A Molecular Biology Database Digest

François Bry and Peer Kröger

Institute for Computer Science, University of Munich, Germany
http://www.pms.informatik.uni-muenchen.de
bry@informatik.uni-muenchen.de peer.kroeger@dbs.informatik.uni-muenchen.de

Abstract

Computational Biology or Bioinformatics has been defined as the application of mathematical and Computer Science methods to solving problems in Molecular Biology that require large scale data, computation, and analysis [18]. As expected, Molecular Biology databases play an essential role in Computational Biology research and development. This paper introduces into current Molecular Biology databases, stressing data modeling, data acquisition, data retrieval, and the integration of Molecular Biology data from different sources. This paper is primarily intended for an audience of computer scientists with a limited background in Biology.

1 Introduction

> *"Computational biology is part of a larger revolution that will affect how all of science is conducted. This larger revolution is being driven by the generation and use of information in all forms and in enormous quantities and requires the development of intelligent systems for gathering, storing and accessing information."* [18]

As this statement suggests, Molecular Biology databases play a central role in Computational Biology [8]. Currently, there are several hundred Molecular Biology databases – their number probably lies between 500 and 1.000. Well-known examples are DDBJ [49], EMBL [6], GenBank [11], PIR [10], and SWISS-PROT [5]. It is so difficult to keep track of Molecular Biology databases that a "meta-database", DBcat [20], has been developed for this purpose. Nonetheless, DBCat by far does not report on all activities in the rapidly evolving field of Molecular Biology databases.

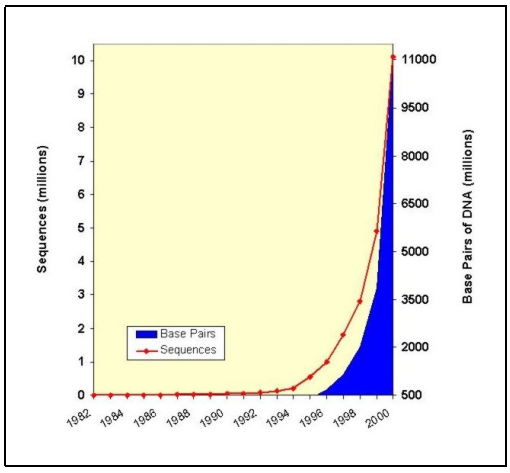


Figure 1: Growth of GenBank [26]

Most Molecular Biology databases are very large: e.g. GenBank contains more than 4×10^6 nucleotide sequences containing altogether about 3×10^{12} occurrences of nucleotides. The growth rate of most of these databases is exponential – cf. Figure 1. Both, the actual size and the growth rate of most Molecular Biology databases has become a serious problem: Without automated methods such as dedicated data mining and knowledge discovery algorithms, the data collected can no longer be fully exploited.

Most Molecular Biology databases rely upon *ad hoc* management methods. Some make use of management systems, e.g. relational database management systems, that were developed for rather different types of applications and that are not fully satisfying for Molecular Biology databases. Many important Molecular Biology databases are just collections of so-called "flat files", e.g. ASCII and GIF files. Flat files are the *de facto* data interchange standard for Molecular Biology data.

Molecular Biology databases are very heterogeneous in their aims, shapes, and usages they have been developed for. While some Molecular Biology databases contain only data gathered on one specific organism (e.g. the Human Genome Database GDB [34] on the Human Genome Project or the MIPS/Saccharomyces [39] database on yeast) and/or are developed and maintained by only one research team, other Molecular Biology databases aim at collecting all data available on biologically interesting concepts (such as SWISS-PROT [5], a database containing information about proteins from all organisms or GenBank [11], a database of all publicly available nucleotide sequences) and are the result of long lasting international co-operations between research laboratories. Furthermore, different approaches are used for data modeling, for storing, and for data analysis and query purposes. Molecular Biology databases have neither a common schema, nor a few widely accepted schemas, although querying different databases is a common practice in Computational Biology. As a consequence, the integration and interoperability of Molecular Biology databases are issues of considerable importance.

In spite of the recent surge of interest in Molecular Biology databases, these databases are rather unknown outside Computational Biology and Molecular Biology. Computer scientists and database experts are rarely knowledgeable about these databases and their uses. This is regrettable because there is a considerable need for further work and more database expertise in Computational Biology. Especially traditional database issues such as data modeling, data management, query answering, database integration as well as novel issues such as data mining, knowledge discovery, ontologies deserve more consideration in Computational Biology. Most Molecular Biology databases are far away from the state-of-the-art in data modeling, data data management, and query answering. They are often implemented using *ad hoc* techniques that do not provide with the services of a database management system. To some extent, this is explainable by specificities of Molecular Biology data and by the specific services (such as sequence analysis, similarity search, identification and classification) Computational and Molecular Biologists expect from Molecular Biology databases. However, the discrepancy between most current Molecular Biology databases and the state-of-the-art in data management also results from a lack of knowledge of two scientific communities for each other's concerns.

This paper aims at introducing into Molecular Biology databases, stressing data modeling, data acquisition, data retrieval, and current efforts in Molecular Biology database integration. The study reported about in this paper results from an investigation of 111 frequently used Molecular Biology databases. This paper is a digest primarily intended for an audience of computer scientists with a limited background in biology.

Following this introduction, Section 2 briefly describes the areas of Computational Biology where Molecular Biology databases are used. Section 3 introduces into the resources and the cross-references stored in Molecular Biology databases. How (computational) biologists use Molecular Biology databases is addressed in Section 4. Section 5 describes how Molecular Biology databases are implemented and the services they provide with. A Grand Table of 111 databases that have been investigated in this study is described in Section 6 and given in Appendix. Section 7 is devoted to current efforts in Molecular Biology database integration. Finally, Section 8 points out research perspectives.

2 Database Use in Computational Biology

Molecular Biology databases pervade all areas of Computational Biology. In the following, the major areas of Computational Biology are briefly introduced stressing the use of Molecular Biology databases.

2.1 DNA Analysis and Sequencing

Proteins are complex molecules that are the building stones of all forms of life. The protein variety is immense: There are e.g. more than one Million different proteins in the human organism. The proteins of an organism are build up of amino acids in a manner which is coded in the DNA (Deoxyribose Nucleic Acid) of the organism. The DNA is a linear polymer, a sequence made of 4 nucleotides. A subsequence of 3 nucleotides is called a codon. Each of the 20 different amino acids is coded by 1 to $4^3 = 64$ codons. Most amino acids have more than one such code. This coding is very complicated: Within a DNA there are non coding areas; the beginning and the end of the coding areas are difficult to recognize; the coding areas are not necessarily connected. Sequencing is the name given to the recognition of coding areas (and also of non-coding areas) in the DNA. Sequencing relies upon both, Computer Science methods and laboratory investigations, and makes use of databases. DNA analysis and sequencing rely upon stochastic methods such as stochastic grammars and hidden Markov applied to large databases of empirical data.

2.2 Protein Structure Prediction

The prediction of the three dimensional structure of the proteins (coded in a DNA) is one of the main goals of life sciences because the protein function depends on its structure. A complete solution to the protein prediction problem would revolutionize medicine and drug engineering. In order to avoid or restrict long lasting and complex laboratory investigations, Computer Science methods are applied for "folding" proteins, i.e. for determining (an approximation of) the three dimensional structure of proteins from their amino acid sequences.

One distinguishes between the primary, secondary, tertiary and quaternary structures of a protein. The primary structure of a protein is its amino acid sequence. The secondary structure of a protein is an abstraction of the three dimensional structure of the protein based upon three dimensional substructures, i.e. typical folding patterns called α-helix, β-fold and turn. The tertiary structure of a protein is the three dimensional structure of certain of its components. The quaternary structure of a protein expresses the spatial organization of the protein's components defined by its ternary structure. Up till now, the primary, secondary, and tertiary structures of only about 9.000 (protein coding) sequences are known.

The so-called "homology based methods" for the prediction of the ternary structure of proteins consist in algorithmic comparisons of (protein coding) sequences, the ternary structure of which is to be determined, with (protein coding) sequences, the ternary structure of which is already known. To this aim, so-called "similarities search methods" are applied to databases. Whether a protein might form a stable complex with some other molecule is called "protein docking problem". "1-1 docking procedures" determine relative positions of the molecules to one another. "1-n docking procedures" search in a molecule database potential docking partners for a given molecule. Homology based protein structure prediction and 1-n docking methods combine techniques from molecular dynamics, discrete mathematics, or genetic algorithms with data mining and knowledge discovery techniques.

2.3 Phylogenetic Trees

As times goes, the evolution modifies the protein codes in the DNA of organisms. Models specify the speed of such modifications. Specific sequence analysis algorithms based on these models compare the DNA of organisms for determining time intervals when the organisms are likely to have diverged from a common ancestor. This way, so-called "phylogenetic trees" are determined. Phylogenetic trees have used with noticeable success e.g. in evolutionary paleontology. In computing such trees, databases are often used.

2.4 Metabolic and Regulatory Pathways

A metabolic pathway is an abstract representation of a metabolism, i.e. of chemical reactions in a cell, listing the proteins and other molecules involved. A regulatory pathway describes the "control flow" for metabolic reactions within cells of a certain kind resulting in some diseases – such as cancer. Pattern matching, similarity search, and sequence analysis methods are applied to databases for discovering new metabolic or regulatory pathways for some organisms that are similar to already known pathways for some other, better known organisms.

2.5 Gene Expression

A gene is an DNA area which "codes" a protein and therefore determines genetic diseases. Within cells of a certain kind a certain gene produces the protein it codes: This process is called "gene expression". Using so-called "DNA chips", the concentration or "expression level" of thousand to ten thousands of genes that cells of a certain type express can be measured. With so-called "differential displays", the differences between the expression levels of healthy and ill cells can be recognized. The extensive data obtained this way are stored in databases that are used for developing new forms of diagnosis and/or therapies.

3 Resources and Cross-References in Molecular Biology Databases

One distinguishes between the *genotype* and the *phenotype* of organisms. The genotype has been compared with the software, the phenotype with the processes specified by the genotype [50]. The genotype of an organism is expressed in its *genome* "stored" or "coded" in its DNA. Data related to genotypes are usually referred to as *genomic data.* The phenotype of an organism consists in the phenomena determined by both, the genotype of the organism and the environment.

