Integration of Biological Sources: Exploring the Case of Protein Homology

Tjeerd W. Boerman[1], Maurice van Keulen[1], Paul van der Vet[1], and Edouard I. Severing[2]

[1]Faculty of EEMCS, University of Twente
POBox 217, 7500 AE Enschede, The Netherlands
[2]Laboratory of Bioinformatics, Wageningen University
Dreijenlaan 3, 6703 HA Wageningen, The Netherlands

Abstract

Data integration is a key issue in the domain of bioinformatics, which deals with huge amounts of heterogeneous biological data that grows and changes rapidly. This paper serves as an introduction in the field of bioinformatics and the biological concepts it deals with, and an exploration of the integration problems a bioinformatics scientist faces. We examine ProGMap, an integrated protein homology system used by bioinformatics scientists at Wageningen University, and several use cases related to protein homology. A key issue we identify is the huge manual effort required to unify source databases into a single resource. Uncertain databases are able to contain several possible worlds, and it has been proposed that they can be used to significantly reduce initial integration efforts. We propose several directions for future work where uncertain databases can be applied to bioinformatics, with the goal of furthering the cause of bioinformatics integration.

1 Introduction

The purpose of bioinformatics is to apply statistics and computer science to information used in—or generated by—the field of molecular biology in order to gain new insights. Such information includes nucleic acid sequences, protein sequences, macromolecular structures and functions, expression patterns, and networks of metabolic pathways [30]. In the last three decades the amount of data generated by biological researchers has increased dramatically, due to increased effort by the scientific community and new high throughput technologies for producing data. A landmark in molecular biology was the Human Genome Project (HGP), which completed the identification of all 20,000-25,000 genes in human DNA in 2003, after 13 years of work[11].

Since that moment the genomic data has been constantly evolving and changing due to new research, and some of the original data has even been proven false. Nevertheless, the initial identification still stands as a major accomplishment. In 2005 the nucleotide sequence databanks contained $80,000 * 10^6$ bases, which is 26 times the size of the human genome [30]. Figure 1 shows the number of entries over time in GenBank, one of the major genetic sequence databases [34].

After a scientific publication, experimental results are curated and annotated by archives that choose to include the data in their next release. Derived databases collect information from one or more archival databases, and extend it with with annotations based on additional analyses and research. Examples of such database analyses are the identification of mutations in DNA and protein sequences, and the derivation of relationships between entries (Lesk 2002). In 2007 the amount of available biomolecular databases had increased to nearly 900 [19]. Integration tasks have become common, and consume a significant part of a molecular biologists' time. Manual integration of different data sources used to be very normal, but with the exponential increase of data and databases manual integration has quickly become infeasible. The scien-

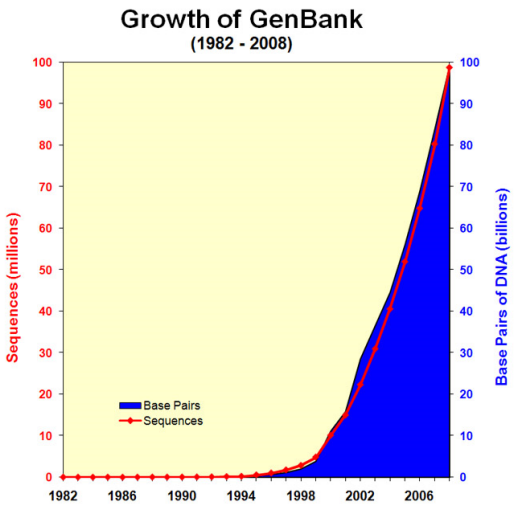


Figure 1: The amount of base pairs and sequences available from the GenBank database has shown enormous growth in the last decade.

tific community has recognized this problem, and it has become a major focus of the data integration field [24].

The complex and dynamic nature of biological data has proven a challenge for integration attempts. Many systems have been developed in an attempt to create the panacea for data integration, but the problem persists to this day. The Pay-As-You-Go Data Integration for Bio-Informatics (PayDIBI) project aims to contribute by developing a data coupling and integration technology that supports bioinformatics scientists in quickly constructing targeted data sets for researching questions that require the combination of information from several databases. This paper is a first step toward that goal, being both an introduction to the relevant bioinformatics concepts and an exploration of the integration problem and its characteristics. Our true goal for this paper is to show what properties of biological data(bases) frustrate the integration process, and to determine possible avenues that can be pursued in further research.

The rest of this paper is organized as follows. Since this paper's intended audience also consists of computer scientists, Section 2 introduces some basic biological concepts that are important for understanding the field of bioinformatics. An understanding of these concepts and the relations between them is required to reason about their integration, and to understand bioinformatics literature. It is also important to understand the information sources a bioinformatics researcher has at his disposal. For this reason bioinformatics databases are described in Section 3, analyzing what types of biological data are stored and in what formats. Next, Section 4 covers some of the approaches that have been suggested for data integration and aggregation. A case study of an actual workflow of bioinformatics scientists at Wageningen University is described and analyzed. From this workflow, three use cases on the subject of protein homology are introduced in Section 5. Section 6 describes how the researchers currently go about using them. It also contains information about ProGMap, the protein homology integration tool the researchers use. Finally, in Section 7 the biggest integration concerns in bioinformatics are enumerated, and we analyze to what extent ProGMap deals with them.

2 Biological Concepts

Molecular biologists are concerned with understanding the interactions between the various systems of a cell, such as those involving DNA, RNA and proteins, as well as learning how these interactions are regulated. In the next sections we describe some of these basic cellular concepts, and the data that is associated with them.

2.1 DNA and RNA

Each organism's genetic material is contained in DNA, or in some viruses in RNA. These are long chains of four possible nucleotides that contain the genetic instructions used in development and functioning. Sequences of nucleotides, the building blocks of DNA and RNA, are represented using the first letter of the bases adenine (A), cytosine (C), guanine (G), thymine (T) and uracil (U). In RNA chains, the uracil base replaces the thymine base. The complete set of genetic material of an organism is called a genome. Segments of DNA that code for proteins or RNA chains that have a function in the organism are called genes.

In a process called transcription, shown in Figure 2, the nucleotide base pairs of a gene are copied onto RNA. In eukaryotes, which include humans, the primary transcript that is formed after the initial transcription is subjected to post-transcriptional modifica-

tions before producing the mature mRNA. In a process called translation the mature mRNA sequence is decoded to produce a protein: triplets of nucleotides from the mRNA sequence, called codons, code for amino acids. Twenty naturally-occurring amino acids exist, and most are associated with several different nucleotide triplets. In bioinformatics databases, each amino acid and DNA nucleotide is identified by a letter, usually the first letter of the full name.

2.2 Proteins

A protein is a complex organic macromolecule that is composed of one or more chains of amino acids. They are responsible for the structure and activities of organisms, and are involved in virtually all cell functions. Proteins help contract our muscles and digest our food, and specialized proteins called antibodies defend the body from antigens. Protein synthesis is a complex process that involves many components inside a biological cell, and is very similar in all living organisms [28]. Typically, proteins are 200-400 amino acids long, and they fold to form a native three-dimensional structure. Unlike protein synthesis, which requires immensely complicated cell machinery, folding is in most cases spontaneous. Proteins show a great variety of three-dimensional conformations.

