A protein structure based annotation of genomes

ARNE MÜLLER

2002

Biomolecular Modelling Laboratory
Cancer Research UK
44 Lincoln's Inn Fields, London, WC2A 3PX

and

Department of Biochemistry and Molecular Biology
University College London
Gower Street, London, WC2E 6BT

Submitted in part fulfilment of the requirements for the degree of Doctor of Philosophy in Biochemistry of the University of London.

Abstract

A strategy for protein structure and function based annotation of genomes was developed, evaluated and applied to the proteins of several genomes including the human genome.

First the performance of the widely-used homology-based sequence comparison program PSI-BLAST to detect distant homologous relationships (<20% sequence identity) was evaluated. The benchmark is based on two sets of sequences from the Structural Classification Of Proteins (SCOP) database for which the homologous relationships are known. About 40% of the test proteome can be annotated via remote homologies. Common sources of errors are identified. PSI-BLAST is applied to assign homologues of known structure and function to proteins of *M. genitalium* and *M. tuberculosis.* From the benchmark, the number of missed assignments and the potential extent of new structural and functional families was estimated.

An automated proteome annotation system was developed to perform large scale annotations based on analyses such as PSI-BLAST. Computationally intensive analyses can be distributed across several computers. The system is based on a relational database serving as a back-end and a software interface as a front-end. Relational storage of results from different analyses permits straightforward evaluation of results and the comparison of annotations across genomes.

The above annotation system was applied to fourteen proteomes including the human proteome. The extent and reliability of structural and functional annotation in these proteomes was evaluated and compared. About 40% of the human proteome can be assigned to protein folds. For 77% of the proteome there is some functional information, but only 26% of the proteome can be assigned to the standard sequence motifs that characterise function. There are substantial differences in the composition of membrane proteins between the proteomes in terms of their globular domains. Commonly occurring structural superfamilies are identified and compared across the proteomes. The frequencies of these superfamilies leads to the estimate that 98% of the human proteome evolved by domain duplication, with four of the ten most duplicated superfamilies potentially specific for multi-cellular organisms. Occurrence of domains in repeats is more common in metazoa than in single-cellular organisms. Superfamily pairs co-occurring in the same protein sequence were analysed and compared across the proteomes. Structural superfamilies over- and under-represented in human disease genes were identified.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The available sequence data from the finished genome projects provides biological science with a huge and valuable source of data. The genetic information together with its derived data such as protein sequences and structures, expression levels and sub-cellular location has to be managed, understood and exploited for human benefit. It is a long and challenging way from the raw sequence data (the genome) to only a basic understanding of how an organism developed in evolution and how it functions. It is not just the sum of the parts that makes life but a complex regulatory network of interactions involving many components. The sequence data is further analysed in large scale experiments such as expression profiles and protein interaction networks which in turn increases the amount data to be analysed dramatically. Bioinformatics organises and integrates all parts of the experimentally generated data as well as connecting them to gain understanding of biological systems.

Bioinformatics is a relatively young discipline as a science with components from software engineering. Bioinformatics aims to analyse and understand biological data, but a hypothesis is not necessarily required when it comes to the description, management and interpretation of the experimentally generated data. Currently, the development of new algorithms, recycling of algorithms from other areas such as natural language processing, data management, the interpretation of data and their relationships as well as supporting biologists working in a specific system is included in bioinformatics.

This work contains a software engineering component, the development of an automated annotation system that integrates existing data and methods to perform a scientific analysis of the integrated data. The results are of interest from the scientific point of view (bringing insight into commonalities and differences between genomes) and from the software engineering point of view (the annotation system may be used to support biologists and could be a platform for further developments).

1.1 Genome sequencing projects

As of November 2001 there were 67 completely sequenced bacterial and archaea bacterial genomes and eleven eukaryotic genomes (for which at least one chromosome has been sequenced) available. The draft human genome sequence with >3,000 mega bases was published in February 2001. Table 1.1 gives an overview of the finished sequencing projects. In addition there are roughly 300 ongoing prokaryotic and about 80 eukaryotic public and commercial sequencing projects (data from Integrated Genomics Inc., http://wit.integratedgenomics.com/GOLD, Bernal *et al.* (2001)). Many of the sequenced genomes are from pathogenic organisms such as the recently published *Yersinia pestis* genome that causes plague (Heidelberg *et al.*, 2000) or the two *Salmonella* strains (Parkhill *et al.*, 2001a; McClelland *et al.*, 2001). The genome sequence reveals many secrets about the organism that may help to identify potential drug targets. The ideal target might be a key protein in an essential pathway specific to the pathogenic organism.

species (+strain)	size	genes
Archaea		
Methanococcus jannaschii DSM 2661 (Bult *et al.*, 1996)	1664 Kb	1750
Methanobacterium thermoautotrophicum delta H (Smith *et al.*, 1997)	1751 Kb	1918
Archaeoglobus fulgidus DSM4304 (Klenk *et al.*, 1997)	2178 Kb	2493
Pyrococcus horikoshii (shinkaj) OT3 (Kawarabayasi *et al.*, 1998)	1738 Kb	1979
Aeropyrum pernix K1 (Kawarabayasi *et al.*, 1999)	1669 Kb	2620
Pyrococcus abyssi GE5 (no reference)	1765 Kb	1765
Halobacterium sp. NRC-1 (Ng *et al.*, 2000)	2014 Kb	2058
Thermoplasma acidophilum (Ruepp *et al.*, 2000)	1564 Kb	1478
Thermoplasma volcanium GSS1 (Kawashima *et al.*, 2000)	1584 Kb	1524
Sulfolobus solfataricus P2 (She *et al.*, 2001)	2992 Kb	2977
Sulfolobus tokodaii 7 (Kawarabayasi *et al.*, 2001)	2694 Kb	2826
Bacteria		
Haemophilus influenzae KW20 (Fleischmann *et al.*, 1995)	1830 Kb	1850
Mycoplasma genitalium G-37 (Fraser *et al.*, 1995)	580 Kb	468
Synechocystis sp. PCC6803 (Kaneko *et al.*, 1996)	3573 Kb	3168
Mycoplasma pneumoniae M129 (Himmelreich *et al.*, 1996)	816 Kb	677
Escherichia coli K12- MG1655 (Blattner *et al.*, 1997)	4639 Kb	4289

continued on next page

continued from previous page

species (+strain)	size	genes
Helicobacter pylori 26695 (Tomb *et al.*, 1997)	1667 Kb	1590
Bacillus subtilis 168 (Kunst *et al.*, 1997)	4214 Kb	4099
Borrelia burgdorferi B31 (Fraser *et al.*, 1997)	1230 Kb	1256
Aquifex aeolicus VF5 (Deckert *et al.*, 1998)	1551 Kb	1544
Mycobacterium tuberculosis H37Rv (lab strain) (Cole *et al.*, 1998)	4411 Kb	4402
Treponema pallidum subsp. pallidum Nichols (Fraser *et al.*, 1998)	1138 Kb	1041
Chlamydia trachomatis serovar D (Stephens *et al.*, 1998)	1042 Kb	896
Rickettsia prowazekii Madrid E (Andersson *et al.*, 1998)	1111 Kb	834
Helicobacter pylori J99 (Alm *et al.*, 1999)	1643 Kb	1495
Chlamydia pneumoniae CWL029 (Kalman *et al.*, 1999)	1230 Kb	1052
Thermotoga maritima MSB8 (Nelson *et al.*, 1999)	1860 Kb	1877
Deinococcus radiodurans R1 (White *et al.*, 1999)	3284 Kb	3187
Ureaplasma urealyticum serovar 3 (Glass *et al.*, 2000)	751 Kb	650
Campylobacter jejuni NCTC 11168 (Parkhill *et al.*, 2000b)	1641 Kb	1654
Chlamydia pneumoniae AR39 (Read *et al.*, 2000)	1229 Kb	1052
Chlamydia trachomatis MoPn Nigg (Read *et al.*, 2000)	1069 Kb	924
Neisseria meningitidis MC58 (serogroup B) (Tettelin *et al.*, 2000)	2272 Kb	2158
Neisseria meningitidis Z2491 (serogroup A) (Parkhill *et al.*, 2000a)	2184 Kb	2121
Bacillus halodurans C-125 (Takami & Horikoshi, 2000)	4202 Kb	4066
Chlamydia pneumoniae J138 (Shirai *et al.*, 2000)	1228 Kb	1070
Xylella fastidiosa CVC 8.1.b clone 9.a.5.c (Simpson *et al.*, 2000)	2679 Kb	2904
Vibrio cholerae serotype O1, Biotype El Tor, strain N16961 (Heidelberg *et al.*, 2000)	4000 Kb	3885
Pseudomonas aeruginosa PAO1 (Stover *et al.*, 2000)	6264 Kb	5570
Buchnera sp. APS (Shigenobu *et al.*, 2000)	640 Kb	564
Mesorhizobium loti MAFF303099 (Kaneko *et al.*, 2000)	7596 Kb	6752
Escherichia coli O157:H7 EDL933 (Perna *et al.*, 2001)	4100 Kb	5283
Mycobacterium leprae TN (Cole *et al.*, 2001)	3268 Kb	1604
Escherichia coli O157:H7. Sakai (Hayashi *et al.*, 2001)	5594 Kb	5448
Pasteurella multocida Pm70 (May *et al.*, 2001)	2250 Kb	2014
Caulobacter crescentus (Nierman *et al.*, 2001)	4016 Kb	3737
Streptococcus pyogenes SF370 (M1) (Ferretti *et al.*, 2001)	1852 Kb	1696
Lactococcus lactis IL1403 (Bolotin *et al.*, 2001)	2365 Kb	2266
Staphylococcus aureus N315 (Kuroda *et al.*, 2001)	2813 Kb	2594
Staphylococcus aureus Mu50 (Kuroda *et al.*, 2001)	2878 Kb	2697
Mycobacterium tuberculosis CDC 1551 (no reference)	4403 Kb	4187
Mycoplasma pulmonis (Chambaud *et al.*, 2001)	963 Kb	782
Streptococcus pneumoniae TIGR4 (Tettelin *et al.*, 2001)	2160 Kb	2094
Clostridium acetobutylicum ATCC 824D (Nolling *et al.*, 2001)	4100 Kb	4927
Sinorhizobium meliloti 1021 (Galibert *et al.*, 2001)	6690 Kb	6205
Streptococcus pneumoniae R6 (Hoskins *et al.*, 2001)	2038 Kb	2043
Agrobacterium tumefaciens C58 (Wood *et al.*, 2001)	4915 Kb	4554
Rickettsia conorii Malish 7 (Ogata *et al.*, 2001)	1268 Kb	1374
Yersinia pestis CO-92 Biovar Orientalis (Parkhill *et al.*, 2001b)	4653 Kb	4012
Salmonella typhi CT18 (Kuroda *et al.*, 2001)	4809 Kb	4600
Salmonella typhimurium,LT2 SGSC1412 (McClelland *et al.*, 2001)	4857 Kb	4597
Listeria innocua Clip11262, rhamnose-negative (Glaser *et al.*, 2001)	3011 Kb	2981
Listeria monocytogenes EGD-e (Glaser *et al.*, 2001)	2944 Kb	2855
Eukaryota		
Saccharomyces cerevisiae S288C (No authors listed, 1997)	12069 Kb	6294
Caenorhabditis elegans (The C. elegans Sequencing Consortium, 1998)	97000 Kb	19099
Drosophila melanogaster (Adams *et al.*, 2000)	137000 Kb	14100
Arabidopsis thaliana (The Arabidopsis Genome Initiative, 2000)	115428 Kb	25498

continued on next page

continued from previous page

species (+strain)	size	genes
Guillardia theta (Douglas *et al.*, 2001)	551 Kb	464
Leishmania major Friedlin Chromosome 1 (Myler *et al.*, 1999)	257 Kb	79
Plasmodium falciparum 3D7 Chromosome 2 (Gardner *et al.*, 1998)	947 Kb	205
Plasmodium falciparum 3D7 Chromosome 3 (Bowman *et al.*, 1999)	1060 Kb	220
Homo sapiens (Lander *et al.* (2001) and Venter *et al.* (2001))	>3000 Mb	35000

Table 1.1: Finished genome projects (status in November 2001). The size of the genome is given in thousand base pairs (Kb) or million base pairs (Mb), *genes* is the number of identified genes. The data of this table is taken from the *GOLD* database at http://wit.integratedgenomics.com/GOLD (Bernal *et al.*, 2001).

1.2 Introduction into genome annotation

A standard component of any genome project is an overall annotation. Having the genome sequence alone does not substantially help to understand the biology of the organism. In the following sections the major steps in genome annotation are represented. Protein sequences are the starting point for any annotation in this work, and therefore the following sections focus on protein sequences.

1.2.1 Finding genes in genomes

The first important step in annotating the genome is to identify the genes within the genomic sequence. It is worth mentioning the basic methods used in identifying genes as well as associated problems and errors, because these can have an effect of 'downstream' analyses (e.g. analyses based on genes and proteins). An introduction into gene finding is given in a review by Stein (2001).

In bacteria, genes may be identified by just looking for the longest open reading frame (ORF) defined by a start and a stop codon. The Shine-Dalgarno sequence, which is a polypurine (adenine and guanine) sequence shorter then ten nucleotides at the 3' end of the gene (about 7 nucleotides 5' of the start codon), helps to identify the location of a gene within the genome. In addition to start and stop codon location, codon usage can be used in gene finding. Similar sequences with a common evolutionary origin (homologues) from already annotated genomes are considered to confirm the location of genes in a newly sequenced genome. The genomic DNA sequence is translated in all three reading frames on both nucleotide strands (in direction of translation, from 3' to 5') to produce long theoretical peptide sequences which are compared to known proteins from other organisms. Nevertheless, Skovgaard *et al.* (2001) showed that the number of genes in bacteria is generally overpredicted (in *A. pernix* they estimated 100% gene overprediction which is by far the most extreme in their analysis).

Gene identification in eukaryotic genomes is far more problematic than in prokaryotic genomes. This is due to the exon-intron structure of genes and the lack of obvious sequence features such as a Shine-Dalgarno sequence to distinguish between coding and non-coding regions . Despite the start codon there is no clear landmark where a gene starts on a eukaryotic chromosome. Rule based *ab initio* gene identification methods such as GeneScan (Burge & Karlin, 1997) or Grail (Uberbacher & Mural, 1991; Roberts, 1991; Xu *et al.*, 1994) that employ statistical methods (for example hidden Markov models, see section 1.3.7), have been shown to identify only 40% of the existing genes with their exon-intron structure. About 70% of these predictions are to some extent wrong, i.e. do not corresponds to the correct gene structure (Reese *et al.*, 2000). On the other hand 90% of the predictions include at least a fraction of the real gene. The use of experimental data as described above for bacterial gene identification improveseukaryotic gene finding. For example, the human genome sequence as defined by the ENSEMBL project version 1.2 (Hubbard *et al.* (2002), http://www.ensembl.org), contains more than 150,000 predicted genes, but only about 25,000 genes are either confirmed by expressed sequenced tags (ESTs derived from mRNA of expressed genes) or homologues in a different organism. Because of the extensive exon-intron structure and the small fraction of actual coding sequences in the human genome (estimated at about 1.5% of the genome, Lander *et al.* (2001)), two predicted genes may in fact be one larger gene, or a larger gene may be in fact several genes. A positive view on the human genome shows that 25,000 of at least 30,000 genes have been identified with the help of experimental data (ESTs and homologues), which corresponds to nearly 85% of the estimated number of genes in the genome.

The expected number of genes in the human genome is between 30,000 and 40,000 (Lander *et al.*, 2001), thus there are theoretically still 5,000 to 15,000 genes missing. The genome sequences of other higher eukaryotes, in particular those of mouse (*M. musculus*), rat (*R. norvegicus*) and the puffer fish (*Fugu rubripes*) will help to identify genes within these genomes and that of human, because of the higher sequence conservation within exons compared to non coding regions. The mouse and rat genome projects were established mainly because these organisms are used as models in biology. The genome sequence (with the confirmed set of genes) will accelerate the progress with which molecular biologists clone and analyse specific parts of the genome. The puffer fish project was deliberately established to enhance gene finding and interpretation of the human genome sequence. A draft sequence of the puffer fish project has been available since October 2001. The extent of the coding sequences is estimated to be similar to that of human, but the overall size of the genome (350 to 400 mega bases) is just about one eighth of the human genome (>3,000 mega bases). The sequence conservation between the dense coding regions of the puffer fish and the corresponding regions in the human genome is expected to reveal currently unidentified genes.

In interpreting results from the analysis of the identified peptide sequence repertoire of a genome one has to keep in mind that the absence of a particular protein does not necessarily mean that the genome contains no coding sequence for this peptide, it may just have been missed in the interpretation of the genome.

1.2.2 Functional classification of genes and proteins

Once the genes are identified within a genome, they have to be functionally characterised. Usually the genes are compared to a set of already functionally characterised genes. Since a protein sequence is more conserved in its amino acid sequence than the corresponding nucleotide sequence of the gene (because of the redundant genetic code), sequence comparisons for functional annotation are performed at the peptide level.

Function, at the level of a functional classification of proteins, is the description of the biochemical function or a combination of several biochemical functions. A functional annotation is generally derived from one or more homologous sequences for which a functional description has been generated previously. However, only for a fraction of annotated proteins has the biochemical activity been proven experimentally (Ursing *et al.*, 2002). Section 1.4.1 discusses the quality and the limitations of functional transfer between homologues.

The majority of proteins in a genome consist of more than one protein domain. A domain can be considered as the smallest functional and evolutionary unit of proteins and is generally found in different proteins in combination with other domains of the same (repeats) or of different type (Apic *et al.*, 2001; Qian *et al.*, 2001a). The potential multi-domain character of proteins may need a list of biochemical functions, which depends on the level detail of the annotation. For example a protein with a NAD(P) binding domain and a dehydrogenase domain may just be described as a dehydrogenase or in more detail as a protein that binds NAD(P) and has a dehydrogenase activity (the NAD(P) binding domain may be a 'helper' domain to fulfil the proteins biochemical function). In most cases the functional annotation does not include the biological function, e.g. a human protease may be found in a different biological context such as digestion, during development or in wound healing. The main concepts in functional protein annotation are:

- Finding a homologous sequence that has been functionally characterised previously, the main databases containing such protein sequences are SwissProt and PIR.
- Identifying domains within a protein sequence via homology. The main domain databases with functional descriptions are PFAM, SMART, ProDom and InterPro. (Structural domain databases are discussed later.)
- Finding conserved patterns or motifs (these motifs are generally shorter than a domain and may not include an independent folding unit). The main databases maintaining collections of patterns or motifs associated with a function are Prosite, Prints and Blocks.

1.2.3 Major resources used in protein annotation

The following sections give a more detailed view of the contents of some of the available databases, including an overview of how these databases are constructed. The first issue each year of the journal *Nucleic Acids Research* (in particular those from 1999 on) contains articles about biological databases. The first 2002 issue describes 112 different specialised biological databases.

The main source database GenBank and EMBL

All the specialised databases described below are based on the basic sequence databases. The major nucleotide sequence databases are GenBank (Benson *et al.*, 2002) and

EMBL (Stoesser *et al.*, 2002). Usually nucleotide sequences (or a nucleotide sequence together with its peptide sequence) are submitted to either of these databases. Also, GenBank and EMBL update each other, so that both databases, with some delay, contain the same sequences. If possible the submitted nucleotide sequences are translated into a theoretical peptide sequence. These peptide sequences generate the TrEMBL database (translated EMBL) and the GenPept database (translations from GenBank). In addition, all publicly available genome sequences are submitted to one of these databases. GenBank and EMBL entries contain information associated with the sequence: literature references, authors, gene or protein names, taxonomic information of the source organism and a feature table that lists all known features (e.g. a ribosomal binding site for a bacterial ORF or an exon for a eukaryotic sequence) with their location in the sequence. GenPept and TrEMBL contain more than 800,000 non-redundant peptide sequences (status 11/2001). EMBL/TrEMBL is available from the EBI (http://www.ebi.ac.uk) and GenBank/GenPept is available from the NCBI (http://www.ncbi.nlm.nih.gov).

The SwissProt protein database

The SwissProt database (Bairoch & Apweiler, 2000) historically collected sequences from protein sequencing experiments, i.e. the sequence information was directly taken from the peptide sequence and not by translating a coding region of a gene. SwissProt (version 40.11) contains 105,322 protein sequences. TrEMBL sequences are transfered to SwissProt if there is sufficient evidence for the existence of the gene product. The procedure for integrating new entries into SwissProt includes reviewing by human experts (database curators) and external consultants with expert knowledge about a particular protein family. A SwissProt entry contains, in addition to the peptide sequence and literature references, comments about the functions associated with the protein (edited by the human experts), keywords that describe the function and a structured feature table that describes regions or positions in the sequence such as post-translational modifications, domains and sites (e.g. an ATP binding site).

The PIR protein database

The Protein Information Resource (PIR, Barker *et al.* (2000)), contains about 200,000 protein sequences (status in 2001). Like SwissProt, the database aims to provide high quality annotation. Automatically generated annotations are reviewed and edited by PIR staff, and consultant scientists who review specific parts of the database. Sequence entries are classified according to their status to which there is evidence of their existence, e.g. for entries that are classified as *experimental* there is some experimental evidence, and predicted proteins from theoretical coding regions are classified as *predicted*. Also the annotation is classified into *validated* or *similarity* according to the available evidence. PIR further clusters sequences in families and superfamilies based on sequence similarity. Because PIR and SwissProt both get their sequences from translated coding regions of the major nucleotide databases, there is redundancy between the two databases.

The PFAM, SMART and ProDdom domain and family databases

The domain and protein family databases described here are generated by splitting protein sequences into domains and then clustering similar domains into a family. Annotating proteins according to their domain composition generally leads to more detail than annotating the protein as a single unit.

PFAM is a database of protein domain families (Bateman *et al.*, 2002), based on protein sequences from SwissProt and TrEMBL. It contains a set of curated multiple sequence alignments, each representing a protein family. From these multiple alignments hidden Markov models (see section 1.3.7) are built, which are in turn used to search the protein sequence databases to find new members and to expand a family. The final database PFAM-A provides a high quality description of the families which can help in annotating newly sequenced genomes. Most of the PFAM-A families also contain a functional text description, cellular location of the members of the family, relevant literature references and links to taxonomic groups in which a family is found. PFAM-A is manually curated. Another part of PFAM (PFAM-B) contains potential domain families for which there is not enough evidence to be placed into PFAM-A. PFAM-B entries are mainly taken from families of the large ProDom database (see below). PFAM-B contains more members and families than PFAM-A but is of lower quality. PFAM-B and ProDom are used to update and curate PFAM-A. PFAM-A version 6.6 (August 2001) contains 3071 families. PFAM is available at The Sanger Centre (http://www.sanger.ac.uk/Software/Pfam).

SMART (a Simple Modular Architecture Research Tool, Letunic *et al.* (2002)), like PFAM, is a domain database but originally focused on domains in eukaryotic signal transduction. Recent SMART versions (November 2001) also include a wide range of other domain types (more than 600 domain families). Domain families are constructed in a similar way to PFAM, but the initial step to create a seed multiple sequence alignment involve manual editing and, if available, consideration of protein structure, or homologues of proteins of known structure. Hidden Markov models are constructed from these alignments that are used to search the protein sequence database to collect new family members. The hidden Markov models are then rebuilt, and the search starts again until no more members are found. In addition each member of a family is compared to the sequence database using the homology search method PSI-BLAST (see section 1.3.5) to collect new family members. Alignments are updated, e.g. when the three dimensional structure of a member is published, to re-assess domain boundaries of the family. SMART is based on sequences from SwisProt and TrEMBL. The database is available at the EMBL (http://smart.embl--heidelberg.de). The web-interface also allows the user to search for proteins of a given domain architecture (domain combinations).

ProDom (Corpet *et al.*, 2000) is a domain database with a larger sequence coverage than PFAM or SMART. Over 75% of the proteins from SwissProt and TrEMBL can be assigned to ProDom families (status 2001). There are about 44,000 ProDom domain families with more than one member. From version 35 onwards, the ProDom database includes manual inspection of protein families by scientific consultants. PFAM-A (see above) is used to increase the quality of ProDom. Domain families are generated via PSI-BLAST homology searches (Sonnhammer & Kahn, 1994). Two proteins may share only one homologous region in their sequence, which can be a single domain or several domains. These regions are then used as queries in subsequent PSI-BLAST searches to find additional significant alignments. This procedure is repeated until the regions cannot be split or truncated anymore because no further homologous regions are found. The identified regions are then considered to be domains, and all homologous regions belong to one family. As a quality control, recent versions of ProDom assign consistency indicators to each family (for example sequence variation within a family). ProDom-GC is a ProDom version that clusters protein sequences from complete genomes into families. Both databases are available at http://prodes.toulouse.inra.fr/prodom/doc/prodom.html.

Motif databases: PROSITE, PRINTS and BLOCKS

The PROSITE database (Falquet *et al.*, 2002) is a collection of pattern descriptions that usually are associated with a biochemical function. These signatures are generated from curated multiple sequence alignments and generally describe conserved positions within a domain family. Signatures are represented as regular expression patterns. Since patterns are not flexible (i.e. a pattern matches a sequence region or it does not), the extent to which patterns identify a particular motif is limited. To overcome this limitation, signature profiles have been developed which assign a score to each of the 20 amino acids at each position of the signature according to the frequency of which each amino acid is found at a particular position. Further, alternative protein structure-based profiles and methods involving hidden Markov models have been employed. A PROSITE entry can be associated with a functional description and reasons that lead the construction of a pattern or profile. PROSITE version 16.50 (November 2001) contains 1103 documents describing 1493 patterns and profiles, and is available at http://www.expasy.org/prosite.html, it is updated in parallel with SwissProt.

PRINTS (Attwood *et al.*, 2002) and PRINTS-S (a recent development of the original PRINTS) is a collection of protein fingerprints. The concept behind fingerprints is that a protein can be represented by several conserved motifs. A fingerprint is an ordered list of these motifs that describes a protein family. PRINTS-S is a database for protein sequences rather than domains, although its components (the single motifs) may be characteristic for a particular type of domain. The procedure to build the fingerprints starts with manual curated multiple sequence alignments, and then a series of conserved regions are extracted to construct motifs. This procedure includes manual intervention. The sequence database is searched iteratively with these motifs to expand and gain confidence of the motifs. PRINTS-S contains its own search software FingerPRINScan. The database is built from SwissProt and TrEMBL. Each entry is associated with bibliographic information, functional descriptions, lists of matching sequences and comments. The database (PRINTS-S version 10, based on PRINTS version 32, November 2001) contains about 9,800 individual motifs and about 1,600 fingerprints. It is available at http://www.bioinf.-man.ac.uk/dbbrowser/PRINTS.