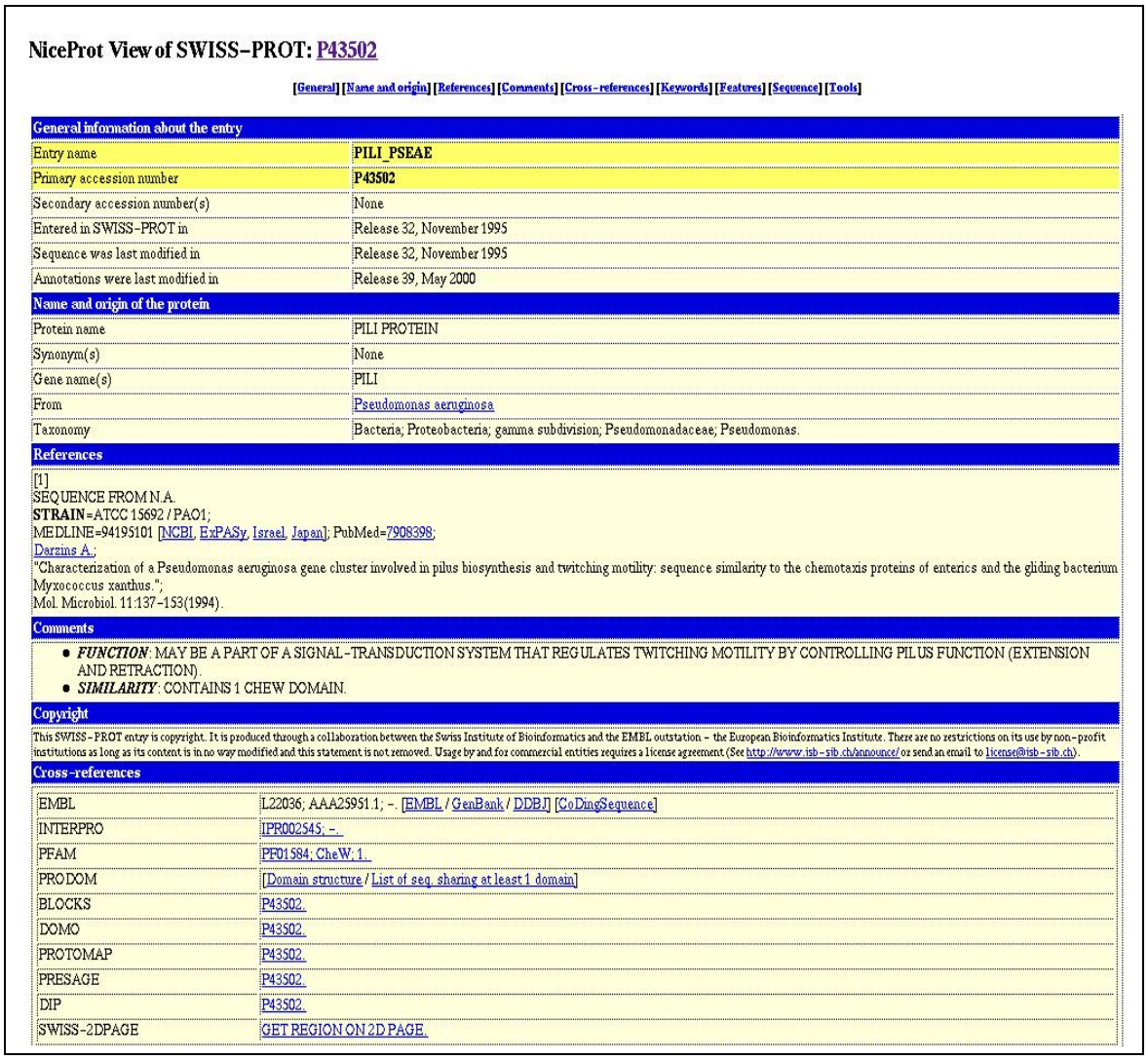
Computational Biology is concerned with both, genotypes and phenotypes of organisms. Thus, in addition to the celebrated genomic data also phenotype data are to be modeled, stored in databases, and queried. Phenotype data range from gene products, to complex interactions between gene products, to the behavior of entire organisms. Thus, Molecular Biology databases contain resources of three types [43]:

1. **Static Data:** Data on genotypes, i.e. biological entities such as nucleic acids, proteins, etc. and on relationships between theses entities.

2. **Dynamic Data:** Data on phenotypes, i.e. the dynamics of biological processes.

3. **Data on Analysis Tools:** Data on biological and computer science methods which can be used to identify the entities and their relationships.

4. **References and Annotations:** References to scientific papers (stored in specialized literature databases) on data of the above mentioned types; references between data of the above-mentioned types; and textual explanations called “annotations” of data items.



NiceProt View of SWISS-PROT: P43502

[General] [Name and origin] [References] [Comments] [Cross-references] [Keywords] [Features] [Sequence] [Tools]

General information about the entry	
Entry name	**PILI_PSEAE**
Primary accession number	**P43502**
Secondary accession number(s)	None
Entered in SWISS-PROT in	Release 32, November 1995
Sequence was last modified in	Release 32, November 1995
Annotations were last modified in	Release 39, May 2000
Name and origin of the protein	
Protein name	PILI PROTEIN
Synonym(s)	None
Gene name(s)	PILI
From	Pseudomonas aeruginosa
Taxonomy	Bacteria; Proteobacteria; gamma subdivision; Pseudomonadaceae; Pseudomonas.

References

[1]
SEQUENCE FROM N.A.
STRAIN=ATCC 15692 / PAO1;
MEDLINE=94195101 [NCBI, ExPASy, Israel, Japan]; PubMed=7908398;
Darzins A.;
"Characterization of a Pseudomonas aeruginosa gene cluster involved in pilus biosynthesis and twitching motility: sequence similarity to the chemotaxis proteins of enterics and the gliding bacterium Myxococcus xanthus.";
Mol. Microbiol. 11:137-153(1994).

Comments

- ***FUNCTION***: MAY BE A PART OF A SIGNAL-TRANSDUCTION SYSTEM THAT REGULATES TWITCHING MOTILITY BY CONTROLLING PILUS FUNCTION (EXTENSION AND RETRACTION).
- ***SIMILARITY***: CONTAINS 1 CHEW DOMAIN.

Copyright

This SWISS-PROT entry is copyright. It is produced through a collaboration between the Swiss Institute of Bioinformatics and the EMBL outstation - the European Bioinformatics Institute. There are no restrictions on its use by non-profit institutions as long as its content is in no way modified and this statement is not removed. Usage by and for commercial entities requires a license agreement (See http://www.isb-sib.ch/announce/ or send an email to license@isb-sib.ch).

Cross-references

EMBL	L22036; AAA25951.1; -. [EMBL / GenBank / DDBJ] [CoDingSequence]
INTERPRO	IPR002545; -.
PFAM	PF01584; CheW; 1.
PRODOM	[Domain structure / List of seq. sharing at least 1 domain]
BLOCKS	P43502.
DOMO	P43502.
PROTOMAP	P43502.
PRESAGE	P43502.
DIP	P43502.
SWISS-2DPAGE	GET REGION ON 2D PAGE.

Figure 2: SWISS-PROT [5] browser

Thus, Molecular Biology resources are rather heterogeneous. Most Molecular Biology databases focus on one of the above mentioned three first resources and also contain references of some kind. Currently, most Molecular Biology databases contain genotype data, referred to as “core data”, extended with annotations to these core data.

Many Molecular Biology databases also refer to other Molecular Biology databases. These references have often the form of Hypertext links within data items making a “point-and-click navigation” [32] possible. For cross-referencing of the data, most databases provide a unique access numbers

for each entry (artificial primary keys). References within a Molecular Biology database or between different Molecular Biology databases can be classified into “similarity links” and “biology links”.

```
ID   PILI_PSEAE     STANDARD;      PRT;   178 AA.
AC   P43502;
DT   01-NOV-1995 (Rel. 32, Created)
DT   01-NOV-1995 (Rel. 32, Last sequence update)
DT   30-MAY-2000 (Rel. 39, Last annotation update)
DE   PILI PROTEIN.
GN   PILI.
OS   Pseudomonas aeruginosa.
OC   Bacteria; Proteobacteria; gamma subdivision; Pseudomonadaceae;
OC   Pseudomonas.
RN   [1]
RP   SEQUENCE FROM N.A.
RC   STRAIN=ATCC 15692 / PA01;
RX   MEDLINE=94195101; PUBMED=7908398;
RA   Darzins A.;
RT   "Characterization of a Pseudomonas aeruginosa gene cluster involved
RT   in pilus biosynthesis and twitching motility: sequence similarity to
RT   the chemotaxis proteins of enterics and the gliding bacterium
RT   Myxococcus xanthus.";
RL   Mol. Microbiol. 11:137-153(1994).
CC   -!- FUNCTION: MAY BE A PART OF A SIGNAL-TRANSDUCTION SYSTEM THAT
CC       REGULATES TWITCHING MOTILITY BY CONTROLLING PILUS FUNCTION
CC       (EXTENSION AND RETRACTION).
CC   -!- SIMILARITY: CONTAINS 1 CHEW DOMAIN.
CC   ---------------------------------------------------------------------------
CC   This SWISS-PROT entry is copyright. It is produced through a collaboration
CC   between  the Swiss Institute of Bioinformatics  and the  EMBL outstation -
CC   the European Bioinformatics Institute.  There are no  restrictions on  its
CC   use  by  non-profit  institutions as long  as its content  is  in  no  way
CC   modified and this statement is not removed.  Usage  by  and for commercial
CC   entities requires a license agreement (See http://www.isb-sib.ch/announce/
CC   or send an email to license@isb-sib.ch).
CC   ---------------------------------------------------------------------------
DR   EMBL; L22036; AAA25951.1; -.
DR   INTERPRO; IPR002545; -.
DR   PFAM; PF01584; CheW; 1.
KW   Sensory transduction.
SQ   SEQUENCE   178 AA;  19934 MW;  634A1A4B135A7E77 CRC64;
     MSDVQTPFQL LVDIDQRCRR LAAGLPAQQE AVQSWSGIGF RMGGRFFVAP MGEVGEVLME
     PRYTQLPGVK TWVKGVANVR GRLLPIMDLC GFLGTELSPL RKQRRVLVVE HLDVFAGLIV
     DEVFGMQMFP VDTFSEQLPP LEAALQPFIH GVFMREQPWL VFSPHALAQH QGFLDVAV
//
```

Figure 3: A SWISS-PROT [5] excerpt

Similarity links connect sequence entries (or data items specifying sequence data) with similar sequences (or with data items specifying similar sequence data). Similar sequences (or data items specifying similar sequences) are often called “neighbors”. Neighbors are detected using similarity search programs such as BLAST [3] and FASTA [40]. Usually, similarity links are not stored in Molecular Biology databases. Instead, they have to be computed by database users using similarity search programs often provided by the database.

Biology links refer to relevant biology information including literature references.

The database SWISS-PROT [5] provides with examples of the different kinds of references. A SWISS-PROT data item on a protein might be linked to a GenBank [11] data item describing the gene encoding this protein and to an article stored in the literature database PubMed [42] – cf Figure 2.

In flat files databases, annotations are in general intertwined with the Molecular Biology data and references are encoded – cf Figure 3.

4 A Biologist's View of Molecular Biology Databases

What a Biologist usually sees from a Molecular Biology database, this is the services it provides – not how the database is implemented. Molecular biology databases usually provide software tools for the analysis of the data it contains. Typically, these tools serve to analyzing newly produced data, in comparing data with formerly collected data, in making new predictions, and in testing hypothesis. The use of mathematical and Computer Science methods is essential, for it makes it possible to avoid or restrict long lasting and expensive "wet lab" work. Interfaces to Molecular Biology databases aim at overcoming the following obstacles: Limited data awareness, complex data retrieval, limited data analysis tools availability, limited literature reference availability.

4.1 Data Awareness

A biologist is in general not aware of all the databases relevant to its investigation. Typically, a biologist uses three to ten Molecular Biology databases he or she is familiar with. The help provided with by similarity and biology links (cf. Section 3) is often insufficient. Furthermore, such links are inefficient to manage: If n databases are to be linked this way, then the information to collect and to update is distributed over the n databases. The "meta-database" DBcat [20] is a better approach, for the linking information is centralized. Keeping such a database up-to-date, however, is extremely time-consuming. Specialized search engines possibly using data mining methods dedicated to Molecular Biology contents, like existing search engines for Molecular Biology *literature* (cf. e.g. [31]) and possibly relying upon ontologies might be promising approaches.

4.2 Complex Data Retrieval

Most Molecular Biology database users are not familiar with database query languages such as SQL. Control of database query languages is not common among biologists. Therefore, Molecular Biology databases in general have form-based query and/or browsing interfaces. This is convenient for simple queries, but significantly restrict data access if complex queries have to be expressed. It is not clear, whether SQL would be a convenient query language for Molecular Biology data, anyway, for the relational data model does not seem appropriate to represent Molecular Biology data. XML query languages such as XPath [16] and XQuery [14] might be more convenient than SQL for retrieving Molecular Biology data since the semistructured data model seem to be appropriate to model such data [1, 2] – cf. infra Section 5.