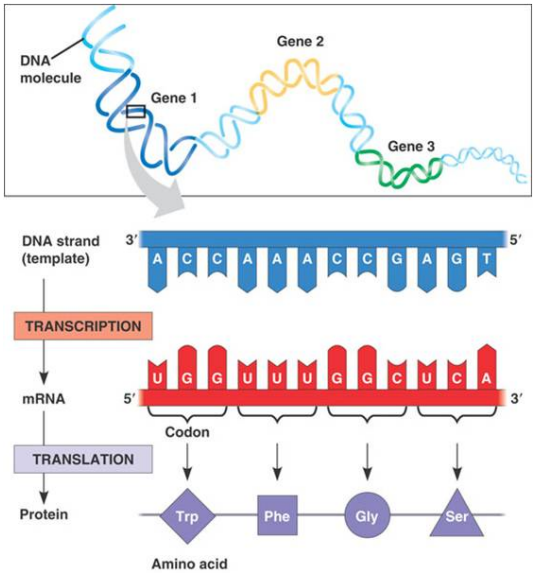


Figure 2: Synthesis of a protein requires transcription and translation of a gene [3].

It is clear that DNA and proteins are strongly related:

- The DNA sequence indirectly determines the sequence of the result protein.
- A protein's sequence usually determines its three-dimensional structure.
- A protein's three-dimensional structure determines its function.

Synthesis of proteins is controlled by regulatory mechanisms that "deliver the right amount of the right function to the right place at the right time." [30].

2.3 Homology

The use cases we examined at Wageningen University are all related to protein homology. Homology is the relationship of two characters that have descended, usually with divergence, from a common ancestral character [18]. A character can be any genic, structural or behavioral feature of an organism. Homology is different from analogy, where the characters are also similar but have descended separately from unrelated ancestral characters. Homology is an abstraction because it is a relationship—common ancestry—which we can only infer with more or less certainty based on the biological evidence we have at our disposal. The above definitions are very general; in molecular biology, homology almost invariably refers to similarities in the nucleotide sequences of DNA and RNA, or the amino acid sequences of proteins.

We distinguish between two major, disjoint subtypes of homology: orthology and paralogy. Genes and proteins that are orthologs or paralogs of each other do not necessarily have identical sequences, but their sequences will be very similar, and they often, but not always have the same function. Sequences are orthologous if they exist in different species and were separated by a speciation event. Such an event occurs when a species diverges into two different species. The resulting two species both carry a version of the original gene, and these versions are orthologous. The common ancestor of the two genes lies in the cenancestor, the most recent common ancestor of the taxanomic group under consideration. In Figure 3 the genes B1 and B2 are orthologous to A1. By contrast, paralogy among genes is the result of gene duplication. Gene duplication occurs when a gene in an organism is copied to occupy two different positions in the same

genome. Due to lack of the original selective pressure on the new instance of the gene, it can easily mutate and possibly acquire new functions. In Figure 3 genes B1 and B2 are paralogous to each other. Paralogs initially belong to the same species but this does not have to remain so [26].

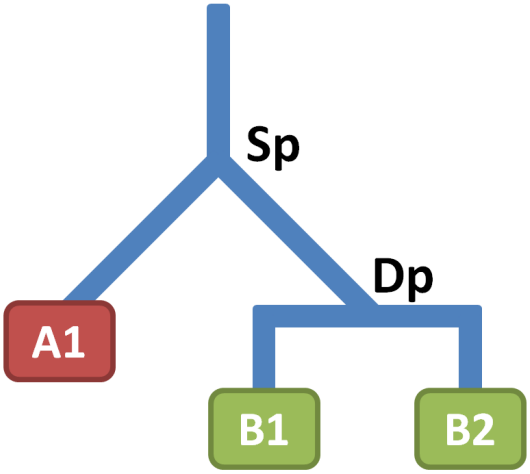


Figure 3: The abstracted evolution of a gene is shown from a common ancestor to populations A and B. The speciation event Sp is denoted by an upside down Y shaped junction, while the gene duplication event Dp is displayed as a horizontal bar. The speciation event results in the inception of species A and B. Two genes whose common ancestor resides at a horizontal bar (duplication event) are paralogs, and homologs in the case where the ancestor resides at a speciation event.

Summarizing, orthologs are genes derived from a common ancestor through vertical descent (or speciation) and can be thought of as the direct evolutionary counterpart. In contrast, paralogs are genes that are the result of gene duplication.

2.4 Sequence Alignment

An isolated protein or genetic sequence does not increase a researcher's understanding of biology, nor does a collection of sequences. The data can be effectively used for medical or biological research only if one can extract functional insight from the sequences. Through statistical analysis of protein sequences and structures, a bioinformatics scientist can predict their function and structure if only a sequence is available. For example, if a new protein's sequence is very similar to a number of fully annotated protein sequences, the new protein is likely to have a similar structure and function.

The process of performing sequence comparison is called sequence alignment: a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences [32]. Sequence alignment is the basic tool of bioinformatics.

Multiple sequence alignment (MSA) algorithm development is an active area of research [16]. A description of the algorithms is not in the scope of this document, but the basic principle is to align the sequence residues that correspond while maintaining their ordering. Gaps may be introduced, but the order of the residues must be maintained and no residues can be deleted to achieve alignment. An example of multiple sequence alignment is displayed in Figure 4.

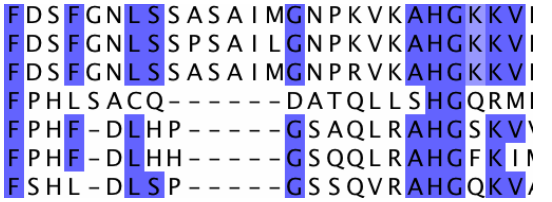


Figure 4: A multiple sequence alignment of protein sequences. Gaps are introduced to achieve alignment. In this case a deletion of five residues possibly occurred in the bottom three sequences [2].

Many algorithms exist, with differing complexities and speeds. A common workflow is to use a very fast algorithm to search a database for roughly similar sequences, then examine these candidate sequences with a slower, more accurate algorithm. Common algorithms are the pairwise comparison algorithm BLAST [4] and MSA algorithm CLUSTAL W [46].

3 Bioinformatics Databases

A databank comprises a database with information, a logical organization of that information, and tools to disclose it. Since the inception of bioinformatics in the nineteen-eighties many databanks have been created, covering nucleic acid and protein sequences, macromolecular structures and functions, expression

patterns and networks of metabolic pathways and control cascades [30]. Most databases are publically available online, but some are privately owned and used for commercial purposes. This section introduces some major types of biological data, the largest public databases that store such data and the data formats that are used in their releases. New releases by most major databases are made after constant cycles that last a handful of weeks or months.

Data enters the scope of bioinformatics when a scientist submits experimental results to an archive. The archives may perform tasks such as curation and annotation on the data before including it in its next version. Annotation is the task of adding layers of analysis and interpretation to raw data. Yakel defines curation as "the active involvement of information professionals in the management, including the preservation, of digital data for future use" [49]. New versions of an archive are called releases, and are periodically made available. Other information-retrieval projects may integrate the data from archives into their databases. They may reorganize the data structure or provide alternate tools or interfaces to access the data. Reorganization may include extracting subsets of the data, deriving new information or reannotating the data [30].

Manual curation is an important tool in assuring the quality of databases. The largest databases employ dozens of biocurators who make sure that the entries are standardized and annotated so that other biologists can understand them. These employees will read the paper that accompanies the data and use their expert knowledge and a lot of cross-referencing to validate and process the data. Extensive domain knowledge and expertise are required to assess the quality of the research and its results [43]. Due to the enormous amount of data that the scientific community generates a lot of data remains untouched by manual curators. Often databases will computationally generate annotations and functional classifications for this data, in an attempt to give at least some meaning to the entries. The TReMBL section of the UniProtKB database employs this method [47].