The BLOCKS database (Henikoff *et al.*, 1999, 2000) is similar to PRINTS. It contains a list of motifs that are representative for a family. Motifs in the BLOCKS database are called blocks. To generate these blocks, protein family databases such as PFAM-A, PRINTS, ProDom and Domo (Gracy & Argos, 1998) are used. Sequences for each family of these databases are re-aligned via a non-gapped multiple local alignment procedure and converted into non-overlapping blocks. Thus, the BLOCKS database identifies local motifs within given protein families but does not find new protein families (because it uses domain families of the existing domains databases as input). The BLOCKS database can be searched with sequences via the BLIMPS (Henikoff *et al.*, 1995) program that identifies individual blocks and then combines hits belonging to the same family. Sequences can also be searched against the database via the IMPALA program (see section 1.3.6). BLOCKS (June 1999) contains about 9,500 individual blocks and more than 2,000 families. It is available at http://www.blocks.fhcrc.org.

InterPro: A combination of databases

InterPro (Apweiler *et al.*, 2001), a recent database development from the EBI (http://www.ebi.ac.uk/interpro), integrates most of the above databases. InterPro itself does not contribute any new information, and its power comes from having all the above databases in one place providing a range of evidence for a protein to belong to a certain InterPro entry. InterPro is divided into families (3,532 entries), domains (1,068 entries), repeats (74 entries) and post-translational modifications (15 entries). A short description and an abstract about the biochemical function, the biological role and matches against the SwissProt and TrEMBL databases are included for each entry. InterPro also contains, like recent PFAM versions, families for which the function is unknown, but where there is evidence for the conservation of this family, domain or motif.

A family can be described by a set of characteristics from the above databases, e.g. the thiolase family (InterPro entry IPR002155) is described by two PFAM entries and three Prosite patterns. Sequences can be searched against InterPro via the InterProScan software package (Zdobnov & Apweiler, 2001).

InterPro is a 'modern' database. It is distributed in XML format and is, together with the integrated search engine InterProScan, a step towards solving common bioinformatics problems such as standardisation, automatisation and distribution.

A list of InterPro families is now commonly reported as an initial analysis of a newly sequenced genome (e.g. Lander *et al.* (2001); Rubin *et al.* (2000) and http://www.-ebi.ac.uk/proteome).

1.2.4 Gene Ontology (GO), a controlled vocabulary for genome annotation

A recent commentary published in the journal Nature (Pearson, 2001) summarises problems and inconsistencies in gene (and protein) nomenclature and stresses the importance of an ontology for gene names and functions to overcome problems in annotation. In GO, descriptive terms and phrases are used to annotate a gene rather than using gene and protein names such *PMS1* or *TFIIA*. These terms are organised in a hierarchy (a tree of terms and phrases) with the more general terms such as *transcription* or *fatty acid metabolism* as the root for more detailed terms or phrases such as *RNA polymerase II transcription factor* or *fatty acid hydrolase*. The set of terms and phrases is stored in a central GO database maintained at Stanford University. However, different GOs may be constructed for special purposes. New terms can be inserted into the GO-tree. GO is also able to cope with synonyms and can describe biological function. Using a system with a controlled vocabulary organised in a tree as in GO allows automatic comparison of annotations between genomes at different levels of the tree (i.e. at different level of detail, for example to test for the existence of enzymatic pathways between genomes). The central GO resource is located at http://www.geneontology.org, see also Lewis *et al.* (2000); Ashburner *et al.* (2000); The Gene Ontology Consortium (2001).

1.2.5 Putting everything together to find pathways

At a higher level, genome annotation aims to identify complete biological subsystems such as metabolic pathways or signalling pathways. The usual approach is to compare all members of a pathway (e.g. for glycolysis) in a model organism to the proteins of a newly sequenced genome. The comparison is carried out via the standard homology search methods (see section 1.3 below). This approach generally identifies the fundamental pathways such as glycolysis in a newly sequenced genome. If members of a pathway cannot be identified, this does not necessarily mean the pathway is incomplete. The homology based comparison may just have missed some members of that pathway because of insufficient similarity (although the homologues are present), or there may be alternative routes bypassing the known proteins of that pathway. There are three major database systems available that implement the above approach for metabolic pathways: The partly freely available WIT system from *Integrated Genomics* (this system is now known as *ERGO* and is no longer freely available for academic use, http://www.integratedgenomics.com/), the KEGG (Kanehisa *et al.*, 2002) database (Kyoto Encyclopedia of Genes and Genomes) freely available for academic use and EcoCyc (Karp *et al.*, 2002), a system that describes metabolic pathways in *E. coli* (this database recently has been made freely available for academic users).

The publication of the genome sequence of the cholera bacterium *V. cholerae* (Heidelberg *et al.*, 2000) contains an overview of some of the identified pathways in this bacterium and can serve as an example of how to represent complex pathways information in a comprehensive way (see figure 1.1).

1.3 Homology based sequence comparison methods

If two genes or proteins have diverged from a common ancestor they are by definition homologues. Further, homologues within the same species are paralogues, and often have different functions due to specialisation. The closest homologues with generally the same biochemical function in two species are orthologues (Tatusov *et al.*, 1997, 2001). Whether two sequences are homologues can be measured by their sequence similarity for which there are different definitions and methods.

As mentioned in the introductory sections above, identifying homologous sequences is often the first step in annotating a newly sequenced gene. The homologue may already have some functional annotation that may then be transfered to the newly sequenced gene (or protein). Section 1.4.1 explains the conditions under which this transfer is considered to be reliable. The sections below explain the most common sequence search methods and their definition of similarity.

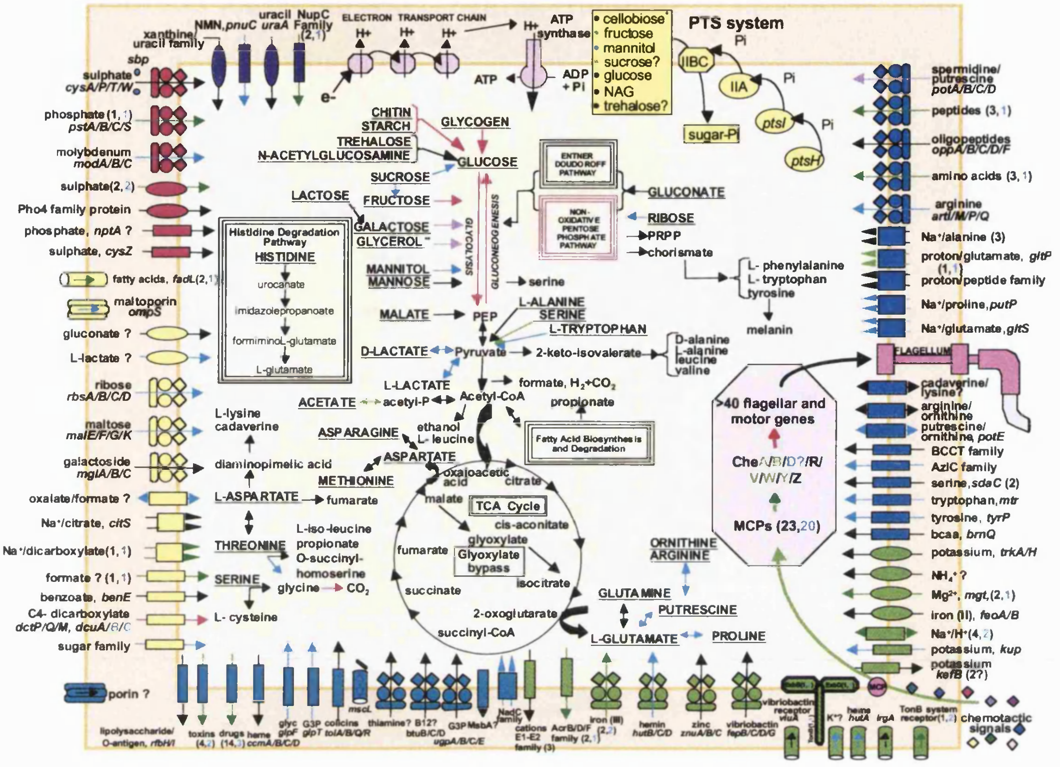


Figure 1.1: Schematic representation of the *V. cholerae* cell with a selection of metabolic pathways and transporters identified in the genome. This figure is an example how the huge amount of information from genome annotation can be represented in a comprehensive and user friendly way. The figure is from Heidelberg *et al.* (2000).

1.3.1 Dynamic programming

The oldest sequence comparison method that is still part of recent methods was developed by Needleman & Wunsch (1970). Their method is based on the general dynamic programming algorithm which was introduced in the 1950s by Bellman (1957), and allows the optimal alignment of two sequences. Two sequences with length n and m form an $n \times m$ matrix. For each position in the matrix $(n[i], m[j])$ a numeric value scores how favourable a replacement of the residue/nucleotide $n[i]$ with $m[i]$ or alternatively a deletion or insertion is. See section 1.3.2 below for a discussion of substitution scores. Generally these are negative for unfavourable substitutions (e.g. aligning tryptophan with a lysine), and positive for conservative substitutions such as lysine to arginine.

Global sequence comparison via dynamic programming aligns two sequences from the first to the last position in both sequences, and produces a global alignment. Even if only a region in the middle of one sequence shares similarity with a region of the other sequences, the algorithm will try to align the sequences over their full lengths. This may result in a drop of the overall score of the alignment, because the ends of the alignment may contribute negative scores, and the sum of the scores may therefore then not be significant.

The local alignment is a development based on the method from Needelman and Wunsch and was introduced by Smith & Waterman (1981). It solves the problem of forcing an alignment over the entire sequence. This method is fundamental to many other methods applied in this work, and is therefore explained in more detail below.

The formal rule to fill each cell of the $n \times m$ matrix is given in equation 1.1. j describes a position in n and i describes a position in m, d is a fixed negative score for a gap (the gap penalty) and *score* is a judgement of the biological significance for aligning residue $n[j]$ with $m[i]$.

$$F(i,j) = max \begin{cases} F(i-1,j) - d & \text{deletion at position j (cell above)} \\ F(i-1,j-1) + score(a,b) & \text{substitution i, j (diagonal cell)} \\ F(i,j-1) - d & \text{insertion at position j (cell to the left)} \\ 0 & \text{stop for local alignment} \end{cases} \tag{1.1}$$

In equation 1.1 scores for a deletion or insertion are fixed. Generally the costs of introducing a gap is set higher than for extending an existing gap. The substitution score is taken from a lookup matrix described in more detail below. If deletion, insertion or substitution gives a negative score, the stop condition holds, and the local alignment is terminated. The matrix can be filled row by row or column by column.

As an example the two sequences `'HEAGAWGHED'` and `'PAWHEAE'` are aligned using the method from Smith and Waterman. The matrix below shows the calculated scores from which the optimal path can be traced back. This is the optimal local alignment. Note that each cell of the matrix contains the sum of its own score and the last highest scoring cell as determined by equation 1.1. Matrix cells of the optimal path are shown in red.

	(j)	**H**	**E**	**A**	**G**	**A**	**W**	**G**	**H**	**E**	**D**
(i)	0	0	0	0	0	0	0	0	0	0	0
P	0	0	0	0	0	0	0	0	0	0	0
A	0	0	0	5	0	5	0	0	0	0	0
W	0	0	0	0	2	0	20	12	4	0	0
H	0	10	2	0	0	0	12	18	22	14	6
E	0	2	16	8	0	0	4	10	18	28	20
A	0	0	8	21	13	5	0	4	10	20	26
E	0	0	6	13	18	12	4	0	4	16	22

The resulting alignment is shown below:

```
(j)  A  W  G  H  E
(i)  A  W  -  H  E
```

Often there can be more than one optimal path through the matrix. If the local alignment method is applied to align two three-domain proteins where the N-terminal and the C-terminal domains of the two proteins are homologous but the central domain is not homologous, there will be two paths with high score sums through the matrix. Distinguishing alignments based on homology from those produced by chance similarity is critical for sequence comparison methods, i.e. it is critical to find paths through the matrix that rely on evolutionary relationships. The basis of local alignment statistics and probabilities are discussed below in section 1.3.4.

Sequence search and alignment methods based on dynamic programming are dependant on the length of both sequences to be compared. Every cell in the matrix has to be filled to find high scoring paths. The runtime of the algorithm is proportional to the product of the length of both sequences to be aligned. Comparing a single sequence with sequences from a protein database with generally several hundreds of thousands of sequences is time consuming, and the algorithm is therefore not applicable for large scale sequences searches.

1.3.2 Substitution matrices

An ideal substitution matrix scores a biologically meaningful alignment with positive scores and all chance alignments with negative scores. A scoring matrix is a 20 × 20 matrix, with each row/column representing a score for a particular amino acid substitution. Each cell contains a score that is based on the probability for exchanging amino acid i with amino acid j. The general formula for all substitution matrices with negative expected score is:

$$S_{ij} = \frac{log\frac{q_{ij}}{p_i p_j}}{\lambda} \tag{1.2}$$

where q_{ij} is the target substitution frequency (the observed frequency with which amino acid i is replaced by amino acid j) usually calculated from homologous proteins. All target frequencies for a given amino acid are $\geqslant 0$ and sum to one; p_i and p_j are background frequencies (the overall frequencies with which i and j are observed). The product of the background frequencies can be thought of as the probability of exchanging i and j by chance. Furthermore, the normalisation by the background frequencies implies that conservative exchanges for rare amino acids are weighted stronger. S_{ij} is multiplied by a factor (10 for the original PAM matrices) and then rounded to the nearest integer. These are the scores that are stored in the substitution matrix as shown in table 1.2 and are usually referred to as 'log-odds' (the log-odds for BLOSUM matrices are based on log_2 whereas the original PAM matrix was based on log_{10}). The logarithm is used for computational reasons to avoid multiplications of the substitution scores of the cells of the optimal path through the dynamic programming matrix. The log-odds are divided by a scaling factor λ that is specific for the scoring system.

A substitution matrix is uniquely determined by its target frequency (the background frequencies are the same for different matrices). The assumption for most scoring matrices is that the expected score S_{ij} for a chance amino acid substitution in a comparison of two random sequences is negative. Otherwise chance alignments gave positive cumulative scores by just extending over a sufficient length.

The most common matrices are PAM and BLOSUM. Generally the choice of the substitution matrix is crucial for the performance of sequence database searches, although no single scoring system is the best for all purposes. The best way to distinguish between real and chance alignments of a given class is to choose a matrix for which the target frequencies specifically characterise this class (e.g. a protein family). This aspect is treated in more detail in a later section.

The PAM matrices

The Point Accepted Mutation (PAM) matrix models the evolutionary distance between sequences of closely related proteins (Dayhoff *et al.*, 1978). A matrix cell gives the probability of amino acid i to be replaced with amino acid j after a given evolutionary interval which is given in PAM. One PAM is the probability of a residue to be mutated during an evolutionary distance in which one point mutation was accepted in 100 residues (i.e. 1% mutations). 100 PAMs do not necessarily mean that all residues are mutated, some residues may have been mutated several times, including mutations that restore the original amino acid, and some residues may not have changed at all. The mutation data to calculate the PAM matrix were collected from closely related proteins.

PAM matrices for longer evolutionary distances can be obtained by multiplying each target exchange frequency of the PAM1 matrix n times with itself to generate a PAMn matrix.

Sequence comparisons using a PAM matrix generally do not perform well in detecting more distantly related sequences. In particular the theoretical extrapolation from the experimentally derived PAM1 matrix to higher order PAM matrices to model a longer evolutionary distance does not take into account the conservation of functionally important sequence regions and may therefore overestimate mutability.

The BLOSUM matrices

The BLOSUM matrices (Henikoff & Henikoff, 1992) were derived from the BLOCKS database (see page 20). The frequencies of amino acids from conserved sequence blocks were tabulated, and the probabilities for target and background frequencies were calculated. To reduce multiple contributions of several closely related proteins, the sequences were clustered within blocks. Each cluster was treated as a single sequence. Clusters for different identity levels were built to produce different matrices allowing sequences $\geqslant n\%$ identity to be included in a cluster. The most commonly used matrices are BLOSUM50, BLOSUM62 and BLOSUM80, where the number

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	R	S	T	W	Y	V
A	5	-4	-2	-1	-4	-2	-1	0	-4	-2	-4	-4	-3	-6	0	1	1	-9	-5	-1
R	-4	8	-3	-6	-5	0	-5	-6	0	-3	-6	2	-2	-7	-2	-1	-4	0	-7	-5
N	-2	-3	6	3	-7	-1	0	-1	1	-3	-5	0	-5	-6	-3	1	0	-6	-3	-5
D	-1	-6	3	6	-9	0	3	-1	-1	-5	-8	-2	-7	-10	-4	-1	-2	-10	-7	-5
C	-4	-5	-7	-9	9	-9	-9	-6	-5	-4	-10	-9	-9	-8	-5	-1	-5	-11	-2	-4
Q	-2	0	-1	0	-9	7	2	-4	2	-5	-3	-1	-2	-9	-1	-3	-3	-8	-8	-4
E	-1	-5	0	3	-9	2	6	-2	-2	-4	-6	-2	-4	-9	-3	-2	-3	-11	-6	-4
G	0	-6	-1	-1	-6	-4	-2	6	-6	-6	-7	-5	-6	-7	-3	0	-3	-10	-9	-3
H	-4	0	1	-1	-5	2	-2	-6	8	-6	-4	-3	-6	-4	-2	-3	-4	-5	-1	-4
I	-2	-3	-3	-5	-4	-5	-4	-6	-6	7	1	-4	1	0	-5	-4	-1	-9	-4	3
L	-4	-6	-5	-8	-10	-3	-6	-7	-4	1	6	-5	2	-1	-5	-6	-4	-4	-4	0
K	-4	2	0	-2	-9	-1	-2	-5	-3	-4	-5	6	0	-9	-4	-2	-1	-7	-7	-6
M	-3	-2	-5	-7	-9	-2	-4	-6	-6	1	2	0	10	-2	-5	-3	-2	-8	-7	0
F	-6	-7	-6	-10	-8	-9	-9	-7	-4	0	-1	-9	-2	8	-7	-4	-6	-2	4	-5
R	0	-2	-3	-4	-5	-1	-3	-3	-2	-5	-5	-4	-5	-7	7	0	-2	-9	-9	-3
S	1	-1	1	-1	-1	-3	-2	0	-3	-4	-6	-2	-3	-4	0	5	2	-3	-5	-3
T	1	-4	0	-2	-5	-3	-3	-3	-4	-1	-4	-1	-2	-6	-2	2	6	-8	-4	-1
W	-9	0	-6	-10	-11	-8	-11	-10	-5	-9	-4	-7	-8	-2	-9	-3	-8	13	-3	-10
Y	-5	-7	-3	-7	-2	-8	-6	-9	-1	-4	-4	-7	-7	4	-9	-5	-4	-3	9	-5
V	-1	-5	-5	-5	-4	-4	-4	-3	-4	3	0	-6	0	-5	-3	-3	-1	-10	-5	6

Table 1.2: PAM70 amino acid substitution matrix. Cells contain the log odds of a particular amino acid substitution probability after 70 PAMs. Note that the matrix is symmetric.

indicates the $n\%$ cut-off.

The BLOSUM matrices perform better in sequence alignments and homology searches than the PAM matrices, especially in detecting more distant homologies (e.g. Henikoff & Henikoff (1993); Russell *et al.* (1998a)). The matrices are constructed from sequences of any evolutionary distance without any theoretical extrapolation. There are substantial differences in the amino acid mutability when comparing BLOSUM and PAM (Henikoff & Henikoff, 1992).

1.3.3 The basics: BLAST and FastA

Several heuristics to speed up sequence searches have been developed. Here the BLAST (Altschul *et al.*, 1990) method is discussed in more detail, because BLAST and its derivatives have been applied extensively in this work. Significant sequence similarity may be found by a simple comparison of short regions of a few amino acids length without performing dynamic programming. If the initial step was successful, more sensitive but time consuming refinement steps are applied (including dynamic programming). Methods based on such simple comparisons are heuristics and do not guarantee an optimal alignment between two sequences. Nevertheless, when comparing a query sequence to a sequence database, generally most of the sequences do not share any homology with the query, and may be skipped by the fast heuristic step, reducing the search space to which the more detailed comparisons are applied.

The FastA heuristic

Wilbur & Lipman (1983) introduced the first heuristic method to search a query sequence against a database of sequences. This method has been subsequently improved in the FastP and later in the FastA methods (Pearson & Lipman, 1988; Pearson, 1990). The FastA method can be applied to nucleotide or peptide sequences. There are five major steps in the algorithm:

1. Identify matching 'words' between two sequences (the query and a database sequence) that share identical pairs of amino acids (*ktup* = 2, a word of two residues).

2. Find regions of high density of identities. This is done by finding the words that are on the same diagonal of a plot between the two sequences. These words are extended to merge with other existing words to form a region if the distance of the previous word or region in residues is smaller than the score of the current region or word match.

3. Re-score the ten highest scoring regions using a PAM250 matrix, and trim or extend the ends of these to optimise their score. This is a partial alignment without gaps.

4. If there are several regions above a given score cut-off, these regions are joined via dynamic programming, producing a gapped alignment if their score can be improved (the overall score is the sum of the scores of the regions minus a penalty score for gaps). This score is called *initn*, and is used as a rank of the database sequence.

5. For the top ranking sequences, a local alignment is constructed with the query sequence using a centred 32 residue window on top of the best *initn* region. The resulting score is the *optimised score* that is reported.

The initial search step may not reduce the number of sequences substantially, but it reduces the subsequent more detailed and time consuming searches to only a few regions of the sequence that have to be compared in more detail. The calculation of the *initn* value reduces the number of regions and sequences for which Smith-Waterman local gapped alignments have to be produced. In summary, the FastA method speeds up sequence database searches by reducing the time consuming dynamic programming to a set of matrices per sequence which are in total smaller than the complete $n \times m$ matrix.

The BLAST heuristic

The original BLAST method (Basic Local Alignment Search Tool, Altschul *et al.* (1990)) uses heuristics similar to FastA to find candidate sequences, but BLAST is even faster then FastA. The original BLAST method produced un-gapped alignments and was refined (Altschul & Koonin, 1998; Schaffer *et al.*, 2001) to gain more sensitivity (including gapped alignments) and speed. The steps of the method implemented in BLAST series 2.0 (Altschul & Koonin, 1998) for amino acid sequences are described below (the steps for nucleotide sequences are similar).

1. Find word pairs of a given length (usually 3 residues for proteins) for which the cumulative score is at least T. A word satisfying this condition is called a hit. Scores are taken from a standard matrix such as BLOSUM or PAM.

2. If the two sequences contain at least two non-overlapping hits within a distance A on the same diagonal then the extension of these matches is triggered. If two hits overlap, the most recent one is ignored. This two-hit method reduces the number of triggered extensions, which is the most time consuming step in BLAST.

3. If the previous conditions are satisfied, the un-gapped bidirectional extension of the second hit is triggered using the same substitution matrix as in the first step. The extension terminates if its cumulative score cannot be improved anymore, and the score is $\geqslant S$. A step in the heuristics to speed up the extension procedure is to terminate an extension if it reaches another hit with a score that falls a certain distance below the previous shorter extension. The extended hit may include other hits. An extended hit is called an HSP (High scoring Segment Pair).

4. The highest scoring HSP with a score $\geqslant S_g$ is further extended in both directions via a gapped alignment. Only the highest scoring HSP is extended because most of the HSPs will be included in this gapped extension.

5. The final alignment for hits for which a gapped extension produced a high score are re-aligned with relaxed alignment parameters. This increases the | extent of the alignment.

BLAST performs far fewer local alignments compared to FastA and is therefore much faster. Like FastA, gapped extensions are only performed on a relatively small region within a sequence.

1.3.4 Basic statistics and probabilities for local alignments

The scoring system is crucial in distinguishing between real and chance alignments, and equation 1.2 gives most of the basic statistics of a scoring system. Sequence search methods employ a scoring system to judge whether similarity could have arisen by chance, and for heuristics such as BLAST whether a more time consuming comparison has to be performed.

The basic statistics for the score distributions from local ungapped alignments has been described by Karlin and Altshul (Karlin & Altschul, 1990, 1993; Altschul & Gish, 1996). The distribution of scores for hits between a real sequence and a set of randomly generated sequences can be approximated with an extreme value distribution. Scores as given in equation 1.2 are summed over the region participating in a hit. Figure 1.2 shows scores that are approximated with an extreme value distribution. Since this score distribution is the result of chance alignments, biologically meaningful scores should be distributed at the long tail end of the distribution, and the location of this score on the distribution can be treated as a confidence level for this score (Karlin & Altschul, 1990). The formal description of this confidence is given in equation 1.3 which is the probability to find at least one random alignment with a score $S \geqslant x$. This probability is also known as a P-value. K is another constant that depends on the scoring system, and mn is the product of the lengths of the sequences that are compared. For database searches mn is the product of the length of the query sequence and the search space of the database.

$$P(S \geqslant x) = 1 - e^{-Kmne^{-\lambda x}} \tag{1.3}$$

The score S depends on the scoring system via K, λ and special scores for the introduction of gaps and gap extensions (λ is the same as in equation| 1.2). It is useful to convert this score into a score S' that is independent of the scoring system

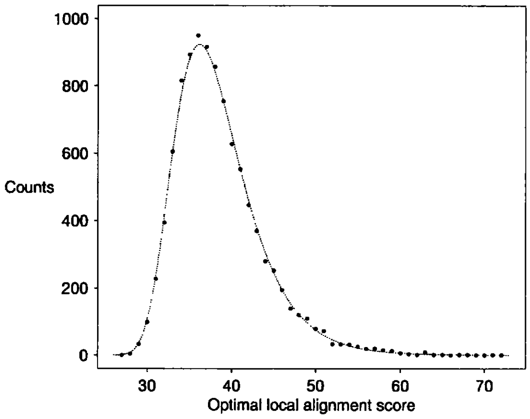


Figure 1.2: Random alignment scores can be approximated by an extreme value distribution. The figure is taken from Altschul & Koonin (1998) (figure 6). A position specific scoring matrix generated by PSI-BLAST (see section 1.3.5) was compared to 10,000 randomly generated protein sequences.

to compare results obtained from searches that use different substitution matrices. A normalised score S' is expressed in bits which can be obtained from the scaling constants of the scoring system and the score distribution. Equation 1.4 gives the formal description of this normalisation.

$$S' = \frac{\lambda S - lnK}{ln2} \tag{1.4}$$

The reliability of an alignment in BLAST and other programs is given as an e-value, described in equation 1.5.

$$e(S') = mn2^{-S'} \tag{1.5}$$

$$e(S') = Kmn \exp(-\lambda S) (\text{directly calculated from the raw score}) \tag{1.6}$$

The e-value is the number of expected chance hits with a score $\geqslant S'$. Doubling the length of the query sequence or database doubles the number of expected chance hits, and the number of expected chance hits decreases exponentially with increasing score. Note that $e(S')$ is found in the exponent of equation 1.3.

