OOPSLA'92
Vancouver, British Columbia, Canada
5 – 10 October 1992

Addendum
to the
Proceedings

Poster Submission—
Integration of Molecular Biology Data Collections Using Object Oriented Databases and Programming

Report by:

Patrick Herde
Peter R. Sibbald
European Molecular Biology Laboratory

Molecular biology is the study of the molecules that make up living things. These molecules include DNA, RNA, proteins, carbohydrates, lipids etc. Biologists seek to understand the function, structure, location, synthesis, regulation, dynamics, interaction and origin of these molecules. Such a discipline is motivated not only by practical problems in fields like medicine and biotechnology but also by curiosity about life itself. There are a number of data collections in molecular biology that would be much more powerful if they could be used in concert. Our ultimate aim is to be able to make arbitrarily complex queries across these data collections. Ultimately the computer naive user who is an expert biologist should be able to readily make any query without having to know where the data reside, how they are accessed or what programs are used. In other words, the biologist should be able to work in an extremely high level way.

There are some imposing barriers to achieving such a goal. Living things are incredibly complex and heterogeneous and this tends to be reflected in the data. For example, a human contains about 10^{13} cells, each one different. The human genome contains on the order of 10^9 nucleotides, subdivided into chromosomes, genes and various functional entities. And humans are just one of the millions of organisms of interest to biologists. Any attempt to use biological data in a meaningful way must cope with complexity inherent in the data. Most of the "facts" have exceptions.

Molecular biologists are generally not near the leading edge of computing. The result is a few hundred data collections quite often lacking in structure. These collections are valuable but difficult to use optimally. When many biological data collections were initiated decisions were made about the formats and structure that have not facilitated change. In particular, quite a lot of code has been written that relies on particular ways of storing data. There is a strong resistance to change since it would require recoding. Biology is theory impoverished. Unlike physics where there is a strong theoretical tradition to provide a framework on which to organize data, in biology there are few theories (evolution, cell theory) and therefore no agreed upon way that the information should be "naturally" structured. This results in each biologist having a different view of the biological data. Any model for data representation must recognize this.

There is also the autonomy problem that occurs in other areas of database research. The access to a database may be restricted, for example it can only be accessed via one particular piece of software. There may be restrictions on redistributing or reformatting the data. It is now common that data may be accessed via telnet but that the number of

users who can have access at one time is limited or the speed of the network problematic. Other data can be accessed via mail servers which by definition are asynchronous. Even when data are freely available they may be updated at a rate that renders impractical the idea of having local copies of the data.

An Approach

This work is in progress and one of the purposes is to describe the current direction with the aim of attracting criticism and input. There are several categories of data collection, each requiring somewhat different treatment. Some data collections are presently more or less static flat files. We are forming an OODB for each of these collections. Some data collections are already under DBMS, typically relational (we are unaware of any collections in a network model). We plan two experimental approaches for these: some we shall convert to OODBs and some we shall attempt to access under their present models. This will allow us to compare the two methods as well as ensuring that in cases where conversion is impractical there is another option. A third category is those data collections that can only be accessed via remote access through other people's software. These shall require another type of accessor as well as special consideration of the networking problems. The problems being faced appear to fit very well into the OO paradigm since many biological entities are already objects and can exploit inheritance and encapsulation.

From a user's viewpoint there a number of steps to make this a reality. The query must be interpreted. The correct items must be accessed in the appropriate databases. The items must be processed in such a way as to answer the query. We concentrate on the last two problems. Briefly, our model is: for each data collection there is an "accessor." A request (message) to an accessor results in the item requested or news that the item cannot be accessed. A list of the items accessible by each accessor is kept up to date. The request router responds to higher level requests (messages) by (a) decomposing them into sub-requests and (b) routing them to the correct accessor based on the list of accessible items. If a data collection leaves the system then the list of accessible items is shortened accordingly. If a new data collection enters the system, an accessor must be put in place and the list of accessible items updated. At the top end there is a user interface which obtains queries from the user and details about how the results of these queries are to be processed. The query interpreter resolves ambiguities and results in high level requests to the router. A logical unit assembles the items (the messages from the accessors) into a response to the query. In particular it will remove redundancies, resolve or report conflicts, and make connections between the items as dictated by the query.

The logical unit is necessarily sophisticated and the subject of current research. One of the main motivations in this work is to be able pose queries where the answer consists of parts that reside in different databases. To do this kind of synthesis a number of problems must be solved. One of the hard problems is that nomenclature is often not standard. For example, "human" might be termed "man," "Mann," "homme," "Homo sapiens," "man-kind" in different collections (or misspelt). Abbreviations are often used: "cyt" instead of "cytochrome." So that a request for information about "human cytochrome c" must deal with the items returned and (minimally) be able to recognize information about identical biological entities from different databases.

Despite our optimism, there remain non-trivial problems to solve. Biological data are globally distributed and often poorly formatted. Putting data where possible into an object oriented database may in itself require some effort, especially where those data are presently in flat files (e.g., there are frequently format violations that must be correctly handled by the parser software). Biological data remain complex and will defy rigid formats. One of the real tests of the proposed model will be how well it can cope with change and exceptions as they emerge. Perhaps the most challenging problem is how to design a system when the only sure constant is further major change.

Contact information:

Patrick Herde
Peter R. Sibbald
EMBL Datalibrary
European Molecular Biology Laboratory
Meyerhofstrasse 1, Postfach 10.2209
6900 Heidelberg, Federal Republic of Germany
Herde@EMBL-Heidelberg.DE
Sibbald@EMBL-Heidelberg.DE