3.1 Nucleotide Sequence Databases

DNA and RNA sequences are undoubtedly the central type of data in bioinformatics. A comprehensive public database is GenBank, which is built primarily from the submission of sequence data from researchers and bulk submissions of sequence centers that use high-throughput methods. It currently comprises over 25 billion nucleotide bases and over 260,000 named species are represented [6]. Similar databases are the EMBL Nucleotide Sequence Database [27] and the Ribosomal Database Project [10].

The central piece of information in these databases is of course the nucleotide sequence of a DNA or RNA region, such as a gene. Such sequences are annotated with additional information, which almost invariably includes:

- An accession number, a unique identifier which is assigned by the database.
- The species in which the sequence was found.
- Taxonomic information.

The annotations may contain many other types of data, including information about the product protein, supporting evidence and crosslinks to other databases, among other things. Most sequence databases support several output formats for the sequences. For example, GenBank supports a plain format, an idiosyncratic GenBank format and the FASTA format. The FASTA format is a very common format for sequence data. It is derived from conventions of FASTA, a program for FAST alignment by W.R. Pearson [30]. A sequence in FASTA format begins with a single-line description. A greater than character (>) must be the first character, and the rest of the title line is arbitrary but should be informative and is often used to contain accession numbers and other identifiers. An example of the FASTA format is:

```
gi>|14456711|ref|NM_000558.3|Homo sapiens
hemoglobin, alpha 1 (HBA1), mRNA
ACTCTTCTGGTCCCCACAGACTCAGAGAGAACCCACCAT
GGTGCTGTCTCCTGCCGACAAGACCAACGTCAAGGCCGC
CTGGGGTAAGGTCGGCGCGCACGCTGGCGAGTATGGTGC
```

This is a truncated FASTA representation of the mRNA sequence of human hemoglobin type alpha 1, commonly referred to as HBA1.

3.2 Ontology

An ontology defines a collection of representational primitives with which to model a domain of knowledge. It is an abstraction of real world objects in the domain into classes, attributes and relationships, creating a common vocabulary [37]. Ontologies are structured in a hierarchical fashion, with general concepts

appearing at the top of the tree, much like a biological taxonomy [44]. In bioinformatics, several ontology databases have been created with the objective of resolving semantic and schematic heterogeneity across databases and species. The use of ontology databases by GenBank and EMBL, for example, facilitates uniform queries across them. The most prominent bio-ontology is Gene Ontology (GO), a web-based, open source database managed by the Gene Ontology Consortium [23]. It attempts to provide a structured vocabulary in three domains: molecular function, biological process and cellular component. A particularly important point is that, being an ontology, it does not name actual genes or gene products. An example ontology is displayed in Figure 5.

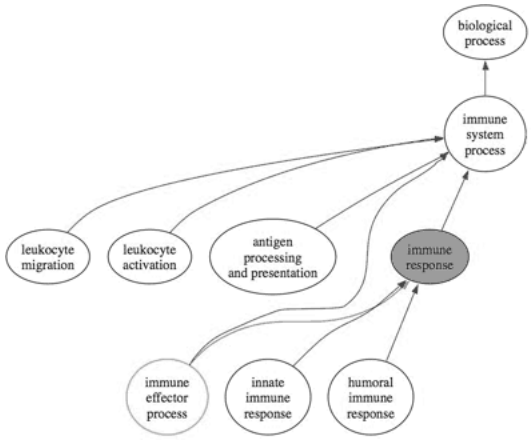


Figure 5: An example of what a small section of an ontology might look like. The image displays a part of the top level structure of the immune system processes in GO. Ontologies usually are directed acyclic graphs, and this is also the case here: the immune effector process has an outgoing relation with two nodes, but there are no cycles [38].

There are many ontology resources apart from GO, although some of them will cross-reference to GO entries. These resources include Sequence Ontology [17] and Protein Ontology [33]. Many exist under the umbrella of the Open Biomedical Ontologies (OBO) consortium, which aims to impose a set of principles to give coherence to the ontological efforts across the community [45]. Openness, common representation and independence are some of the criteria the OBO emphasizes in order to prevent massive syntactic and semantic heterogeneity.

An ontology is only useful when database entries are annotated with its terms, and this annotation needs to have a reasonable degree of accuracy. This is not the case in every bioinformatics database. For example, while the Gene Ontology Annotation (GOA) database is a major project that annotates Uniprot Knowledgebase entries with GO terms [9], the Ribosomal Database Project does not provide any form of GO annotations.

3.3 Protein Sequence

Protein sequences databases contain entries with the sequence of amino acids in a protein as the central piece of data. One of the largest databases in this domain is the Universal Protein Resource (or UniProt), which was formed by uniting the Swiss-Prot, TrEMBL and PIR protein database activities [5]. It is web-accessible, and with millions of entries it is one of the largest—if not the largest—databases of its kind. The central hub of protein knowledge in UniProt is the UniProt Knowledge Base (UniProtKB). It provides a unified view of protein sequence and functional information [31]. It consists of two sections: UniProtKB/Swiss-Prot and UniProtKB/TrEMBL. UniProtKB/Swiss-Prot is manually curated and contains annotations based on experimental results, computated features and scientific conclusions that are manually entered by field experts, resulting in highly reliable information. Computationally generated annotations are used in UniProtKB/TrEMBL in an attempt to give meaning to sequences that have not been manually annotated yet.

Other protein sequence databases include the Transporter Classification Database, which is a web-accessible and curated database containing information on proteins in membrane transport systems [42] and COMBREX, which includes experimentally determined and computationally predicted functions for more than three million microbial genes [41].

3.4 Protein Structure

Proteins mostly automatically fold into three-dimensional structures. This process is displayed in Figure 6. Such a shape, or conformation, usually determines the chemical action a protein performs, and with that its biological function. The comparison of 3D protein structures may reveal biologically

interesting similarities that are not detectable by comparing the protein sequence.

The tertiary structures—the atomic coordinates in 3D—of many different proteins have been solved with methods such as X-ray crystallography and nuclear magnetic resonance (NMR) spectroscopy [8]. Researchers have created dozens of databases that store this structure information [20]. The central repository in use today is the Protein Data Bank (PDB) that was established in 1971 [7]. It is web-accessible and currently contains over 70,000 structures.

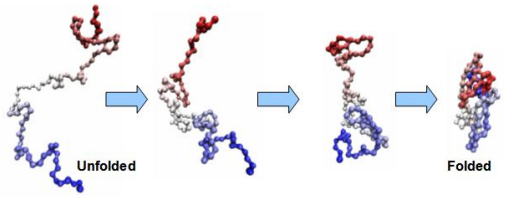


Figure 6: A protein exists as an unfolded polypeptide after synthesis. The general behavior of such polypeptides is that amino acids interact with each other to produce a well-defined three-dimensional structure, known as the native state. Many environmental factors can influence the folding process, and failure to fold into native structure can be toxic: several neurogenerative and other diseases are believed to result from misfolded proteins [1].

3.5 Protein Classification

In the previous sections we established a sense of the basic information units a molecular biologist deals with. Many data sources do not focus on these basic objects, however. There are many databases that process such basic information to provide meta-data or relationships. If we intend to research improvements to bioinformatics databases, we must understand the semantics and origin of this information as well.