Another confidence measure that requires a substantial sample of the score distribution is the z-score. It is defined as the distance of an the alignment score S from the mean μ of the distribution of all scores of the analysis divided by the standard deviation σ of the score distribution ($score = (S - \mu)/\sigma$). The normalisation by the standard deviation of the distribution ensures that even high scores with a short distance to the mean get relative low z-scores if the score distribution is flat, e.g. if there are many chance hits. A z-score is as defined above is only informative for normally distributed scores. However, it is possible to calculate P-values for z-scores that are derived from an extreme value distribution of scores (personal communication with William Pearson). Therefore z-scores may be used as confidence measures for local alignments such as in the FastA (Pearson, 1990).

All equations in this section and equation 1.2 have only been proven to hold for ungapped local alignments, but computational analysis and some analytical work suggest the same applies to gapped local alignments (Karlin & Altschul, 1990, 1993; Altschul & Gish, 1996; Altschul *et al.*, 2001). Extreme value distributions fit scores from gapped local alignments of randomly generated sequences well using standard background frequencies (Robinson & Robinson, 1991) and a standard substitution matrix such as BLOSUM62 with standard gap opening and extension scores (Waterman & Vingron, 1994; Altschul & Koonin, 1998; Altschul & Gish, 1996), from which the scale parameters λ and K are derived. These parameters cannot be determined analytically for gapped local alignments. However, Mott (2000) derived an empirical formula from a large number of simulation with different scoring systems to calculate λ. For ungapped local alignments these parameters are analytically derived from the scoring system (Karlin & Altschul, 1990). The FastA method generates enough optimal gapped local alignments between unrelated sequences for each run to have a basis from which to λ and K can be estimated. The BLAST program generates gapped alignments only for potentially related sequences and cannot estimate the parameters from these scores. Therefore BLAST uses pre-estimated parameters from simulations for different standard matrices and gap opening and extension costs (Altschul *et al.*, 1997).

1.3.5 Sequence specific profiles and PSI-BLAST

As mentioned at the beginning of section 1.3.2, none of the standard substitution matrices optimally describes the target frequencies of a particular class of sequences. A position specific scoring matrix (PSSM) or sequence profile is specifically constructed for a particular class of proteins. A PSSM has the dimensions $n \times 20$, where n is the length of the sequence. At each position n_i of the matrix, a substitution score for each of the 20 amino acids is given. The main difference to the standard substitution matrices is that the score for the same amino acid type can differ depending on the position within the sequence. Usually a PSSM is constructed from a multiple sequence alignment, for example from a set of already identified homologues and may be subsequently refined by pulling in more distant homologues when a database is searched with the PSSM. Earlier profile methods (e.g. Patthy (1987); Gribskov *et al.* (1987); Taylor (1986); Yi & Lander (1994); Tatusov *et al.* (1994)) used rather complex procedures involving several programs with substantial user intervention.

The PSI-BLAST method (Altschul *et al.*, 1997; Schaffer *et al.*, 2001) combines all the required steps, automatically constructs a PSSM and uses this profile to search a sequence database. A comparison of several sequence database search methods showed that PSI-BLAST is about three times more sensitive than BLAST or FastA in detecting remote homologues (Park *et al.*, 1998).

Figure 1.3 shows the basic steps of the PSI-BLAST procedure. First, a standard BLAST, as described in section 1.3.3, is performed using a standard substitution matrix (e.g. BLOSUM62) and a sequence database. From this run those sequences satisfying a given e-value cut-off are stored, and a multiple sequence alignment is constructed from these sequences. This multiple alignment is converted into a PSSM which is then used in the second search round instead of the query sequence and the standard substitution matrix to search the sequence database via the BLAST algorithm. The difference between this step and the original BLAST is just that the PSSM itself contains the information about the query sequence and the substitution matrix. The procedure of searching the database and re-constructing a new PSSM after every round is repeated until no more sequences with sufficient e-value can be added to the list of sequences of the previous round or a given maximum number of rounds has been reached. The result is a list of sequence alignments of the last round that are of sufficient e-value.

Construction of a Position Specific Scoring Matrix

A multiple alignment is constructed by stacking all sequences found in a search round with an e-value $\leq$ the cut-off. Sequences identical to the query are skipped,

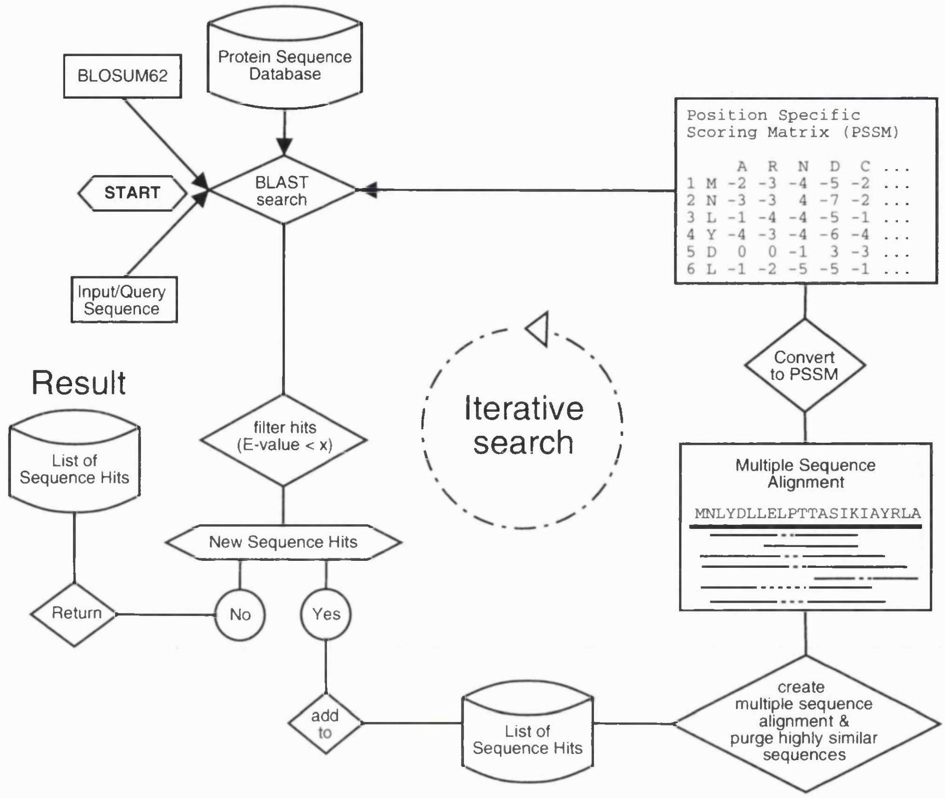


Figure 1.3: Overview of the PSI-BLAST procedure. The procedure starts by running BLAST for a query sequence against the sequence database using a standard matrix (here BLOSUM62). In the next round the PSSM, instead of the query sequence and the BLOSUM62 matrix, is used for the database search. A new PSSM is constructed in every round until no new sequences can be found. A search cycle is called *iteration*. See text for more details.

and for sequences with very high sequence identity ($> 97\%$ in PSI-BLAST version 2.0 and $> 93\%$ in version 2.1) only one representative sequences is kept. The final multiple sequence alignment M has residues or gap characters in every column and row. For the calculation of the sequence weight for a column in the PSSM only those rows (sequences) are considered that contribute a residue or gap to that row.

Sequences contributing to a column of the multiple alignment are weighted in a similar way as for the construction of the BLOSUM matrices described in (Henikoff & Henikoff, 1992). Closely related sequences can bias the PSSM. This bias can be avoided by weighting each sequence according to its individual information content. Gaps are treated as the 21^{st} distinct character of the amino acid alphabet, and any column consisting of identical characters are ignored for calculating the individual weight factor for a sequence. This weight scales the raw observed residue frequency for a given column i of the PSSM, giving the weighted residue frequency f_i. Further the relative number of independent residue observations N_C is calculated as the mean of the number of different amino acid types observed at a position. The maximum of N_C is 21, but for most columns in the multiple alignment N_C is much smaller. N_C is a per column scaling factor reflecting alignment variability.

A general frequency probability Q_i/P_i with Q being the target frequency and P being the standard background frequency on which equation 1.2 is based on is not appropriate for the probability estimation for the PSSM, because of the weighting issues discussed above. A small sample size (some alignments may just have a few sequences at some columns) and the necessity for the prior knowledge of the relationships among the residues requires a different probability scheme. The calculation of Q_i for a position in the PSSM includes the target frequency q_{ij} that was used for the initial substitution matrix (see equation 1.2) to make use of the prior knowledge of the residue relationships. Equation 1.7 calculates a *pseudocount* (Tatusov *et al.*, 1994) for a given column in the PSSM where q_{ij} is the target frequency for the standard substitution matrix from equation 1.3.2.

$$g_i = \sum_{j=1}^{20} = \frac{f_j}{P_j} q_{ij} \tag{1.7}$$

$$Q_i = \frac{\alpha f_i + \beta g_i}{\alpha + \beta} \tag{1.8}$$

The target frequency Q_i for a position in the PSSM is then given via equation 1.8 which combines the scaled observed frequency with the pseudocount. Therefore a PSI-BLAST PSSM is a position specific scaled version of the initial substitution matrix that was used. The factor α is defined as $N_C - 1$ to account for the alignment variability mentioned above. The two equations above imply that for positions in the query for which the multiple alignment does not have any sequences the initial substitution score is used. The β factor can be used to increase or decrease the weight of the initial substitution matrix. Gaps do not have any position specific scores, constant gap opening and gap extension scores are applied as for the standard substitution matrices. The actual substitution score is calculated from Q_i using equation 1.2.

Applying BLAST to a position specific search

The BLAST method is applied in the same way to the PSSM as for a query sequence and a standard substitution matrix, assuming the same statistics holds for a position specific search. The calculation of the normalised score S' for hits includes the scaling parameters λ and K for which Altschul *et al.* use the same values as for the initial substitution matrix that was used in the first round (e.g. BLOSUM62). They showed that the employed scoring system fits well the observed score distribution. The score distribution from comparisons of random sequences with a PSSM derived from a real sequence can be fitted by an extreme value distribution (figure 1.2) with the calculated parameters λ and K close to those for gapped simulations for a BLOSUM62 matrix.

By employing the pseudocount PSI-BLAST makes use of the statistics from BLAST and the underlying substitution matrix which assumes a standard amino acid composition of the query sequence and the database. Although the initial analysis of PSI-BLAST has shown that its statistics fits the observed score distribution, and the calculation of the *e*-value approximates the observed error rate within a range of 20%, there have been problems with the PSI-BLAST statistics for a range of query sequence the more the sequence differs from the assumed standard amino acid composition. A BLAST comparison between a query and a database sequence of similar biased composition may produce a hit with significantly high score because the standard BLAST statistics does not apply for this sequence pair. Recent changes in the BLAST and PSI-BLAST algorithms (Schaffer *et al.*, 2001) implemented in the 2.1 series of the program consider biased amino acid compositions. Especially for PSI-BLAST, biased sequences have a strong impact because in every iteration the PSSM itself will be biased towards the amino acid composition of the query, producing even more unreliable results in the next search round (Schaffer *et al.*, 2001; Altschul & Koonin, 1998).

The most important change to cope with different amino acid compositions is a PSSM specific λ. For composition biased sequence pairs the standard λ (e.g. that for the BLOSUM62 scoring system) is generally too big and results in a lower *e*-value (lower *e*-values give more confidence) than justified (Schaffer *et al.*, 2001). A composition dependant λ' is therefore generally smaller than the standard λ. It is computationally too intensive to estimate λ' by fitting the score distribution for each query or PSSM and database sequence pair. Since λ_u can be determined analytically (Karlin & Altschul, 1990) for ungapped alignments (it is the unique solution to sum the scores for a matrix colum given in equation 1.2 to one), a composition specific λ'_u for scores from ungapped alignments is calculated using the amino acid frequencies of the database sequence and the query. The composition rescaled score for a matrix cell in the PSSM is then given by $\frac{\lambda'_u}{\lambda_u} Sij$, where S_{ij} is the non-scaled score of the PSSM.

As mentioned in section 1.3.4 the statistics for ungapped alignments has been shown to approximate score distributions for gapped alignments, too. Matrix rescaling is time consuming because it has to be performed for every query database sequence pair. Rescaling is only triggered if an alignment produces a significantly high score using the non-scaled scoring system. The alignment for the sequence pair (or a PSSM and the sequence) is then recalculated. *e*-Values as the common confidence measure for BLAST and PSI-BLAST alignments are more conservative with the rescaled scoring system and have been shown to be more realistic than the original *e*-values (Schaffer *et al.*, 2001).

To avoid the application of the BLAST algorithm to highly biased sequences with a low *amino acid entropy*, for which re-scaling may not be sufficient to stop a corrupted search, a *low complexity* filter can be applied to remove regions from the database or query sequence that differ markedly from the standard amino acid composition. Positions in these *low complexity* regions are replaced by the 'X' character and are ignored by the BLAST search procedure. Such a filter is implemented in the BLAST 2.0 and 2.1 series (Wootton, 1994).

Finally, it is worth mentioning that the sensitivity of PSI-BLAST, the ability to detect even distantly related homologues, depends on the diversity and size of the sequence database that is used for the search. Generally in every iteration more distantly related sequences are identified and added to the PSSM. After every round the PSSM explores evolution a step backward. PSI-BLAST would not be able to detect the relationship between a query sequence A and a distantly related sequence B in the database if there were no evolutionary intermediates present in the database, see e.g. Aravind & Koonin (1999).

1.3.6 Using sequence profiles with IMPALA

The IMPALA method (Schaffer *et al.*, 1999) compares a query sequence against a library of PSSM produced by PSI-BLAST. This is particularly useful if one wants to find the protein or domain family to which a given query belongs. Each family is represented as one PSSM in the library. Such a library may be constructed by searching a large sequence database with a member of a characterised protein family using PSI-BLAST. The final PSSM produced by PSI-BLAST may then be used as a representation of the protein family.

The comparison of the query sequence with each PSSM is performed via the Smith-Waterman procedure (see equation 1.1 and text in that section), so that optimal local alignments are guaranteed. The time consuming Smith-Waterman procedure is acceptable because a profile library generally contains only a few hundred members representing families or domains rather than hundreds of thousands of single protein sequences from a database that is used within e.g BLAST and PSI-BLAST searches. IMPALA faces the same statistical problems calculating significance for scores between the query and a PSSM as PSI-BLAST. In fact the re-scaling procedure to scale a PSSM by λ'_u (mentioned in the previous section) was initially developed for IMPALA and later adapted by PSI-BLAST version 2.1. IMPALA performs similarly to PSI-BLAST version 2.0 and 2.1 in terms of sensitivity and error rate. Since IMPALA and PSI-BLAST version 2.1 use the same re-scaled scoring system, *e*-values are very similar, whereas *e*-values generally differ from those calculated by the older PSI-BLAST version 2.0.

A recent development is the RPS-BLAST program (Reversed Position Specific, Marchler-Bauer *et al.* (2002)) that is a derivative of IMPALA. The query is compared to the query PSSM via the BLAST heuristics instead of using a Smith-Waterman dynamic programming as in IMPALA (the program is part of the NCBI BLAST package).

1.3.7 Hidden Markov Models

Hidden Markov models are a commonly used technique in genome annotation, for example to identify known protein families (Krogh *et al.*, 1994). An overview of this technique and its application in sequence comparison is given in a review by Eddy (1998). A hidden Markov model (HMM) associates different states and the transition between these states with probabilities. Protein sequences generated randomly by an HMM for a particular family should then contain members of this family, or from a different point of view, sequences with a high probability to be derived from this model should belong to the family the model describes. HMM based methods have been used in this work.

Sequences can be represented by first order Markov chains. A letter in a sequence is not independent, it depends on the previous letter, but does not depend on the full list of previous letters in the sequence. An HMM contains different states which are for example biological meaningful descriptions, such as hydrophobic H and polar P, to describe different regions within a protein. Between these states there are transitions, each associated with a probability t to go from one state to another. All transition probabilities from one to another state must sum to one. Each state contains emissions which are the 20 amino acids for a protein sequence. The probabilities of the emissions per state must sum to one. Only the emission symbols (the amino acid letters) of the model are directly observed, but the states and the transitions between them are hidden, therefore such a Markov chain is called a hidden Markov chain. Having introduced the terms *transition* and *emission*, the dependency of a letter in a sequence on the letter of the previous position is in fact the transition state between two emissions. Inferring a hidden state sequence (such as the above hydrophobic and polar states) from a protein sequence labels the protein sequences with biological information of higher order than just the residue letters in the protein sequence.

Figure 1.4 represents the two state HMM for hydrophobic and polar with the transitions between these states. The probability that a sequence FYK is modelled via $H \longrightarrow H \longrightarrow P$ is then given by equation 1.9, the first probability in each term is t, the second is e.

$$P(HHP) = (1 * 0.25) * (0.9 * 0.1) * (0.1 * 0.5) \tag{1.9}$$

The sum of the probabilities to find the sequence in any of the states is the probability with which the sequence can me modelled by this HMM. Usually dynamic programming is used to find the optimal path for a given input sequence through the HMM, where the rows and the columns of the matrix contain the sequence letters and the states.

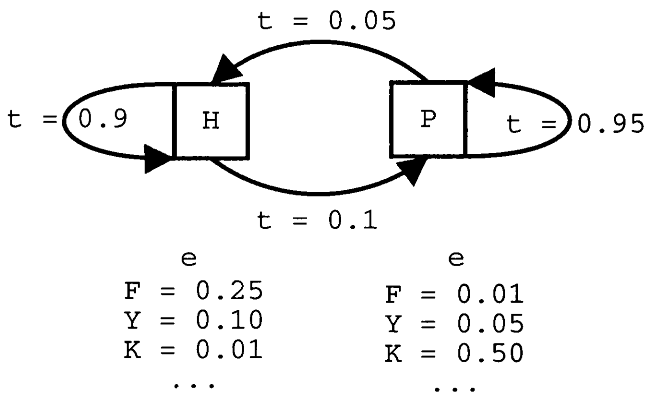


Figure 1.4: Schematic representation of a two state hidden Markov model, to assign a residue in a protein sequence to either the hydrophobic H or the polar P state. t is the transition probability, e gives the probability for emitting a particular amino acid type from this state.

HMMs are used in a wide range of bioinformatics applications, such as (i) gene prediction where a gene is modelled with different states such as exon-intron structure (see section 1.2.1), (ii) transmembrane helix prediction of protein sequences (e.g. Sonnhammer *et al.* (1998); Krogh *et al.* (2001); Tusnady & Simon (2001)) where a helix may get states for the helix caps and states for the hydrophobic core and (iii) the identification of homologous sequence families (Bateman *et al.*, 1999). Homology based sequence searches using carefully constructed HMMs for protein families perform better than PSI-BLAST (Park *et al.*, 1998) in detecting distantly related proteins, but the construction of high quality HMMs on which the performance relies is difficult and usually requires several steps and manual inspection (Bateman *et al.*, 1999, 2002; Letunic *et al.*, 2002; Gough & Chothia, 2002). The key aspect for the performance of any HMM based application is the design of the HMM which includes a definition of the states and the associated probabilities e and t.

Profile HMMs that describe a protein or domain family such as in PFAM and SMART (see section 1.2.3) usually derive the probabilities for e and t from multiple sequence alignments. An initial HMM is constructed that may just contain a limited number of rather closely related members of the family. This HMM is then iteratively refined in a similar way PSI-BLAST refines its PSSMs (Bateman *et al.*, 1999). A HMM in database search round n will detect more divergent members of the family than in round $n - 1$, and the new HMM that is constructed after round

n is used to search the sequence database in round $n + 1$. The most commonly used profile HMM packages are HMMer (Eddy, 1998) and SAMT99 (Karplus *et al.*, 1998). These methods contain programs to construct, refine and manage HMMs and to search libraries of HMMs with a query sequence.

The states for a sequence profile HMM are (a) the residue positions of the protein family (from one to the sequence length of members of the family), referred to as match states, (b) a deletion state between each match state that allows bypassing a match, and (c) an insertion state between each match state to allow residues to be inserted between two matches. Figure 1.5 represents a model for a three residue sequence motif (Eddy, 1998). The two major differences between sequence profiles such PSI-BLAST PSSMs and HMMs is that a PSSM does not score gaps in a position specific way whereas a HMM contains the deletion (gaps) state. Further, in a HMM a state is dependant on the previous state, whereas a position in a PSSM is mathematically independent.

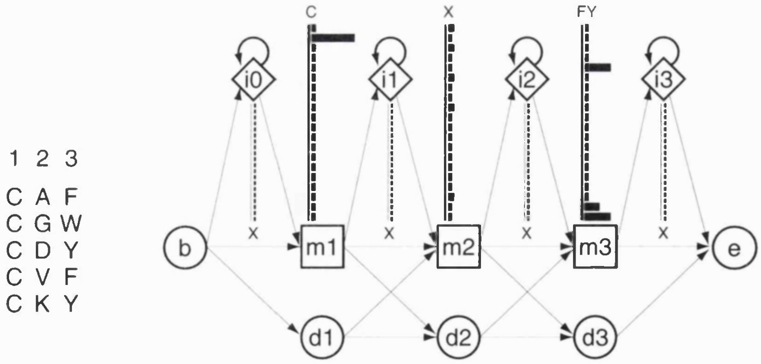


Figure 1.5: A small profile HMM (right) representing a short multiple alignment of five sequences (left) with three consensus columns. The three columns are modelled by three match states (squares labelled m1, m2 and m3), each of which has 20 residue emission probabilities, shown with black bars. Insert states (diamonds labeled i0-i3) also have 20 emission probabilities each. Delete states (circles labeled d1-d3) are 'mute' states that have no emission probabilities. A begin and end state are included (b, e). State transition probabilities are show as arrows. The figure and the legend are from Eddy (1998) (figure 2).

1.4 Protein structure and genome annotation

This section explains why knowledge of the three dimensional structure of proteins is important. There is a huge discrepancy between the availability of protein sequences and their 3D-structures. Currently there are more than 800,000 different sequences in the public databases (12/2001, ftp://ftp.ncbi.nlm.nih.gov/blast/db/), but there are less than 16,000 experimentally determined protein structures in the Protein Data Bank (PDB, 12/2001, http://www.rcsb.org, Berman *et al.* (2000)), and these contain redundancies such as structures with point a mutation. Despite the difference in absolute numbers, the sequence and the structure databases both grow exponentially.

1.4.1 Functional and evolutionary insights from protein structure

The 3D-structure of a protein determines its biochemical function. Homology based sequence comparisons and motif searches to identify the function of a protein are therefore simplifications because these searches only consider 1D-information. However, divergent sequences often share a similar 3D-structure that accepts to some extent a range of amino acid substitutions. The 3D-structure is generally more conserved than the 1D-structure (the sequence), see e.g. Chothia & Lesk (1986) and Murzin *et al.* (1995). Figure 1.6 shows the dependency of the structural similarity measured as the root mean square of C_α distances of homologous protein domains and the sequence identity between these domain pairs. At about 20-25% sequence identity the 3D-similarity starts to decrease dramatically. Distantly related sequences with less than 20% sequence identity (the *twilight zone*) generally only share a similar structural scaffold, a common fold, with differences in structural details which usually determine the biochemical function (Hegyi & Gerstein, 1999; Wilson *et al.*, 2000). However, an analysis from Wood & Pearson (1999) using z-scores for a sequence-structure comparison showed a linear relationship between z-scores of the sequences members of a fold and the z-scores of their structural alignments.

Wilson *et al.* (2000) analysed the relationship between sequence identity and function, and structural similarity and function. For enzyme domains with an RMSD

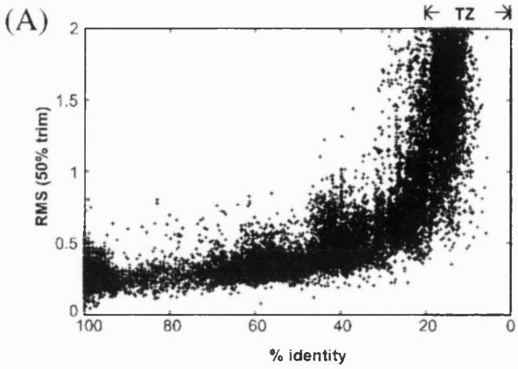


Figure 1.6: Relationship between sequence identity and structural similarity. RMS deviation of superimposed structural domains as a function of percentage identity. Scatter plot of homologous superfamily domain pairs from the SCOP database (see section 1.4.4). The plot is similar to an earlier presentation by Chothia & Lesk (1986) but considers 1,000 times more domain pairs (30,000 in total). TZ denotes the twilight zone of sequence similarity where inferring structural similarity gets unreliable. Only the best 50% of superimposed C_α atoms per pair where included in the RMS calculation (50% trim). Figure 2(a) from Wilson *et al.* (2000).

of 1Å 90% of the domains pairs have the same broad function. This structural similarity can be mapped to the start of the *twilight zone* sequence similarity (about 25% sequence identity) in figure 1.6. For a 90% chance of a precise match of function of two structures a similarity of about less than 0.6Å RMSD is required corresponding to 40% sequence identity. These thresholds of sequence identity are also supported by other work (Devos & Valencia, 2000; Todd *et al.*, 2001). Hegyi & Gerstein (1999) showed with their analysis, that the functional diversity of protein domains decreases approximately as a function of the exponent of the *e*-value threshold of the alignment between a protein domain and its functionally annotated homologues in the SwissProt database (see section 1.2.3 for a description of SwissProt). The plot of this sequence/function relationship is shown in figure 1.7.

The analysis described above is based on single domains. For multi-domain proteins function is less conserved between proteins than for single domain proteins, and even proteins with the same domain combination may not have the same function (Hegyi & Gerstein, 2001). This renders functional flexibility of folds of domains in a different context.