4.3 Data Analysis Tools Availability

Most Molecular Biology databases provide with dedicated data analysis tools implementing – e.g. the similarity search methods BLAST [3] or FASTA [40]. Such tools are essential for data interpretation. Some of them are difficult to use, in general because of large numbers of parameters to set up. It might also be difficult to estimate whether a tool implements an algorithm appropriate to the data retrieval task considered. Finally, many such tools are poorly documented.

4.4 Literature Reference Availability

As mentioned in Section 3, most Molecular Biology databases contain literature references. These references, however, might be inaccurate or out-of-date. In Computational Biology in general, and in Molecular Biology databases in particular, there is a considerable need for advanced, dedicated electronic library databases such as PubMed [42] and for literature data mining. More and more computational biologists consider data documentation by means of references (e.g. to articles describing how the data have been collected) a premier objective.

4.5 Interfaces to Molecular Biology Databases

Interface systems have been developed that provide with unified, in general Web-based interfaces to several Molecular Biology databases, e.g. BioKleisli [19], DBGET/LinkDB [25], Entrez [23], Tambis [7], and SRS [24, 47].

SRS [24] is such a system offering rather comprehensive functionalities. It provides a unified WWW acess to about 500 Molecular Biology databases. Its query answering facilities exploit the Hypertext references between data items available in most Molecular Biology databases and can also compute additional references. It has both, a form-based query interface and an advanced query language using which complex queries – possibly accessing Hypertext references – can be expressed. SRS also provide with standard Computational Biology data analysis methods and support their application to the data returned as answers to queries. SRS is discussed in more detail in Section 7.

5 A Computer Scientist's View of Molecular Biology Databases

This section is devoted to how current Molecular Biology databases are build up and managed, considering successively, data models and data management systems, data retrieval methods, and data acquisition.

5.1 Data Modeling and Data Management

Following [36], Molecular Biology databases can be classified as follows:

1. Databases using a **standard database management system**, i.e. a relational, object, or object-relational system.

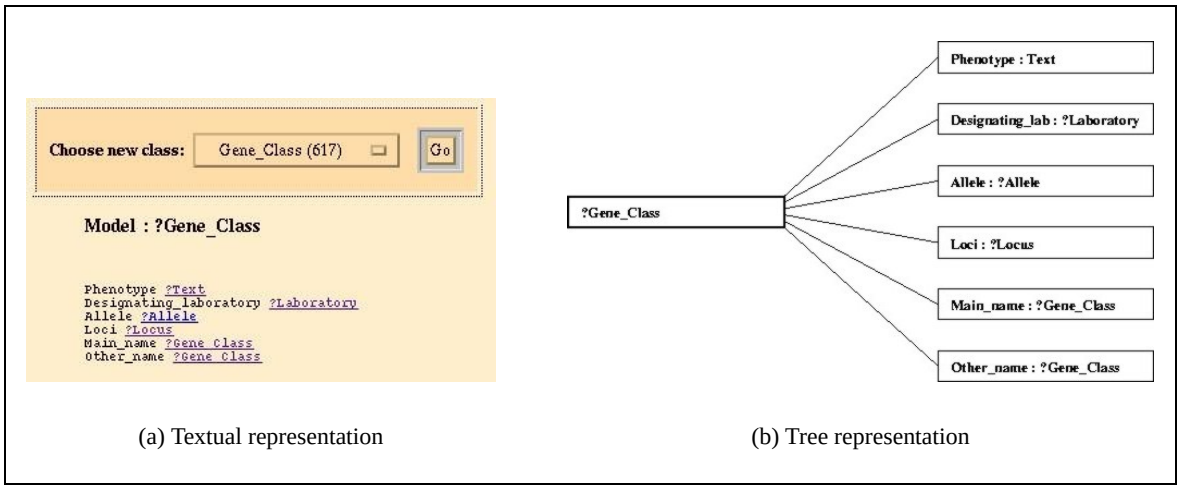
2. Databases using the database management system **ACEDB** [2]. ACEDB (note the upper case 'E') is a DBMS which was originally implemented for the Molecular Biology database called "A *C.elegans* Data Base (ACeDB)" (note the lower case 'e').

3. Databases using the **Object Protocol Model (OPM)** [15] together with a relational or object database management system. OPM is a data model combining standard object-oriented modeling constructs with specific constructs for the modeling of scientific experiments.

4. Databases implemented as **flat file collections**.

Standard Database Management System Most Molecular Biology databases have been first implemented as flat file collections. Later, in general in the mid nineties, many of them were reimplemented using a relational, object, or object relational database management system (DBMS). The object model is more suitable than the relational model to model Molecular Biology data. Molecular Biology databases based on the relational model often have very complex schemas which, in general, are no longer intuitive. Therefore, they are often difficult to administrate and to query. Nevertheless, a significant number of Molecular biology databases are nowadays implemented using widespread relational DBMS – such as Oracle, Sybase or MySQL – cf. Section 6.

ACEDB ACEDB [2] (with upper case 'E') is a database management system initially developed by for a database called "A *C.elegans* Data Base (ACeDB)" (with lower case 'e') containing data on the organism (a small worm) called *C. elegans*. Later, ACEDB has been extended so as to also manage other such specialized databases. In the literature, the database management system ACEDB and the database ACeDB are often confused.



(a) Textual representation

(b) Tree representation

Figure 4: An ACEDB object (of the class "Gene_Class") from the database ACeDB [22]

ACEDB resembles an object database management system. With ACEDB, data are modeled as objects that are organized in classes. However, ACEDB supports neither class hierarchies, nor inheritance. An ACEDB object has a set of attributes that are objects or atomic values such as numbers or strings. ACEDB objects are represented as trees where the (named) nodes are object or atomic values and arcs express the attribute relationship cf. Figure 4. An ACEDB class has a "class model" specifying the maximal set of attributes an object of the class may have and the class or type of the objects and of their attributes. An object of a class may have only part of the attributes, i.e. of the branching pattern, permitted by the class model. This reminds of the semistructured data model [1]. In addition to the object classes, ACEDB also provides with arrays. ACEDB's arrays allow for a less flexible, but more efficient storage of data like DNA sequences. ACEDB's arrays consist of tables with variable length tuples.

Like the semistructured data model and for the same reasons, the ACEDB data model has the following advantages: First, it accommodates irregular data items. This is useful for accommodating exceptions, as often occur in empirical data. Second, extensions of the schema can be easily achieved by adding attributes to objects because class model do not require every object of the class to have instances for all class attributes. With ACEDB, it is possible to extend a database schema without

having to restructure the database, for existing objects need not to be modified. The semistructured data model is richer than the ACEDB data model because it also has multiple inheritance. Multiple inheritance, however, can be simulated with ACEDB [2].

Basic services of a DBMS such as transaction, recovery and indexing are supported by ACEDB. In addition, ACEDB provides a powerful, high level query language called AQL. The source code of ACEDB is public and can therefore be modified to fit specific requirements of some application.

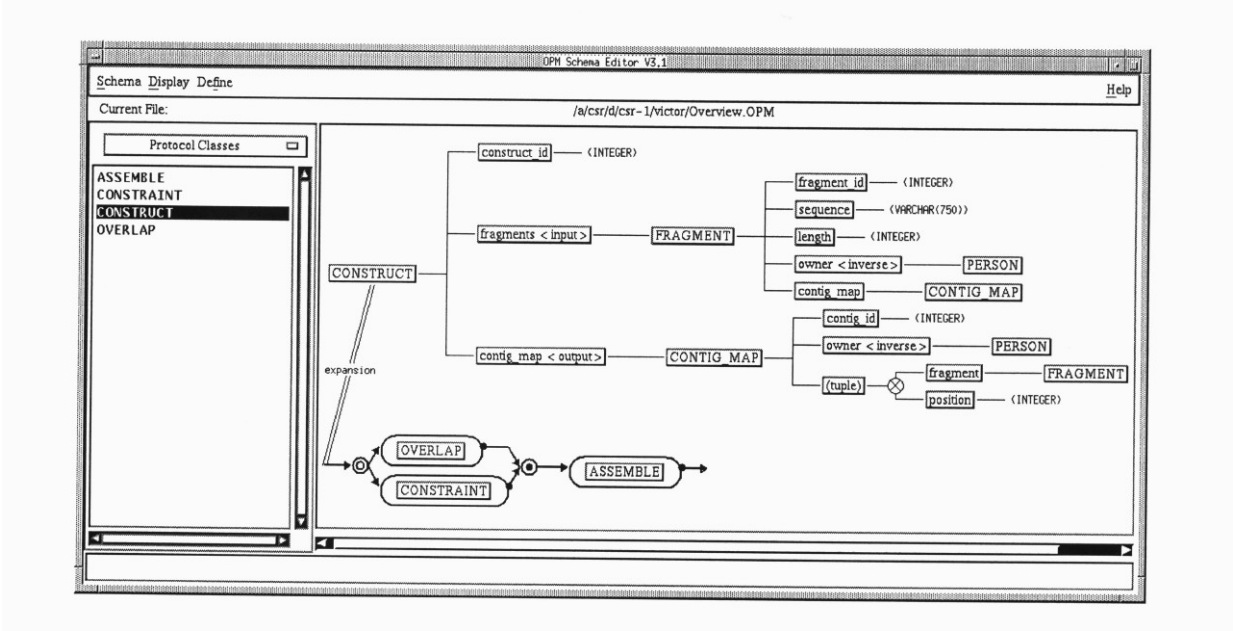


Figure 5: An OPM schema [15]

OPM The Object Protocol Model (OPM) [15] has been developed for modeling both biology data and the event sequences in scientific experiments. These event sequences are referred to as "protocol". OPM is similar to an object model but, in contrast to standard object models, OPM also provides with specific constructs for the modeling of scientific experiments cf. Figure 5. The OPM objects are similar to that of the Semantic Database Model (SDM) [29] and of O_2 [9]. OPM has derived object classes as well as inheritance mechanisms [15].

The development of OPM has been motivated by the observation that the relational and object models are inadequate to the modeling of scientific experiments [15]. This comes from the fact that experiments not only refer to static but also to dynamic data – cf. Section 3.

Using OPM, experiments can be accurately described. So-called "protocol classes" are similar to object classes. Protocol modeling is characterized by the recursive specification of generic protocols in terms of component protocols (or "sub-protocols"). A complex protocol can consist in a sequence of sub-protocols or in optional sub-protocols. "Input and output attributes" are associated with a protocol class in addition to regular attributes, such as the attribute of a non-protocol object, and "connection attributes". Input and output attributes express the resources consumed and produced of directly related protocols. Protocol relationships are expressed using delete rules associated with "connection" and "system attributes". Derived protocol classes can be generic protocol classes used

for representing experiments that are constructed from instances of existing protocol classes, or sub-protocol classes used for representing parts of existing experiments. A derived sub-protocol inherits the attributes of its generic protocol.