One major example is protein classification. A protein family is a group of proteins that are related evolutionary. Proteins that have descended from a common ancestor are homologous, typically have similar three-dimensional structures and functions, and show significant sequence similarity. While it is difficult to prove homology irrefutably, scientists have made great progress in using sequence alignment methods to infer these phylogenetic relationships: proteins that do not share a common ancestor are very unlikely to show statistically significant sequence similarity. There are many databases available that classify proteins using different group terms such as families, domains or clusters, based on different criteria.

The PIRSF database is a network classification system based on evolutionary relationships of whole proteins. It provides clustering of proteins in the Protein Information Resource, which is part of the UniProt consortium [48]. PIRSF is publically available on the web and is being updated continuously, containing around 32,000 clusters in 2004 and 35,000 in 2006 [36]. It uses several classification methods including multiple sequence alignment and Hidden Markov Models [15]. Practically this means that if one has a PIR accession number that refers to a protein, it is possible to look up the family/families that the protein belongs to. PIRSF uses a system with superfamilies, families and subfamilies that are arranged in a hierarchical fashion [40]. Each family level is based on different principles, with the superfamilies being the most general.

HomoloGene is a database of both curated and calculated gene orthologs and homologs for 20 organisms including human, mouse, rat, zebrafish and cow [35]. Sequences are compared using a BLAST algorithm, then matched up and put into groups, using a taxonomic tree built from sequence similarity. We were unable to find any literature authored by the designers of HomoloGene.

The Ensembl project is a genome information system that is freely available on the web [25]. It includes an extensive comparative genomics pipeline that clusters proteins in Ensembl along with proteins from UniProtKB, and the web interface can provide information about orthologues, paralogues and protein families. Relationships between sequences can also be displayed graphically, Figure 7 shows a part of the gene tree the web interface can show.

All of the databases discussed in this section provide downloads for their classification data. It is important to realize that there are many additional classification databases that we did not describe.

4 Integration Research

The goal of this paper is both to serve as an introduction, and to explore the integration problems in bioinformatics. Having described the fundamentals

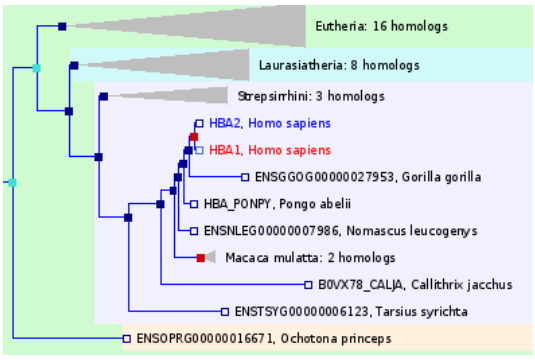


Figure 7: Part of a gene tree that Ensembl can generate. The image shows genes related to the human HBA1 gene, which is responsible for a type of hemoglobin.

of bioinformatics and the information that molecular biologists deal with, we will now describe some additional research on integration. In a survey of integration of biological and genomic sources, Hernandez and Kambhampati describe the main integration approaches that have been adopted [24]. They identify warehouse integration, mediator-based integration and navigational integration. Several characteristics of bioinformatics sources are described, and it is proposed that these properties make it difficult for biologists to use the sources in combination with each other. The characteristics include:

- The highly diverse nature of the data.
- The representational heterogeneity of the data.
- The autonomous and web-based character, and the different ways in which the data is published.
- The various interfaces and querying capabilities of the sources.

Hammer and Schneider distinguish two commonly accepted approaches to integration systems. Query-driven integration or mediation means that no data is stored in the integration system, and queries are answered by individually querying the underlying source databases. The data warehousing approach entails a system that fetches the data from the source databases, integrates it, and stores it locally. The approaches differ in many ways including performance, querying capabilities, ownership of data. In their research Hammer and Schneider propose a unified database that integrates the source databases in the form of a unified data warehouse, but have no clear solution for the semantic heterogeneity problem [22].

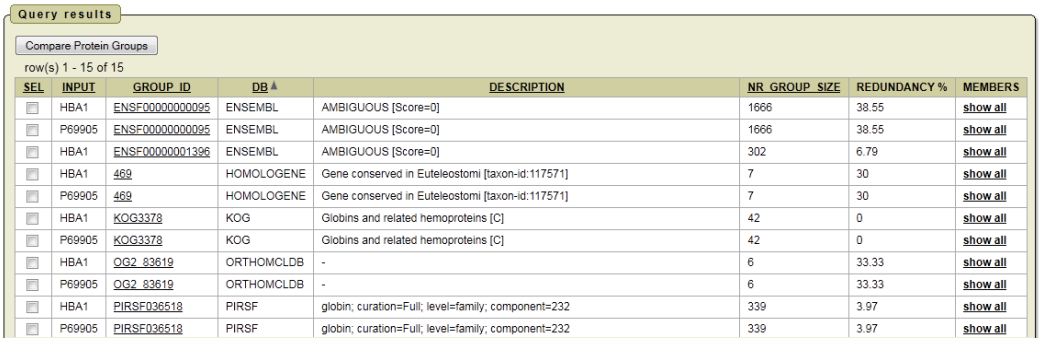
A key step in data integration is schema matching or mapping. A 2001 survey of automatic schema matching approaches by Rahm et al. distinguishes between schema- and instance-level, element- and structure-level, and language- and constraint-based matchers [39]. Some of the discussed schema matchers can handle mixing different types of schemas—such as XML and relational schemas.

Work by Halevy et al. recognizes the challenge that integration poses in today's organizations [21]. There is often an information need that relies on large numbers of diverse, interrelated data sources, but no means of managing them in a convenient, integrated or principled fashion. They propose dataspaces as a data management abstraction for these diverse applications, and associated Dataspace Support Platforms (DSPPs) that provide the required services over dataspaces. A DSPP can start off very simple, with limited semantic integration that can be improved as time progresses. Data integration of multiple data sources can result in uncertainty. An interesting aspect of this research is that DSSPs have a mechanism that models inconsistencies and states of uncertainty about data.

This approach to data conflicts is further explored in work by De Keijzer and Van Keulen, who suggest a "good is good enough" approach, where unresolved conflicts and uncertainty are stored in the database [14]. Such an uncertain database stores *possible worlds* instead of having one view of the data that is considered to be true. Feedback on query answers can be used to remove impossible worlds, and improve data quality. We were not able to find any literature that applies this approach to bioinformatics databases.

5 Use Case Descriptions

In order to get an understanding of the integration problems bioinformatics scientist face "in the wild", we describe several use cases related to protein homology in this section. Proteins are homologous when they are derived from a common "ancestor", we described several variations of homology in Section 2.3. Analysis of homology relations can help reveal information about a protein's function, structure or phylogeny. We consulted bioinformatics scientists at Wageningen University in order to learn about common homology use cases, and the tools they use in their homology research. The use cases are:


Query results

Compare Protein Groups

row(s) 1 - 15 of 15

SEL	INPUT	GROUP_ID	DB	DESCRIPTION	NR_GROUP_SIZE	REDUNDANCY %	MEMBERS
	HBA1	ENSF00000000095	ENSEMBL	AMBIGUOUS [Score=0]	1666	38.55	show all
	P69905	ENSF00000000095	ENSEMBL	AMBIGUOUS [Score=0]	1666	38.55	show all
	HBA1	ENSF00000001396	ENSEMBL	AMBIGUOUS [Score=0]	302	6.79	show all
	HBA1	469	HOMOLOGENE	Gene conserved in Euteleostomi [taxon-id:117571]	7	30	show all
	P69905	469	HOMOLOGENE	Gene conserved in Euteleostomi [taxon-id:117571]	7	30	show all
	HBA1	KOG3378	KOG	Globins and related hemoproteins [C]	42	0	show all
	P69905	KOG3378	KOG	Globins and related hemoproteins [C]	42	0	show all
	HBA1	OG2_83619	ORTHOMCLDB	-	6	33.33	show all
	P69905	OG2_83619	ORTHOMCLDB	-	6	33.33	show all
	HBA1	PIRSF036518	PIRSF	globin; curation=Full; level=family; component=232	339	3.97	show all
	P69905	PIRSF036518	PIRSF	globin; curation=Full; level=family; component=232	339	3.97	show all

Figure 8: A part of the ProGMap web-interface, showing the results of a query: a list containing information about the different groups in which a protein was found.