The relationship between structure and function raises the question whether

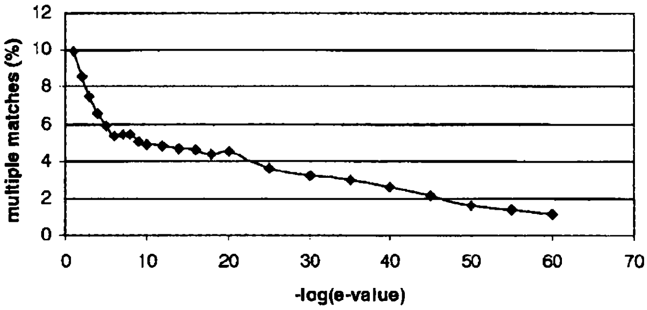


Figure 1.7: Multi-functionality of protein domains versus e-value threshold. A domain has multiple functions if at least two homologues of different function from the SwissProt database can be identified for this domain. The e-value of the alignment between homologous pairs is plotted as the negative logarithm to the base of 10 against the fraction of domains with multiple functions (i.e. increasing values on the x-axes indicates more confidence in the homologous relationship). Starting from an e-value of 10^{-5} ($log_{10} - 5$) multi-functionality decreases exponentially. Figure 7 from Hegyi & Gerstein (1999).

there is a relationship between a particular function and a fold. Studies from Martin *et al.* (1998) showed only little preference of a function to be associated with a particular protein fold. However, other results (Hegyi & Gerstein, 1999; Wilson *et al.*, 2000) show a significant bias of certain folds with a particular group of functions. E.g., mixed α/β-folds are often associated with enzymatic domains whereas all-α domains are biased towards non-enzymatic function. On the other hand there are a few folds such as the TIM (Triose-phosphate Isomerase) barrel that provides a generic scaffold to fulfil a broad range of enzymatic functions.

Todd *et al.* (2001) showed that 25% of the homologous superfamilies of similar structure have different enzymatic function, highlighting the divergent evolution within these superfamilies. Most functional changes within a related set of sequences are due to a change in the substrate but maintain the same reaction mechanism (Holm & Sander, 1997; Todd *et al.*, 2001).

Due to the structural conservation of proteins the number of distinct 3D-architectures for globular proteins has been estimated to be limited between 1,000 and 7,000 (Brenner *et al.*, 1997; Govindarajan *et al.*, 1999; Zhang & DeLisi, 1998; Wolf *et al.*, 2000). This means that many proteins have the same or a very similar general architecture of secondary structure elements (α-helices and β-sheets), although their peptide sequences may not show obvious similarity. Considering this structural 'limitation', functional diversity has to be generated by adopting an existing structural scaffold to a particular function. Functional changes within the same structural fold is often related to critical local sequence changes Todd *et al.* (2001); Aloy *et al.* (2001), and in difficult cases may be traced to differences of a few critical atoms.

An overview about the relationships between sequence, structure, function and evolution is given by Orengo *et al.* (1999); Thornton *et al.* (1999, 2000). Generally protein structure is more conserved than its function (and its sequence).

1.4.2 Examples for protein structure/function relationships

Glycogen synthase kinase 3β

The recently published structure of the glycogen synthase kinase 3β (GSK3β, Dajani *et al.* (2001)) is represented as an example of how protein structure reveals insight into biochemical function, supporting and guiding functional studies. The GSK3β plays a regulatory role in two distinct signalling pathways, the insulin induced signalling pathway to regulate glycogen synthesis and the Wnt (Wintbeutel) signalling pathway involved in cell proliferation and development. The default for GSK3β is to phosphorylate and thereby inhibit its target proteins.

GSK-3β contains an N-terminal *activation segment* that is also found in other kinases such as ERK2 MAP kinase (Zhang *et al.*, 1995), forming a β barrel structure that opens a substrate specific binding cleft and positions the active site residues for the phosphorylation reaction. This activation itself is enhanced by the phosphorylation of the *activation segment* (tyrosine 216 in GSK-3β). A feature specific for GSK3β is the P+4 phosphorylation pattern. The kinase efficiently phosphorylates substrates at a position with a serine or threonine if the residue 4 positions towards the C-terminus has already been phosphorylated (*primed phosphorylation*). Additional serine or threonine residues can be phosphorylate in +4 steps in a C-terminal to N-terminal direction (*hyper-phosphorylation*, Fiol *et al.* (1994)).

The crystal structure was analysed to suggests a model by which the requirement for *primed phosphorylation* and the substrate specificity is explained. The structure of GSK3β shows the active from of the protein, with an open cleft between the *activation segment* at the N-terminus and the C-terminal domain. Figure 1.8 (A) shows the surface of GSK3β with the functionally key residues labelled. The cleft from the positively charged patch formed by R96, R80 and K205 to the left, passing the active site residues R220 and D181, is the substrate binding site. The positively charged patch is stabilised by either a phosphorylated tyrosine at position 216 forming a hydrogen bonding network with the three positively charged residues or by a free phosphate or sulphate from the surrounding buffer *in vitro* (as it is found in the crystal structure) and the cytosol *in vivo*. The modelled protein substrate complex in 1.8 (B) explains the requirement for P+4 *primed substrates*, and the specificity for substrates containing a serine or threonine at 'P(0)' and 'P(+4)'.

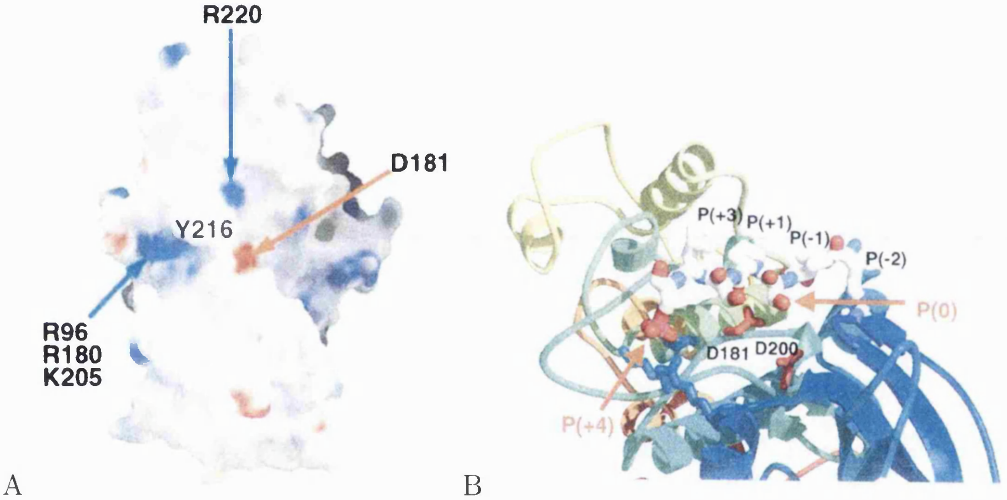


Figure 1.8: GSK3β surface and active site. From Dajani *et al.* (2001), figures 3a and 4a. (A) The solvent-accessible surface of GSK3β coloured according to electrostatic potential (red, negative, blue: positive). The intensive positive patch generated by the basic side chains of Arg 96, Arg 180 and Lys 205 is indicated, as is the location of the catalytic Asp 181 and Arg 220 which could interact with a phosphorylated Tyr 216. The N-terminal mainly neutral *activation segment* is located towards the bottom of figure. (B) Phospho-Substrate bind model. Model of substrate binding (peptide sequence PPSPSLS) to GSK3β. Phosphorylation of a serine at P(0) by the active site residues (red) depends on a 'priming' phospho-serine at P(+4) interacting with residues of the positively charged patch (blue sidechains) shown in (A) fitting the substrate into the binding pocket.

The authors further suggest an autoinhibition mechanism to interpret the inhibition of GSK3β when serine 9 is phosphorylated in the insulin pathway (Cross *et al.*, 1995). The 35 residue N-terminal peptide, which is distorted in the crystal structure and therefore not visible, was modelled into the substrate binding site serving as a *pseudo primed* substrate analogue with the phosphorylated serine 9 as 'P(+4)' and a proline 5 in 'P(0)' occupying the pocket at the catalytic residues. The authors showed experimentally that inhibition depends on the sequence context of the serine 9, and is in fact specific to the sequence N-terminal fragment of GSK3β itself.

The structure of GSK3β from Dajani *et al.* (2001) does not reveal any insights into how GSK3β acts differently in the two signalling pathways (insulin and Wnt). However, recently a structure of a complex between GSK3β and a peptide from an interacting regulatory protein required in the Wnt pathway was published (Bax *et al.*, 2001), showing that the interaction site is close to the substrate binding site but without any overlap. This structural complex explains why GSK-3β can be inhibited in the Wnt pathway while staying active in the insulin pathway.

Similar structure and function - different sequence

As figure 1.6 shows and is further discussed in section 1.4.3 below, similar sequences generally have a similar 3D-structure which in turn determines the biochemical function of the protein, although, as explained in section 1.4.1, it is not straightforward to identify these relationships. In this section two protein structures with such a difficult relationship are discussed.

The structures of the core domain from different viral integrase proteins Dyda *et al.* (1994) are similar to ribonuclease H (RNaseH, Katayanagi *et al.* (1990); Davies *et al.* (1991)), but their sequences do not show significant similarity (Yang & Steitz, 1995; Dyda *et al.*, 1994). The integrase inserts the viral DNA into the host DNA, whereas RNaseH hydrolyses RNA strands of RNA-DNA hybrids. Despite the difference of their biological function, both enzymes perform a similar trans-esterifiaction reaction that requires either Mg^{2+} or Mn^{2+} ions and three carboxylates. Overall the reaction mechanism of both enzymes has been proposed to be similar Yang & Steitz (1995).

The topology of the core folds for the integrase and the RNaseH are the same, but the length and twist of the secondary structure elements are different, also both folds contain additional secondary structure elements. Figure 1.9 shows a superposition of both structures. The three residues of the catalytic site that provide the carboxylates for the chelated metal-ion are in similar relative positions (coloured in magenta and green). In integrase glutamate 157 (magenta) does not interact directly with the magnesium-ion, although mutagenesis has shown that this position requires a glutamate (Kulkosky *et al.*, 1992). Further, glutamate 157 is in an opposite position relative to glutamate 48 of the RNaseH. It has to be pointed out that the fold of the Avian Sarcoma Virus (ASV) integrase shown in the figure is similar to the HIV-1 integrase (Bujacz *et al.*, 1996) with a sequence identity of 24% but the relative orientation of the three active site residues are different (Bujacz *et al.*, 1996).

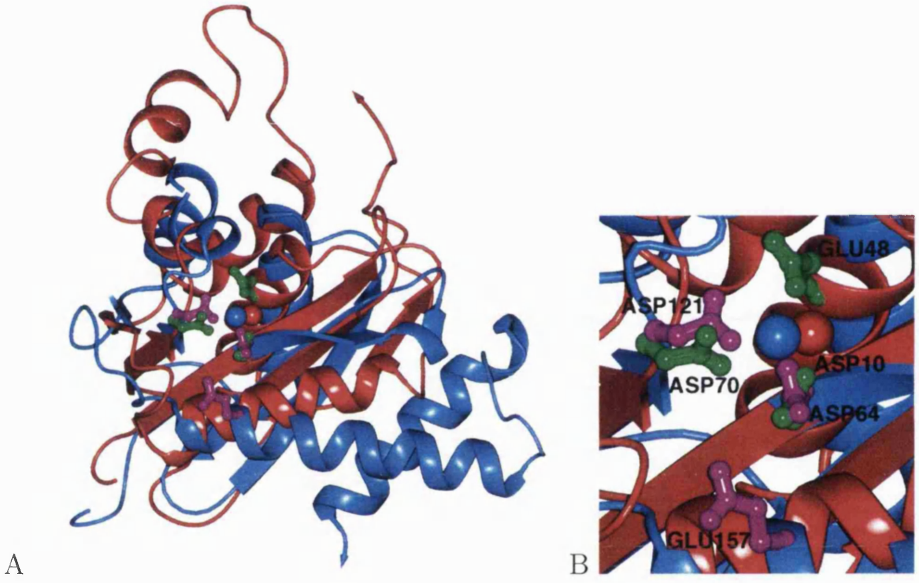


Figure 1.9: Superposition of ribonuclease H from *E. coli* (PDB code 1RDD, red structure, Katayanagi *et al.* (1993)) and integrase from Avian Sarcoma virus (PDB code 1VSD, structure shown in blue, Bujacz *et al.* (1996)). (A) The RMSD of the superposition is 3.9Å. Most similarity is found in the 5 stranded sheet, both structures contain additional secondary structure elements, although their general topology is the same. (B) Mg^{2+} binding site of both enzymes (integrase in magenta, and RNaseH in green). The two aspartates occupy similar positions whereas the two glutamates are on opposite sites of the metal ion.

The similarity between both protein domains and the proposal of a common enzymatic mechanism was identified only because their 3D-structures are available, pointing out the limitations of sequence based comparisons, and raising the question of how many of these hidden relationships there are in the protein universe.

Similar sequence and structure - different function

The sequence and structure of lysozyme and α-lactalbumin are very similar (36% sequence identity and an RMSD of 1.3Å between the structures, see figure 1.10), although their biochemical functions are different. The first 3D-structure of lysozyme was described by Blake *et al.* (1965), and was derived from Hen egg. Lysozyme is also found in other birds, mammals and insects Jolles *et al.* (1984). It degrades bacterial cell walls by cleaving the β-1,4 glycosidic linkage between N-acetylmuramic acid and N-acetylglucosamine of polysaccharides. α-lactalbumin is mainly found in mammary glands and milk. The protein changes the substrate specificity of the enzyme galactosyltransferase in the lactating mammary gland from N-acetylglucosamine to glucose to produce lactose. The first α-lactalbumin structure was published by Phillips and co-workers (Smith *et al.*, 1987). A review about the discovery, analysis and comparison of α-lactalbumin and lysozyme is given by McKenzie & White (1991).

In addition to their sequence and structural similarity, both enzymes have a similar exon-intron structure (McKenzie, 1996) suggesting a common ancestor. The different biochemical functions, despite different substrates, are rendered by two major features: (i) α-lactalbumin binds calcium, whereas only a few lysozymes have been reported to bind calcium (e.g. Nitta *et al.* (1988); Nitta (2002)), and (ii) α-lactalbumin interacts with galactosyltransferase, this interaction has not been found for lysozymes. Figure 1.10 shows a structural superposition of both proteins, highlighting the calcium binding site of α-lactalbumin (red) and the catalytic residues the lysozyme (blue).

Although α-lactalbumin and lysozyme have developed different functions, it is commonly accepted that they are homologous. However, it is not clear when in evolution the gene duplication event took place (lysozyme is believed to be the ancestor of α-lactalbumin). Some authors suggest the event happened before the divergence of birds and mammals (Prager & Wilson, 1988) while others suggest a more recent event, after birds and mammals have diverged (Shewale *et al.*, 1984). The functional divergence of both proteins cannot be explained by structural data alone, but needs careful sequence analysis and experimental work. Similar sequences and structures do not necessarily imply similar function. This is an important aspect in functional genome annotation which was discussed in section 1.4.1.

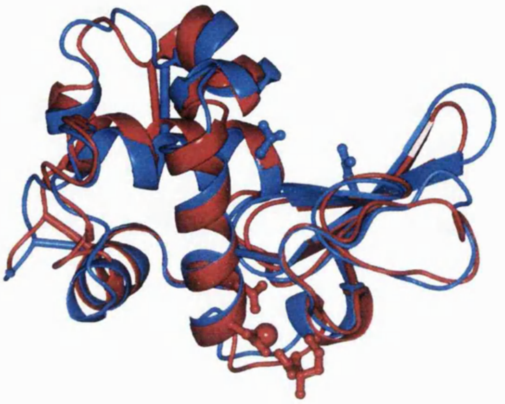


Figure 1.10: Superposition of lysozyme (PDB code 1LYZ, blue, Diamond (1974)) and α-lactalbumin (PDB code 1ALC, red, Acharya *et al.* (1989)). The catalytic sidechains ASP52 and GLU35 of lysozyme are shown. The calcium (red sphere) and the sidechains of the residues LYS79, ASP82, ASP87 and ASP88 involved in calcium binding are shown in red.

1.4.3 Structural genomics projects

Automated large scale structural genomics projects have been setup around the world to determine large numbers of protein structures (Sanchez *et al.*, 2000). There are at least fifteen such projects in North America, four in Europe using X-ray crystallography and one in Japan that uses NMR technology. Generally the aim of structural genomics projects is to solve protein structures without the focus on a particular protein. Targets may be selected carefully including those of special interest such as potential drug targets, protein families or a representative set of proteins from a particular organism. An important aspect is to have a wide range of possible protein targets so that a protein that is difficult to express or to crystallise may be skipped or suspended from the processing pipeline without having any impact on the entire project. This philosophy which is often referred to as *grabbing for the low hanging fruit* aims for the easy targets. However, the current lack of protein structures supports this point of view, and advances in technology based on the experience of ongoing projects may allow future exploration of targets that cannot be handled at this time. Nevertheless, there are projects such as the one at the Midwest Center For Structural Genomics, that include difficult targets such as membrane proteins.

As mentioned at the beginning of section 1.4, there is a large discrepancy between the number of available sequences and structures. However, structural genomics projects do not need to provide experimental structures for every single sequence, because the number of distinct 3D-architectures for globular proteins is limited to a relatively small number of folds, allowing the modelling of the structures of many proteins from a limited number of homologues for which the structures were determined experimentally.

Recent work by Vitkup *et al.* (2001) suggests that a number of 16,000 structures may be required to have representative structures for 90% of all proteins. To cover 90% of all protein families in PFAM (version 4.4 with 2,000 families, see section 1.2.3) about 4,000 structure determinations are required. More than one structure per family has to be solved if the sequence identity between members of a family is low (< 30%). Assuming that reliable homology based model building for protein structures requires at least 30% sequence identity between the target (the protein of unknown structure) and the template (the homologue of known structure), one could model all members of a protein family with a minimum number of template structures. This minimum number is determined so that all members of the family share at least 30% sequence identity to at least one template. On average a quarter of a genome is covered by PFAM (version 4.4), and so the extrapolated number of structure determinations rises to 16,000. This is the estimated number of protein structures to cover 90% of the sequence space. About 10% of these structures are already available. Targeting a 100% coverage of the protein sequence space requires four times more protein structures to be solved, and therefore a 90% coverage cut-off is a good ratio of completeness to costs. This theoretical estimate does not consider membrane proteins and technical difficulties with certain protein families, although difficulties with individual target proteins from families can be bypassed by choosing an alternative candidate target protein of the same family (e.g. from a different organism).

Target selection is critical for the success of structural genomics and has to be coordinated to avoid redundant work. Lists of targets from various projects are maintained at http://presage.berkeley.edu/ (Brenner *et al.*, 1999) and http://-www.structuralgenomics.org.

The expected benefits from having a large set of available structures (including those derived from homology modelling, see section 1.4.5) are combinations of 'new/old' folds (3D-architectures) and 'known/unknown' functions (Burley, 2000). The examples in 1.4.2 already highlighted the benefits of knowing the structure of a protein. Structures will be used for guiding experimental work such as site directed mutagenesis, protein-protein interaction studies and identification of possible ligands (e.g. inhibitors). Having a larger number of proteins with the same or a similar fold but different function sheds light into the evolutionary history of a fold. This allows the exploration of the differences between proteins that have diverged from a common ancestor, and how proteins with the same structural scaffold evolved new functions. As discussed in section 1.4.1, the structure/function relationship is complex, and there is still a lack of structural data to extract reliable rules for this relationship. New folds of proteins with known function will allow to elucidate the function of a fold, which in turn may allow to propose a function for all those members (proteins) of this fold. For a known fold with an unknown function the structure may be used to propose a function, e.g. by screening this fold for 3D-sites extracted from existing structures (Wallace *et al.*, 1997; Russell, 1998; Jonassen *et al.*, 1999).

1.4.4 Structure based classification of proteins

The protein family and domain databases discussed in section 1.2.3 derive their relevant information to cluster proteins mainly from sequence information. Another type of domain database uses protein structure to identify and cluster similar domains. Protein structure supports the identification of domain boundaries for a sequence family. A comparison of protein structures also allows the identification of structurally similar domains in the absence of obvious sequence similarity as the structural similarity of the integrase and the ribonuclease in section 1.4.2 shows.

The most commonly used structural domain databases are SCOP (Murzin *et al.* (1995); Conte *et al.* (2002), see also http://scop.mrc-lmb.cam.ac.uk/scop/) and CATH (Orengo *et al.* (1997); Pearl *et al.* (2001), see also http://www.biochem.ucl.ac.uk/bsm/cath/). Both databases are based on the PDB database which is the central repository for protein structures. SCOP (*Structural Classification Of Proteins*) has been employed extensively in this work, and therefore its architecture is described in detail. Proteins are classified via a tree with six branch levels. The top level is the *class* that summarises domains according to their secondary structure content. In SCOP version 1.53 there are five main classes, all-α, all-β, mixed α/β and $\alpha + \beta$ (domains contain a separated α and β part) and *small* domains (dominated by short domains that usually contain a complexed metal or disulphide bridges). The next level is the *fold*, that groups domains for which the secondary structure elements are arranged in a similar topology but without the need of sequence similarity. Each fold contains one or more *superfamilies* which aims to group domains for which the evidence suggests there is be a common ancestor, therefore members of the same superfamily are homologues. The evidence that two domains belong to the same superfamily can be similarity in sequence, structure and function, but may be a combination of similar structure and function without detectable sequence similarity (as for the integrase and ribonuclease H examples in section 1.4.2). Domains in the same fold but from different superfamilies are considered to be analogues, their similar structural framework is believed to have evolved independently. Since the discrimination between analogy and homology is not straightforward, a common evolutionary origin cannot be excluded for some domains within the same fold but in different superfamilies. SCOP decides conservatively, and places domains without clear evidence for common ancestry in different superfamilies. Each superfamily contains at least one *family* that groups closely related domains with at least 30% sequence identity or in some cases less identity but very similar structures and function. A *domain* itself is the next level within a family, followed by the *species*, i.e. the same domain may be present in different species. The SCOP database is constructed and maintained mainly manually, some steps of the analysis are automated.

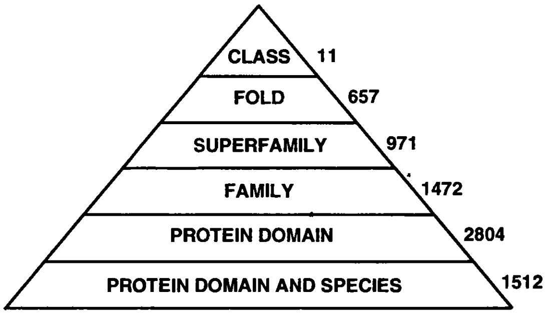


Figure 1.11: The SCOP classification. The CLASS level at the top of the triangle is the most general classification level. Several entries from a level can be summarised by the next higher level (e.g. a FOLD contains one ore more SUPERFAMILIES). The lowest level is the PROTEIN DOMAIN IN A SPECIES, i.e. the same domain may be found in different species. The numbers of distinct entries at each level are given, in total there are 26,174 domains (including the same domain in different species) in SCOP version 1.53

The CATH database is organised similarly to SCOP, it contains five levels: (i) the *class*, similar to SCOP, and contains the entities mainly-α, mainly-β and $\alpha - beta$, (ii) the *architecture* level groups domains with similar arrangements of secondary structure elements but ignoring their connectivity, (iii) the *topology/fold family* level that considers secondary structure topology (grouping analogues), (iv) the *homologous superfamily* and (v) the *sequence family* levels for similar sequences. CATH is constructed and maintained mainly automatically with some manual intervention.

1.4.5 Methods for assigning a 3D-structure to protein sequence

The previous sections have demonstrated the benefit of protein structure for the understanding of function and evolutionary relationships. Clear homologous relationships between sequences can be identified straightforward via sequence comparison e.g. using BLAST (see section 1.3.3). Thus way one can identify a close homologue of known structure for a sequence of unknown structure. However, because the structure is usually more conserved than the sequence, and similar structures often share a broad similar biochemical function (see section 1.4.1), different methods have been developed to make use of the knowledge that is derived from structure, such as physical interactions between residues distantly apart in the sequence. The aim is not only to detect distant homologous relationships but also those for which the structures share similar physical constraints which may have arisen by convergent evolution. These methods are generally summarised as *fold recognition* or *threading*[1], and were reviewed by Jones (1997); Sippl (1999); Sternberg *et al.* (1999).

One of the earliest fold recognition methods compares a template sequence with a library of profiles from proteins of known structure (Bowie *et al.*, 1991). The profiles contain observed secondary structure states and solvent accessibility for each residue position. A statistical analysis of all 20 amino acids with their states is performed for all proteins of known structure, calculating a score for each amino acid type in each state, which is used to score each residue of a target sequence in the templates residues states.

One of the most successful methods developed was THREADER (Jones *et al.*, 1992) which uses pair-potentials to evaluate an energy function for the target residues

[1] *Threading* in this context means to *thread* the residues of a sequence of unknown structure onto the backbone conformation of a template structure

in a template structure. Pair-potentials introduced by Sippl (1990); Hendlich *et al.* (1990) are derived by analysing the surrounding residues in a given radius in space for a given residue. This is a measure for the preferred amino acid environment for a given residue.

Advances in secondary structure predictions based on multiple sequence alignments and neural networks (Rost & Sander, 1993b,a; Jones, 1999b) enhanced fold recognition (similar 3D-structures have the a similar secondary structure content and topologies) and were frequently incorporated into fold recognition methods.

In the 4^{th} CASP competition (Critical Assessment of Structure Prediction) in 2000, a blind trial to predict the fold of structures that were held back temporarily from publication for the purpose of CASP, the 3D-PSSM method performed best under the fully automated methods (Kelley *et al.*, 2000). Different methods are combined to score the compatibility of a target sequence with each library sequence represented by a set of profiles that are derived from superimposed structures, solvent-potentials, secondary structure prediction and sequence homology.

If more information than just the general fold is required and a homologue of known structure is available, homology based modelling can be applied to build an accurate structural model that includes sidechains. The assumption for homology modelling is that the target sequence will have a similar fold, and therefore a similar backbone conformation for the main secondary structure elements. The backbone conformation of the homologue of known structure is used as a template onto which the sidechains of the target are placed. The model may be refined using different force fields (e.g. Sali & Blundell (1993); Sanchez & Sali (1997b)), see Sanchez & Sali (1997a); Moult (1999) for a review on comparative modelling. Flexible loops and gaps are difficult to model, and special methods have been developed to tackle this problem (Bates *et al.*, 1997). The quality of homology models strongly depends on the accuracy of the alignment between the target and the template. Reasonable models that include sidechains and flexible loops require at least 30% sequence identity (Sanchez & Sali, 1998; Bates *et al.*, 1997; Fischer *et al.*, 1999). Structural genomics projects benefit from the conservation of protein structure by building reliable models for closely related sequences (see section 1.4.3 on page 53). The growth of the sequence database and the expected growth of the protein structure database will increase the number of relationships with >30% sequence identity, increasing template selection via straightforward sequence search methods such as BLAST.

