entities within a given source. Internal links can be seen as foreign keys in relational databases or, more generally, as a way of obtaining information on one entity from another entity in the same source.

We provide below a graphical representation of the sources and links. Each node represents a data source and is divided with respect to the entities it contains. The focus of each source is indicated in bold typeface. Arrows indicate the links between a given entity in a data source and another entity (in the same source or another source). For the sake of clarity, figure 2 presents only the sources and links required for the example dealt with in section 3. Figure 2 is therefore just a part of the complete graph of sources.

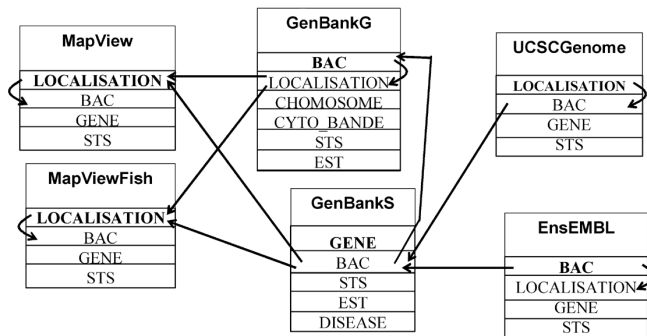


Fig. 2. Graph of sources

3 EXAMPLE

Our example (see figure 3) concerns the process of positioning genomic BACs (Bacterial Artificial Chromosomes) on the draft of the human genome sequence. BACs are used in CGH array experiments as a means of detecting gains and losses in the DNA of tumor samples. This process leads to the definition of lost or gained regions in the genome of tumors, referred to as deletions and amplifications, respectively. It has been shown in many cancers that the deletion of regions containing tumor suppressor genes or the gain of regions containing oncogenes is associated with and may cause tumorigenesis and tumor progression (see Hanahan

and Weinberg, 2000 and Albertson *et al.*, 2003, for a good introduction). CGH array experiments aim to identify new cancer-related genes in the regions lost or gained. It is therefore of the utmost importance to map BAC precisely onto the genome sequence and to compare their positions with those of the genes. This can only be achieved by carrying out thorough searches to identify the position of each BAC as described in public data sources.

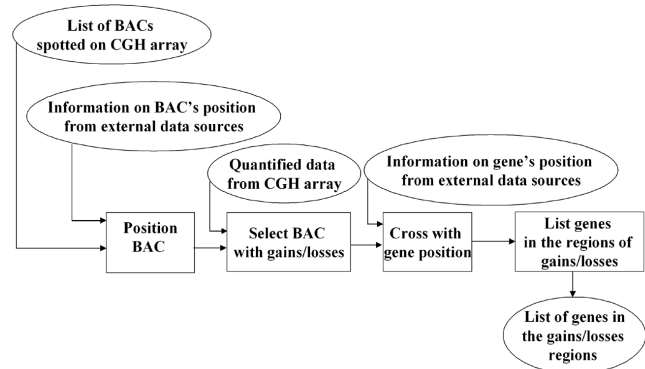


Fig. 3. BAC augmentation scenario

4 DATA SOURCE SELECTION ALGORITHM

4.1 According to the process followed by HKIS biologists

The *DSS* algorithm described below was designed on the basis of the way in which HKIS biologists search for information in different sources.

At each step of an HKIS scenario, the user may ask questions, such as “which are the genes possibly involved in breast cancer?” or “where is the BAC identified by *CTD-2012D15* located?”. The biologist can map the various components of his or her specific queries (e.g. “breast cancer”, “BAC number *CTD-2012D15*”) to higher level biological objects (DISEASE, BAC), corresponding to the entities of the conceptual model introduced in section 2. The underlying entities are GENE and DISEASE for the first query and BAC and LOCALISATION for the second query. Note that a given entity may be present in several sources which give different set of instances.

Once a biologist has chosen the entities for which he or she is seeking information, he or she tries to find a group of sources linked by cross-references that could provide instances of these entities. Each source may offer only instances of some of the entities sought, but the group of sources queried should provide information about all of the entities. It is worth noticing that each group of sources queried may give different sets of results. This is why it is very important to provide the biologist with the opportunity of considering alternative groups of sources.

More precisely, the biologist follows a process consisting of two main stages. The first step involves searching for in-

formation about each of the entities, one by one. In this case, the biologist may follow cross-references to the same entity across several banks, to collect as much information as possible on that entity. He or she will then move on to consider the next entity, and so on. The same source may be consulted several times if it provides information about several entities. The second step involves linking entities by means of cross-references or internal links. The biologist considers all of the possible permutations between entities to ensure that the search is exhaustive.