Interaction networks Many biological functions are performed by protein-complexes rather than by individual proteins. Consider the case where high throughput experiments have resulted in the availability of a large set of protein-complexes for species A. An initial strategy for determining whether a particular protein-complex in species A also exists in species B is to determine whether species B has orthologs for each of the interacting proteins in species A. If orthologs can be found in species B for all interacting proteins in species, then this is a clue that the interaction network is also present in species B.

Function annotation A new genomic region has been sequenced, and a protein sequence has been predicted from this data. Finding sequentially related proteins or protein clusters in a homology database can reveal information about the new sequence: if it is very similar to a group of homologous proteins, then the associated protein is likely to have similar function.

Phylogeny A homologous protein can be used to study the evolutionary relations, or phylogeny, between species. If a protein can be found that has orthologs in several species, then the sequence difference between these proteins can give clues about the relations between species and their place in the evolutionary tree of life.

6 ProGMap

To help answer their homology queries, researchers at Wageningen University developed ProGMap, an integrated annotation resource for protein orthology [29]. It is a single-entry web-tool that allows for simultaneous searches in several protein grouping databases including UniProt, RefSeq, Ensembl, COG, KOG, OrthoMCL-DB, PIRSF and Homologene. The ProGMap designers recognized that current databases use a variety of classification schemes and methods, but that unifying the protein databases would require a major effort and it would lead to a loss of information. For these reasons two important design goals were:

- To preserve all the information present in the underlying datasets.
- Creating a database that can be maintained in a largely automated fashion.

Any database-specific identifier can be used as the entry point for a homology search. For example, identifiers such as P69905, NP_000549 and ENSP00000251595 used by UniProt, RefSeq and Ensembl databases, respectively, are all allowed. A search is executed through the web-interface displayed in Figure 8. We will now turn to the internals of ProGMap, and describe its integration mechanisms.

ProGMap uses a centralized data warehouse approach to store all protein-to-protein, protein-to-group and group-to-group mappings, as well as functional descriptions of proteins and groups. A relational database stores this information, which is retrieved from the source databases using a Sequence Retrieval Server, as well as modules written in Perl. Protein sequences are hashed using the MD5 algorithm to form uniquely labeled Protein Identity Groups (PIG). Each PIG corresponds with entries in one or several source databases. A simplified database schema is displayed in Figure 9, and the full version is included in Appendix A.

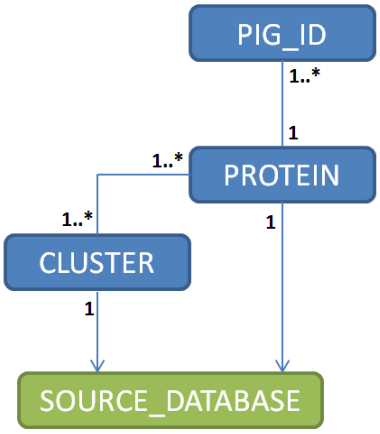


Figure 9: The central elements of the ProGMap database are the tables that contain protein and cluster information. Each protein has a single corresponding PIG_ID, but one PIG_ID can refer to multiple protein entries that originate in different databases. Protein entries with the same PIG_ID invariably have the same sequence. Every protein and cluster entry contains a reference to its source database.

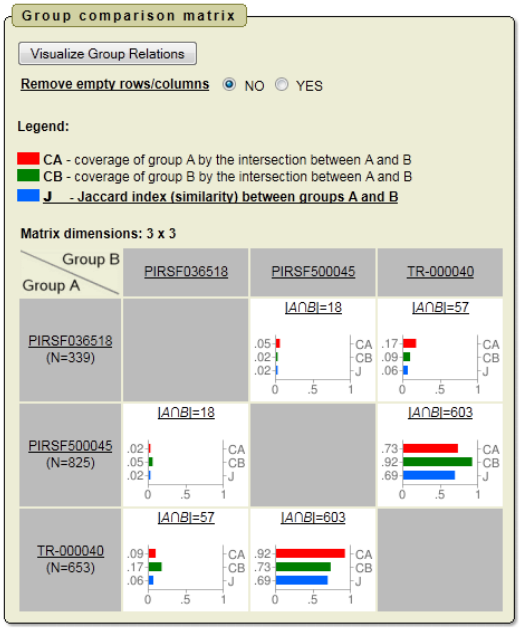


Figure 10: A pair-wise comparison of a set of three hemoglobin protein groups. The results show that the overlap between PIRSF500045 and TR-000040 is very large, while other overlaps are smaller.

Once the database has been established, additional mappings are established by determining which groups share at least one common member. Such mappings are purely conceptual: they are not stored in the database but determined at query-time. These group-to-group relations are used in the web-interface to display comparison of groups, see Figure 10.

6.1 Use Cases

6.1.1 Interaction Networks

To see if species B has a protein interaction network that is known to exist in species A, a researcher needs to search for an ortholog in species B for every protein in the species A's interaction network. This search problem is displayed in Figure 11. Using the ProGMap web interface it is possible to run a query on multiple input proteins, but this is semantically equal to running a search on each individual protein and simply combining the search results. Also, there is currently a lack of support for taxonomic information in the web interface. The database does contain taxonomic information (see Appendix A), but custom scripts that directly access the database are required to disclose this information. This is complicated by the fact that ProGMap currently does not have a public API to facilitate custom scripts [29].

6.1.2 Function Annotation

Next we look at the functional annotation case: a new protein sequence has been identified and a researcher wants to investigate the function of this protein. The first step is to run a similarity search to find out which known proteins are most similar to the new sequence. The ProGMap provides a page to perform BLAST queries using the protein sequence in FASTA format. After a candidate protein is selected, a simple ProGMap query for that protein will find all the clusters containing that protein, as displayed in Figure 12. These clusters have functional annotations, originating from the source database. These functional annotations are often unreliable: they can be too general, incorrect, or simply not present. Also, annotations from different databases often do not match (completely). It is up to the bioinformatics scientist to analyze and value the annotations based on knowledge about the

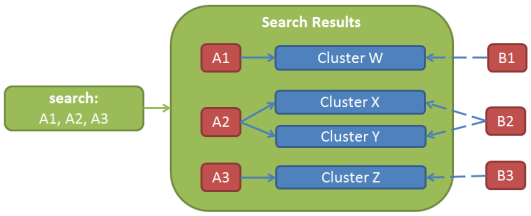


Figure 11: A search is executed on three proteins from species A using ProGMap, returning several homology clusters. The search for homologous proteins in species B (B1, B2 and B3) is not possible with the current web interface, and bioinformatics researchers have to write custom scripts to verify if, for example, B1 is actually present in cluster W. The A2 protein is found to be present in two clusters: cluster X and Y. These clusters might both contain B2, complicating matters further. Real life searches can be much more complicated, making the scripting approach tedious and time-consuming.

databases and domain knowledge.