1.5 Scope and outline of this thesis

The methods used in genome annotation as described in the previous sections together with the vast amount of data that is already available requires a systematic integration. To perform comparisons across genomes, a unified annotation protocol has to be applied to all sequences of each genome. Such a cross-genome comparison highlights the differences shaping the nature of a particular organism or a group of organisms (e.g. metazoans). Commonalities between genomes reveal evolutionary relationships as well as conserved functions. Several comparative genomics projects with different aims have been developed by others which are discussed in the later chapters and are compared to this work. Here, a comparative annotation system and its application based on the protein repertoire of fully sequenced genomes is described with a focus on domains of known structure. Below the main aspects of this work are introduced.

- Chapter II describes the development of a benchmark for the protein sequence database search program PSI-BLAST (see section 1.3.5) to evaluate its performance in protein based genome annotation. For the benchmark an artificial genome is constructed from domains of the SCOP database (for which the annotation is known, see section 1.4.4), so that the ideal structural and functional annotation can be compared to PSI-BLAST results. The well characterised genome of *M. genitalium* and the genome of *M. tuberculosis* (at that time just published) are annotated via PSI-BLAST sequence comparisons. The extent of new folds and proteins of potentially new function within these genomes is estimated.

- Chapter III describes the development of a computer based annotation system that is capable of performing an automated analysis of a vast amount of protein sequences with structured storage and retrieval of the results. The annotation system is based on a relational database and an object oriented software interface to this database. Standard protein sequence based analysis tools such as those described in the previous sections (e.g. PSI-BLAST) are integrated as a part of the annotation pipeline.

- Chapter IV analyses the proteins of 14 genomes from archae, bacteria and eukaryota including proteins from the draft human genome. The extent of structural and functional annotation within these genomes is analysed and compared. The extent of domain duplications within SCOP superfamilies in the processed proteomes is analysed, including a comparison of the most abundant superfamilies, repetitiveness of domains and the co-occurrence of superfamilies in the same sequence. Membrane proteins are analysed for globular domains, and SCOP superfamilies found in membrane proteins are compared across the proteomes. Further, SCOP superfamilies found in proteins from human disease genes are compared to those found in non-disease genes. Results from other projects that analyse the fold distribution across different proteomes are discussed.

- The thesis closes with a summary and discussion of the results and suggestions for possible future developments, in particular possibilities for the annotation and analysis system described in this work.

Chapter 2

Benchmarking PSI-BLAST in genome annotation

2.1 Summary

The recognition of remote protein homologies is a major aspect of the structural and functional annotation of newly determined genomes. This work presents a benchmark for the coverage and error rate of genome annotation using the widely-used homology-searching program PSI-BLAST (position specific iterated basic alignment tool). The study evaluates the one-to-many success rate for recognition, as often there are several homologues in the database and only one needs to be identified for annotating the sequence. In contrast, previous benchmarks considered one-to-one recognition in which is was required that a single query should find a particular target. The benchmark constructs a model genome from the full sequences of the structural classification of protein (SCOP) database and searches against a target library of remote homologous domains (<20% identity). The structural benchmark provides a reliable list of correct and false homology assignments. PSI-BLAST successfully annotated 40% of the domains in the model genome that had at least one remote homologue in the target library. This coverage is more than three times that obtained if one-to-one recognition is evaluated (11% coverage of domains). Although a structural benchmark was used, the results equally apply to just sequence homology searches. Accordingly, structural and sequence assignments were made to the sequences of *Mycoplasma genitalium* and

Mycobacterium tuberculosis (see http://www.bmm.icnet.uk/PsiBench). The extent of missed assignments and of new superfamilies can be estimated for these genomes for both structural and functional annotations. The work described in this chapter has been published in _Journal of Molecular Biology_ (Muller _et al._, 1999).

2.2 Introduction

At the start of this work in 1998 it was clear that over the next few years a major activity in molecular biology would be the assignment of protein structure and function to ORFs in newly determined genomes (Bork *et al.*, 1998; Bork & Koonin, 1998). A standard approach is to perform database searches to identify homologous protein sequences which will have similar three-dimensional structures and often a related function (Bork & Koonin, 1998; Chothia & Lesk, 1986; Hegyi & Gerstein, 1999; Karp, 1998; Martin *et al.*, 1998). Indeed an initial report of a newly determined genome nearly always reports the results of homology searches. However, despite the importance of the methodology, there has only been limited systematic evaluation of the accuracy, both in terms of coverage and errors, of the procedure (Brenner *et al.*, 1998; Park *et al.*, 1998). This work uses a structural benchmark developed by Chothia and coworkers (Brenner *et al.*, 1998; Park *et al.*, 1998) from the SCOP (Structural Classification of Proteins) database (Murzin *et al.*, 1995) to assess the accuracy of homology based annotation[1] of ORFs. The results of the benchmarking will be used to interpret assignments of protein structures to ORFs in two bacterial genomes. Although a structural benchmark is used, the conclusions of the study relate to the accuracy of genome annotation by homology to other proteins irrespective of whether these proteins have a determined structure.

The SCOP database employs sequence, structural and functional relationships between protein domains of experimentally determined three dimensional conformation (Murzin *et al.* (1995), see section 1.4.4 for details); In summary: protein domains of similar three-dimensional structure are classified into the same superfamily if there is substantial evidence to propose that they are homologues (i.e. the result of divergent evolution). A key feature is that without structural information,

[1] Here, annotation is defined as the assignment of a functionally or structurally characterised homologue to an uncharacterised protein sequence

many homologous relationships between proteins in the same superfamily could not have been established. Domains that lack strong evidence for divergence but share a common structure are assigned to the same fold family. In general, domains with a common fold are presumed to be structural analogues (i.e. the result of convergence) but a homologous relationship remains a possible explanation.

Chothia and coworkers established a structural benchmark for sequence homology search algorithms based on recognising superfamily relationships in SCOP (Brenner *et al.*, 1998; Park *et al.*, 1998). A database of sequences with less than 40% identity was derived from SCOP. An optimal homology algorithm should identify all pairs of sequences for domains within the same superfamily (i.e. total coverage) without detecting any erroneous relationships between different superfamilies (i.e. zero errors per query). In practice, algorithms are not optimal and different methods can be compared from their different coverage at a chosen observed error rate. Park *et al.* (1998) showed that the iterative profile approach of PSI-BLAST (Altschul *et al.*, 1997) and the hidden Markov models implemented in SAMT98 (Karplus *et al.*, 1998) were found to identify three times as many remote homologues as the sensitive pairwise algorithm FASTA (ktup=1) (Pearson & Lipman, 1988).

The evaluations of the accuracy of different homology search algorithms by Chothia and coworkers (Brenner *et al.*, 1998; Park *et al.*, 1998) and the related studies by Salamov *et al.* (1999), evaluate a one-to-one success rate in terms of whether a single probe identifies a particular homologue in the library (see table 2.1). This measure, appropriate for comparison of the performances of different algorithms, is not the most useful to benchmark actual genome assignment. A better measure for genome annotation is the one-to-many success rate as there are several potential homologues in a database and only one needs to be identified to propose a common three-dimensional structure and probable related function. One would expect that the presence of multiple homologues would increase the accuracy of genome assignment for populated homologous families. In addition, these previous benchmarks considered recognition of protein domain probes and targets whilst often the actual genome will be a multi-domain protein. Not only could this lead to additional problems in assignment, but it also raises the question of how well domain boundaries can be identified.

It is important therefore that the benchmark for genome assignment represents

Probe	Targets	Found< e-value	one-to-one success	one-to-many success
A	B	√	1	1
	C	√	1	
	D	X	0	
B	A	√	1	1
	C	X	0	
	D	X	0	
C	A	√	1	1
	B	X	0	
	D	X	0	
D	A	X	0	0
	B	X	0	
	C	X	0	
TOTAL SUCCESS RATE			4/12	3/4

Table 2.1: One-to-one and one-to-many assignment Sequences A, B, C and D are homologues (i.e. the same SCOP superfamily). In a benchmark, each sequence would be taken as probes in turn and their success at identifying the remaining target homologues determined (i.e. 'Found < e-value'). In a one-to-one benchmark the success of finding each pair is considered. In one-to-many only one correct assignment is needed to classify the probe. This highlights the difference in the two methods of assignment. In the approach of Brenner *et al.* (1998) and Park *et al.* (1998), the observed error rate is evaluated and is the basis for comparison of algorithms (see text).

the actual situation. Accordingly, in this work a model genome (the SCOP genome) is constructed from a selection of the entire protein sequences forming protein domains in SCOP. The performance of PSI-BLAST for genome assignment will be evaluated since this program is exceptionally widely-used and can be readily installed at any site (see e.g. Aravind & Koonin (1999); Koehl & Levitt (1999); Sternberg *et al.* (1999)). Indeed, today, PSI-BLAST is the standard tool for an initial, state-of-the-art analysis of newly determined genomes. The results of the benchmark are then used to interpret the PSI-BLAST analysis of the fold composition in the *Mycoplasma genitalium* and *Mycobacterium tuberculosis* genomes.

2.3 Development of the SCOP genome benchmark

For details of the materials and methods, see section 2.7 on page 80.

2.3.1 SCOP1625 - representative target domain library

Structural information was taken from SCOP release 1.37 (Murzin *et al.*, 1995). Each SCOP entry consists of a structural domain. These domains can be continuous or discontinuous (i.e. in which the same structural domain is formed from two or more discontinuous sequence segments) (Wetlaufer, 1973). The unit used in this study is referred to as a 'region' which is defined as one domain or a segment of a discontinuous domain and represents one segment of the protein sequence.

To generate a representative library[2], SCOP entries have been excluded if they did not have coordinates in the protein data bank (Abola *et al.*, 1997; Berman *et al.*, 2000), any errors in residue numbering, an X-ray resolution of >3.5Å or undefined residues, length <20 residues, C^{α} trace only, more than 15 C^{α}-C^{α} separations of >4.0Å or more than five undefined residues. From 11,373 domains, a set of 1,560 domains was generated so no pair shared >40% identity. These domains contain 1,625 regions which is the SCOP1625 target library.

2.3.2 SCOP genome probe

The SCOP genome was constructed to have complete chain sequences. Any sequence in SCOP1625 that was only part of a chain was replaced by the entire chain sequence. This yielded 1,300 different sequences comprising 934 single domain chains and 366 multi domain chains. The sequences are from a range of different organisms. The SCOP query genome contained 1,845 regions. In this genome there are 224 regions that cannot be annotated (i.e. these are the only representatives for a SCOP superfamily), and this provides a model for the types of errors that can occur in actual genome assignment when there are no homologues in the database. For example, the identification of domain boundaries may be subject to more errors if there are no homologues for parts of a protein. However the SCOP genome is limited as it only includes a few transmembrane and coiled-coiled domains, and real genomes tend to have a higher fraction of these types of structures.

[2] This library was created by R.M. MacCallum

2.3.3 Assignment of structural regions to the SCOP genome

In outline (see section 2.7 for details), PSI-BLAST (Altschul *et al.*, 1997) performs iterative searches against a non redundant sequence database (NRPROT-SCOP) that includes every non identical representative from the standard sequence databases together with the sequences of all the regions in SCOP1625. The benchmark is to evaluate the accuracy and coverage of detecting remote homologues to the SCOP genome.

In PSI-BLAST, the confidence in a particular sequence hit to the query is quantified by an e-value that indicates the theoretically expected number of erroneous matches per query (also see section 1.3.4). Up to 20 iterations of PSI-BLAST were performed and all hits to SCOP1625 from any iterations are stored. For hits to the same region within query, the one with the best (lowest) e-value is taken. Hits that overlap within a similar region in the SCOP protein are clustered. Two parameters determine which match is taken as the assignment. First the percentage of the target (i.e. known) SCOP region that is included in the PSI-BLAST match must be greater than a cut-off value *t*. Thus one can exclude a match to a small fraction of the target that may be erroneous. After this, the match with the best e-value is taken.

For the benchmark only matches to remote homologues are considered. Here 20% identity for long alignments (>350 residues) is used to distinguish between close and remote homologues with a progressively higher identity required for shorter alignments based on the relationship derived by Rost (1999).

The proposed annotation generated using PSI-BLAST is then compared to the real assignment of the query. This is performed by associating the mid point of each proposed region with its nearest mid point of the real region of the query. If the SCOP superfamily of the real and proposed region is the same, then this is a correct assignment. If there are more proposed regions than real regions in the query, one or more of the proposed regions are flagged as 'over-assignments' in the benchmark.

2.3.4 Accuracy measures

The accuracy of genome assignment can be considered in terms of two measures: coverage and the error rate. The coverage of true positives is the number of correctly assigned regions divided by the number of regions in the SCOP genome that have a homologue (1621). The assignment of a target to a region within the query that is from a different superfamily than the target is defined as a false positive. The error rate is the number of false positive assignments divided by the number of SCOP query regions (1845).

A correct assignment is when a region in the SCOP genome is matched by PSI-BLAST to a target region of the same SCOP superfamily. Sequence based profile methods can detect analogous folds in addition to homologues (Fischer *et al.*, 1999) which would lead to erroneous functional assignments (although members of a diverse superfamily can have different function). Thus, in our study assignment to the same SCOP fold but different superfamily is taken as an incorrect result. However, the SCOP classification of domains into the same superfamily is conservative. In preliminary work, several errors occurred when there was an assignment to the correct fold but the wrong superfamily for a β/α TIM-barrel. This suggested that the SCOP classification was too conservative for these superfamilies. Accordingly, any correct assignment to the TIM-barrel fold irrespective of superfamily is taken as correct. In addition, any assignment between a nucleotide-binding domain and a FAD/NAD(P)-binding domain (two different SCOP folds) is not treated as an error. In the benchmark, there were four such assignments to different superfamilies for TIM barrels and four for nucleotide-/FAD/NAD(P)-binding domains.

2.3.5 Parameter selection

First suitable parameters for the percentage t of the target that needs to be identified by PSI-BLAST and the standard e-value cut-off were determined. Figure 2.1 plots the coverage and error rate against different t-values for three different e-value cut-offs (5×10^{-6} , 5×10^{-4} and 5×10^{-2}). When the t cut-off is above 50%, the coverage begins to decrease markedly. In contrast, errors tend to accumulate when t is less than 50%. Accordingly we chose a value of t of 50% as optimal. A commonly used PSI-BLAST e-value of 5×10^{-4} (i.e. 0.05%) yields an observed error rate in our final assignment of 0.9%. Note that the PSI-BLAST e-value relates to the estimated error rate from a single iteration. The observed error rate is the result of several iterations and the subsequent structural assignment that includes a length requirement. The benchmark therefore provides an estimate of the relationship between a PSI-BLAST e-value and the resultant error rate in genome annotation.

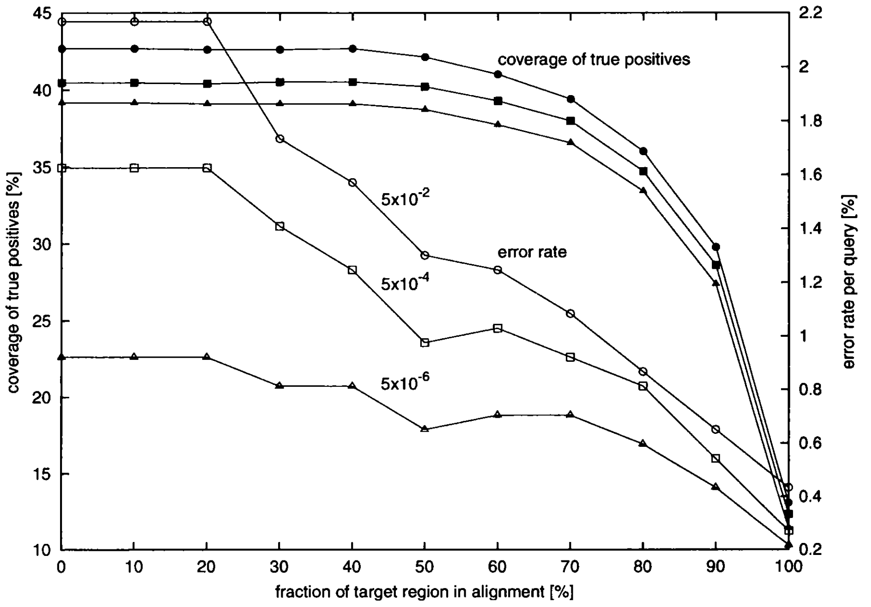


Figure 2.1: Coverage and errors for genome assignment for different parameters. The graphs show the percent coverage of true positive matches divided by the total number of possible assignments (left ordinate and filled symbols) and the error rate per query region (right ordinate and open symbols). These values are plotted for the different percentages of the target domain region included in the alignment and at different e-values

2.4 Results of the SCOP genome benchmark

2.4.1 Assignment coverage

Table 2.2 presents the results of the evaluation of the accuracy of genome assignment at the PSI-BLAST e-value of 5×10^{-4}. To recapture, the 1,300 sequences in the SCOP genome contained 1,845 regions (domain segments, see section 2.3.2). There were 1,254 sequences that had at least one potential remote homologue in the target database. There were 1,621 query regions that could be assigned and PSI-BLAST correctly identified 652 of these regions. Thus the percent coverage for assigning remote homologues (<20% identity) in the model genome is 40%. There were 16 false positive assignments and two over-assignment (see below). Table 2.2 also gives the results of genome assignment in terms of sequences with at least one region recognised, and with this measure the percent coverage remains at 40%. However, on a per residue basis the percentage coverage falls to 32%. This lower coverage is due to alignments not including the complete query sequence but still having the correct assignment.

	Sequences	Regions	Residues
No. in SCOP genome	1,300	1,845	299,910
No. with at least one region that can be assigned	1,254	1,621	263,863
No. correctly assigned	503	652	84,827
Coverage of correct assignment	40%	40%	32%
No. of false positive assignments	13	16	1,985
No. of over assignments	2	2	163

Table 2.2: Accuracy of genome assignment. Sequences refer to each chain, i.e. model ORFs; region refers to a domain segment. For sequences, correctly assigned means that at least one region has been correctly assigned (i.e. there is some correct information about the sequence) irrespective of whether other regions are not assigned or have been erroneously characterised. Similarly, errors for sequences are reported irrespective of whether another region in the sequence has been correctly assigned.

An important aspect of genome assignment is that for many of the queries there are several database homologues and only one needs to be identified to assign the protein superfamily. The importance of this is demonstrated if the accuracy of one-to-one assignment is evaluated. This corresponds to the benchmark used previously (Brenner *et al.*, 1998; Park *et al.*, 1998; Salamov *et al.*, 1999) when accuracy is considered in terms of each query recognising a correct one-to-one relationship between database entry. In this study at the PSI-BLAST e-value of 5×10^{-4}, there are 15,469 potential pairwise relationships between regions that could be identified (this corresponds to the query-target space for a one-to-one evaluation) and only 1,671 (11%) were correctly assigned. Thus identification of remote homologues (<20% identity) in structural genome analysis has 3.6 times more true positive coverage than obtained in detecting pairwise relationships.

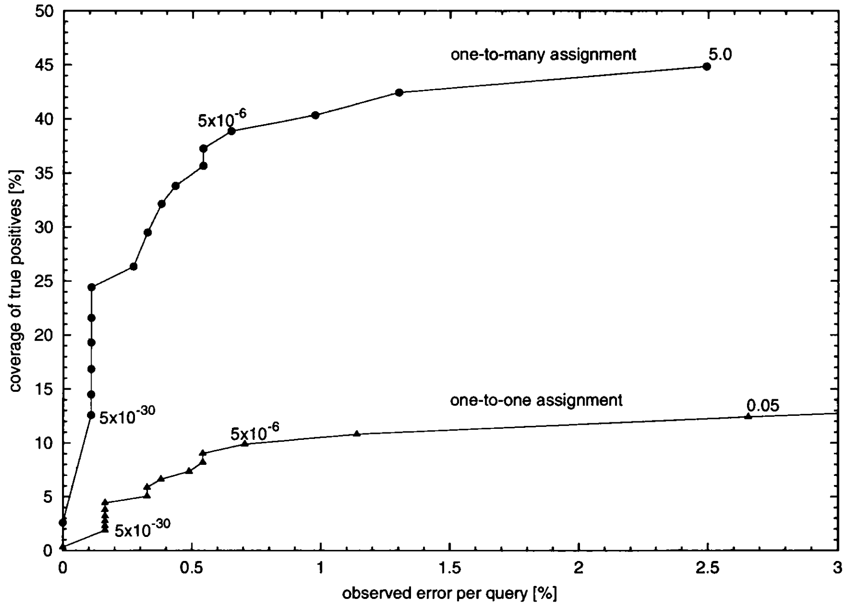


Figure 2.2: Coverage plotted against observed error rates. The cumulative coverage and observed error rate corresponding to different PSI-BLAST e-values are plotted for one-to-many and one-to-one evaluations. The smallest e-value is 5×10^{-60}.

The above comparison of one-to-many and one-to-one coverage is made at a particular PSI-BLAST e-value. As demonstrated by Brenner *et al.* (1998) and Park *et al.* (1998), comparisons of approaches should be performed by consideration of plots of the coverage of true positives against the observed error rate. For each approach, the cumulative coverage and observed error are plotted as the theoretical error-rate from the approach increases. Figure 2.2 presents these plots for the one-to-one and one-to-many assignments. At any observed error rate per query, there is a several fold greater coverage in annotation via one-to-many compared to pairwise recognition measured by one-to-one.

For each superfamily in the SCOP genome, the average percent coverage of superfamily assignment from one-to-many recognition was calculated and then plotted against the average number of cross-validated members in the superfamily (figure 2.3). One might expect that for one-to-many superfamily assignment (figure 2.3 (a)), there would be a tendency that the percent coverage would improve as the size of the superfamily increases, but this is not observed. This is explained by figure 2.3 (b) which shows that the percent coverage for detecting remote one-to-one relationships tends to decrease with increasing superfamily size. Some large superfamilies, such the immunoglobulins and the Rossmann fold, contain a diverse set of members and even sensitive search methods such as PSI-BLAST have difficulty in detecting many of the one-to-one relationships.

2.4.2 Length of region assignment

In the assignment of domain regions to multi-domain query sequences, there could be substantial errors in delineating the domain boundaries. In this study for each region in a multi-domain query the offset of the assigned location of the domain boundary to that reported in SCOP has been evaluated. A perfect assignment would have a zero offset. No offsets were calculated for the N- and the C-termini as these are easier to determine. Figure 2.4 is a histogram of the frequency of each offset length. 65% of the domain boundaries are correctly determined to within 5 residues and 86% to within 20. This shows a high accuracy in automatically delineating domain boundaries given that the query and the target are remote homologues.

Figure 2.4 is helpful in both theoretical and experimental studies to characterise a sequence. For example, in structural studies in which the domain will be cloned and expressed, it is helpful to know the likelihood of a domain boundary being correct.

2.4.3 Analysis of errors

There were 16 false positive classifications and two over-assignments where two regions are assigned to a query protein that has only one continuous domain. It is useful to examine these errors to identify commonly occurring problems.

Six classification errors are due to short cysteine rich regions, for example false assignments between tumour necrosis factor receptor and EGF/Laminin superfamilies. The problem caused by cysteine rich regions has been noted previously (Huynen *et al.*, 1998; Park *et al.*, 1998). Three of the errors are introduced by the algorithm we used to identify the positions of regions in the query. For a query protein with a discontinuous domain the target spans both of the two regions of the discontinuous domain and the intervening one, consequently the target is erroneously assigned to the intervening domain although the assignment to the flanking regions of the dis-

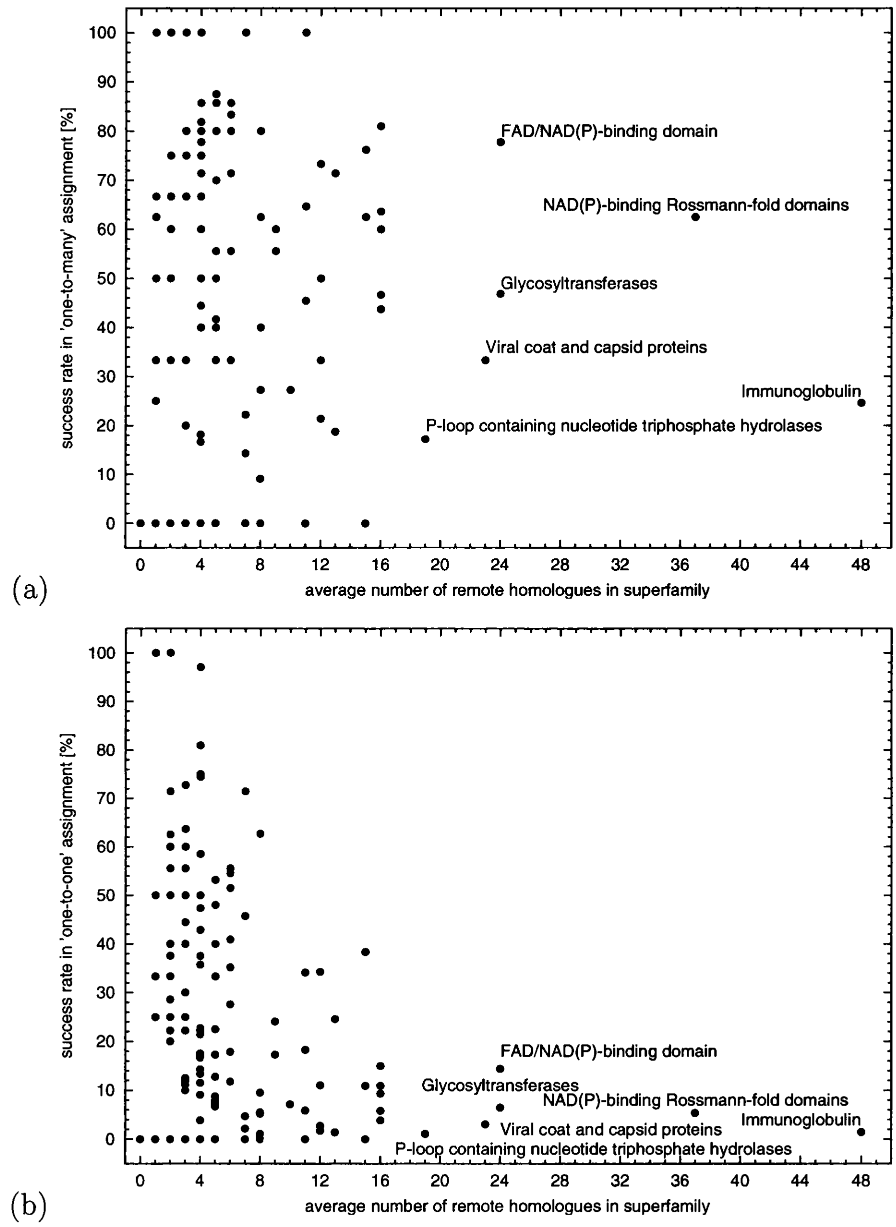


Figure 2.3: Relationship between assignment accuracy and superfamily size. The percent coverage of genome assignment (one-to-many) is plotted against the average number of members in the cross-validated superfamily in the target library (a). Results for evaluation of one-to-one assignment are shown in (b).

continuous domain was correct. PSI-BLAST did not produce two separate sequence pairs but one long gapped one. If this gap is longer than 25 residues, a warning is generated by the program developed for this analysis. This occurred 25 times, and

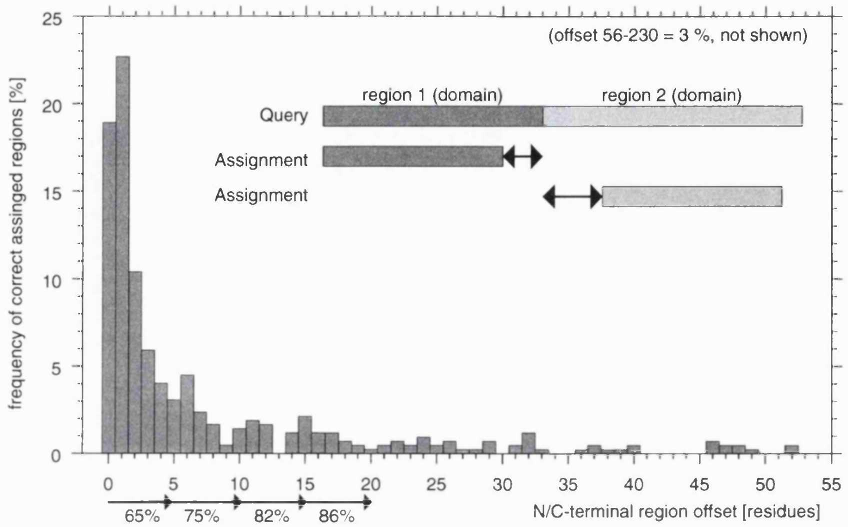


Figure 2.4: Accuracy of domain identification. Histogram of the normalised frequency of the offset error in domain identification. Offset is the number of residues error in the delineation of a domain boundary. The N- and C- terminal boundaries of the full sequence are not included. The diagram includes 97% of the observed offsets. The included scheme shows two possible errors when assigning sequences to regions in a the query. Percentages below the arrows give the cumulative frequency of offsets included.

three of these warnings correspond to these erroneous assignments. The presence of long gaps provides a flag for possible errors.