OPM gives rise to defining views. The SQL-like query language of OPM supports the kind of nested queries prevalent in scientific applications, path expressions and set predicates. OPM also offers an ontology of scientific terms. OPM has a suite of data management tools providing with an interface to relational database management systems like Sybase and Oracle. These tools also include an OPM schema editor, a translator of OPM schemas into relational definitions and procedures, a generic WWW-based graphic query browsing and data entry interface, and a translator of relational database schemas into OPM schemas. OPM and its data management tool suite are commercial products.

```
>HSPM3|HSPM3 histone H3 - garden pea. [ Pisum sativum ]|peaH3
ARTKQTARKSTGGKAPRKQLATKAARKSAPATGGVKKPHRFRPGTVALREIRKYQKSTEL
LIRKLPFQRLVREIAQDFKTDLRFQSSAVSALQEAAEAYLVGLFEDTNLCAIHAKRVTIM
PKDIQLARRIRGERA

>HSPM4|HSPM4 histone H4 - garden pea. [ Pisum sativum ]|peaH4
SGRGKGGKGLGKGGAKRHRKVLRDNIQGITKPAIRRLARRGGVKRISGLIYEETRGVLKI
FLENVIRDAVTYTEHAKRKTVTAMDVVYALKRQGRTLYGFGG

>HSPG4|HSPG4 histone H4 - pig. [ Sus scrofa domestica ]|porH4
SGRGKGGKGLGKGGAKRHRKVLRDNIQGITKPAIRRLARRGGVKRISGLIYEETRGVLKV
FLENVIRDAVTYTEHAKRKTVTAMDVVYALKRQGRTLYGFGG
```

Figure 6: A HDB [48] excerpt

Flat files In the early days of Molecular Biology databases, data base management systems were rarely used. Instead most Molecular Biology databases were built up as (more or less) indexed ASCII text files, called "flat files" – cf. Figures 6 and 3. Later, in the eighties and nineties, as database management systems especially relational database management systems were used more and more frequently for Molecular Biology databases, many Molecular Biology databases remained collections of flat files. It has been argued that database management systems are dispensable in Computational Biology because Molecular Biology data in general are not expected to change, because multiple-user access is rarely required, and because the cost of porting an existing flat-file databases into a relational database would often be too high. Another, maybe more convincing reason is that Molecular Biology data are often very complex. The typical data type subjacent to many flat files includes deeply nested records, sets, lists and variants. Such data types cannot easily be represented in existing relational and object database management systems [19]. Arguably, data management still has to be established in Computational Biology.

Molecular Biology databases implemented as flat files in general have no explicit data model. Their entries (i.e. data items) are usually structured either implicitly (cf. Figure 6) or explicitly by search indexes (cf. Figure 3). Most flat file collections are explicitly structured using keywords (to be used as search indexes). The term "line type" is often used for these keywords. The keywords may be two-character strings or variable length words. The flat files used in Computational Biology seem to have no common semantic structure: The keywords and indices used in distinct flat files often differ

not only in their syntax, but also semantically.

Sequence databases are often flat file collections, for the modeling and efficient storing of long sequences (of nucleotides or amino acids) has not been much investigated. Some databases (e.g. the celebrated database GenBank [11]) use ANS.1 to define the structure of their data items. The "Abstract Syntax Notation No. 1 (ANS.1)" has been originally defined for the data transmitted by telecommunication protocols [4].

Nowadays, flat files are the *de facto* data exchange standard in Molecular Biology. Many tools biologists are accustomed to (e.g. BLAST [3] and FASTA [40]) work only with flat files. As a consequence, most Molecular Biology databases provide their entire contents in one or more flat files (cf. infra "Data Retrieval").

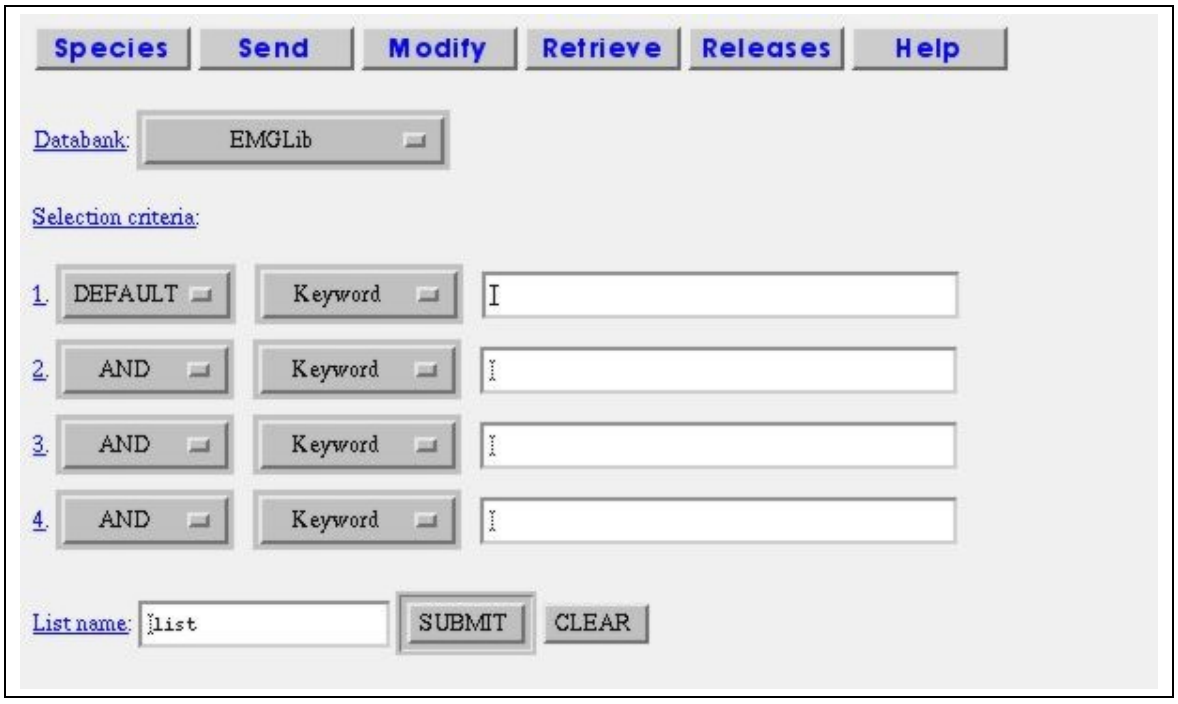


Figure 7: Fixed-form query interface of EMGLib [41] (at PBIL)

5.2 Data Retrieval

In general, a Molecular Biology databases provides with at least on of the following data retrieval approach:

1. Query interface.
2. Indirect data retrieval with database browsers.
3. Database (as flat file) downloading.

The query interfaces to be found in Molecular Biology databases can be classified in "free-form/*ad-hoc*" query interfaces and "fixed-form" query interfaces.

Free-form/*ad-hoc* query interfaces provide the possibility to express a query in a query language depending on the underlying data model. Although the query languages used often powerful, free-form/*ad-hoc* query interfaces have the following drawbacks: Biologists are usually not familiar with the principles of these languages and of database query languages in general, and a user of such a language must have a detailed knowledge of the schema of the database.

Fixed-form query interfaces provide one or several views on the database cf. Figure 7. With such a query interface, queries can only be posed against a predetermined set of tables, classes, or other database components, and in queries only a predetermined set of attributes for each database component can be used. The view underlying a fixed-form query interface to a Molecular Biology database not necessarily reflects the internal, i.e. storage, structure of the database. Fixed-form query interfaces do not have the above-mentioned drawbacks of free-form/*ad-hoc* query interfaces – at the price of strongly restricting data retrieval.

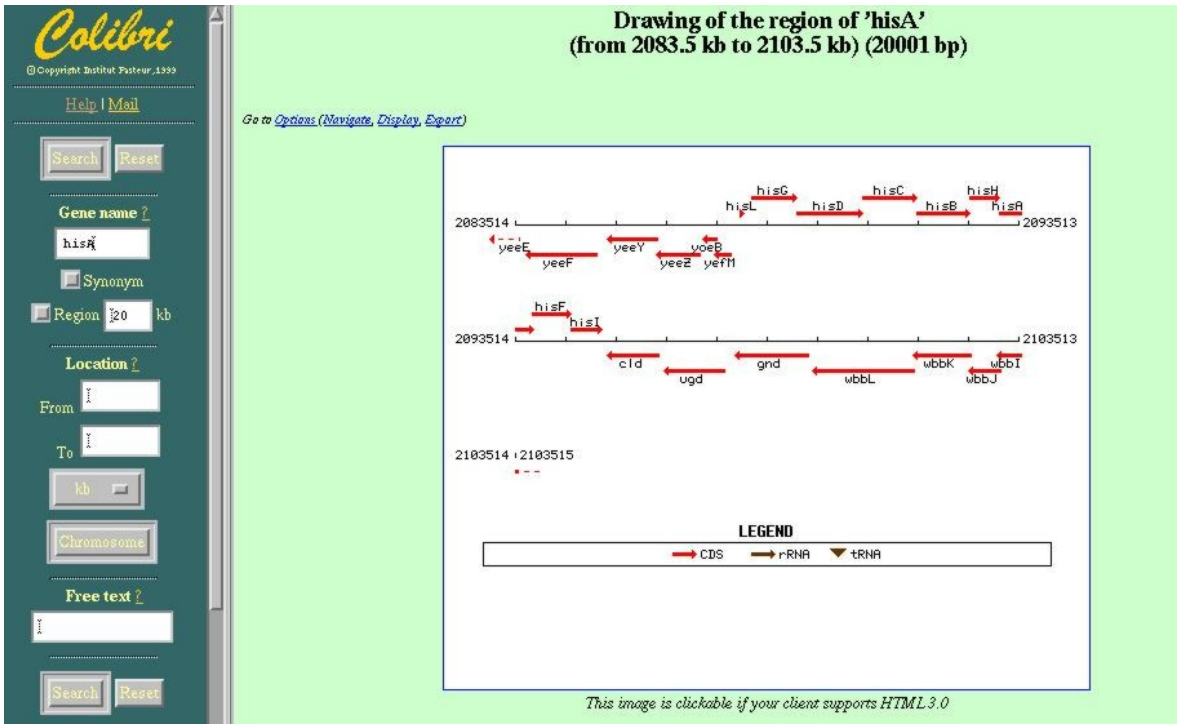


Figure 8: Browser of Colibri [37]

In some Molecular Biology databases, hierarchical classifications of the data can then be browsed for data retrieval cf. Figure 8. This approach to data retrieval has been called "indirect data retrieval". Interestingly, browsers are also available for flat file databases cf. Figure 2 (compare with Figure 3).

Most Molecular Biology databases, support flat file download via FTP, even if they are implemented with a database management system. Recall that flat files still are the *de facto* data interchange standard in Molecular and Computational Biology.

5.3 Data Acquisition

Molecular Biology databases collect their data using some of the following approaches:

1. **From other databases.** The collected data in general have to be reformatted.

2. **From the research community:** Many Molecular Biology databases acquire their data from submissions by researchers. Some databases restrict the data submission rights (in general to some research teams). Fill-in forms often make sure that the data fit the database schema. Problems often arise from errors in and inconsistencies between submissions. An *a posteriori* "cleaning" of the submitted data do not always takes place.

3. **From the literature:** Usually, data acquisition from the scientific literature is done manually and is therefore work intensive.

The update frequency is an interesting aspect of a Molecular Biology database, for it considerably varies between databases. Some Molecular Biology databases are updated daily or many times a day. Other Molecular Biology databases are no longer updated (in some cases because the database was built as a by-product of a research project now completed or interrupted).

6 The Molecular Biology Databases Investigated

For this study, 111 randomly selected Molecular Biology databases have been considered between Autumn 2000 to Summer 2001. This database selection contains major Molecular Biology databases as well as more specialized and less known databases. Inclusion in (and ommission from) this selection should not be misinterpreted as an appreciation of a database's quality.

A Grand Table given in Appendix briefly describes the 111 databases investigated in this study. The legend of this table is given in Figure 9. In this table, ? denotes an unknown value. Following a value, ? expresses that this value is uncertain. A few databases are accessable only through SRS (cf. Sections 4 and 7). This is indicated by the mention "via SRS" under "Querying/Data Retrieval".

Database	database short name in alphabetical order (digits before letter)
Contents	Molecular Biology nature of the data
DB-Links	References to other databases as HT: Hypertext links TR: textual references
Implementation	flat files rel. DBMS: relational database management system obj. DBMS: object database management system o.-r. DBMS: object-relational database management system
Acquisition	C: submissions from the research community D: collected from other databases L: collected from scientific literature
Querying/Retrieval	FF: fixed-form query interface – cf. Section 5 AH: *ad hoc* query interface – cf. Section 5 FTP: download of files (usually via FTP) Ind.: indirect data retrieval – cf. Section 5 via SRS

Figure 9: Legend of the Grand Table of Molecular Biology databases

Interestingly, 96 (i.e. 86%) of the 111 considered databases have Hypertext references to other databases, 40 to 44 (i.e. 36% to 40%) are implemented as flat files, 41 (or 42) (i.e. 37%) are implemented using a relational database management system, 7 (i.e. 6%) use an object database management system, 3 (.i.e. 3%) use an object-relational database management system, and all databases collect data from different sources.