4.2 Specification and presentation of the DSS algorithm

We present here the DSS algorithm, which provides the list of the sources to be accessed to obtain information about the entities underlying the user's query. The outputs of the DSS algorithm are paths consisting of the parts – *i.e.* *views* – of data sources which concern the underlying entities. In such paths, views of data sources are linked by internal links or cross-references.

Let us introduce the following notations. Let $E = \{e_1, \dots, e_n\}$ be the set of the n nodes of the graph of entities. Let $E_Q = \{e_{q_1}, \dots, e_{q_m}\}$ be the set of entities underlying the user query Q ($E_Q \subseteq E$) and $S = \{s_1, \dots, s_m\}$ be the set of the m nodes of the graph of sources. We will call *src_ent_path* a sequence of pairs $(s, e) \in S \times E$ such that entity e is in source s and such that: if $(s_{i1}, e_{i1}), (s_{i2}, e_{i2})$ are two consecutive pairs then either $s_{i1} = s_{i2}$ and there is an internal link from (s_{i1}, e_{i1}) to (s_{i2}, e_{i2}) , or there is a cross-reference from (s_{i1}, e_{i1}) to (s_{i2}, e_{i2}) in the graph of sources. Intuitively, each pair (s, e) of such a path suggests using a view of the source s over the entity e to collect instances of e . Moreover, the order of pairs in each path indicates the way in which data from sources should be combined.

More precisely, the DSS algorithm builds the set of all the *complete_src_ent_paths* which are the *src_ent_paths* that satisfy the three properties below :

Let $L = \{\text{path}_1, \dots, \text{path}_k, \dots, \text{path}_t\}$.

- (1) Each path of L concerns all of the underlying entities: for each path_k of L , $1 \leq k \leq t$, for each underlying entity e , there exists in path_k (at least) one pair $(s, e) \in S \times E$;
- (2) Each path of L gathers information about the same entity once for all: in a given path, between 2 pairs related to the same entity e , there is no pair related to another entity e' with $e \neq e'$;
- (3) Any pair (s, e) appears at most once in a path of L .

It should be stressed that the paths are not built while searching in the graph of entities because the relationships between the underlying entities in the biological model are not considered. Instead, the paths are built while examining the entities one by one. The algorithm is not a basic search

in the graph of sources either as it is entity-related. Indeed, the DSS algorithm consists of two steps, like the process followed by HKIS biologists. Firstly, the *Ent_Related_paths* procedure builds every entity-related path, that is, every *src_ent_path* in which each pair concerns the same entity. Secondly, the *Rec_Build* procedure recursively builds all the *complete_src_ent_paths*, which are combinations of entity-related paths.

DSS output therefore provides a means of obtaining information about the underlying entities of the user query as a whole, across several biological data sources, by exploiting relationships between entities within sources. The complete algorithm is presented elsewhere (Cohen Boulakia et al., 2004) and it is available for use from www.lri.fr/~cohen/dss/dss.html.

4.3 Back to the example

We illustrate the behavior of the DSS algorithm by studying the query introduced previously “Where is the BAC identified by CTD-2012D15 located?”. Let B and L denote the underlying entities of this query, namely BAC and LOCALISATION, respectively. We consider the set of sources in figure 2 and the entities contained in the sources, as indicated in the figure. In this subsection we provide a few examples of paths generated by DSS.

The first step of DSS involves building the set of Entity-Related paths: $ER(L)$ and $ER(B)$ for LOCALISATION and BAC, respectively. $ER(L)$ contains seven paths including $[(UCSCGenome, L)]$ and $[(GenBankG, L), (MapView, L)]$. These paths suggest querying the view over LOCALISATION in *UCSCGenome* or to follow the cross-reference from the view over LOCALISATION in *GenBankG* to the view over LOCALISATION in *MapView*, as a means of collecting information about LOCALISATION. $ER(B)$ contains 11 paths including $[(UCSCGenome, B)]$, $[(UCSCGenome, B), (GenBankS, B)]$ and $[(UCSCGenome, B), (GenBankS, B), (GenBankG, B)]$.