The causes of the remaining errors are not obvious but several may be due to the incorrect construction of the PSI-BLAST profile. These errors can be identified, and accordingly all PSI-BLAST annotations in which more than one superfamily was assigned to the same query segment were considered as these are conflicting assignments. There were three occurrences of this, two correspond to an actual erroneous assignment. These two erroneous assignments were in two queries from the same superfamily. Thus in the benchmark, conflicting superfamily assignments can be used to indicate a potential error.

2.5 Application to bacterial genomes

Structural annotation based on the SCOP1625 library was performed on two bacterial genome sequences. Firstly, this serves to relate the results from the model SCOP genome to real genomes and thereby evaluate the usefulness of the benchmark. Secondly, structural assignments provide valuable insights into the function and evolution of the organism.

In this work the *Mycoplasma genitalium* (MG) and *Mycobacterium tuberculosis* (TB) genomes are considered. MG is a relatively small genome with 479 ORFs and has been widely studied for structural annotation by several groups (Fischer & Eisenberg, 1997; Huynen *et al.*, 1998; Rychlewski *et al.*, 1998; Teichmann *et al.*, 1998, 1999). In contrast, TB is far larger (3,924 ORFs) and has not been extensively studied in terms of structural annotation[3] (see Frishman *et al.* (2001), http://-pedant.mips.biochem.mpg.de). Details of the assignments can be on our Web page, see http://www.bmm.icnet.uk/PsiBench.

2.5.1 Structural annotation using SCOP1625

For the MG genome with 479 ORFs (174,566 residues) sequences of the SCOP1625 database are assigned to all or a part of 136 ORFs (28% of the ORFs). These 136 MG sequences represent 201 domains with 208 regions (21% of the residues). There are 7 discontinuous domains with two regions each. Of the 208 regions, 88 (10% of the residues) were assigned by close homologues (i.e. >20% identity based on the Rost (1999) cut-off) whilst 120 regions (11%) are assigned via a remote homology.

The TB genome is 7.6 times larger than that of MG with 3,924 ORFs and 1,331,539 residues, and it is important to evaluate whether the structural assignment is similar to that of MG. Of the 3,924 ORFs in TB, 1,079 could be assigned completely or in part to a sequence in the SCOP1625 database (27% of the ORFs). The assignments represent 1,566 domains with 1,639 regions (23% of the residues). There are 73 discontinuous domains with 2 regions each. Of the 1,639 regions, 448 (7% of the residues) were assigned by close homologues and 1,191 regions (16% of the residues) by remote homologues. Thus at the general level of structural assignment

[3] Between 1998 and 1999 when this study was carried out.

MG and TB are similar although there is a smaller percentage of close homologues in TB than in MG.

When, however, the most commonly occurring superfamilies are considered there are major differences between the two genomes (table 2.3). The most common superfamily in MG is the P-loop nucleotide triphosphate hydrolase yet this occurs at rank 10 with 36 matches in TB. In contrast the most common superfamily in TB is the NAD(P)-binding Rossmann domain with 123 matches compared to its rank 11 with 3 matches in MG. The general observation is that certain superfamilies tend to occur roughly a fixed number of times in the bacterial genomes irrespective of the genome size (e.g. the class I amino acid (aa) -tRNA synthetases catalytic domain). In contrast, other superfamilies such as the Rossmann fold undergo duplication and diversification of function in the larger TB genome. Certain superfamilies were not observed in MG but are common in TB. In particular, the thiolase superfamily occurs at rank 4 in TB, probably due to its important role in fatty acid metabolism which may be linked to the complex cell envelope rich in lipids. The acetyl-CoA dehydrogenase and luciferase like domains may also be linked to fatty acid metabolism in TB and were not found in MG. The general observations about the frequencies of superfamilies in these two genomes are in agreement with the pedant database (Frishman *et al.* (2001), http://pedant.mips.biochem.mpg.de) although there are differences in the exact numbers due to differences in the methodologies of assignment.

Several other groups have analysed superfamily populations (Gerstein, 1997, 1998b; Gerstein & Levitt, 1997; Wolf *et al.*, 1999; Teichmann *et al.*, 1998, 1999). Work by Teichmann *et al.* (1998) using PSI-BLAST first with the MG sequence and then with the known structures as the queries (i.e. two-way PSI-BLAST) identified more occurrences of the superfamilies in MG than obtained in this work. However, the two studies give the same results for rank one and for the top five ranking superfamilies. Thus the observations in this work about the relative populations of superfamilies between MG and TB are likely to remain after adding the additional hits obtained from two-way PSI-BLAST.

Teichmann *et al.* (1998) describe how the rate of domain duplication can be calculated from the number of homologous domains in a genome. The basic assumption is that all domains within the same superfamily have arisen via duplication from

Superfamiliy description	MG		TB	
	rank	freq	rank	freq
P-loop nucleotide triphosphate hydrolases	1	20	10	36
Class II aaRS and biotin synthetases	2	10	39	10
Nucleic acid-binding proteins	3	9	21	17
Class I aa-tRNA synthetases (RS), Catalytic domain	4	8	39	10
FAD/NAD(P)-binding domain	4	8	2	57
α/β-Hydrolases	6	4	3	53
Anticodon-binding domain of Class II aaRS	6	4	76	4
Thiamin-binding	6	4	13	26
Adenine nucleotide alpha hydrolases	6	4	65	5
Actin-like ATPase domain	6	4	31	12
NAD(P)-binding Rossmann domain	11	3	1	123
Thiolase	-	-	4	48
S-adenosyl-L-methionine-dependent Methyltransferases	11	3	5	43
Luciferase	-	-	5	43
TetR/NARL DNA-binding domain	-	-	7	42
Acyl-CoA dehydrogenase (flavoprotein), N-terminal and middle domains	-	-	8	39
Acyl-CoA dehydrogenase (flavoprotein), C-terminal domain	-	-	8	39

Table 2.3: Popular superfamilies in MG and TB. The table lists all SCOP superfamilies which occur in the top 10 ranks in MG and/or TB.

a common ancestor. A superfamily with e.g. ten domain members in a genome therefore was duplicated nine times. Results from this work give figures for the percentage of protein domains that arose by duplication in MG and TB as 49% and 84%. Thus as suggested by others (Teichmann *et al.*, 1999), the larger genome of TB shows a far greater extent of domain duplication. Teichmann and coworkers using two-way PSI-BLAST on calculated a domain duplication rate for MG of 58%. Thus the precise figures for domain duplication obtained in this work will need to be revised using two-way PSI-BLAST, but the general observation about the relative rates of duplication should remain valid.

2.5.2 How much of the genome can be classified

A further consideration of this work is how much of the MG and TB genomes have either structural or both sequence and structural homologues in the databases. For structural assignment, the SCOP1625 data set was updated by including PSI-BLAST matches to a sequence of the PDB (Abola *et al.*, 1997; Berman *et al.*, 2000). This resulting structural database includes proteins with coordinates deposited after SCOP was compiled, and accordingly a larger fraction of the genomes will be structurally annotated than described in section 2.5.1 that used only SCOP1625 data.

For sequence assignments one needs to include any match to any sequence that has a useful annotation. To consider this, any match with the text description that includes the words 'probable' or 'hypothetical' was excluded, although this is only a first approximation to evaluate what corresponds to a functionally useful annotation. In addition matches of species name (MG or TB) between query and database were ignored as a useful annotation. Segments were identified as low complexity regions if they were longer than 24 residues using the SEG program with default parameters (Wootton & Federhen, 1996). Coiled-coil region were found using MULTICOIL with defaults (Wolf *et al.*, 1997). Transmembrane regions were identified using the 'certain' assignment in TOPPRED (von Heijne, 1992).

Figure 2.5 presents pie-charts of the results in terms of residues and represents the results in 1999. In the SCOP benchmark for remote homologues, 28% of the SCOP genome was annotated and 59% was missed (undetected homologues) so there are 2.1 times as many potential remote homologues in the database as detected by PSI-BLAST (figure 2.5 (a)). To consider the potential for structural assignment in genomes, first close and then remote homologues of known structure were identified. From the benchmark the scaling factor of 2.1 was taken and applied to the fraction of remote structural matches. Thus there are 32% of missing structural matches in MG and 36% in TB (figures 2.5 (b) and (c)). Enhanced methods such as two-way PSI-BLAST, hidden Markov models and threading have a major role to play in structural annotation of genomes (see Jones (1999a) for another approach to estimate missing structural matches in genomes). As there are very few coiled-coils, transmembrane and low complexity regions in SCOP and the PDB, these must be added to the pie-chart for structural assignment in MG and TB (see figures 2.5 (b) and (c)), there is <1% of coiled-coils in TB). Therefore, as an estimation, there remains 31% of the residues in MG and 22% of TB that are in new superfamilies from globular proteins.

To evaluate the potential for functional annotation, first matches to close homologues of either structure or just sequence were identified and then the remote matches were considered. Many short low complexity regions, coiled-coils and transmembrane proteins will be matched by PSI-BLAST to homologues in the sequence database. Therefore, unlike the pie-charts for structural assignment, we do not indicate separately coiled-coils and transmembrane regions, (see legend to figure 2.5 for more details). Assignments to low complexity regions longer than 24 residues will generally not be matched by PSI-BLAST and are indicated in the pie-charts, (1% in the MG and 5% in the TB genome). The correction factor of 2.1 can then be applied to the remote homologues to estimate the missed homologues in the databases (17% for MG and 11% for TB).

Figure 2.5(d) shows that in MG if all the missed homologues were identified, there is only a small fraction of the MG genome left to annotate. Although homologous proteins can have different functions, this remains a rare event for the broad function (Hegyi & Gerstein, 1999; Russell *et al.*, 1998b). Thus the pie-chart suggests that nearly all the gene functions of MG are described in annotations of the present sequence databases. Indeed it has been suggested that the MG genome is not much larger than the minimal required for cellular life (Mushegian & Koonin, 1996).

For TB (figure 2.5(e)), after allowing for missed homologues, there remains roughly 14% of the genome that is formed from genes that are not homologous to annotated genes of known function. Thus there may well be several genes of previously unrecognised function in TB.

The above calculations are based on the assumption that the ratio of detected to undetected remote homologues found from the SCOP benchmark will apply to the actual genomes. Although this ratio varies for the different superfamilies (see figure 2.3(a)), the overall trend is that the ratio is not dependent on the size of the superfamily, and for many genomes the value from the SCOP benchmark should provide a valid first approximation. Note that the pie-charts are based on fractions of residues annotated and some other workers (Mushegian & Koonin, 1996; Teichmann *et al.*, 1998; Jones, 1999a) take a different approach and consider there is structural / functional annotation for an ORF if any part of that ORF is homologous to a database protein of known structure / function.

2.6 Discussion and Conclusions

This study benchmarked the coverage and error rate of PSI-BLAST when applied to the recognition of remote homologies in the annotation of a genome. The evaluation was based on recognising remote homologies (<20% identity) between protein domains of known structure. The critical aspect of the evaluation is that it included

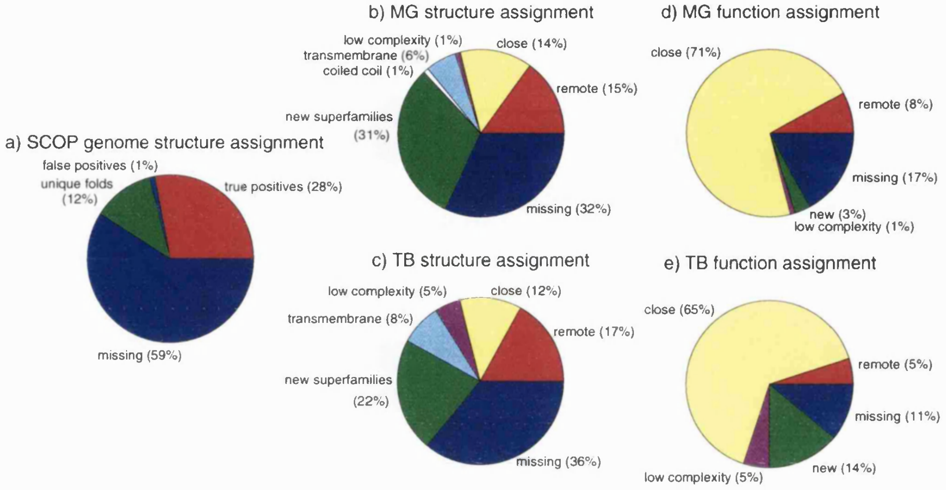


Figure 2.5: Identified and missed homologues. Results are on a per residue basis. (a) The results of the SCOP benchmark. For remote homologues (<20% identity), the data in table 2.2 is plotted as a pie-chart. The figure shows the percentage of the SCOP genome that (i) was correctly assigned, (ii) incorrectly missed, (iii) erroneously assigned and (iv) that were in a unique superfamily with no target assignment possible. The ratio of (ii) / (i) provides the correction factor used in the other charts to estimate the missed remote homologues. (b) The results of structural assignment for MG. The chart shows the percentage of the genome that has a close structural homologue, a remote structural homologue and the estimation of the missed structural remote homologues. (c) As (b) but for TB, coiled-coils are < 1%. (d) The results of functional assignment for MG. Matches are to sequences with functional annotation. Missing are undetected homologues. New are ORFs with no previously known homologous. (e) As (d) but for TB. Transmembrane regions are 6% of the residues in MG and 8% in TB, 1% of the residues are in coiled-coil regions in MG and < 1% in TB. As figures (b) and (c) show these regions are not matched by any sequence of known structure (in fact there are a few matches but without impact on the percentage figures). In figures (d)-(e) transmembrane helixes and coiled-coils are not shown in separate fractions in the pie-charts because about 2/3rd of the transmembrane regions and nearly all of the coiled-coils are matched by sequence hits of known function (data not shown). That means the remaining 1/3rd (2% of the residues for MG and 3% for TB) of the transmembrane regions are distributed in the fractions for missing and new functions. Low complexity regions longer than 24 residues are indicated in (b)-(e) because these regions cannot matched by any sequence.

the requirement that only one out of several possible homologies needs to be identified to assign the query to a homologous superfamily. In addition, the multi-domain structure of queries is included in the evaluation. Thus the model used is close to the actual aspects of genome annotation.

Although a structural benchmark is used, the results are particularly relevant to evaluate the accuracy of assigning proteins to any homologous sequences (including those of unknown structure), which is the standard first step in the interpretation of a genome. In particular, methods such as two-way PSI-BLAST become computationally prohibitive if a representative sequence (rather than structural) database becomes the probes. Profile methods such as IMPALA (Schaffer *et al.*, 1999) provide an alternative to the two-way PSI-BLAST approach. A query sequence is compared to a library of profiles each representing a protein family (e.g. a SCOP superfamily). Clearly fold recognition methods cannot be applied when there are no structural homologues. In one respect, the benchmark does not carry over to just sequence annotation as we used the structure based domain information that is not available for all sequences without coordinates. However, domain assignment can still be obtained from databases such as PRODOM (Corpet *et al.*, 2000), SMART (Letunic *et al.*, 2002) and PFAM (Bateman *et al.*, 2002) for many sequences without known structure (see section 1.2.3 for an introduction into domain databases).

The key results of the study are:

- Genome assignment is based on one-to-many identification and successfully recognises around 40% of the remote homologies (<20% identity) between protein domain regions. This corresponds to recognition of 32% on a per residue basis.
- Previous benchmarks evaluating one-to-one rather than one-to-many identification would suggest a three-fold lower success rate.
- In general, the more populated superfamilies do not have improved success rates for genome identification.
- Domain boundaries determined from the alignment of the query to the target are well characterised, 65% are correctly found to within 5 residues.
- There are major differences between the most common superfamilies in the minimal bacterial genome of MG compared to that in TB.
- Based on the success rate for detecting remote homologies, about 30-40% of the residues in the analysed bacterial genomes do not correspond to a protein of known structure.

- There are very few proteins in MG that do not have a homologue of annotated function in the databases but there probably are far more ORFs in TB with novel function.

2.7 Materials and Methods

2.7.1 Sequence database for PSI-BLAST profiles

A non-redundant protein sequence database (NRPROT) containing 302891 entries was generated by progressively taking sequences from the Protein Data Bank (Abola *et al.*, 1997; Berman *et al.*, 2000), TrEMBL-NEW, TrEMBL, SWISSPROT-NEW, SWISSPROT (Bairoch & Apweiler, 2000) and PIR (Barker *et al.*, 2000) but excluding any sequences that are 100% identical[4]. Next, the SCOP1625 target library was added to NRPROT so that hits to known structures can readily be identified. To ensure the optimal generation of sequence profiles (but not for structural matches), to the above sequence library the concatenated regions of discontinuous domains and the entire chains from multi domain proteins were added. This database is called NRPROT-SCOP.

2.7.2 PSI-BLAST

The sequence similarity search algorithm PSI-BLAST was benchmarked (Altschul *et al.*, 1997). An important parameter in the procedure is the e-value, which is the theoretically calculated number of errors per query, for details see section 1.3.4, in summary: PSI-BLAST first searches the sequence database using the gapped BLAST algorithm to collect obvious homologues defined as sequences with an e-value $<$ a chosen cut-off (h) and here $h = 0.0005$. These sequences are collected and aligned to generate a profile that is converted to a position specific scoring matrix (a PSSM). The PSSM is used in subsequent iterations to identify more remote sequences that are added to the PSSM if their e-value is below the cut-off h. PSI-BLAST is run for 20 iterations. Sequence hits are scored by their e-value. Low complexity regions that can introduce erroneous matches were removed from the query and NRPROT-SCOP database using SEG with default parameters (Wootton

[4] This database was provided by A. Stewart from the Computational Genome Analysis Laboratory from Cancer Research UK

& Federhen, 1996).

As noted by others (e.g. Park *et al.* (1998)), sometimes sequences can be erroneously added to the PSSM causing PSI-BLAST to drift from the original set of homologues. To check for this, the sequences included in the PSSM for an iteration were checked to ensure that they always included all the sequences found in the first search with gapped-BLAST. If the PSSM drifted away from including all the original set of sequences, then the PSI-BLAST run was restarted with an h values of 0.1 the previous value. This is repeated until the h value is 5×10^{-16} or no drift is detected.

Sequence hits from iterations other than the first could still drift out of the final profile and not be identified as homologues. Thus each iteration of the PSI-BLAST output was parsed. A sequence listed in an iteration was collected if it was not already found in a previous iteration or if the e-value of that hit was below the e-value of the previous collected one (in this case the new alignment replaced the old one). All hits with their individual position of the alignment, percent sequence identity, e-value, first and last residue of the alignment together with the full length query were stored in a file as a stacked multiple sequence alignment sorted from lowest (best) to highest e-value.

2.7.3 Identification of regions and domains in the query sequence

The percent overlap between two hits in the stacked multiple sequence alignment is defined as the length of the overlap in residues as a percentage of the shorter sequence. Two homologous sequences are defined as overlapping if their percent overlap is at least 50%.

The first step in the identification procedure is a clustering of sequence hits (figure 2.6). The hit of lowest (i.e. best) e-value is progressively compared to hits of higher e-values and the two hits are clustered if they overlap. A hit can only join an existing cluster if it overlaps with every member of the existing cluster. This is then repeated for the hit of the second lowest e-value against all the remaining hits and subsequently for the remaining hits of lower e-value. Next, all hits that cannot be clustered are considered as a cluster with one member. Finally regions are assigned to the query sequence using only the member of lowest e-value of each cluster. The structural classification of this hit is assigned to the appropriate region in the query.

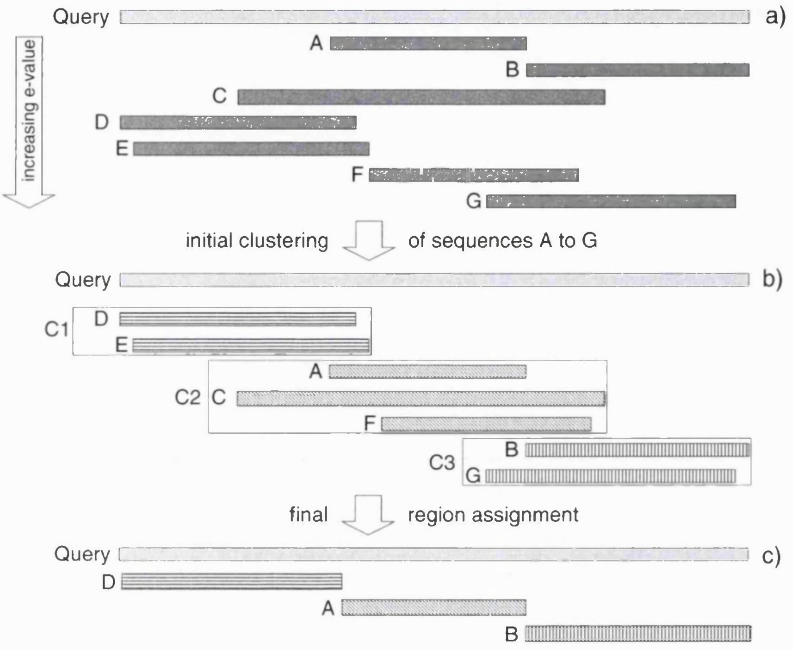


Figure 2.6: Annotating the SCOP genome on the domain level. The flow chart shows the methods to identify domains in a query sequence. (a) Sequences are schematically represented as bars. Homologues of the query sequence found by PSI-BLAST (A to G) are represented as a stacked multiple sequence alignment sorted by increasing e-value. (b) The target sequences are clustered (see text). Sequences of the same cluster are indicated by a common pattern. Three clusters (C1 to C3) have been generated. (c) Finally the target of lowest (best) e-value of each cluster is taken for the domain assignment (annotation) of the query. These best targets are truncated at the N- and C-terminus so that domain boundaries do not overlap.

2.7.4 Benchmark of remote homologues

The aim is to consider each sequence in the SCOP genome in turn and to evaluate the success of finding a remote homologue of known structure using PSI-BLAST. Therefore it is necessary to define when remote homology begins in terms of difficulty in being recognised by PSI-BLAST.

Rost (1999), extending previous work by Sander & Schneider (1991), derived an equation relating both sequence identity and alignment length to distinguish between true homologues and false positives for low levels of sequence identity (see figure 2.7). Very short alignments require a much higher percentage identity to be confident that they truly represent homologous relationships. The identity falls off exponentially and for alignment lengths of more than 350 residues, there is roughly a fixed identity cut-off. The actual equation is taken from the Web site http://-www.embl-heidelberg.de/˜rost/ and is:

$$p_{cut} = 510 * L^{(-0.32*(1.0+\exp(-L/1000)))} \tag{2.1}$$

where p_{cut} is the required percent identity for an alignment and L is the length of the alignment. This corresponds to defining alignments of over 350 residues as remote homologues if they have less than 20% identity and for simplicity we refer to this as the 20% identity cut-off.

The validity of using this cut-off is shown in figure 2.7. From an independent study the following data has been derived[5]: First, each single domain protein in SCOP1625, all homologous pairs (i.e. the same superfamily) of less than 40% identity, were structurally superimposed using the method of Orengo *et al.* (1992). From these structural superposition, the number of residues equivalenced and the percent identity were taken. The capacity for PSI-BLAST to recognise each pair was evaluated using an acceptance e-value of 0.0001 and up to 20 iteration but without saving intermediate matches that drift out of the profile.

Figure 2.7(b) shows that above 20% identity given by the cut-off from equation 2.1 there are only 11 homologous pairs that could not be identified by PSI-BLAST in a one-to-one evaluation. These 11 pairs correspond to 4% of all the possible pairs above the 20% sequence identity. The one-to-many success rate for PSI-BLAST above this cut-off can only be better than this level of success.

In the evaluation of the assignment accuracy for a particular SCOP sequence, that sequence was searched against all the SCOP entries in NRPROT-SCOP using gapped BLAST (Altschul *et al.*, 1997) (not PSI-BLAST). Matches with a percent identity ($\geq p_{cut}$ were excluded as they are close homologues of the SCOP protein.

[5] This data was provided by R.M. MacCallum

2.7.5 Genome data

The genome of Mycoplasma genitalium (isolate G37) has 479 ORFs (Fraser *et al.*, 1995) and was downloaded from The Institute For Genome Research (TIGR, http://-www.tigr.org/). The list of translated ORFs of the Mycobacterium tuberculosis genome (strain H37Rv) was down loaded from The Sanger Centre (http://www.-sanger.ac.uk/Projects/M_tuberculosis). The genome contains 3924 ORFs (Cole *et al.*, 1998).