7 Molecular Biology Database Integration

A widespread practice in Molecular Biology is that a research team first analyzes some data it has generated or collected (e.g. from databases or from the literature), then makes these data available to the research community through a database. Many Molecular Biology databases have been developed in this manner. As a consequence, Molecular Biology databases are highly distributed and heterogeneous, reflecting the distribution and heterogeneity of the Molecular Biology research community [7, 32]. Collecting and integrating data from different Molecular Biology databases is an issue of increasing importance in Computational Biology, for the detection of similarities between data from distinct origins (e.g. from different organisms) is prevalent in Molecular Biology – cf. Section 2.

7.1 Importance of Semantic Conflicts in Molecular Biology Database Integration

Integrating data from distinct origins leads to so-called "descriptive", "heterogeneity", and "semantic conflicts" [46]. Descriptive conflicts occur when the same semantic objects are differently modeled in distinct databases. Heterogeneity conflicts result from distinct data models and management systems used in distinct databases. Semantic conflicts occur when naming conventions differ in distinct databases. In standard, e.g. managerial databases, semantic conflicts can in general be quite easily overcome with so-called data dictionaries. In Molecular Biology, semantic conflicts are much more difficult to deal with, for they usually reflect distinct scientific viewpoints. Molecular Biology semantic conflicts make an automatic data retrieval from distributed, heterogeneous Molecular Biology databases very difficult.

The concept of "gene" illustrate semantic conflicts: In GDB [34], a gene is defined as a DNA fragment which can be transcribed and translated into a protein. For GenBank [11], a gene is in contrast a DNA fragment carrying a genetic trait or phenotype (including non-structural coding DNA regions like introns or promoters).

The notion of "biological functions" illustrates how semantic conflicts can make data retrieval difficult. Biological functions may be described at different levels. E.g. the function of a protein can be described at the molecule level, one speaks of "molecular function" of the protein, or at the cell level, one speaks of the "cellular function" of the protein. The molecular function of an enzyme such as aspartokinase is the catalysis of a certain reaction, whereas the (documented) cellular function of aspartokinase in bacteria is the catalysis of the first step in the common biosynthetic pathway [51]. Both, the molecular and the cellular function of a protein often have to be considered together because a protein with a given molecular function is often involved in cellular processes. The definition and modelization of biological function in a Molecular Biology database reflects the database's focus of interest. It might happen that in a Molecular Biology database the molecular function of a protein is described in an attribute named "biological function", while the cellular function of that protein is explained in a "comment" attribute. In such a case, an automatic recognition of the definition of the cellular function might be almost impossible.

Integrating Molecular Biology data from different origins in general require to "curate" the data utilizing specific knowledge about the database's field. This can be done manually by expert curators and also automatically using computational approaches. Usually, both forms of data curation take place.

7.2 Updates in Molecular Biology Database Integration Systems

In order to keep data originating from different databases up to date, frequent (e.g. daily) updates are necessary. With Molecular Biology databases, this is especially computing intensive because flat files are the *de facto* exchange format in the field – cf. Section 5. Structured models are preferable for data interchange. The semistructured data to data modeling and data management [13, 1] seem to be especially promising for Molecular Biology database integration, for it supports irregular data items and exceptions – cf. Section 5. Several research activities are concerned with using XML for modeling Molecular Biology data – cf. e.g. [52, 35, 53]. Some Molecular Biology databases can be downloaded in XML format e.g. Entrez (cf. `http://ncbi.nlm.nih.gov/entrez/`) and PIR (cf. `ftp://nbrfa.georgetown.edu/pir/databases/pir_xml/`).

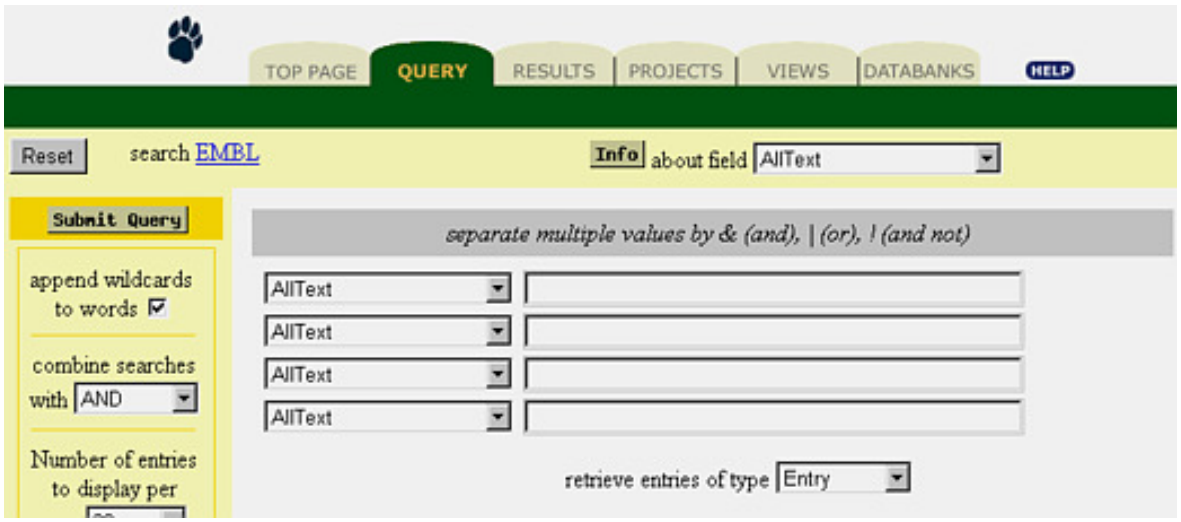


Figure 10: A SRS Standard Query Form [47]

7.3 Dedicated Integration Systems for Molecular Biology Databases

A few systems have been developed for the integration of Molecular Biology databases e.g. BioKleisli [19], DBGET/LinkDB [25], Entrez [23], Tambis [7], and SRS [24]. As an example, SRS is described in more detail.

SRS is worth describing in more detail, for it has interesting features like a query language using which Hypertext links can be followed. SRS is described in its user guide [47] as a "*data integration, analysis and display tool for bioinformatics, genomic and related data.*"

SRS offers a WWW portal to about 500 Molecular Biology databases. Using it, a same "standard query form" (cf. Figure 10) can be used for accessing data from different databases. Answers to SRS queries are listed as Hypertext links in "query result" web pages (cf. Figure 11). SRS exploits the Hyperlink cross-references almost all Molecular Biology contain: With an answers to an SRS query,

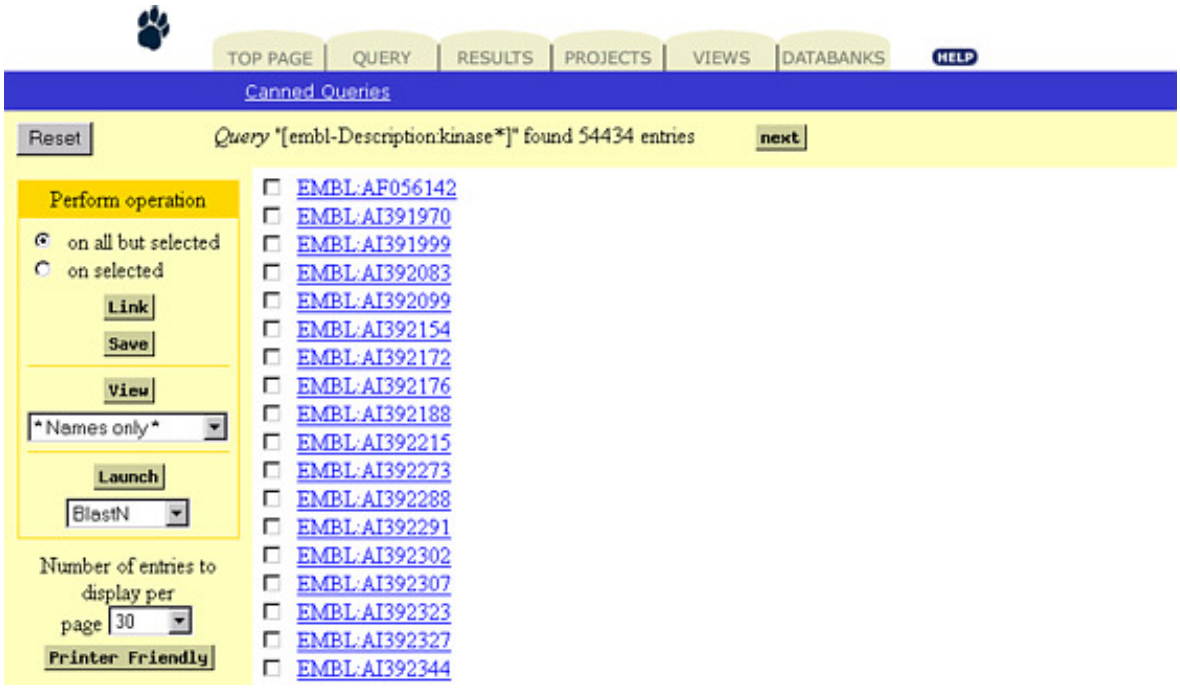


Figure 11: A SRS Query Result [47]

a SRS query result web page also displays the Hypertext links contained in this answer to related data items in other database). Following such a link result in augmenting the SRS query result web page originally returned by SRS.

User profiles make it possible to customize both, query forms (e.g. by pre-selecting databases) and query result web pages. Also, SRS makes it possible to save queries for later re-use. Answers to queries can also be downloaded.

Another feature of SRS is the support of Computational Biology data analysis methods. The methods applicable to an answer can be listed on demand (using a button on the query result web pages). They are mentioned as Hypertext link. Activating such a link displays a "launch" (cf. Figure 12) web page using which parameters can be set up for an application of the selected method to the answer this method was associated which in the query result web page. For simplifying the use, default values are provided for the parameters as the "launch" page is displayed. The result of applying a method on an answer is displayed on a web page (cf. Figure 13). SRS provide many different ways to display method results.

SRS also provides with a query language, called "SRS query language", using which database and data selections, operations on sets obtained as answers from other queries can be expressed and a crawler function (accessible through so-called "link operators") so as to automatically follow Hypertext links associated by SRS with answers.