INPUT	GROUP_ID	DB	DESCRIPTION
NP_247002	COG1460	COG	Uncharacterized protein conserved in archaea [S]
NP_247002	OG2_105968	ORTHOMCLDB	-
NP_247002	PIRSF005053	PIRSF	RNA polymerase, subunit F; curation=Full/Desc.; level=family; component=10024
NP_247002	TR-009241	TRIBES	Hypothetical protein

Figure 12: A search for the protein M. jannaschii (RefSeq: NP_247002) returns four different group annotations. The PIRSF group annotates the protein as "RNA polymerase subunit F", and the annotation indicates manual curation, which can be an argument for accepting the annotation as the answer to the query. There is a high degree of overlap between the groups, but still the annotations are all different. This is the kind of problem bioinformatics scientist deal with on a daily basis: different sources say different things; can they be combined; and which source do you trust the most?

6.1.3 Phylogeny

A protein that contains homologs in many different species can be used to study the species' phylogeny. Identifying a protein that has a homologous counterpart in many species is straightforward: a protein of each species should be represented in the same homology cluster. However, answering such a query is currently not possible with the ProGMap web interface. One might think custom scripts can be used since the cluster information from the source databases is stored in the ProGMap database. The problem with this approach is the lack of a unified view: different clusters contain different proteins, so there might be many possible candidates for the phylogenetic study. Which set of proteins is the best to proceed with? In the next section we look at this issue, as well as other integration concerns.

7 Integration Concerns

In this section we look at the integration problems that we encountered when examining the use cases to answer the following question: where does the information need not match with the information ProGMap can provide? After identifying these situations, we attempt to connect them with research from the integration field that could help mitigate the problems.

First we analyze the integration method used by ProGMap. A small amount of data is currently stored for proteins and clusters in a single relational database. This includes the protein and cluster accession number and the source database the accession number refers to. There is some extra data such as the cluster description and a protein's taxonomic information, but most of the information remains in the source databases. When answering queries ProGMap does not act as a mediator to the source databases: it is both the integration system and the data repository, which would mean it is a pure data warehouse. However, the proteins that ProGMap returns are simple identifiers and hyperlinks to the original database. This means that even though the data required for the integration is stored in the data warehouse, the source databases are still required to interpret the search results. If the sources disappear then ProGMap is still heavily affected, albeit indirectly.

Next we look at the semantics of the clusters. Clusters from the different source databases are based on different clustering methods, and it is not unlikely that some clusters are based on specific forms of homology, such as paralogy, or involve some functional or structural properties into the grouping algorithm. Even if two source database's grouping methods seem similar, the clusters might contain different (possibly con-

flicting) annotations. All of the downloadable cluster annotations each database provides are imported into ProGMap, and when we examined the description field we identified many conflicts. Uncertainty resulting from informational heterogeneity among source databases is currently left completely to the ProGMap user, who has to choose between different clusters.

We identify four main problems in bioinformatics data integration [12][24][22]:

1. *Similar data can be contained in several sources, but represented in a variety of ways.* This representational heterogeneity encompasses structural, naming and semantic differences. Each source may refer to the same semantic concept with its own identifier, for example in the case of protein accession numbers. This leads to entity identification problems across sources. A related problem arises when different databases use the same identifier (i.e. an ontological term) for different semantic concepts.
2. *Two or more databases might hold additive or conflicting information.* Even if it is possible to resolve all schematic and semantic differences, sources that contain data for the same semantic object might contain different data, or have some missing data, creating possible inconsistencies between sources. This uncertainty about the distribution of information across source databases also leads to apprehension that essential information will be overlooked.
3. *A familiar database can disappear or change at any time.* Most databases operate autonomously, so they are free to modify their structure or functionality with prior "public" notice. Such changes will most likely have significant impact on any integration systems that depend on them. Data warehouses become outdated when a source database updates its content, as often happens with biological sources. If a source database disappears completely then any mediator-based integration systems based on that source will malfunction, and data warehouses can no longer update their data.
4. *Differing querying capabilities.* Some databases do not provide (all of) their data in a downloadable format, and many databases do not have public API's. Automated querying through the web-interface is possible but tedious and possibly very limiting.

Now we analyze how ProGMap deals with these problems:

1. The representational heterogeneity is manually resolved at design-time. An importer or driver component exists for each source database; it retrieves the semantically correct pieces of information and integrates them into the ProGMap data warehouse, resolving any schematic differences. The only entity identification mechanism is applied to proteins: the PIG identifier. This means proteins are distinguished based on their amino acid sequence rather than their database accession number.
2. Data conflicts and inconsistencies are left completely untouched by ProGMap. Result clusters from all source databases are returned separately in a list, and analyzing the semantic similarities and differences is left to the user. This would be the place to start if one wants to provide a more unified answer to the query.
3. ProGMap uses a data warehousing approach, so if a database becomes inaccessible it can still return result clusters for that database. However, in this event it is no longer possible to look up the full annotations of the cluster or its proteins in the database's web interface, so the data that remains in ProGMap is not of much use.
4. If a source database changes its schema then the ProGMap importer needs to be changed.
5. The web interface of ProGMap provides a unified way of querying, but only databases that provide a downloadable archive can be integrated. One positive effect of this restriction is that when a source database changes its web interface functionality this does not impact the ProGMap system. Ideally, it should be possible to integrate information from all databases, even those without downloadable archives.

It is clear that ProGMap does not solve all major integration problems. One might think that unifying and/or reclassifying source databases to provide a single truth would make for a promising integration system, see Figure 13. The ProGMap designers identified two major drawbacks to this approach. First, the effort required would be immense, similar to establishing and maintaining a new, curated protein database. Second, the individual classification schemes of the databases represent a very important

added value which would go at least partly lost if replaced with a new classification scheme [29]. These are valid reasons, but it does not change the fact that ProGMap's current integration is very limited, and that the system does not fully meet the information need of its users. Is there a middle ground between the current system and a fully integrated and unified database?

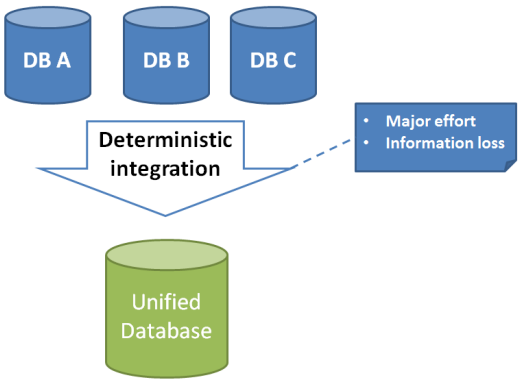


Figure 13: In an ideal world a biologist uses a single, accurate, up to date data source. Doing a full, deterministic integration of all source databases into a unified database with only truths is a huge task, however. The information loss problem is very prominent in homology integration, since every source database uses its own classification scheme. Since the semantics of the clusters differ, it is impossible to merge them without losing information.

Work by de Keijzer and van Keulen on uncertain databases leads us to believe there are opportunities. Posing queries to an uncertain database means that it may return several possible answers, with associated probabilities [13]. The advantage of this approach is that conflict resolution can be postponed until after the initial integration. It would be very interesting to see how this can be applied to ProGMap, where conflict resolution is currently an untouched problem that is left up to the user: uncertain databases have a potential to significantly decrease the effort required for an (initial) integration. The main point here is that a system that returns a single answer with 90% certainty can be more useful than a system that returns several answers and no hints on which answer is true. The cluster annotation use case could benefit from this, for example. Certainly, there is no substitute for a biologists' expertise, but it is a direction worth exploring.