2.8 Remarks about recent PSI-BLAST enhancements

The benchmark described in this chapter was carried out in 1998/99, and since then the PSI-BLAST method has been enhanced (Schaffer *et al.*, 2001) based on evaluations from different research groups including the benchmark described in this work.

The PSI-BLAST version used in this work belongs to the 2.0 series that uses a pre-calculated λ for the initial substitution matrix (here BLOSUM62 was used) and for the position specific search (see sections 1.3.4 and 1.3.5 for details). The bit score and the therefore the e-value is dependent on the scoring system (and in particular λ) that is used. The PSI-BLAST 2.1 series (Schaffer *et al.*, 2001) contains several enhancements such as a position specific scoring system that generally produces higher e-values, representing a better estimation of the real (observed) error rate (also see section 1.3.5). In addition the new scoring scheme reduces the 'drift' effect that may be induced by corruption of the PSSM as described in section 2.7.2.

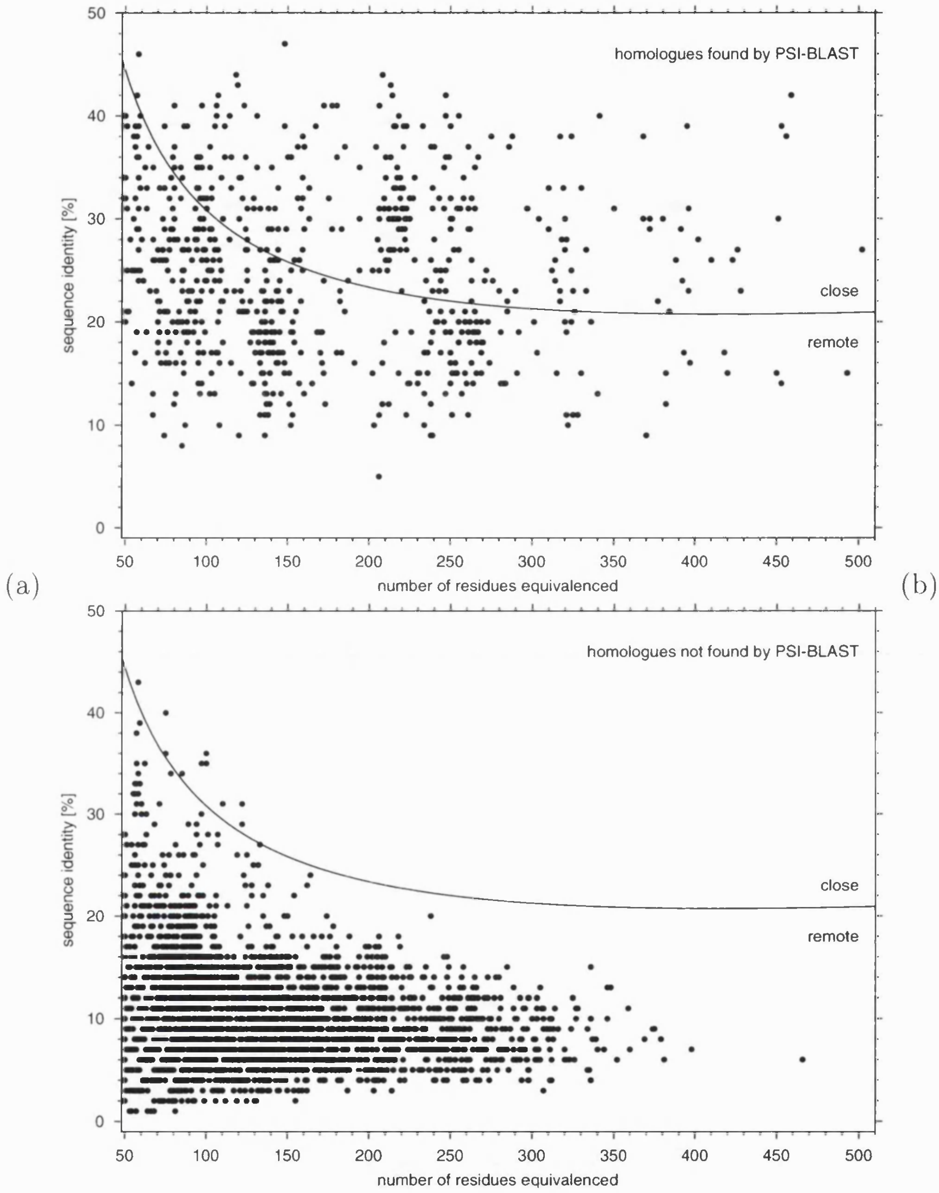


Figure 2.7: Identification of homologues by PSI-BLAST. Equation 2.1 is plotted as a function of structural equivalenced residues. All pairs of the same superfamily of the SCOP1625 database were structurally superimposed (see text) to identify structurally equivalenced regions (this is used as the sequence alignment length) and the percent sequence identity for each pair. Homologous pairs that can also be identified with PSI-BLAST are plotted as points in (a), pairs that cannot be identified in (b). Pairs on and above the curve are defined as close homologues and those below as remote homologues. There are only very few close homologues which cannot be identified by PSI-BLAST. The SCOP1625 database includes only pairs of proteins of <40% identity calculated by sequence (not structural) alignment.

Chapter 3

3D-GENOMICS: A proteome annotation pipeline

3.1 Summary

An automated proteome annotation system has been developed. The back-end is a relational database for data storage such as protein sequences and results from different protein based analyses. The database is interfaced by an object-oriented software API (Application Programming Interface) that allows for easy access for the analysis of the stored data. The API is used to run different analyses such as PSI-BLAST based sequence comparisons and to store the results as objects within the database. Several versions of an analysis can be managed. The analysis of a set of sequences can be automatically distributed over several computers. Several levels of inheritance within the database scheme and the API allow for straightforward integration of new analysis tools. This chapter explains the principles on which the database and the API are based.

3.2 Introduction

This chapter describes the database and software system that has been developed to perform the analysis described in chapter 4 and has also been used for other projects within the Biomolecular Modelling Laboratory at Cancer Research UK and the Structural Bioinformatics Group at Imperial College. The system is referred to as 3D-GENOMICS.

The objectives of the 3D-GENOMICS project are:

- To provide an abstract back-end research platform that can be employed in different projects related to the comparative analysis of genomes. On top of this platform software can be developed to perform specific tasks.
- To develop the software that is necessary for the comparative analysis of protein sequences described in chapter 4.
- To provide a back-end for a web based proteome annotation and information system that can be updated on a regular basis.

The last point of the objectives is not fully implemented for reasons discussed at the end of this chapter. However, there is a web-interface to 3D-GENOMICS accessible at http://www.sbg.bio.ac.ic.uk.

The initial objective was to develop a platform for large scale, mainly structure based bioinformatics projects including large scale homology modelling, which is the main justification for the name 3D-GENOMICS.

Following the analysis of PSI-BLAST in genome annotation and the application to the genomes of *M. genitalium* and *M. tuberculosis* described in chapter 2, 3D-GENOMICS has been developed as a re-usable and automated system for comparative analysis of genomes (the proteins of fully sequenced genomes in particular). This chapter therefore describes the general architecture of 3D-GENOMICS. Chapter 4 is an application of this system, and contains its own methods section describing parameters and other specificities of the analysis.

3D-GENOMICS contains pre-calculated results from different analyses, such as sequence comparisons, for a range of proteomes. The overall architecture of 3D-GENOMICS is a relational database, to store data such as protein sequences, domains and alignments. An object-oriented application programming interface (API) written in object-oriented Perl encapsulates this database layer. Once the analysis pipeline has been completed, access to the pool of data can be performed on demand via the API without having to perform any of the often time-consuming analysis, and with a minimum of code development. Several versions of the same type of analysis (e.g. using different parameters) can be stored. Changes to the database scheme are encapsulated by the API, so that front-end scripts do not have to be modified every time the database scheme is changed. On top of the API, scripts for automated data analysis and visualisation of results have been developed, including web based applications.

This chapter does not include a complete description of the database scheme, nor does it provide a manual or a tutorial for the API and the applications developed during this work. This chapter gives an overview of the principles that have been used to handle the objectives described above.

3.3 Resources

As pointed out above, results are pre-calculated. A set of standard sequence analysis software packages is run for a set of protein sequences. The software that is currently integrated in 3D-GENOMICS, and therefore part of the sequence processing pipeline, is listed in table 3.1. The integrated source databases are listed in table 3.2.

3.4 Architecture of the 3D-GENOMICS system

This section describes the architecture of the relational database and briefly describes the front-end API that was developed to process and retrieve data from the 3D-GENOMICS system[1]. Although the API is meant to be a stable interface to the database, independent of changes to the database scheme, in the current version of 3D-GENOMICS there is a close link between the database and the API.

3.4.1 The core scheme of the relational database

Figure 3.1 shows an entity relationship diagram (Chen, 1976; Connolly *et al.*, 1998) of the 3D-GENOMICS relational database. An *entity* is physically implemented as

[1] R.M. MacCallum contributed to the development of the core database scheme and the core API

Program	VN	Description	Reference	URL
BLAST	2.0.14	protein sequence homology search	Altschul *et al.* (1997)	`http://www.ncbi.nlm.nih.gov/-BLAST/`
PSI-BLAST	2.0.14	homology search for remote homologues via profiles	Altschul *et al.* (1997)	`same as for BLAST`
IMPALA	2.0.14	homology search for remote homologues using PSI-BLAST profiles	Schaffer *et al.* (1999)	`ftp://ftp.ncbi.nih.gov/blast`
3D-PSSM	-	Search for remote homologues of known structure	Kelley *et al.* (2000)	`http://www.sbg.bio.ic.ac.uk/-3dpssm/`
HMMer	2.1.1	HMM based homology search (hmmpfam)	Eddy (1998)	`http://hmmer.wustl.edu/`
Coils	2.2	prediction of coiled-coils in protein sequences	Lupas *et al.* (1991)	`ftp://ftp.ebi.ac.uk/-pub/software/unix/coils-2.2/`
TMHMM	2.0	HMM based prediction of transmembrane helices	Sonnhammer *et al.* (1998)	`http://www.cbs.dtu.dk/-services/TMHMM/`
HMMTOP	1.0	HMM- and neural network based prediction of transmembrane helices	Tusnady & Simon (2001)	`http://www.enzim.hu/hmmtop/`
SignalP	1.2	neural network based prediction of signal peptides	Nielsen *et al.* (1997)	`http://www.cbs.dtu.dk/-services/SignalP-2.0/`
SEG	-	detection of regions of biased amino acid composition (low complexity)	Wootton & Federhen (1996)	`ftp://ftp.ncbi.nih.gov/pub/seg/`
PSI-Pred	1.01	secondary structure prediction using neural networks and profiles	McGuffin *et al.* (2000)	`http://bioinf.cs.ucl.ac.uk/-psipred/`
Prospero	1.3	finding repeats in protein sequences	Mott (2000)	`http://www.well.ox.ac.uk/-rmott/ARIADNE/`

Table 3.1: External programs integrated in the 3D-Genomics processing pipeline. *VN* denotes the version number if available.

Database	VN	Description	Reference	URL
NRROT	20/01/01	non-redundant protein sequence database from the NCBI (translated GenBank, PDB, PIR, SwissProt)	Benson *et al.* (2002)	`ftp://ftp.ncbi.nih.gov/-blast/db/nr.Z`
genomes	20/01/01	protein sequences from completed genome projects (from NCBI GenBank)	Benson *et al.* (2002)	`ftp://ftp.ncbi.nih.gov/genomes/`
ENSEMBL	0.8.0	protein sequences and other data from the human genome	Hubbard *et al.* (2002)	`http://www.ensembl.org`

SCOP	1.53	Structural Classification of Proteins (structural protein domains)	Conte *et al.* (2002)	`http://scop.mrc-lmb.cam.ac.uk/-scop/ &`
				`http://astral.stanford.edu/`

ASTRAL	1.53	supplement to SCOP (such as sequences)	Chandonia *et al.* (2002))	`http://astral.stanford.edu/`
PFAM	6.2	HMMs and annotation for protein domain families	Bateman *et al.* (2002)	`http://www.sanger.ac.uk/-Software/Pfam/`
Prosite	16	patterns and annotation for protein sequence motifs	Falquet *et al.* (2002)	`http://www.expasy.org/prosite`
taxonomy	15/01/02	taxonomic database (taxonomic trees) from the NCBI	-	`ftp://ftp.ncbi.nih.gov/-pub/taxonomy/`
OMIM	15/01/02	hereditary human disease genes (from the NCBI)	Antonarakis & McKusick (2000)	`http://www.ncbi.nlm.nih.gov/omim/`

Table 3.2: External databases integrated in 3D-GENOMICS. If no version number (*VN*) is available the date of the integration into 3D-GENOMICS (day/month/year) is given.

a database table, and usually has a *primary key* that is unique for the entity, i.e. it identifies a particular entity. A *weak entity* depends on another *(strong)* entity, and usually does not have its own primary key, but uses the primary key of the strong entity it depends on (the *weak key* within the weak entity). The diagram is simplified, showing only the most important tables, attributes and keys of the entities, and most of the entity inheritance (superclass-subclass relations) is not shown. The diagram only demonstrates the principles on which the 3D-GENOMICS database scheme is built. The paragraphs below describe each of the entities and their relations.

The green part of the diagram represents part of the database scheme related to protein sequences. A *Pseq* entity represents a protein sequence that has the *Seq* attribute, which is the amino acid sequence string and the primary key *PseqId.* One protein sequence can have several descriptions, so that the same sequence may be present in several sequence databases (having different accession numbers). A sequence may have slightly different descriptions in different source databases such as 'protein kinase (type A)' and 'protein kinase A'. Furthermore, the description has a relation to the taxonomy database provided by the NCBI via the *TaxId.* If a protein sequence has several descriptions, these may be from different organisms (i.e. different organisms with exactly the same sequence). A protein description cannot exist without a protein sequence, and therefore the *Pdesc* entity is weak, although for technical reasons it has its own primary key (*PdescId*). Each protein description may have a list of associated keywords (*Tag* entities). Several descriptions may share a set of keywords. This relation is implemented via the helper table *PdescTag.* A *Tag* has a *Name* (the keyword), and a *Type* which is either *user* (the tag has been inserted manually to label a protein description or a set of descriptions), *static* (usually tags automatically set by scripts that insert sequences into the 3D-GENOMICS database) or *db* (an abbreviated name of a source database). A description entity may have *Tags* of the same name but different type. Associating descriptions with *Tags* allows the selection of a sets of sequences with a common label. All sequences from the ensembl version 0.8.0 dataset of human proteins may have the tags *human* (type *user*), *ensembl* (type *db*) and *v0.8.0* (type *user*). *Pseq* and *Pdesc* entities also contain attributes keeping track of the date of data integration and modification.

The blue part of the diagram shows entities that store information about the integrated analysis programs that have been run. The central entity is the *Run*, which keeps basic information about an analysis. This includes an error string returned by the analysis software. The *Run* entity is abstract, i.e. it is a superclass from which other entities such as *BlastRun* (not shown in figure 3.1) inherit. Therefore the name of the subclass to which this run belongs (*BlastRun*) has to be stored, so that an instance of the correct entity (or object on the API level) can be recreated from the stored data. The *Params* entity stores an optional set of parameters that was used to run the analysis (e.g. an e-value cut-off and the name of the sequence database for a *BlastRun* object). A run can have several parameters, and the same set of parameters can be used by different runs. *Params* entities with the same *ParamsId* define a set of parameters that belong together.

A *Run* is the superclass (the same as a baseclass) of specialised run entities such as a *GenomeRun* shown in figure 3.4 that treats a genome or proteome as a whole or a *PseqRun* that represents an analysis that was performed on a protein sequence or a protein sequence fragment (given by the start and stop attributes). A sequence may be subject to many *PseqRuns*. The *PseqRun* entity itself is the superclass of more specialised sequence based analyses such as *BlastRun*.

The red part of the diagram shows the results of *PseqRuns*. These are *Features*, that describe a region of the protein sequence (given by the *Start/Stop* attributes) of the corresponding run (referenced by the *RunId*). A *Feature* is a weak entity, because it cannot exist without a *Run*, although this entity has its own primary key for technical reasons. A *Feature* may also be produced by other instances inheriting from *Run* which are not *PseqRuns*, e.g. a gene feature representing the location of a gene on a chromosome. However, in the current version of 3D-GENOMICS only *PseqRun* based features are implemented. Specialised entities such as an *Alignment* inherit from *Feature* to extend its list of attributes (and methods on the API level). Like the *Run* entity, the *Feature* entity is abstract, and the class/entity name of the feature has to be stored in the database to reconstruct an API-object of the correct class.

The special *PerlObject* entity is explained later together with figure 3.9.

The complete 3D-GENOMICS database currently contains 65 tables of which 42 tables are of relevance to this work. Of these tables 18 may be counted as core tables, 21 as subclasses that totally participate in a superclass, and 3 tables for the OMIM

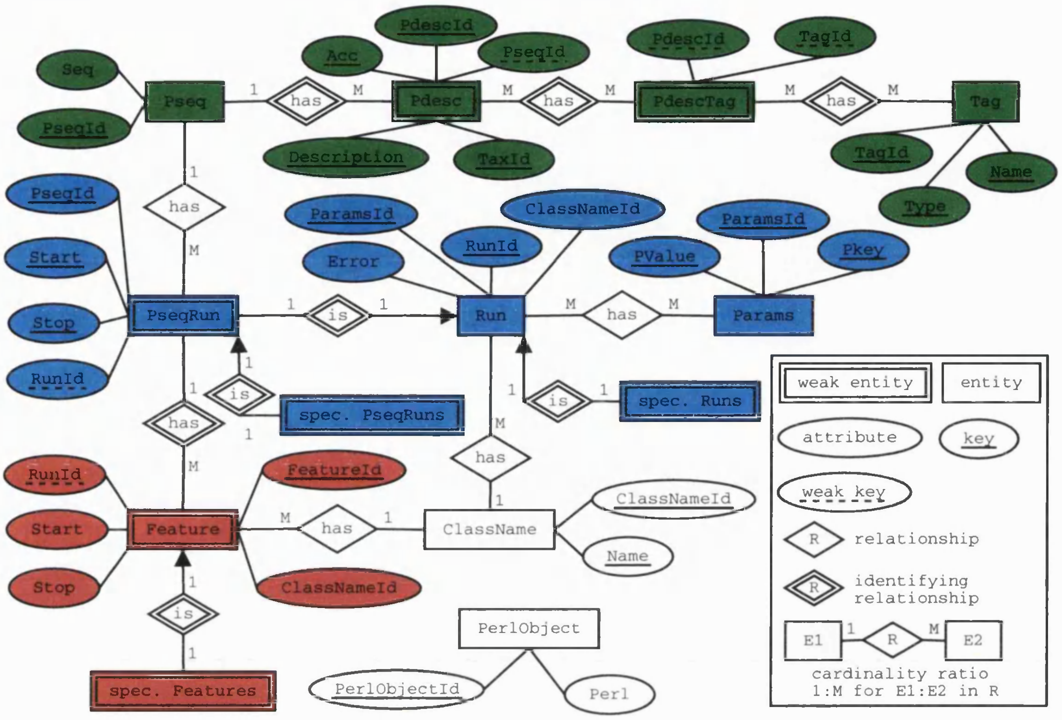


Figure 3.1: Simplified entity relationship diagram of the 3D-GENOMICS database. Protein sequence related entities are coloured in green, *Run* related entities (representing entities coupled with analyses software) are coloured in blue and *Features* (results from an analysis) are coloured in red. 'Helper' entities and relations are shown in white. The legend inside the figure explains the meaning of the symbols, see text for details.

disease database (part of the 3D-GENOMICS database). In addition the taxonomy database is implemented in its own database which can be obtained from the NCBI (see table 3.2) and imported into a relational database system. The SCOP database is provided in flat files via the URL given in table 3.2 and is converted into a simple relational database that is linked to 3D-GENOMICS via accession numbers (in the *Pdesc* table) and tags (see table A.2 in the appendix for the table definitions that have been chosen to represent SCOP). In addition a *scratch* database is required to write temporary tables for the web-service and for some analysis scripts. Table A.1 in the appendix explains the important tables and their attributes.

3.4.2 Inheritance is a major aspect of the database architecture

As mentioned above and indicated in figure 3.1, the *Run* and the *Feature* entities are superclasses for several specialised entities (*subclasses*). Figure 3.2 schematically shows the inheritance as a flow-chart. In the current version of 3D-GENOMICS, all *Feature* 'producing' objects (indicated by lines without arrow head) are *PseqRun* objects.

The *PsiBlastRun* and *PsiBlastHit* subclasses have the deepest inheritance in the 3D-GENOMICS system. A *PsiBlastHit* is a *BlastHit* and adds the *iteration* attribute (in which this hit was found) to the *BlastHit.* The PSSM of the last iteration is an attribute specific to a *PsiBlastRun,* but it is a *BlastRun.* A *BlastHit* is a special type of *Alignment,* it has a score and an e-value. The *Alignment* stores information that is required to reconstruct the complete sequence alignment. It contains a reference to the subject sequence of the alignment, the start and stop of the alignment within the subject, the percent sequence identity and insertions and deletions within the query and the subject sequences. The last level of inheritance is the *Feature* (the superclass on the database level), that has a start and a stop attribute that is used to describe the location of the feature within the sequence that was subject to the analysis. The *Feature* references a *PseqRun* entity, from which the protein sequence for which the analysis was run can be obtained.

The *PSSM3dHit* indicates that there are *Feature* types that do not have a specialised entity that inherits from *PseqRun* (there is a direct connection between *PSSM3dHit* and *PseqRun* in figure 3.2). However, on the 3D-GENOMICS API level there is always a corresponding specialised *Run* class (for example the *PSSM3dRun* class) that at least provides a method to perform the analysis. On the database level there is only a specialised entity if information has to be made persistent, for example a *PsiBlastRun* has its own entity because the last PSSM of the PSI-BLAST run has to be stored.

The *CoilRun/Coil* entities are given as examples of other *Features* that are not *Alignments.* In the current version of 3D-GENOMICS there are eight such entities (and classes on the API level, see tables A.1 and A.3 of the appendix).

Inheritance is implemented by referencing the different tables that represent the different levels of specialisation by the same primary key, which is the *FeatureId* for *Features* and the *RunId* for *Runs*. There is total participation between the *PseqRun* entity and the *Feature* entity, i.e. all *Features* have a *PseqRun* they come from. The *Run* entity is also a superclass of other specialised entities that are not *PseqRuns*. The *GenomeRun* is a superclass for analysis that treat a proteome as a whole, i.e. that do not consider individual protein sequences.

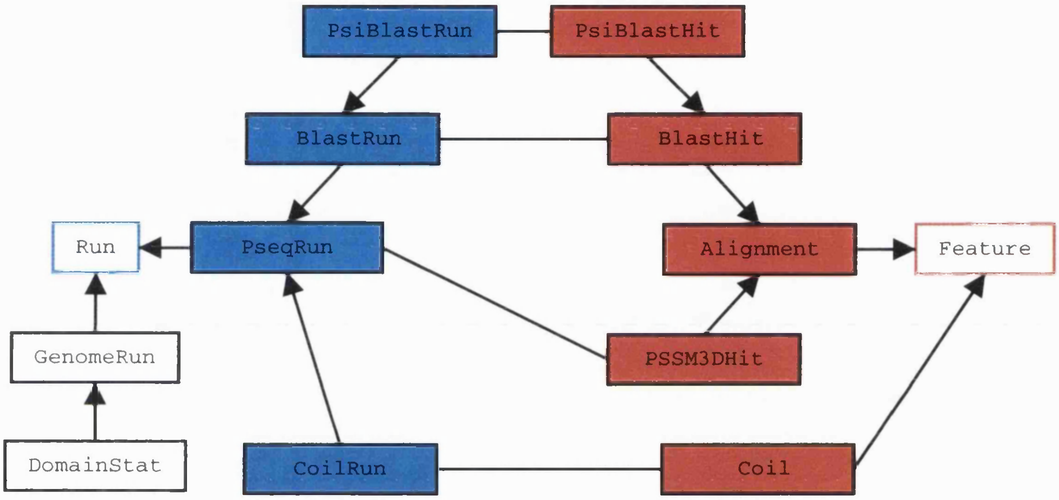


Figure 3.2: Flow-chart of inheritance in the 3D-GENOMICS database. Entities inheriting from the *Run* superclass are shown with blue background, and those entities inheriting from the *Feature* superclass are with red background. The basic superclasses have blue and red outlined boxes. Inheritance is shown as arrows, where the arrowhead points to the entity the other entity inherits from (subclass → superclass), lines without arrowheads indicate that the run produces a particular kind of *Feature*. The same level of indentation of entities of the same colour (red and blue) shows the same level within the inheritance tree, e.g. *Alignment* and *Coil* directly inherit from *Feature*. The *GenomeRun* subclass is a special *Run* class that does not produce *Feature* objects (it manages analyses that treat a proteome as whole), *DomainStat* is a specialisation of *GenomeRun* that is specifically designed for web purposes.

3.5 Post-processing and summary of primary results

For fast data retrieval from the database the results from the different types of analysis are summarised by reducing the complexity of the database queries and the amount of data that has to be retrieved. The three steps of data summary implemented in the current version of 3D-GENOMICS are:

1. Clustering aligned regions from BLAST, PSI-BLAST or IMPALA runs within a query sequence, so that a protein sequence can be described with a small set of regions rather than a huge number of alignments which often do not contribute much additional information.
2. Summarising region clusters and other features such as transmembrane domains to produce a genome wide annotation overview.
3. The above steps are used to generate specialised data warehouses for fast and simple data access required for e.g. web based applications.

The sections below describe the summary steps as a processing pipeline. The underlying database scheme that implements the data summary is explained together with examples.

Different levels of analysis reduce the complexity of data

Figure 3.3 shows the flow of data and results within the 3D-GENOMICS processing pipeline starting after the basic analysis has been run. The results of these analyses are symbolised inside the triangle as 'Atomic Features' (in red). These basic analyses include BLAST and PSI-BLAST runs, assignments to PFAM, prediction of transmembrane helices, signal peptides etc ... (see table 3.1 and 3.2 for a list of integrated resources). The amount of stored basic (atomic) data is huge, e.g. for the human protein dataset (29,000 protein sequences) more than 17,000,000 PSI-BLAST alignment objects are stored.

The red rectangles of the left part of the figure show the atomic features. These are stored per analysis and per sequence. There are several homologues sequences per query, symbolised by the thin coloured lines. These homologues can be clustered according to their position within the query sequence (thick black line) and their sequence type, symbolised by a common colour of the thin lines (e.g. sequences of known structure, homologues from the SwissProt database, etc ...). This produces different region types per sequence. The clustering is explained in section 3.5. This step reduces the number of alignments to less than 87,000 overlapping regions for the human proteome without reducing the annotation quality markedly.