The second step of the algorithm involves building the set of *complete_src_ent* paths from $ER(B)$ and $ER(L)$, using cross-references and internal links. Thus, the set of answers contains 26 paths including $[(UCSCGenome, L), (UCSCGenome, B)]$, $[(UCSCGenome, B), (GenBankS, B), (GenBankG, B), (GenBankG, L)]$ and $[(UCSCGenome, B), (GenBankS, B), (GenBankG, B), (GenBankG, L), (MapView, L)]$.

4.4 Complexity

The time complexity order of the algorithm is clearly greater than the number of paths generated. The worst case occurs when the graph of sources is complete because all the combinations between entity-related paths are then possible. Nevertheless, we do not assume that each source provides all the entities. In this case, the number of paths built by the algorithm is given by the following formula:

$$C = (nr!) * \prod_{i=1}^{nr} \sum_{k=1}^{nbe_i} A_{nbe_i}^k$$

where nr is the number of underlying entities, and nbe_i is the number of the sources that contain e_i ($1 \leq i \leq nr$, $1 \leq nbe_i \leq m$). In this worst case, time complexity is therefore very high. However, in real applications, we can expect that the number of paths is quite small as far as biologists queries involve only a small number of entities at each step of a scenario. Moreover, in the implementation of the DSS algorithm, the paths are generated immediately.

4.5 Preference criteria

As there may be too many paths, we have introduced into the DSS algorithm the possibility of taking into account user preferences to filter and sort these paths. Other kinds of preference criteria are still being studied and could be incorporated into the algorithm with ease. We show examples of such criteria below.

In section 2, we saw that each data source was focused on one entity and provided information about several entities, and that the reliability of this information was variable. We have also pointed out that the reliability of cross-references should be taken into account. Here, we allow the user to set the reliability level associated with entities in the sources and with links between these sources. We also show how this information can be used to limit path length or to access sources with the aim of obtaining information about their focus only. Thus, in the DSS algorithm, the user may set four kinds of filtering criteria, as indicated in table 1. Let us define the length of a path as the number of cross-references between two different consecutive sources in that path. For example, the lengths of the last three paths in subsection 4.3 are 0, 2 and 3, respectively.

Table 1. Preference criteria

| Topic | Criteria for each path |
|----------------|--|
| Length | Path length does not exceed <code>max_length</code> . |
| Focus | At most <code>max_focus</code> sources are consulted for an entity other than their focus. |
| ReliableSource | At least <code>max_fval(ni)</code> sources of reliability level <code>ni</code> are consulted. |
| ReliableLink | At most <code>max_unreliable_links</code> cross-references are followed. |

We will see in subsection 5.2 how the use of these criteria provides the user with the possibility of considerably reducing the number of paths and sorting them. This point will be illustrated by a concrete example in which filtering reduces the number of paths from 26 to 6.

5 IMPLEMENTATION AND RESULTS

5.1 Implementation of the BAC augmentation scenario

User preferences can be used to decrease the number of paths generated by the DSS algorithm. Nevertheless, the number of paths may still be high. Each path indicates which sources should be accessed and how they should be combined. The results of a path are the instantiated answers provided by the sources to the specific user query. We will show how the results of the paths can be implemented in the HKIS platform.

In the context of the lack of standard characterizing biological data (see *Workshop Report on Bioinformatics-Structures for the future*, 2003), the HKIS platform is an efficient solution to the data access and crossing problem. Thanks to a local cache mechanism, it provides transparent access to any biological data source and makes it possible to cross-check any given source with any other in seconds. As such, and because it is an open integration platform facilitating the integration of tools, the HKIS platform provides an opportunity to test the DSS algorithm rapidly. Note that some of the obtained paths may yield no result because not every data source contains answers to the specific user query. As the HKIS platform is based on ISoft AMADEA data morphing technology¹¹ making it possible to handle large volumes of data in real-time, the cost of studying such paths is very low.

In the HKIS platform biologists can build bioinformatics experimentation processes called scenarios and implemented by dataflows. All dataflows are designed graphically in AMADEA, without programming, and can be easily replayed at any time if necessary, in the same context or in new experimental configurations. We provide below an example of an HKIS dataflow implementing part of the BAC augmentation scenario introduced in section 3.

Figure 4 shows how results of the DSS application can be easily implemented to set up a scenario and obtain the result of any crossing immediately: e.g. the sources used by the different steps of the scenario (*Position BAC*, *Cross with gene position* etc.) were identified by using the DSS algorithm. Thus, note that each path generated by the DSS algorithm could be represented in the platform in the same way. Results for the whole CGH scenario are obtained in less than 10 minutes on a standard PC.