E.g. the following query [47]

```
[swissprot-id:acha_human] > prosite > swissprot
```

first retrieves the entry "`acha_human`" from the SWISS-PROT database [5] as well as the entries from the PROSITE database [30] that are refered to (through Hypertext links) in the returned "`acha_human`" entry of SWISS-PROT. With the rightmost link operator `>`, the answer is augmented

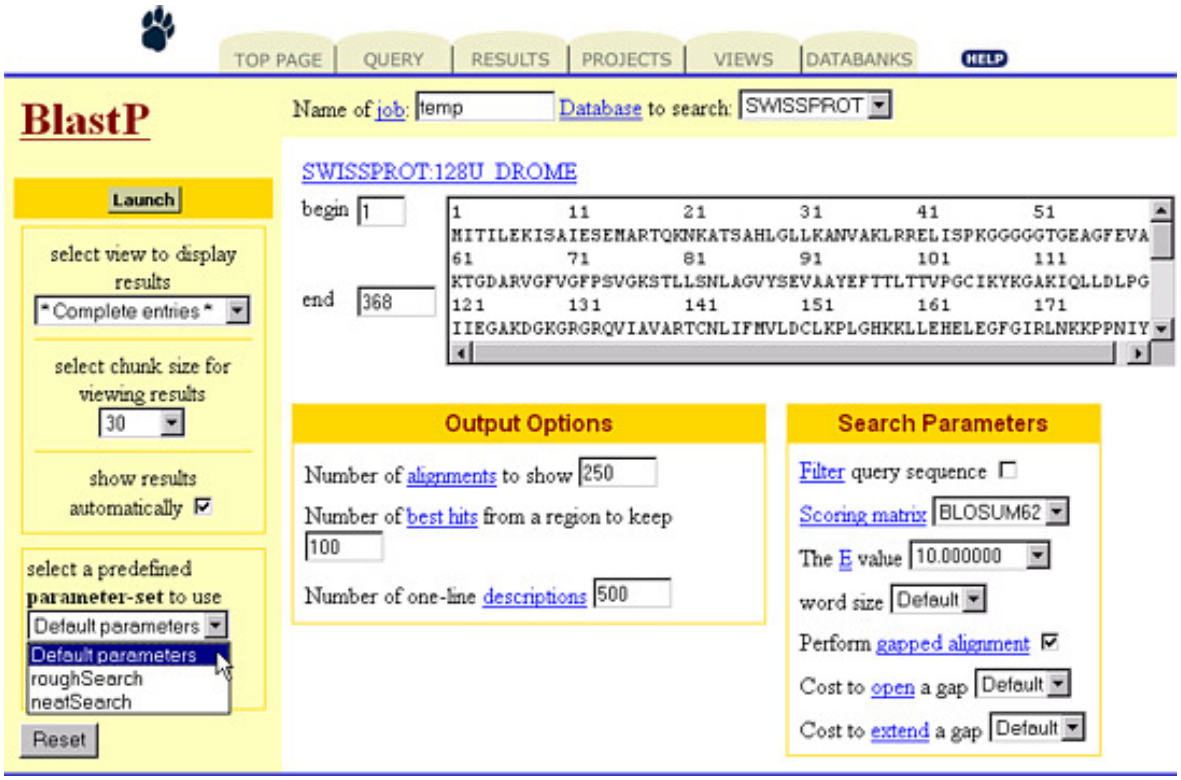


Figure 12: A SRS Launch Form [47]

with all SWISS-PROT entries that are refered to (through Hypertext links) the retrieved PROSITE entries. This way, all SWISS-PROT data items documenting members of the protein families to which "`acha_human`" belongs are retrieved.

Thus, the link operators of the SRS query language make it possible to use this language for (a limited form) of Web crawling.

The SRS query language combines navigational aspects reminding of XPath [16] and of CSS selectors [12] with boolean connectives and set operations. Using the "multiple linking" feature of the SRS query language, one can find information related to a data item in other databases this data item does not refer to with Hypertext links. The SRS query language also has constructs for restructuring answers.

There are worldwide about 30 distinct SRS servers accessing each from 0 to more than 100 "libraries", i.e. databases or parts of databases. Altogether, these SRS servers access about 500 different libraries. These SRS servers support about 30 Computational Biology data analysis methods. The SRS servers, the libraries they access, and the methpods they support are listed at `http://www.lionbio.co.uk/publicsrs.html`.

7.4 Related Issues

Further current integration approaches for Molecular Biology databases consist in the definition of "thesauri" and "ontologies" e.g. [7]. Thesauri and ontologies aim at developing standardized vocabularies, naming convention, and sometimes data interchange formats. Early attempts in the field

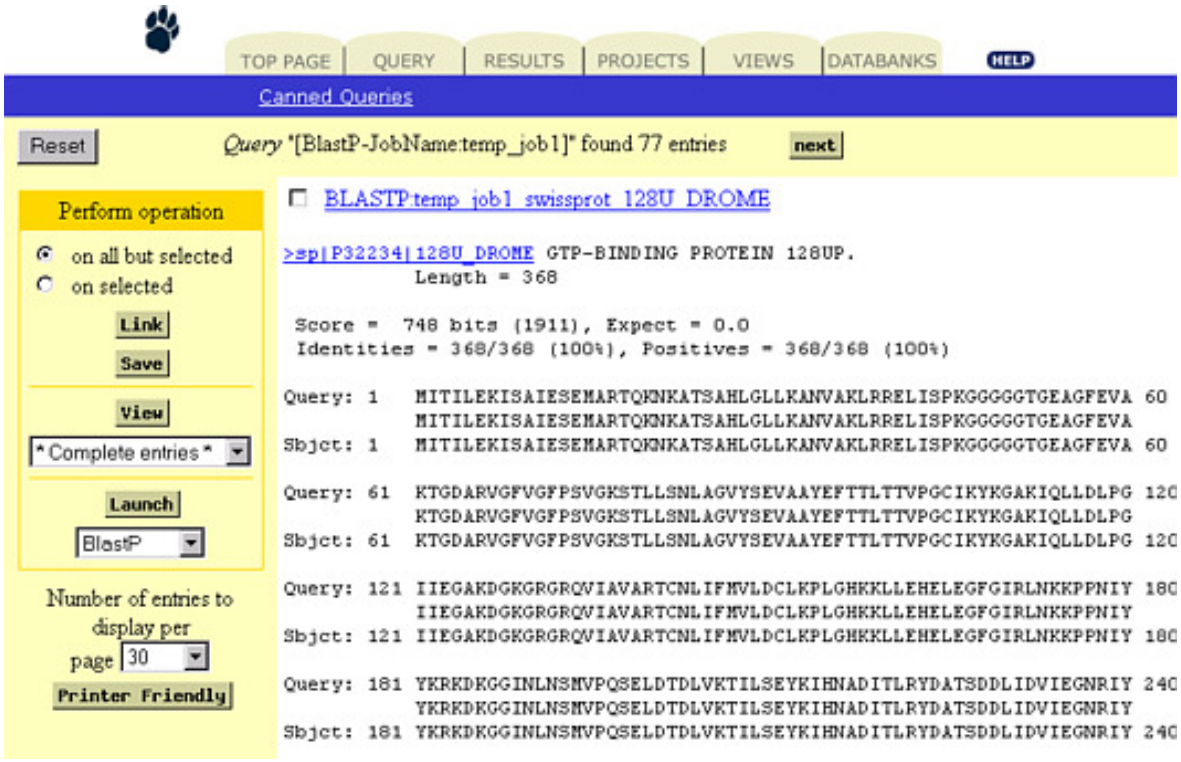


Figure 13: Result of a Method Application with SRS [47]

are reported in [27, 4]. [38] gives an overview on ontologies and interchange formats for Molecular Biology.

Recall that cross-referencing through Hypertext links within data items is a widespread approach to (a lightweight form of) database integration in Molecular Biology databases – cf. Section 3.

Finally, it is worth noting that standard approaches to database integration, i.e. "federated databases" [45], integration through materialized views e.g. in "data warehouses" [28], and "multi-database query systems" [33, 44], are rarely applied to Molecular Biology databases. Tambis [7] can be seen as a federated database system. A few research institutions have collected data from several of their projects into systems reminding of data warehouses e.g. MIPS [39]. BioKleisli [19] can be seen as a multi-database query system for Molecular Biology.

8 Database Research Perspectives

Molecular Biology databases are challenging database applications because their management, querying and integration call for new solutions.

Database integration is a premier research issue in Molecular Biology databases. Standard database integration methods do not seem to be sufficient for Molecular Biology databases. Original approaches have been developed for integrating Molecular Biology databases, in particular cross-referencing (of databases and data items) using Hypertext links (cf. Section 3) and crawling constructs in query languages (cf. Section 7). Interestingly, XQuery [14] does not have specific construct for an automatic traversal of Hypertext links. Both approaches, cross-referencing with Hypertext links and

crawling constructs in query languages, seem to be relevant to databases from other fields, too, and deserve further investigations.

Most Molecular Biology databases integrate databases on scientific literature and databases on Molecular Biology data. This reminds of "data dictionaries" investigated in the eighties – cf. e.g. [21]. The need for integrating text data with other data also exists in scientific and managerial databases. Text mining techniques, e.g. as considered in information retrieval, as well as other approaches, e.g. based on thesauri and/or ontologies, are promising research directions.

Search engines are already applied to finding scientific *literature* in the field of Molecular Biology. It is an open question whether similar techniques could be also applied to Molecular Biology *data.*

Finally, note that the application of the object and semistructutred data models to Molecular Biology data, and the definition of (e.g. XML-based) markup languages for Molecular Biology data, are active areas of research.

Acknowledgments

This digest could not have been written without the kind and patient support of the following persons: Rolf Backofen (Computer Science and Bioinformatics, University of Munich, Germany), Peter Clote (Computer Science and Biology, Boston College, USA), Stefan Conrad (Computer Science, University of Munich, Germany), Antoine de Daruvar (Bioinformatics at LaBRI, University of Bordeaux, France, and Lion Bioscience, Germany), Johannes Herrmann (Chemistry, University of Munich, Germany), Hans-Werner Mewes (Munich Information Centre for Protein Sequences and Bioinformatics, Technical University of Munich, Germany), François Rechenmann (Computer Science and Genomics, INRIA, Grenoble, France), and Thomas Seidl (Computer and Information Science, University of Constance, Germany). The authors thank them all for their help.

This work has been supported by a visiting professorhip of the University of Bordeaux (LaBRI) and a grant of the Bayerisch-Französisch Hochschulzentrum/Centre de Coopération Universitaire Franco-Bavarois.

References

[1] S. Abiteboul, P. Buneman, D. Suciu (2000). *Data on the Web: From Relations to Semistructured Data and XML*. Morgan Kaufmann Publishers, San Francisco.

[2] ACEDB Documentation Library.
http://genome.cornell.edu/acedocs/

[3] S. Altschul, W. Gish, W. Miller, E.W. Myers, D.J. Lipman (1990). *Basic Local Alignment Search Tool*. Journal of Molecular Biology, Vol. 215, pp. 403-410.

[4] ASN.1 Standard. Web Site.
http://asn1.elibel.tm.fr

[5] A. Bairoch, R. Apweiler (2000). *The SWISS-PROT Database and its Supplement TrEMBL in 2000*. Nucleic Acids Research, Vol. 28, No. 1, pp. 45-48.
http://www.expasy.ch/sprot/

[6] W. Baker, A. van den Broek, E. Camon, P. Hingamp, P. Sterk, G. Stoesser, M.A. Tuli (2000). *The EMBL Nucleotide Sequence Database*. Nucleic Acids Research, Vol. 28, No. 1, pp. 19-23.

[7] P. Baker, A. Brass, S. Bechhofer, C. Goble, N. Paton, R. Stevens (1998). *Tambis – Transparent Access to Multiple Bioinformatics Information Sources*. In: *Proceedings of the 6th International Conference on Intelligent Systems in Molecular Biology* (ISMB'98), pp. 25-34.

[8] P. Baker, C. Goble, S. Bechhofer, N. Paton, R. Stevens, A. Brass (1999). *An Ontology for Bioinformatics Application*. Bioinformatics, Vol. 15, No. 6, pp. 510-520.

[9] F. Bancilhon, C. Delobel., P. Kanellakis (1992). *Building an Object-Oriented Database System: The Story of* O_2. Morgan Kaufmann.

