

# A Semantic Mediation Approach for Problems in Computational Molecular Biology

M. Chagoyen<sup>1</sup>, M.E. Kurul<sup>2</sup>, P.A. De-Alarcon<sup>3</sup>, S. Santini<sup>4</sup>, B. Ludäscher<sup>2</sup>, J.M. Carazo<sup>1</sup>, A. Gupta<sup>2</sup>

<sup>1</sup>*Biocomputing Unit, Centro Nacional de Biotecnología, Madrid, Spain*

<sup>2</sup>*San Diego Supercomputer Center, U.C. San Diego, San Diego CA, USA*

<sup>3</sup>*Integromics S.L., Parque Científico de Madrid, Madrid, Spain*

<sup>4</sup>*Biomedical Bioinformatics Coordination Center, U.C. San Diego, San Diego CA, USA*  
*monica.chagoyen@cnb.uam.es*

## Abstract

*As great amount of data are being produced and accumulated in Life Sciences, information access and consequent integration in analytical and computational methods become critical issues in order to conduct research. In this work we describe how the synergic combination of a high-level programmable planner engine and formal semantics are suitable to resolve many information integration problems in molecular biology.*

## 1. Introduction

In many different fields of science, a *computational* method of problem solving needs to interleave *information access* and *algorithm execution* by putting them together in a problem-specific “workflow”. In a complex domain like molecular biology, such a workflow often involves executing a number of algorithms where each algorithm may require access to multiple information sources. To translate this requirement into reality, one needs to develop a general-purpose system from scientific workflows that contains an information management component providing access to a wide variety of information sources (local and public databases, web sites, computational resources). Equally important is the need to ensure that the different pieces of information acquired are *semantically compatible*.

## 2. Scientific computation

In this work, we present two system components that are parts of a larger computation infrastructure to execute a scientific workflow. The first is a *programmable integrator* that can access information from multiple, heterogeneous sources in a consorted manner. The second component is a *semantic mediator* that can integrate information from different information sources by bridging over the semantic differences among them.

## 2.1. A motivating example

The study was conducted in [1] to assess the conservation of residues at protein-protein interfaces compared with other residues on the protein surface. To accomplish this, a set of solved protein structures is chosen, from which a group of functionally equivalent homologues is identified and aligned multiply.

The requirements for choosing the structures can be translated into a set of steps of information access to different Molecular Biology data sources (PDB, CATH, SwissProt, etc.) and post-processing. We point out the repeated occurrence of *search-collect-filter* patterns in this workflow, and that grouping structures by protein has the standard group-by semantics in database systems – thus, for every value of protein, we need to collect the structures that belong to the protein. The *semantic gap* here is that this step assumes we can evaluate the “equality” between two proteins, without specifying *how* such an equality predicate will actually be evaluated.

## 2.2. A language for a programmable integrator

PLAN is a simple XML-based language – it assumes that every data source is represented in XML. This assumption is reasonable because, today there are many software tools that convert web pages to XML, and present object-relational databases as XML objects. For the work described, we used the Minerva wrapper for web pages [2] and our own wrapper [8] for commercial relational DBMS – however, the choice of wrappers does not affect the PLAN language and the PLAN execution model. The execution of a PLAN program produces the result as an XML document. The execution model of PLAN is based on a stack engine where each of the data sources to be processed is pushed to the main execution stack by its unique link and executed one by one with the query assigned to it. Queries are modeled after the XQuery language [11], the query language for XML data standardized by the World Wide Web Consortium.

Besides web sources as illustrated in the resolved example, PLAN also access relational databases.

### 2.3. A declarative integrator for domain Modeling

PLAN is almost an “assembly-level” language to control information fetching, filtering and result construction of an execution engine. Very often, it is more convenient to approach the information access problem from a higher level, where the user would want to specify the retrieval and filtering task (over heterogeneous sources) in a more declarative fashion. The high-level request would then need to be automatically translated into a series of lower-level requests for which a PLAN program would have to be generated. The middleware that provides these transformations is called a *mediator* [13]. A mediator enables a user to access multiple information systems as though they were a single system with a uniform way to retrieve information and perform computations.

An additional difficulty of current information integration efforts lies in the semantic differences between sites. A full exploitation of data integration systems requires the reconciliation of information at the semantic level. Molecular Biology is a very rich discipline in semantics, due not only to the existence of diverse types of data (DNA and protein sequences, structures, expression data, etc.), but also to the broad scope and complex nature of the molecular mechanisms of the cell.

The need to bring in inter-source semantics has been underscored in a number of research efforts in biological information integration [15, 17]. Our semantic mediator is an adaptation from [17, 23, 8], generalizing from the original Neuroscience application to the domain of molecular biology. In the most general case, every source has an object-oriented schema (containing a class hierarchy and inter-class relationships) for the data, a terminological graph (containing the ontology of terms and inter-term relationships), and a set of references (from the terms in the source to terms known to the mediator) called ontological grounding of the source. The central technique is to define a mapping procedure that relates the classes, terms, and relationships of the sources to those known to the mediator. For example, mapping a subset of the Swiss-Prot keywords to the mediator term “neurodegenerative\_disease” can be done by

```
map swp_diseases (keyword,
neurodegenerative_disease)
IF keyword:concepts(SwissProt)
AND My_Disease:concepts(Mediator)
AND keyword in {Huntington's disease,
Alzheimer's disease, ...}
```

### 2.4. Combining Declarative and Procedural Techniques for Problem Solving

The difference between the two approaches of information integration presented so far is that in the first, the exact procedure of retrieving a data object is specified, whereas in the second, one only has to specify the query in a high-level query language. In one mode of operation, a high-level query in the mediator’s language may produce a low-level execution plan specifiable in the PLAN language and executable by the PLAN execution engine. Since it might seem that a high-level query would completely hide away the first, we show the need for making parts of the PLAN explicit even in specifying the high-level mediator query.

### References

- [1] W.S.J. Valdar and J.M. Thornton, “Protein–Protein Interfaces: Analysis of Amino Acid Conservation in Homodimers” *Proteins*, 2001, 42:108–124
- [2] V. Crescenzi and G. Mecca. “Grammars Have Exceptions”. *Information Systems*, 1998, 23:539-565
- [8] A. Gupta, B. Ludäscher, M.E. Martone et al *BIRNM*: “A Semantic Mediator for Solving Real-World Neuroscience Problems”. *Proc. ACM SIGMOD*, 2003
- [11] XQuery 1.0: An XML Query Language W3C Working Draft. <http://www.w3.org/TR/xquery/>
- [13] Widerhold G. “Mediators in the Architecture of Future Information Systems”. *IEEE Computer*, 1992, 25:38-49
- [15] M. Peim, E. Franconi, N. Paton, and C. Goble. “Query Processing with Description Logic Ontologies Over Object-Wrapped Databases”. *Intl. Conf. on Scientific and Statistical Database Management*, 2002.
- [17] B. Ludäscher, Gupta A, Martone ME. “Model-Based Mediation with Domain Maps”. *Proc. 17<sup>th</sup> Int Conf Data Eng, Heidelberg, Germany. IEEE Computer Society*, 2001:81-90
- [23] A. Gupta, B. Ludäscher, M.E. Martone. “Registering Scientific Information Sources for Semantic Mediation”. *21<sup>st</sup> Int Conf Conceptual Modeling (ER), Tampere, Finland*. 2002:182-198.

**Acknowledgements.** This work has been partially funded by CICYT grant BIO2001-1237 and NIH grant 1R01HL67465-01.