The use of uncertain database opens up opportunities for user feedback to reduce the uncertainty of the data, by removing possible worlds [13]. We are unsure if a biologist has enough knowledge to remove possible worlds at query-time, but it is likely that some bogus annotations that the initial integration missed can be corrected. The combination of user feedback and uncertain biological data is complicated by the regular database updates that occur in bioinformatics, but definitely represent an opportunity for future research.

Reducing the effort of the initial integration is one concern; the other concern is losing information when integrating the homology databases. Since the semantics of the clusters differ between databases, it is impossible to merge them without losing information. However, there is significant overlap in the cluster semantics that could be capitalized on to create clusters with more general semantics. Such a system can still be very useful. It is also worth looking at other domains in bioinformatics where there might be more agreement on semantics.

8 Conclusions

In this paper we introduced the field of bioinformatics, the biological concepts it deals with and the data that is used. The data is contained in hundreds of different databases that are very diverse, containing information ranging from protein and DNA sequences to ontological and phylogenetic information. This huge amount of data also grows and changes rapidly. Efforts to integrate this data are frustrated by the autonomy of the databases as well as their diversity: many databases use different semantics and structure.

We looked at the ProGMap tool, an integrated resource for protein orthology that was developed at Wageningen University, and studied how bioinformatics scientists use the system. In an ideal world a researcher would use a single authoritative resource, and ProGMap is an effort in that direction. We found that the integration the system does is very limited however: many data conflicts are left up the user to solve, and many use cases still require a lot of manual work. Work in the data integration field proposes the concept of uncertain databases, where data conflicts are left unresolved and a single database can contain several possible worlds. No research combining bioinformatics databases and uncertain databases seems to exist, but we see opportunities for decreasing the man-

ual work required for bioinformatics integration. We suggest several directions for future research:

- Studying the effect of uncertain databases on the effort required to achieve integration homology databases.
- Examining processes through which uncertain biological data can be improved. For example, when an uncertain database is in place, user feedback can be used to improve the data quality, by removing "impossible worlds".
- There are numerous biological data domains with more agreement on semantics than among protein homology databases. Intuitively, such domains require less effort to integrate into a unified database. The application of uncertain databases to such types of biological data seems a direction worth exploring.

References

[1] Dimensionality reduction methods for molecular motion. http://cnx.org/content/m11461/latest/, June 2011.

[2] Introduction to protein structure and function - ELLS torino jan 2009. http://www.embl.de/~seqanal/courses /proteinEvolutionEllsEmblSept2009/intro ToProteinStructureAndFunction.html, June 2011.

[3] Protein synthesis. http://www.ifm.ac.tz/cim/studentis /registration/fees/protein-synthesis&page=7, June 2011.

[4] S.F. Altschul, W. Gish, W. Miller, E.W. Myers, and D.J. Lipman. Basic local alignment search tool. *Journal of Molecular Biology*, 215(3):403–410, October 1990.

[5] A. Bairoch. The universal protein resource (UniProt). *Nucleic Acids Research*, 33(Database issue):D154–D159, December 2004.

[6] D.A. Benson, I. Karsch-Mizrachi, D.J. Lipman, J. Ostell, and D.L. Wheeler. GenBank. *Nucleic Acids Research*, 36(Database):D25–D30, December 2007.

[7] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, et al. The protein data bank. *Nucleic Acids Research*, 28(1):235–242, January 2000.

[8] A.T. Brunger, P.D. Adams, G.M. Clore, W.L. DeLano, P. Gros, et al. Crystallography & NMR system: A new software suite for macromolecular structure determination. *Acta Crystallographica Section D: Biological Crystallography*, 54(5):905921, 1998.

[9] E. Camon. The gene ontology annotation (GOA) database: sharing knowledge in uniprot with gene ontology. *Nucleic Acids Research*, 32(90001):262D–266, January 2004.

[10] J.R. Cole, B. Chai, T.L. Marsh, R.J. Farris, Q. Wang, et al. The ribosomal database project (RDP-II): previewing a new autoaligner that allows regular updates and the new prokaryotic taxonomy. *Nucleic Acids Research*, 31(1):442 – 443, January 2003.

[11] F.S. Collins, M. Morgan, and A. Patrinos. The human genome project: lessons from large-scale biology. *Science*, 300(5617):286, 2003.

[12] S.B. Davidson, C. Overton, and P. Buneman. Challenges in integrating biological data sources. *Journal of Computational Biology*, 2(4):557–572, January 1995.

[13] A. de Keijzer and M. van Keulen. User feedback in probabilistic integration. In *Database and Expert Systems Applications, 2007. DEXA '07. 18th International Workshop on*, pages 377–381, 2007.

[14] A. de Keijzer and M. van Keulen. IMPrECISE: good-is-good-enough data integration. In *Data Engineering, 2008. ICDE 2008. IEEE 24th International Conference on*, pages 1548–1551, 2008.

[15] S.R. Eddy. Hidden markov models. *Current opinion in structural biology*, 6(3):361365, 1996.

[16] R.C. Edgar and S. Batzoglou. Multiple sequence alignment. *Current Opinion in Structural Biology*, 16(3):368–373, June 2006.

[17] K. Eilbeck, S.E. Lewis, C.J. Mungall, M. Yandell, L. Stein, et al. The sequence ontology: a tool for the unification of genome annotations. *Genome biology*, 6(5):R44, 2005.

[18] W.M. Fitch. Homology:: a personal view on some of the problems. *Trends in genetics*, 16(5):227231, 2000.

[19] M.Y. Galperin. The molecular biology database collection: 2007 update. *Nucleic Acids Research*, 35(SUPPL. 1):D3–D4, 2007.

[20] M.Y. Galperin and G.R. Cochrane. The 2011 nucleic acids research database issue and the online molecular biology database collection. *Nucleic Acids Research*, 39(Database):D1–D6, December 2010.

[21] A. Halevy, M. Franklin, and D. Maier. Principles of dataspace systems. In *Proceedings of the twenty-fifth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, page 19, 2006.

[22] J. Hammer and M. Schneider. Genomics algebra: A new, integrating data model, language, and tool for processing and querying genomic information. In *Proceedings of the 2003 CIDR Conference*, 2003.

[23] M.A. Harris, J. Clark, A. Ireland, J. Lomax, M. Ashburner, et al. The gene ontology (GO) database and informatics resource. *Nucleic Acids Research*, 32(Database issue):D258–261, January 2004. PMID: 14681407.

[24] T. Hernandez and S. Kambhampati. Integration of biological sources: current systems and challenges ahead. *ACM SIgmod Record*, 33(3):5160, 2004.

[25] T.J.P. Hubbard, B.L. Aken, S. Ayling, B. Ballester, K. Beal, et al. Ensembl 2009. *Nucleic Acids Research*, 37(Database):D690–D697, January 2009.

[26] R.A. Jensen. Orthologs and paralogs - we need to get it right. *Genome Biology*, 2(8):interactions1002.1–interactions1002.3, 2001. PMID: 11532207 PMCID: 138949.

[27] C. Kanz. The EMBL nucleotide sequence database. *Nucleic Acids Research*, 33(Database issue):D29–D33, December 2004.

[28] M. Kozak. Comparison of initiation of protein synthesis in procaryotes, eucaryotes, and organelles. *Microbiology and Molecular Biology Reviews*, 47(1):1, 1983.