[10] W. C. Barker, J. S. Garavelli, Zhenglin Hou, Hongzhan Huang, R. S. Ledley, P. B. McGarvey, H.-W. Mewes, B. C. Orcutt, F. Pfeiffer, Akira Tsugita, C. R. Vinayaka, Chunlin Xiao, Lai-Su L. Yeh, C. Wu. (2001). *Protein Information Resource: a community resource for expert annotation of protein data*. Nucleic Acids Research, Vol. 29, pp. 29-32.
http://pir.georgetown.edu/pirwww/aboutpir/doc/nar-pir-2001db.pdf

[11] D. Benson, I. Karsch-Mizrachi, D.J. Lipman, J. Ostell, B.A. Rapp, D.L. Wheeler.(2000). *GenBank*. Nucleic Acids Research, Vol. 28, No. 1, pp.15.-18.
http://www.ncbi.nlm.nih.gov/Genbank/

[12] B. Boss, H. Wium Lee, C. Lilley, I. Jacobs (1998). *Cascading Style Sheets, level 2, W3C Recommendation*.
http://www.w3.org/TR/REC-CSS2/

[13] P. Buneman (1997). *Semistructured data*. Tutorial in: *Proceedings of the 16th ACM Symposium on Principles of Database Systems*.
http://citeseer.nj.nec.com/buneman97semistructured.html

[14] D. Chamberlain, J. Clark, D. Florescu, J. Robie, J. Siméon, M. Stefanescu (2001). *XQuery 1.0: An XML Query Language*, W3C Working Draft.
http://www.w3.org/TR/xquery/

[15] I.-M. Chen, V. Markowitz (1995). *An Overview of the Object Protocol Model (OPM) and the OPM Data Management Tools*, Information Systems, Vol. 20, No. 5, pp. 393-418.

[16] J. Clark, S. DeRose (1999). *XML Path Language (XPath) Version 1.0*, W3C Recommendation.
http://www.w3.org/TR/xpath

[17] Clote P., Backofen R. (2000). *Computational Molecular Biology, an Introduction*. John Wiley & Sons, Ltd., Chichester, New York, Weinheim, Brisbane, Singapore, and Toronto.

[18] M. Clutter (1996). *Hearing on Computational Biology*. Statement before the subcommittee on Science, Technology and Space Committee on Commerce, Science, and Transportation, U.S. Senate.
http://www.nsf.gov/od/lpa/congress/cluttes2.htm

[19] S.B. Davidson, C. Overton, V. Tannen, L. Wong (1997). *Biokleisli: A Digital Library for Biomedical Researchers*. International Journal on Digital Libraries, Vol. 1, No. 1, pp. 36-53.

[20] C. Discala, X. Benigni, E. Barillot, G. Vaysseix (2000). *DBcat: a Catalog of 500 Biological Databases*. Nucleic Acids Research, Vol. 28, No. 1, pp. 8-9.
http://www.infobiogen.fr/services/dbcat

[21] D. R. Dolk (1988). *Model Management and Structured Modeling: The Role of an Information Resource Dictionary System*. Communications of the ACM, Vol. 31, No 6, pp. 704-718.

[22] R. Durbin, J. Thierry-Mieg (1994). *The ACEDB Genome Database*. In: S. Suhai (ed.) (1994). *Computational Methods in Genome Research*. Plenum Press, New York.

[23] Entrez Online Dokumentation.
http://www.ncbi.nlm.nih.gov/Database/index.html

[24] T. Etzold, A. Ulyanow, P. Argos (1996). *SRS: Information Retrieval System for Molecular Biology Data Banks*. Methods in Enzymology, Vol. 266, pp. 114-128.

[25] W. Fujibuchi, S. Goto, H. Migimatsu, I. Uchiyama, A. Ogiwara, Y. Akiyama, M. Kanehisa (1997). *DBGET/LinkDB: An Integrated Database Retrieval System*. In: *Pacific Symposium on Biocomputing* (PSB'97), pp. 683-694.

[26] GenBank Growth.
http://www.ncbi.nlm.nih.gov/Genbank/genbankstats.html

[27] D. George, H.-W. Mewes, H. Kihara (1987). *A Standardized Format for Sequence Data Exchange*. Protein Sequence Data Analysis, Vol. 1, pp.27-39.

[28] A. Gupta, H. V. Jagadish, I. S. Mumick (1996). *Data Integration Using Self-Maintainable Views*. In *Proceedings of the International Conference on Extending Database Technology* (EDBT), LNCS, Vol. 1057, Springer Verlag, pp. 140-144.

[29] M. Hammer, D. McLeod (1981). *Database Description with SDM: A Semantic Database Model*. ACM Transactions on Database Systems, Vol. 6, No. 3.

[30] K. Hofmann, P. Bucher, L. Falquet, A. Bairoch (1999). *The PROSITE Database, its Status in 1999*. Nucleic Acids Research, Vol. 27. No. 1, pp. 215-219.
http://www.expasy.ch/prosite/

[31] T.K. Jenssen, A. Laegreid, J. Komorowski, E. Hovig (2001).*A Literature Network of Human Genes for High-Throughput Analysis of Gene Expression*. Nature Genetics, Vol. 28, No 1, pp. 21-28.

[32] P. Karp (1995). *A Strategy for Database Interoperation*. Journal of Computational Biology, Vol 2, No. 4, pp. 573-586.

[33] L. V. S. Lakshmanan, F. Sadri, I. N. Subramanian (1996). *SchemaSQL: A Language for Interoperability in Relational Multidatabase Systems*. In *Proceedings of the International Conference on Very Large Databases* (VLDB'96), pp. 239-250.

[34] S. Letovsky, R.W. Cottingham, C.J. Porter, P.W.D. Li (1998). *GDB: the Human Genome Database*. Nucleic Acids Research, Vol. 26, No.1, pp. 94-99.
http://www.gdb.org

[35] D. C. McArthur. *An extensible XML Schema definition for automated exchange of protein data: PROXIML (PROtein eXtensIble Markup Language)*.
http://www.cse.ucsc.edu/~douglas/proximl/

[36] V. Markowitz, I.-M. Chen, A. Kosky, E. Szeto (1997). *Facilities for Exploring Molecular Biology Databases on the Web: A Comparative Study*. In: *Pacific Symposium on Biocomputing* (PSB'97), pp. 256-267.

[37] C. Medigue, A. Viari, A. Henaut, A. Danchin (1992). *Colibri: a functional data base for the Escherichia coli genome*. Microbiology and Molecular Biology Reviews, Vol. 57, No. 3, pp. 623-654.

[38] R. McEntire, P. Karp, N. Abernethy, D. Benton, G. Helt, M. DeJongh, R. Kent, A. Kosky, S. Lewis, D. Hodnett, E. Neumann, F. Olken, D. Pathak, P. Tarczy-Hornoch, L. Toldo, T. Topaloglou (2000). *An Evaluation of Ontology Exchange Languages for Bioinformatics*. In: Proc. Int. Conf. Intelligent Systems in Molecular Biology (ISMB'00), pp. 239-50.

[39] H.-W. Mewes, D. Frishman, C. Gruber, B. Geier, D. Haase, A. Kaps, K. Lemcke, G. Mannhaupt, F. Pfeiffer, C. Schüller, S. Stocker, B. Weil (2000). *MIPS: a Database for Genomes and Protein Sequences*. Nucleic Acids Research, Vol. 28, No. 1, pp. 37-40
http://www.mips.biochem.mpg.de

[40] W. Pearson, D. Lipman (1988). *Improved tools for biological sequence comparison*. In: *Proceedings of the National Academy of Science USA (PNAS)*, Vol. 85, pp. 2444-2448.

[41] G. Perrière, P. Bessières, B. Labedan (2000). *EMGLib: the Enhanced Microbial Genomes Library (update 2000)* . Nucleic Acids Research, Vol. 28, No. 1, pp. 68-71.

[42] PubMed Database:
http://www.ncbi.nlm.nih.gov/PubMed/

[43] F. Rechenmann (1995). *Knowledge bases and computational biology*. In: N. Mars, editor, *Towards Very Large Knowledge Bases*, pp. 1-12. IOS Press.

[44] K.-U. Sattler, S. Conrad, G. Saake (2000). *Adding Conflict Resolution Features to a Query Language for Database Federations*. Australian Journal of Information Systems, Vol. 8, No. 1, pp. 116-125.

[45] A. P. Sheth, J.A. Larson (1990). *Federated Database Systems for Managing Distributed, Heterogeneous, and Automated Databases*. In: ACM Computing Surveys, Vol. 22, No. 3, pp. 183-196.

[46] S. Spaccapietra, C. Parent, Y. Dupont (1992). *Model Independent Assertions for Integration of Heterogeneous Schemas*. VLDB Journal, Vol. 1, No. 1, pp. 81-126.

[47] SRS User Guide (2000).
/srs6/doc/srsuser.pdf

[48] S. A. Sullivan, L. Aravind, I. Makalowska, A. D. Baxevanis, D. Landsman (2000). *The Histone Database: a comprehensive WWW resource for histones and histone fold-containing proteins* Nucleic Acids Research, Vol. 28, No. 1, pp. 320-322.

[49] Y. Tateno, S. Miyazaki, M. Ota, H. Sugawara, T. Gojobori (2000). *DNA Data Bank of Japan (DDBJ) in collaboration with mass sequencing teams*. Nucleic Acids Research, No. 28, Vol. 1, pp. 24-26.

[50] S. Tsur (2000). *Data Mining in the Bioinformatics Domain*. In: *Proceedings of the 26th Conference on Very Large Databases* (VLDB'00).

[51] J. van Helden, A. Naim, R. Mancuso, M. Eldridge, L. Wernisch, D. Gilbert, S. J. Wodak (2000). *Representing and Analysing Molecular and Cellular Function in the Computer*. Biological Chemistry, Vol. 381, pp. 921-935.

[52] XEMBL Project.
http://www.ebi.ac.uk/xembl/

[53] G. Xie, R. DeMarco, R. Blevins, Y. Wang (2000). *Storing Biological Sequence Databases in Relational Form*. Bioinformatics, Vol 16, No. 2, pp. 288-289.

Grand Table of 120 Molecular Biology Databases (Legend cf. Figure 9)

Database	Contents	DB-Links	Implementation	Acquisition	Querying/Data Retrieval				URL
					FF	AH	FTP	BH	
3DBase	prot. struct.	?	OPM	D	√				http://pdb.weizmann.ac.il/pdb-bin/pdbmain
AAindex	mixed type	HT	flat files	L			√		http://www.genome.ad.jp/dbget/aaindex.html
AARSDB	nucl. sequ.	TR	flat files	L				√	http://rose.man.poznan.pl/aars/index.html
ALFRED	genetic	HT	rel. DBMS	C, L	√			√	http://alfred.med.yale.edu/alfred/index.asp
aMAZE	pathways	HT	obj. DBMS	D	under construction				http://www.ebi.ac.uk/research/pfbp
AMmtDB	nucl. sequ.	HT	flat files	D	√	√	√		http://bio-www.ba.cnr.it:8000/srs6
ASDB	genetic	HT	**?**	D	√				http://devnull.lbl.gov:8888/alt/index.html
Axeldb	genomic	HT	ACEDB	C, D, L	√	√	√	√	http://www.dkfz-heidelberg.de/abt0135/axeldb.htm
BMRB	prot. struct.	HT	rel. DBMS	C, D, L	√		√	√	http://www.bmrb.wisc.edu
BRENDA	mixed type	HT	rel. DBMS	C, L	√				http://www.brenda.uni-koeln.de/
CATH	taxonomy	HT	**?**	D	√		√	√	http://www.biochem.ucl.ac.uk/bsm/cath_new/
COG	proteins	HT	flat files	D	√		√	√	http://www.ncbi.nlm.nih.gov/COG
Colibri	proteomic	HT	rel. DBMS	C, D	√	√	√		http://genolist.pasteur.fr/Colibri/
COMPEL	genetic	HT	rel. DBMS	D, L	√		√	√	http://compel.bionet.nsc.ru
CSNDB	pathways	HT	ACEDB	L	√	√	√	√	http://geo.nihs.go.jp/csndb/
CyanoBase	genomic	HT	rel. DBMS	C,D	√		√	√	http://www.kazusa.or.jp/cyano/
DAtA	proteomic	HT	rel. DBMS	D	√		√	√	http://luggagefast.Stanford.EDU/group/arabprotein/
DBcat	literature	HT	flat files	C, D, L	√	√	√	√	http://www.infobiogen.fr/services/dbcat
dbSNP	nucl. sequ.	HT	rel. DBMS	C, D	√		√		http://www.ncbi.nlm.nih.gov/SNP/