¹¹ http://www.alice-soft.com/html/prod_amadea.htm

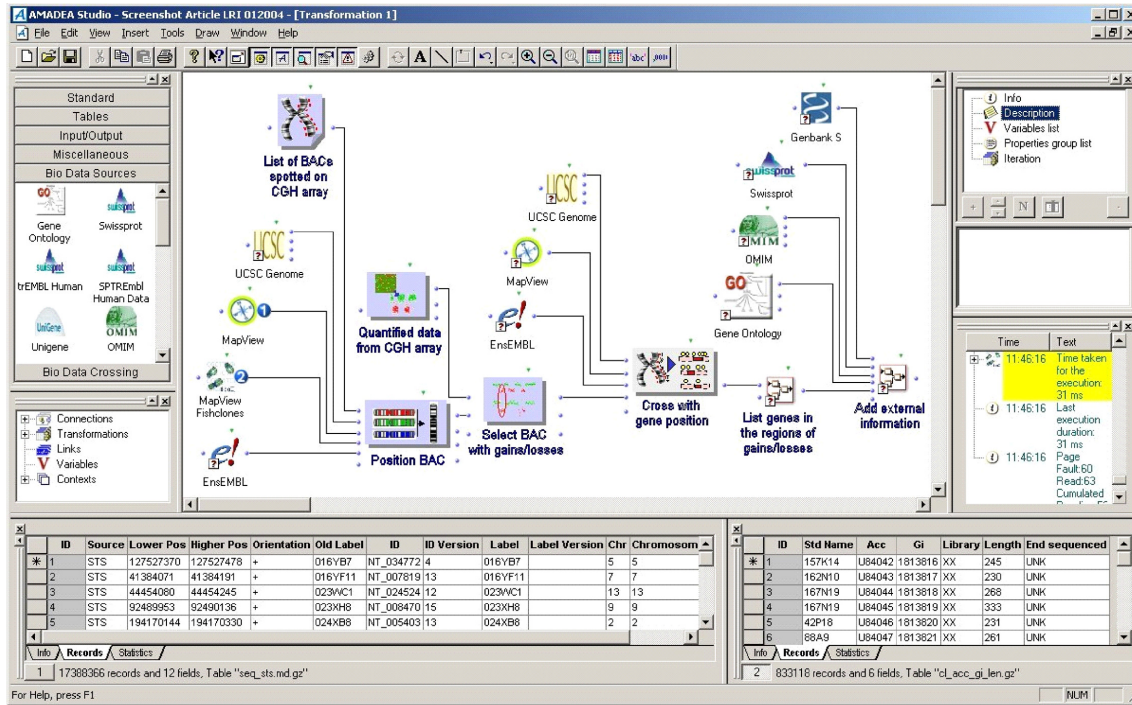


Fig. 4. Implementation in the HKIS Studio of the scenario described in figure 3.

AMADEA therefore provides an elegant way of obtaining results for an instantiated path by combining information from the data sources given by the DSS algorithm.

5.2 Analysis of the biological significance of the results

We also assessed the significance of the results given by the paths generated by the algorithm. Our goal is to highlight the differences that may appear depending on the path considered, showing how important it is to obtain several paths. We assume, by example, that the user assigns to every entity of the data sources *MapView*, *MapViewFish*, *UCSCGenome*, *GenBankS* and *GenBankG*, a level of reliability of 6, 9, 9, 4, and 4, respectively. Moreover, we assume that the user does not really know the source *ensEMBL* and therefore assigns to every entity of this bank a low level of reliability, such as 2. The user may also consider links from *GenBankS* to be unreliable because they are completely automatic.

Now, we consider that the user has indicated the following selection criteria: no unreliable links or sources with a reliability level less than three are accepted and only one source with a reliability level of four is accepted per path. We suppose that the user has also indicated that results should be sorted by taking into account two criteria, length and then reliability, with greater length and higher reliability

preferred. Based on these criteria, the algorithm yields only the six paths given below.

- (1)[(*MapViewFish*,L),(*MapViewFish*,B)],
- (2)[(*UCSCGenome*,L),(*UCSCGenome*,B)],
- (3)[(*MapView*,L),(*MapView*,B)],
- (4)[(*GenBankG*,L),(*MapViewFish*,L),(*MapViewFish*,B)],
- (5)[(*UCSCGenome*,L),(*UCSCGenome*,B),(*GenBankS*,B)],
- (6)[(*GenBankG*,L),(*MapView*,L),(*MapView*,B)].