[29] A. Kuzniar, K. Lin, Y. He, H. Nijveen, S. Pongor, and J.A.M. Leunissen. ProGMap: an integrated annotation resource for protein orthology. *Nucleic Acids Research*, 37(Web Server):W428–W434, June 2009.

[30] A.M. Lesk. *Introduction to bioinformatics*. Oxford University Press, 2002.

[31] M. Magrane and U. Consortium. UniProt knowledgebase: a hub of integrated protein data. *Database*, 2011(0):bar009–bar009, March 2011.

[32] D. Mount. *Bioinformatics sequence and genome analysis.* Could Spring Harbor, New York, 2000.

[33] D. Natale, C. Arighi, W. Barker, J. Blake, T.-C. Chang, et al. Framework for a protein ontology. *BMC Bioinformatics*, 8(Suppl 9):S1, 2007.

[34] NCBI. GenBank growth, May 2011.

[35] NCBI. HomoloGene home. http://www.ncbi.nlm.nih.gov/homologene, June 2011.

[36] A.N. Nikolskaya, C.N. Arighi, H. Huang, W.C. Barker, and C.H. Wu. PIRSF family classification system for protein functional and evolutionary analysis. *Evolutionary bioinformatics online*, 2:197, 2006.

[37] N.F. Noy, D.L. McGuinness, et al. *Ontology development 101: A guide to creating your first ontology*. Citeseer, 2001.

[38] The Gene Onthology Project. GO annotation for the immune system. http://www.geneontology.org/GO.immunology.shtml, June 2011.

[39] E. Rahm and P.A. Bernstein. A survey of approaches to automatic schema matching. *The VLDB Journal*, 10:334–350, 2001. 10.1007/s007780100057.

[40] Protein Information Resource. A proposal for the PIRSF classification system, May 2003.

[41] R.J. Roberts, Y.-C. Chang, Z. Hu, J.N. Rachlin, B.P. Anton, et al. COMBREX: a project to accelerate the functional annotation of prokaryotic genomes. *Nucleic Acids Research*, 39(Database):D11–D14, November 2010.

[42] M.H. Saier. TCDB: the transporter classification database for membrane transport protein analyses and information. *Nucleic Acids Research*, 34(90001):D181–D186, January 2006.

[43] K. Sanderson. Bioinformatics: Curation generation. *Nature*, 470(7333):295–296, February 2011.

[44] N. Schuurman and A. Leszczynski. Ontologies for bioinformatics. *Bioinform Biol Insights*, 2:187–200, 2008. PMID: 19812775 PMCID: 2735951.

[45] B. Smith, M. Ashburner, C. Rosse, J. Bard, W. Bug, et al. The OBO foundry: coordinated evolution of ontologies to support biomedical data integration. *Nat Biotech*, 25(11):1251–1255, November 2007.

[46] J.D. Thompson, D.. Higgins, and T.J. Gibson. CLUSTAL w: improving the sensitivity of progressive multiple sequence alignment through sequence weighting, position-specific gap penalties and weight matrix choice. *Nucleic acids research*, 22(22):4673, 1994.

[47] C.H. Wu. The universal protein resource (UniProt): an expanding universe of protein information. *Nucleic Acids Research*, 34(90001):D187–D191, January 2006.

[48] C.H. Wu, A. Nikolskaya, H. Huang, S.L Yeh, D.A. Natale, et al. PIRSF: family classification system at the protein information resource. *Nucleic acids research*, 32(suppl 1):D112, 2004.

[49] E. Yakel. Digital curation. *OCLC Systems & Services*, 23(4):335340, 2007.

A ProGMap DB Schema

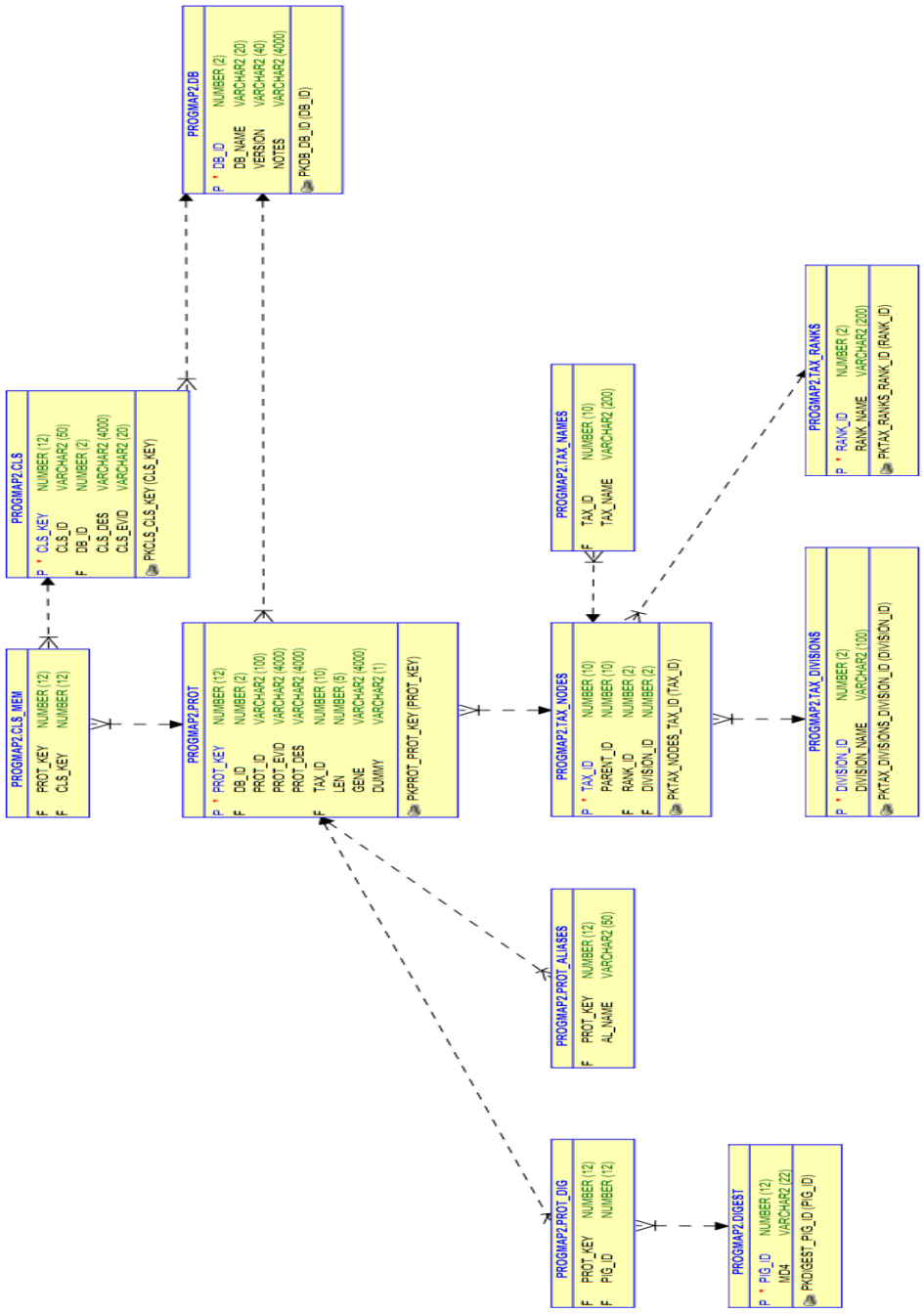


Figure 14: ProGMap DB Schema