Database	Contents	DB-Links	Implementation	Acquisition	Querying/Data Retrieval				URL
					FF	AH	FTP	BH	
DDBJ	nucl. sequ.	HT	rel. DBMS	C, D	√		√		http://www.ddbj.nig.ac.jp/
DIP	proteins	HT	rel. DBMS	L	√		√		http://dip.doe-mbi.ucla.edu/
DSMP	prot. struct.	HT	flat files	D	√	√			http://www.cdfd.org.in/dsmp.html
EcoCyc	met. pathw.	?	obj. DBMS	D, L	√	√	√	√	http://ecocyc.pangeasystems.com/ecocyc/
EcoGene	proteomic	HT	flat files	–	√		√		http://bmb.med.miami.edu/EcoGene/EcoWeb
EID	genomic	TR	flat files	D			√		http://mcb.harvard.edu/gilbert/EID
EMBL	nucl. sequ.	HT	rel. DBMS	C, D	√		√	√	http://www.ebi.ac.uk/embl/
EMGLib	nucl. sequ.	HT	flat files	D	√		√		http://pbil.univ-lyon1.fr/emglib/emglib.html
ENZYME	mixed type	HT	flat files	C, D, L	√		√	√	http://www.expasy.ch/enzyme/
EPD	genetic	HT	flat files	?	√		√		http://www.epd.isb-sib.ch/
ExInt	genomic	HT	flat files	D	√		√		http://intron.bic.nus.edu.sg/exint/exint.html
FIMM	mixed type	HT	flat files	D, L	√				http://sdmc.krdl.org.sg:8080/fimm/
FlyBase	genomic	HT	rel. DBMS	C, D, L	√		√	√	http://flybase.bio.indiana.edu/
GDB	genomic	HT	OPM	C	√	√		√	http://www.gdb.org
GenBank	nucl. sequ.	HT	rel. DBMS	C, D	√		√		http://www.ncbi.nlm.nhi.gov/Genbank
GIMS	genomic	?	obj. DBMS	D	under construction				http://img.cs.man.ac.uk/gims
GSDB	nucl. sequ.	HT	rel. DBMS	D	√	√	√		http://www.ncgr.org
GXD	genomic	HT	rel. DBMS	C, L	√	√	√	√	http://www.informatics.jax.org
HDB	mixed type	HT	flat files	D	√		√	√	http://genome.nhgri.nih.gov/histones/
HGBASE	genomic	HT	rel. DBMS	C, D, L	√	√	√		http://hgbase.cgr.ki.se
HGMD	genetic	HT	flat files ?	L	√				http://archive.uwcm.ac.uk/uwcm/mg/hgmd0.html

Database	Contents	DB-Links	Implementation	Acquisition	Querying/Data Retrieval				URL
					FF	AH	FTP	BH	
HOX Pro	genetic	HT	rel. DBMS ?	?				√	http://www.mssm.edu/molbio/hoxpro/new/hox-pro00.html
IDB/IEDB	mixed type	HT	rel. DBMS	D			√		http://nutmeg.bio.indiana.edu/intron/
IMB	structure	HT	flat files ?	D	√			√	http://www.imb-jena.de/IMAGE.html
IMGT	nucl. sequ.	HT	rel. DBMS	D	√	√	√	√	http://imgt.cines.fr:8104
InBase	mixed type	HT	flat files ?	C				√	http://www.neb.com/neb/inteins.html
INTERACT	proteins	HT	obj. DBMS	C, D, L	√	√			http://bioinf.man.ac.uk
InterPro	proteomic	HT	rel.DBMS	D	√		√		http://www.ebi.ac.uk/interpro/
IXDB	genomic	HT	rel. DBMS	C, D, L	√			√	http://ixdb.mpimg-berlin-dahlem.mpg.de
KEGG	met. pathways	HT	flat files	D	√		√	√	http://star.scl.genome.ad.jp/kegg/
KinMutB.	mixed type	HT	flat files ?	C				√	http://www.uta.fi/imt/bioinfo/KinMutBase/
KMDB	mixed type	HT	flat files	?			√	√	http://mutview.dmb.med.keio.ac.jp
LIGAND	mixed type	HT	flat files	D, L	√		√	√	http://star.scl.genome.ad.jp/dbget/ligand.html
MAGEST	genomic	HT	rel. DBMS	D	√		√		http://star.scl.genome.ad.jp/magest
MaizeDB	genomic	–	rel. DBMS	?	√	√		√	http://www.agron.missouri.edu/
MDDB	mixed type	?	rel. DBMS	?	?	?	√	?	http://www-bm.cs.uni-magdeburg.de/iti_bm/bmbf/mdcave.html
MEROPS	taxonomy	HT	flat files	D	√		√	√	http://www.merops.co.uk/merops/merops.htm
MGD	genomic	HT	rel. DBMS	C, D	√	√	√	√	http://www.informatics.jax.org/
MIPS	proteins	HT	diverse	C	√			√	http://www.mips.biochem.mpg.de
MitBASE	genomic	HT	rel. DBMS	C, D, L	√			√	http://www3.ebi.ac.uk/Research/Mitbase/mitbase.pl

Database	Contents	DB-Links	Implementation	Acquisition	Querying/Data Retrieval				URL
					FF	AH	FTP	BH	
MitoNuc	genetic	HT	flat files	D, L		via SRS			http://bighost.area.ba.cnr.it/srs/
MITOP	proteomic	HT	flat files	D, L	√		√	√	http://www.mips.biochem.mpg.de/proj/medgen/mitop
MMDB	prot. struct.	HT	flat files	D	√		√		http://www.ncbi.nlm.nih.gov:80/Structure/MMDB/mmdb.shtml
ModBase	prot. struct.	HT	flat files	D	√				http://pipe.rockefeller.edu/modbase/
MTB	genomic	HT	rel. DBMS	C, L	√				http://informatics.jax.org
NDB	structure	HT	rel. DBMS	C, D, L	√			√	http://ndbserver.rutgers.edu:80/
OMIM	genetic	HT	**?**	C	√		√		http://www.ncbi.nlm.nih.gov:80/entrez/Omim/
ooTFD	genetic	HT	obj. DBMS	D, L	√		√		http://www.ifti.org/
ORDB	mixed type	HT	o-r. DBMS	C	√			√	http://ycmi.med.yale.edu/senselab/ordb/
PDB	prot. struct.	HT	rel. DBMS	C	√		√		http://www.rcsb.org/pdb/
PEDB	proteomic	HT	rel. DBMS	D, L	√			√	http://www.pedb.org/
Pfam	prot. sequ.	HT	flat files	D	√		√	√	http://www.sanger.ac.uk/Software/Pfam/
PIR/PSD	prot. sequ.	HT	o.-r. DBMS	C, D, L	√		√		http://pir.georgetown.edu
PLMItRNA	mixed type	TR	flat files	C, D, L	√				http://bio-www.ba.cnr.it:8000/srs6/
PombePD	proteomic	HT	rel. DBMS	C, L	√			√	http://www.proteome.com/databases/index.html
PRINTS-S	taxonomy	HT	o.-r. DBMS	D	√	√	√	√	http://www.bioinf.man.ac.uk/dbbrowser/PRINTS/
ProClass	taxonomy	**?**	rel. DBMS	D	√		√		http://pir.georgetown.edu/gfserver/proclass.html
ProTherm	mixed type	HT	flat files ?	**?**	√				http://www.rtc.riken.go.jp/jouhou/Protherm/protherm.html

Database	Contents	DB-Links	Implementation	Acquisition	Querying/Data Retrieval				URL
					FF	AH	FTP	BH	
ProtoMap	taxonomy	?	?	D	√			√	http://www.protomap.cs.huji.ac.il/search.html
PseudoBase	genetic	HT	flat files	C				√	http://wwwbio.LeidenUniv.nl/~Batenburg/PKB.html
ProDom	taxonomy	HT	flat files ?	D	√		√		http://www.toulouse.inra.fr/prodomCG.html
PROSITE	proteomic	HT	flat files	D	√		√		http://www.expasy.ch/prosite/
PubMed	literature	HT	?	?	√			√	http://www.ncbi.nlm.nih.gov/PubMed/
RDP	nucl. sequ.	?	obj. DBMS	C, D			√		http://www.cme.msu.edu/RDP
REBASE	mixed type	HT	?	D, C	√		√	√	http://rebase.neb.com/rebase/rebase.html
RegulonDB	genetic	HT	rel. DBMS	D, L	√				http://www.cifn.unam.mx/ComputationalBiology/regulondb/
RHdb	mixed type	HT	rel. DBMS	C	√		√		http://www.ebi.ac.uk/RHdb
SacchDB	genetic	TR	ACEDB	C, L	√	√	√	√	genome-ftp.stanford.edu
SBASE	prot. sequ.	HT	flat files	D, L	√		√		http://sbase.abc.hu/sbase
SCOP	prot. struct.	HT	flat files	D	√		√	√	http://scop.mrc.lmb.cam.ac.uk/scop/
SELEXdb	nucl. sequ.	–	flat files	L		via SRS			http://wwwmgs.bionet.nsc.ru/mgs/systems/selex/
SENTRA	mixed type	HT	flat files	D	√				http://wit.mcs.anl.gov/WIT2/Sentra/HTML/sentra.html
SGD	genetic	HT	rel. DBMS	C, L	√	√	√	√	http://genome-www.stanford.edu/Saccharomyces/
SMART	mixed type	HT	rel. DBMS	D	√				http://smart.embl-heidelberg.de/
SRPDB	mixed type	–	flat files	D				√	http://psyche.uthct.edu/dbs/SRPDB/SRPDB.html
SWISS-PROT	prot. sequ.	HT	flat files	D	√		√	√	http://www.expasy.ch/sprot/

Database	Contents	DB-Links	Implementation	Acquisition	Querying/Data Retrieval				URL
					FF	AH	FTP	BH	
SWISS-2dPAGE	proteomic	HT	flat files	?	√		√	√	http://www.expasy.ch/ch2d/
TAIR	genetic	HT	obj. DBMS	C, D	√		√		http://www.arabidopsis.org/
TIGR	mixed type	HT	rel. DBMS	D	√		√		http://www.tigr.org/tdb/
tmRDB	nucl. sequ.	HT	flat files	D, L				√	http://psyche.uthct.edu/dbs/tmRDB/tmRDB.html
TRANSFAC	genetic	HT	rel. DBMS	L	√	√	√	√	http://transfac.gbf.de/TRANSFAC/index.html
Transterm	genetic	–	rel. DBMS	D	√		√		http://biochem.otago.ac.nz/Transterm
TrEMBL	prot. sequ.	HT	flat files	D	√		√		http://www.expasy.ch/sprot/sprot-top.html
TRIPLES	genetic	HT	rel. DBMS	C	√		√		http://ygac.med.yale.edu
TRRD	mixed type	HT	flat files	?				√	http://wwwmgs.bionet.nsc.ru/mgs/dbases/trrd4/
UK CropNet	genetic	HT	ACEDB	C	√	√	√	√	http://ukcrop.net/
UTRdb	nucl. sequ.	HT	flat files	D	√	√	√	√	http://bigarea.area.ba.cnr.it:8000/EmbIT/UTRHome/
WIT	met. pathw.	HT	?	D	√				http://wit.mcs.anl.gov/WIT2/
WormPD	proteomic	HT	rel. DBMS	C, L	√			√	http://www.proteome.com/databases/index.html
YIDB	genetic	HT	flat files ?	D, L				√	http://www.EMBL-Heidelberg.DE/ExternalInfo/seraphin/yidb.html
YPD	proteomic	HT	rel. DBMS	C, L	√			√	http://www.proteome.com/databases/index.html
ZmDB	genetic	HT	ACEDB	C, D	√	√	√	√	http://zmdb.iastate.edu/