In the following, we compare the results given by these six paths for the BAC identified by CTD-2012D15. Queries were made on January 5 2004. Firstly, the various paths indicate different locations for this BAC. According to paths (3), (4) and (6), the BAC is located on chromosome X, whereas paths (1), (2) and (5) indicate that it is located on chromosome 11. Faced with this conflicting information, the user may be guided by the confidence he has in entities from sources. Here, as the reliability levels of (*MapView*,L), (*MapView*,B) and (*GenBankG*,L) are lower than the reliability levels of (*MapViewFish*,L), (*MapViewFish*,B), (*UCSCGenome*,L) and (*UCSCGenome*,B), the user is likely to consider it more probably that BAC CTD-2012D15 is located on chromosome 11.

Secondly, it should be stressed that path (5) complements the answers given by path (2), rendering them more precise. Indeed, in path (2), *UCSCGenome* provides information about all the entities of the query - BAC and LOCALISATION - by indicating that CTD-2012D15 is located on the 11q22.3 band of chromosome 11, and giving four cross-references to

GenBankS. Path (5) suggests that the user should follow these links to obtain more precise information on BAC-end sequences.

Finally, the information provided by sources depends on the way the source is reached. For example, *GenBankS*, when reached from *UCSCGenome* in path (5), localizes the BAC to chromosome 11 in the entries B58231, B58232, B666573 and AQ225240 whereas *GenBankS*, when directly accessed, returns the entry NT_025319.14, which localizes the BAC to chromosome X.

6 DISCUSSION

Several approaches and systems have been proposed to deal with the problem of integrating data from life science sources. Examples of such systems include SRS (Etzold *et al.*, 1996), DiscoveryLink (Haas *et al.*, 2001), Tambis (Backer *et al.*, 1999) and myGrid (Stevens *et al.*, 2003), all of which are based on different kinds of architecture. As the DSS algorithm is independent of any architecture and of any source format, it could be used in any integration system. For example, in SRS, the DSS algorithm could help the user to choose which data sources to access. DSS informs the SRS user of all the cross-reference paths that may provide answers to the query, enabling the user to choose between these paths before instantiated results are retrieved.

The biologist's preferences were taken into account in the Tambis mediator as early as 1999 and this aim was strengthened further in the recent myGrid¹² project. Mygrid is one of the largest bioinformatics projects aiming to develop the necessary infrastructural middleware for use over existing Web services & Grid infrastructure to support scientists in making use of complex, widely distributed resources. However, none of these projects proposes a well-identified module for handling these preferences in the process of selecting sources.

Our work was carried out in the same spirit as the projects of Mork *et al.*, 2002 and Lacroix *et al.*, 2003 which addressed the problem of building source paths. Mork *et al.* introduced the query language PQL, which is used in the Biomediator data integration project. This language is based on XML and can be used to express high-level constraints governing the construction of complex paths across XML sources. Lacroix *et al.* reviewed certain challenges in the exploration of life science sources, and illustrated ways of exploring the search space of links between biological data sources. Nevertheless, neither of these solutions provide a means of obtaining the whole combination of data sources to be accessed according to the user query. Instead, they directly provide the complete list of instantiated results from sources. Thus, as in SRS, no filtering occurs and the paths are not sorted before the results are obtained.

Lastly, we compare our study with other studies on meta-data. The work of Cheung *et al.*, 1998, Köhler *et al.*, 2003 and the Medical Core Metadata Project¹³ aimed to describe the content of life science sources (the complex biological entities) rather than to propose quality criteria specific to biomedical data.

We will now sum up the key ideas behind the biomedical data sources selection module presented. This module is based on the new DSS algorithm, which was designed to reflect the way in which HKIS biologists search for information in public data sources. We also carried out a thorough study of the content of and the relationships between about thirty life science data sources. The algorithm is available for use from www.lri.fr/~cohen/dss/dss.html. This current implementation should be considered as work in progress because we are studying new kinds of preference criteria to be taken into account in our algorithm and are developing new menus for the user interface to facilitate the addition and configuration of new sources or new entities.

The main advantages of this module can be summarized as follows:

- The user does not need to know a priori which data sources can answer his query because the sources are selected automatically according to the underlying entities of his query.
- The module yields, by means of a set of data source paths, a list of all the possible ways of obtaining information about the underlying entities of the query. The different paths obtained can be used, in particular, to exploit the complementary aspects of the data sources. The user also knows the order in which to combine the data from these sources.
- User preferences are taken into account, making it possible to filter and to sort the various paths obtained. Thus, the user can be guided in analysis of the collected results. This is critically important if the data from the different sources conflict.

We have shown how useful this module may be by highlighting the biological relevance of the alternative paths obtained, through the example of the BAC augmentation scenario used in the CGH analysis scenario.

¹² <http://mygrid.man.ac.uk/>

¹³ <http://medir.ohsu.edu/~metadata/>

ACKNOWLEDGEMENTS

We are particularly grateful to Bastien Rance and Nicolas Lebas for their implementation of the algorithm and to the HKIS partners for fruitful discussions. We acknowledge Céline Rouveïrol for her valuable comments. We also thank anonymous reviewers for their pertinent suggestions.

This work is supported in part by the European Project HKIS IST-2001-38153.

REFERENCES

- Albertson,D.G., Collins,C., McCormick,F. and Gray,J.W. (2003) Chromosome aberrations in solid tumors. *Nat. Genet.*, **34**, 369-376.
- Backer,P.G., Goble,C., Bechhofer,S., Paton,N.W., Stevens,R. and Brass,A. (1999) An ontology for bioinformatics applications. *Bioinformatics*, **15**, 510-520.
- Ben Miled,Z., Webster,Y., Li,N. and Liu,Y. (2003) An ontology for the semantic integration of life science web databases. *Int. Jour. of Coop. Inf. Sys., IJCIS*, **12**, 275-294.
- Cheung,K., Nadkarni,P.M., and Shin,D. (1998) A metadata approach to query interoperation between molecular biology databases. *Bioinformatics*, **14**, 486-497.
- Cohen Boulakia,S., Froidevaux,Ch., and Lair,S. (2004) Interrogation de sources biomédicales : gestion des préférences de l'utilisateur. *Proc. of EGC'2004, Extraction et Gestion des Connaissances*, 53-64.
- Cornell,M., Paton,N.W., Hedeler,C., Kirby,P., Delneri,D., Hayes,A., and Oliver,S.G. (2003) GIMS: An integrated data storage and analysis environment for genomic and functional data. *Yeast*, **20**,1291-1306.
- Davidson,S.B., Crabtree,J., Runk,B., Schug,J., Tannen,V., Overton,G.C. and Stoeckert,C.J. (2000) K2/Kleisli and GUS : Experiments in integrated access to genomic data sources. *IBM Systems Journal*, **40**, 512-531.
- Etzold,T., Ulyanov,A. and Argos,P. (1996) SRS: information retrieval system for molecular biology data banks. *Methods Enzymol*, **266**, 114-128.
- Haas,L.M., Schwarz,P.M., Kodali,P., Kotlar,E. Rice,J.E. and Swope,W.C. (2001) DiscoveryLink: A system for integrated access to life sciences data sources. *IBM Systems Journal*, **40**, 263-269.
- Hanahan,D. and Weinberg,R.A. (2000) The Hallmarks of Cancer. *Cell*, **100**, 57-70.
- Hendrix,G. (1979) Encoding Knowledge in Partitioned Networks. In *Associative Networks*, ed. Findler,N., 51-92. New York: Academic Press.
- Köhler,J., Philippi,S. and Lange,M. (2003) SEMEDA: ontology based semantic integration of biological databases. *Bioinformatics* **19**, 2420-2427
- Lacroix,Z. Naumann,F., Raschid,L. and Vidal,M.E. (2003) Exploring Life Science Data Sources. *Proc. of IJCAI-03 Workshop on Information Integration on the Web (IIWeb-03)*, 203-210.
- Mork,P., Halevy,A., and Tarczy-Hornoch,P. (2001) A model for data integration systems of biomedical data applied to online genetic databases. *Proc. AMIA Symp*, 473-377.
- Mork,P., Shaker,A., Halevy,A., and Tarczy-Hornoch,P. (2002) PQL: A declarative query language over dynamic biological schemata. *Proc. AMIA Symp*, 533-537.
- Stevens,R.D., Robinson A.J., and Goble,C.A. (2003) myGrid: personalised bioinformatics on the information grid. *Bioinformatics*, **19**, 302i-304i.
- Workshop Report on "Bioinformatics-Structures for the future" (2003):http://imgt.cines.fr/textes/PDF/Bioinformatics/bioinf_workshoprpt_2003_06_30_final.pdf