The region information together with some of the basic non-alignment features, such as transmembrane helices, are then summarised as genome-wide statistics describing the extent of the different types of annotation (blue part of the triangle and blue boxes to the left). It contains the annotation extent as the number of sequences with a particular type of annotation (e.g. the number of sequences with at least one homologue of known structure, or the number of membrane proteins), the number and types of annotated regions within a proteome (e.g. the number of SCOP domains or regions with functional annotation, the number of transmembrane domains, etc ...) and the number of amino acid residues that are covered by an annotation type. These annotation categories can be easily accessed, and individual sequences or regions for a category can be retrieved. There are 4,200 of these annotation summaries for the proteome wide summary for human.

For comparative analysis one can compare genome summaries between different genomes. Usually this is straightforward and fast using the 3D-GENOMICS API. However, for more specific comparative analyses such as the different frequencies of SCOP superfamilies in globular parts of transmembrane proteins in different proteomes (as discussed in section 4.4.7), an additional summary step that uses information from all three of the above analysis levels is generated. This last summary step was developed in a relatively short period after most of the 3D-GENOMICS system was already in use for ongoing research. The interest in a particular research project, the comparison of SCOP domains in different contexts, required this additional step to make some of the 3D-GENOMICS data even more easily accessible. This shows that the 3D-GENOMICS system is rather abstract and may not always allow direct solutions, but also demonstrates that on top of this general and abstract core, specialised objects and applications can be developed with relatively little effort. This specialised data summary further reduces the amount of data from the genome wide summary described above (4,200 annotation descriptions) to 546 SCOP domain descriptions for the human proteome.

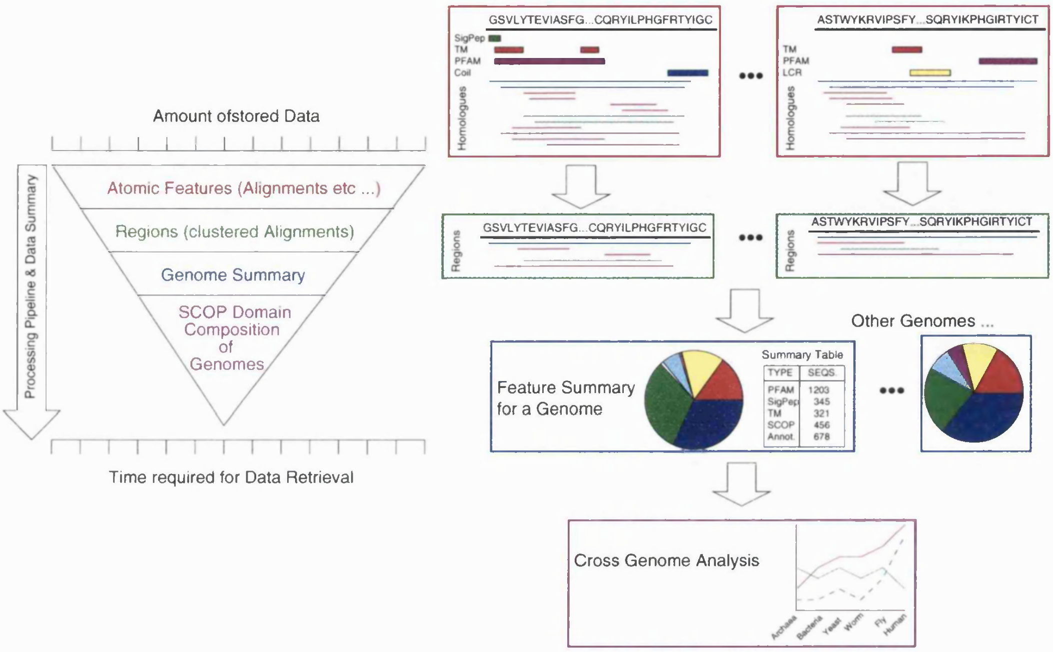

Amount ofstored Data
Processing Pipeline & Data Summary
Atomic Features (Alignments etc ...)
Regions (clustered Alignments)
Genome Summary
SCOP Domain Composition of Genomes
Time required for Data Retrieval
GSVLYTEVIASFG...CQRYILPHGFRTYIGC
SigPep
TM
PFAM
Coil
Homologues
ASTWYKRVIPSFY...SQRYIKPHGIRTYICT
TM
PFAM
LCR
Homologues
Regions
Regions
Other Genomes ...
Feature Summary for a Genome
Summary Table
TYPE SEQS
PFAM 1203
SigPep 345
TM 321
SCOP 456
Annot. 678
Cross Genome Analysis
Archaea
Bacteria
Yeast
Worm
Fly
Human

Figure 3.3: Steps to summarise data and intermediate results. Steps in a particular colour in the triangle represent the summary steps and are detailed in the left part of the figure with steps framed in the same colour as in the triangle. See text for details.

Supplementary entities and relations for the data summary

The summary of alignments into clusters that describe the same region within a query sequence that was introduced above, is performed in a similar way as the clustering of SCOP domains described in the methods section of chapter 2. There are currently four alignment based region types that are relevant to this work (these are used in chapter 4). Regions of the same type do not overlap, and ends are adjusted in the same way as described in section 2.7.3. Different region types may overlap, and an alignment may participate in different region types. The four region types are explained below.

1. **SCOP regions**. Clusters of alignments with sequence subjects corresponding to SCOP domains.

2. **PDB regions.** Clusters of alignments with sequences subjects of known structures (PDB chains). These chains may contain more than one domain.

3. **Annotated regions.** Alignments with sequence subjects from any of the source databases SCOP, PDB, PIR or SwissProt, and with a textual description of the biochemical or biological function. Entries with descriptions containing the substrings 'hypothetical', 'probable', 'putative' or 'predicted' are excluded.

4. **Homology regions.** These regions contain any homologous sequences including conserved hypothetical sequences without any useful functional description. This implies that every member of an annotated region is automatically a member of a homology region.

In general the biological information content of these regions decreases starting with SCOP domains providing most information with structural and often functional information available on the domain level, followed by PDB regions with similar biologically useful information but without distinguishing between domains, and, with least information, the homology region that, in the absence of an annotated sequence, just highlights the conservation of this region without providing direct insight into any biochemical function.

Non-domain regions (all but the SCOP regions), are generated using a greedy version of the clustering described in the methods of chapter 2. A new member can join an existing cluster if it overlaps with at least one member of that cluster by at least one residue. This produces single linkage clusters. If alignment A overlaps with alignment B, and A does not overlap with C but B overlaps with C, then A, B and C are put into the same region. Before clustering, alignments are sorted decreasingly by start position within the query to speed up the clustering. Once a cluster is complete, its members are sorted by increasing e-value with the alignment of best e-value taken as the representative for this region. In many cases the longest sequence of a non-domain cluster defines the expansion of the region over the query sequence, and also may often be the closest homologue of the query sequence. The methods section of chapter 4 describes the actual constraints that were used to define regions.

For SCOP domains the clustered alignments roughly correspond to domains (except for discontinuous domains, i.e. in which a structural domain is formed from two or more discontinuous sequence segments). The other region types must not be thought of as domains, but instead as summaries of alignments that may be used as a general description of the query protein or a part of the query protein. The benefit is to speed up the analysis and comparisons of complete proteomes as discussed above in section 3.5.

Figure 3.4 shows how regions are stored in the database, and how regions and other features are used to generate a genome wide summary. For comparative analysis of genomes a possible starting point may be to compare the frequency and the fraction of sequences or residues within the proteome that can be assigned to a particular feature. The *GenomeRun* entity with its related entities provides the storage for this kind of analysis. Data retrieval is fast and straightforward (in terms of the code that has to be written for an application that uses the 3D-GENOMICS API).

The upper part of figure 3.4 shows that a *Region* inherits from a *Feature*, because a *Region* has a location within a sequence. A *Region* has a list of members (*RegionFeatures*), and because all *Regions* are currently built by clustering alignments, this list is in fact a list of *Alignments* (not shown in figure 3.4), which are in turn *Features. Regions* for a protein sequence are generated by a *SummaryRegionRun* object of the 3D-GENOMICS API, for which there is no corresponding entity in the database. The different *Region* types have specialised classes in the API (*ScopRegion, PdbRegion, ...*) which inherit from the *Region* baseclass. Currently no *Region* type specific information has to be stored that cannot be retrieved easily via the core scheme, so there are no corresponding entities in the database.

The lower part of figure 3.4 shows how the *Region* information is summarised via a *GenomeRun*, which inherits from *Run* and performs a genome wide analysis to summarise the available information (see also section 3.5). The genome or the list of genomes for which this summary is created is stored within the *Tags* attribute of the *GenomeRun* entity, which can be multi-valued (e.g. it is possible to store a genome summary for a set of genomes such as *E. coli* and *B. subtilis*). Global annotation counts or numbers for a *GenomeRun* are stored as *GSCounts* ('Genome Summary Counts'), with the frequency given by the *Number* attribute. The *Type* of the number describes whether the number refers to a protein sequence, a region or amino acid residues. The *Name* is a description of the number, e.g. 'total', 'Non-globular' or '003.003.001' for a SCOP superfamily accession number. For technical reasons a special primary key *GSCountId* has been put into the *GSCount* entity. For most region or sequence based *GSCount* entries the list of members can be accessed via the *MemberId* which is either a *FeatureId* if the member is a *Region* or a *PseqId* if it is a protein sequence. There is a many-to-one relationship between *GSMember* and *Region* or *Pseq* because different versions of a *GenomeRun* entity may reference the same *Region* or sequence. In addition, if the *MemberId* is a *PseqId* one sequence can be part of several *GSMember* types. For example a sequence can have structural annotation as well as functional annotation.

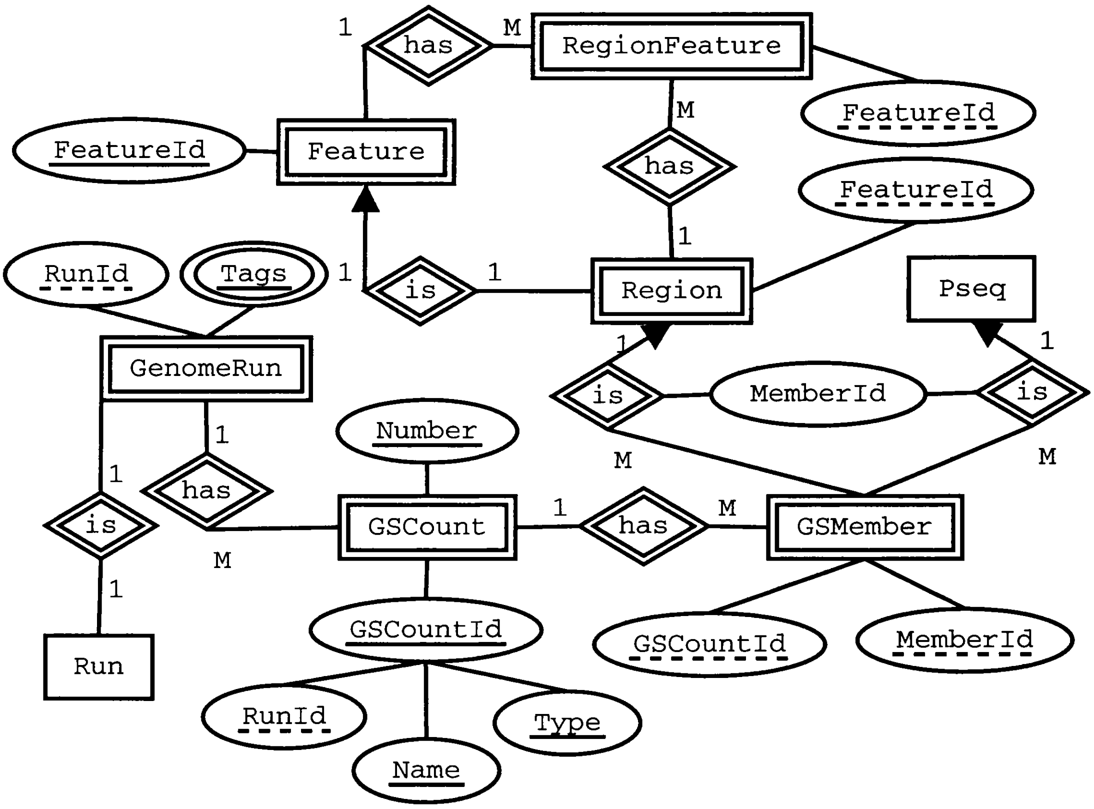


Figure 3.4: Entity relationship diagram of the data and result summary part of the 3D-GENOMICS scheme. This part of the scheme does not belong to the core scheme. The *MemberId* attribute above the *GSMember* that connects the relations from *GSMember* to *Region* and *Pseq* indicates that the *MemberId* can be a *FeatureId* or a *PseqId*. See figure 3.1 for an explanation of the symbols (the *Tags* attribute of the *GenomeRun* entity can store a list of values)

Usage and examples of the data summaries in 3D-GENOMICS

To demonstrate how to use the summary information represented in figure 3.4 a simple code example is given in figure 3.5. The *GenomeSummary* object gs (which inherits from *GenomeRun*) automatically connects to the database server when the `readCount` method is called. Once an object is connected to the database, this connection will be re-used for all subsequent database requests by this object. The parameter `'latest'` for the construction of the *GenomeSummary* object automatically generates the latest version of the analysis, alternatively a *Params* object can be provided to specify a particular version. As mentioned in the introduction and explained in section 3.6 on page 105, several versions of an analysis can be stored and retrieved. `gs->readCount('003.032.001', 'Regions')` returns the number of SCOP domains with the superfamily accession code '003.032.001' (P-loop), a `gs->readCount('003.032.001', 'Residues')` call would retrieve the number of residues that are in P-loop domains.

In the loop to calculate the average P-loop length, a *ScopRegion* object is generated using the *MemberId* from the array that was returned from the `gs->getMemberIds` call. The optional *Parent* attribute for the construction of the object will be used to borrow the database connection from the *gs* object, so that only one database connection is established for the whole script.

Figure 3.6 shows a screen-shot of the summary for the human proteome from the 3D-GENOMICS web-page (http://www.sbg.bio.ic.ac.uk/). The page is generated dynamically on request, so that the summary pages do not have to be updated manually after database updates (i.e. if a new *GenomeSummary* has been produced). All information is requested from the 3D-GENOMICS system in a similar way as shown in figure 3.5 using the API which accesses the underlying tables shown in figure 3.4. The links within the page (blue text) are generated via the *GSMember* entity and allow immediate access to the regions and sequences corresponding to the different annotation categories. From these lists individual sequences and sequence alignments can be accessed.

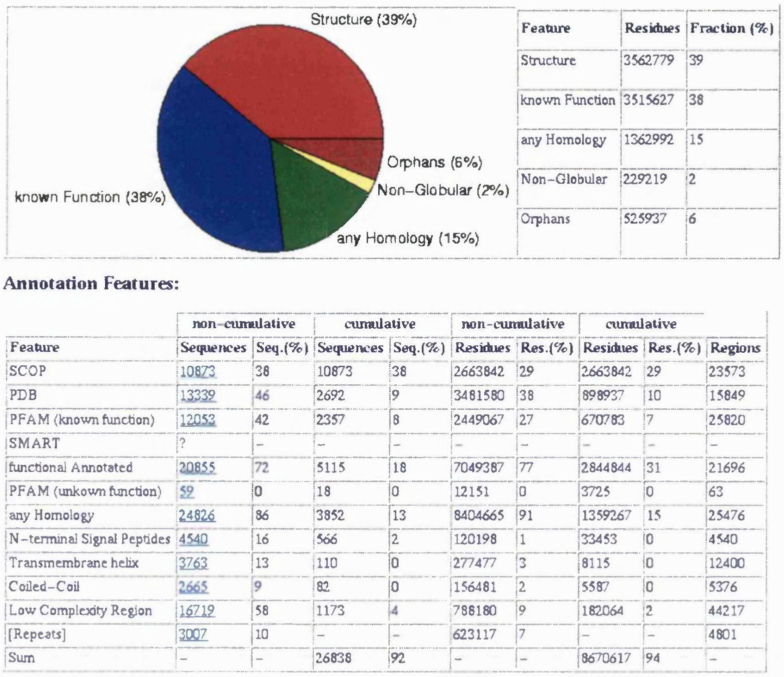
The different categories (rows) in the table in figure 3.6 correspond to different *Names* in the *GSCount* entity shown in figure 3.4, and the columns ('Sequences', 'Residues' and 'Regions') correspond to the *Type* attribute in *GSCount.* SMART domains have not been included in this analysis, and repeats have been excluded from the cumulative analysis. See the legend to figure 3.6 for an explanation of 'non-cumulative' and 'cumulative'.

```
#!/usr/bin/perl -w

use GenomeSummary; # the GenomeSummary class
use ScopRegion;    # The ScopRegion class

### get the most recent GenomeSummary object ($gs) that
### corresponds to the ’Ecoli’ sequence set
my $gs = new GenomeSummary(Tags => [’Ecoli’], Params => ’latest’);
printf "%d SCOP domains found in E. coli\n",
  $gs->readCount(’ScopRegion’, ’Regions’);

### get the IDs for all SCOP regions with superfamily accession
### 003.032.001 (P-loop)
my @memberids = $gs->getMemberIds(’003.032.001’, ’Regions’);
### calculate the average E. coli P-loop domain length
my $len = 0;
my $n = 0;
foreach my $id ( @memberids ) {
  my $region = new ScopRegion(FeatureId => $id, Parent => $gs);
  $len += $region->len();
  $n++;
}
$len /= $n;
print "average length of E. coli P-loop domains is $len ($n domains)\n";
```

Figure 3.5: Code example demonstrating the use of the 3D-GENOMICS summary information via the object-oriented Perl API.

3.6 Principles of the 3D-GENOMICS API

3D-GENOMICS stores data from the included source databases and the results from the different analyses as objects in a relational database by mapping the objects onto the relational scheme. This mapping includes the decomposition of each object into its attributes and relations that may be stored across different tables. An alignment for example contains a subject sequence (a homologue of the query) which is stored as a reference to an entry in the Pseq table. The database is at least in the 1st nor-



Feature	Residues	Fraction (%)
Structure	3562779	39
known Function	3515627	38
any Homology	1362992	15
Non-Globular	229219	2
Orphans	525937	6

Annotation Features:

	non-cumulative		cumulative		non-cumulative		cumulative		
Feature	Sequences	Seq.(%)	Sequences	Seq.(%)	Residues	Res.(%)	Residues	Res.(%)	Regions
SCOP	10873	38	10873	38	2663842	29	2663842	29	23573
PDB	13339	46	2692	9	3481580	38	898937	10	15849
PFAM (known function)	12053	42	2357	8	2449067	27	670783	7	25820
SMART	?	–	–	–	–	–	–	–	–
functional Annotated	20855	72	5115	18	7049387	77	2844844	31	21696
PFAM (unkown function)	59	0	18	0	12151	0	3725	0	63
any Homology	24826	86	3852	13	8404665	91	1359267	15	25476
N-terminal Signal Peptides	4540	16	566	2	120198	1	33453	0	4540
Transmembrane helix	3763	13	110	0	277477	3	8115	0	12400
Coiled-Coil	2665	9	82	0	156481	2	5587	0	5376
Low Complexity Region	16719	58	1173	4	788180	9	182064	2	44217
[Repeats]	3007	10	–	–	623117	7	–	–	4801
Sum	–	–	26838	92	–	–	8670617	94	–

Figure 3.6: Screen-shot from the 3D-GENOMICS web-page showing a part of the analysis summary for the human proteome. The pie-chart shows the extent of assignments in different annotation categories. The pie-chart is residue based, i.e. the fraction of the proteome in residues was calculated. The table below the pie-chart gives details of the generated annotation. 'non-cumulative' means that the actual number of sequence, region or residue assignments are calculated by allowing every sequence, region or residue to be counted more than once across the different categories (e.g. a residue of a protein sequence may be part of a SCOP and a PFAM domain). 'cumulative' means that sequences, regions or residues are counted only once across annotation categories with 'SCOP' having priority followed by 'PDB' etc. to avoid exceeding 100% (e.g. sequences assigned to a SCOP domain and a PDB chain are only counted for SCOP and not for PDB).

mal form, so that there is no obvious redundancy, and most relations of the database core are also in the 2nd and 3rd normal form (Connolly *et al.*, 1998). Although the API should be the interface to the database, for fast access it is possible to bypass the API and to access the contents (the stored objects with their relationships) directly via SQL.

The most central class of the 3D-GENOMICS API is the *Run* class with all its specialised subclasses. A run object can be executed locally or submitted to a computer farm as shown in figure 3.9 of section 3.7. It also contains a *Params* object which gives details about the parameters that are specific for the analysis. From a *PseqRun* object the list of features that are specific for this run and a particular protein sequence object can be retrieved. The usual way of getting sequence features is to get the available *PseqRun* objects from a protein sequence object, and then to request the list of features from each of these *PseqRun* objects.

The *Params* object for a *Run* object allows several different versions of a particular run to be created, e.g. one can have several *PsiBlastRun* objects for the same sequence that are distinguished by their *Params* object (these may for example define different e-values). Figure 3.7 shows a simple example of how to get the objects for a particular type of analysis, and from these objects the *Feature* objects.

```
my $pseq = new Pseq(PseqId => 123);
my $p = new Params(%BlastRun::default_params, blast_e => 0.1);
my @runs = $pseq->getRuns('BlastRun', $p);
foreach my $run ( @runs ) {
  my @hits = $run->getFeatures();
  # do something  with the hit objects ...
}
```

Figure 3.7: A simple example to demonstrate how to access sequence features. The protein sequence object with the *ID* 123 is retrieved from the database. A parameter object (**$p**) is generated that contains the default attributes for a *BlastRun* (this is a class attribute), but overrides the `blast_e` attribute (the e-value cut-off). All *BlastRun* objects for this sequence that were run with the requested parameter object are retrieved, and for each of these objects the feature objects (type *BlastHit*) are retrieved. Note that several *BlastRun* objects may be available because several fractions of the sequence may have been subject to the BLAST analysis.

The integration of new analysis software is straightforward, mainly due to the different levels of inheritance. The *hmmpfam* program of the HMMer software package (see table 3.1) to identify PFAM domains in protein sequence via hidden Markov models was integrated on demand after most of the API was already developed. The *HMMRun* class inherits from *PseqRun*. The output consists of *Features* of the special type *HMMHit*. The integration of *hmmpfam* was straightforward. Usually most development has to be spent on the *run* routine that performs the actual analysis, including the parsing of the program output. For the HMMRun the parser of the BioPerl project (http://www.bioperl.org) is used.

The API is implemented in the Perl language. Perl may not be the ideal language for bigger object-oriented software projects, it has for example no strict data typing, and many developers complain about unreadability of the code. However, Perl is a popular programming language within the biology and bioinformatics community, and is the consensus language of those people who showed interest in the project. An initial objective of the API was to provide some basic compatibility with the BioPerl project. 3D-GENOMICS uses some BioPerl modules, and can also convert a 3D-GENOMICS sequence object into a BioPerl equivalent, but at this time there is no extended and consistent compatibility between the two systems. However, it is possible to implement appropriate export routines on demand.

The 3D-GENOMICS API contains nearly 80 Perl modules with more than 17,000 lines of code defining classes and non-object-oriented code. In addition there are about 40 scripts for database maintenance and evaluation, containing more than 3,000 lines of Perl code. There are 40 CGI scripts for web based applications with more than 6,500 lines of code. In addition there are more than 1,500 lines of Python code included to manage and parse BLAST and PSI-BLAST runs. Table A.3 explains the different modules and classes with their methods and functions that are currently implemented in the API.

The base class from which most objects are built, is *DbConnection.* Objects that are generated via the annotation pipeline (*Run* or *Feature* objects) or objects from a source database (e.g. protein sequences) have to be stored persistently in the database for later retrieval and analysis. Therefore such an object is a *DbConnection* object, that is able to insert itself at the correct place within the database, update its attributes, retrieve its data and delete itself from the database.

To construct an object from the database the identifier is needed (see line 1 in figure 3.8 for an example). The constructed *TMH* object (transmembrane helix object) is empty, and can be filled with its attributes by either calling the *sync* routine (line 4) or by just calling the *get* routine (see lines 5, 6 and 17), that internally performs the complete read synchronisation with the database and returns the requested attributes, which stay within the object, so that subsequent *get* calls do not need to query the database. A new object can be generated by providing all required attributes but no unique identifier as shown in line 8. The new object writes itself to the database with the next *sync* call (line 9). The *set* method (line

11) sets attributes which must not already exist in the object (an empty object was constructed in line 10), the next *sync* call writes the filled object to the database (lines 12). Note, that an object usually has a defined set of attributes that have to be set. An existing attribute can be modified via a *modify* call, shown in lines 13 to 14 (only a few classes allow attribute modification). If the *sync* method is not called before the object is destroyed, all changes, including a complete newly created object will be lost.

Lines 2 and 3 shows the usage of the *clone* method. If the *FeatureId* is known but the special class of the feature is unknown, *clone* will produce a read-synchronised copy of the object of the correct type.

Lines 15 to 18 show how an object (line 18) can be constructed that uses another object as a *Parent*. The *Parent* provides the database connection, so that two objects can share the same connection. This avoids overhead of frequent connect and disconnect requests to and from the database server. This technique is also used in the example in figure 3.5.

```
[ 1] $f = new TMH(FeatureId => 1001)
[ 2] $f = new Feature(FeatureId => 1002)
[ 3] $f = $f->clone()
[ 4] $f->sync()
[ 5] ($begin, $end) = $f->get('Start', 'Stop')
[ 6] $tmrun = $f->get('Run')
[ 7] $params = $tmrun->get('Params')
[ 8] $f = new TMH(Start => 5, Stop => 24, Ori => 'out', Run => $run)
[ 9] $f->sync()
[10] $f = new TMH()
[11] $f->set(Start => 5, Stop => 24, Ori => 'out', Run => $run);
[12] $f->sync()
[13] $f->modify(Start => 8)
[14] $f->sync()
[15] $f = new TMH(FeatureId => 1003)
[16] $f->dbConnect();
[17] print $f->get('Ori')
[18] $f2 = new TMH(FeatureId => 2, Parent => $f)
```

Figure 3.8: Code examples to demonstrate the connectivity with the database. Note, this is not a program, but just a collection of examples to show how objects can be generated from the database, filled with data, be modified and how newly generated objects can be written into the database. See text for explanations.

3.7 Principles of the analysis pipeline: a parallel distributed system

The *PerlObject* entity shown in figure 3.1 plays a central role for the data production process of the analysis pipeline that is schematically represented in figure 3.9. The main annotation script (upper left box) contains code to generate different kinds of *Run* objects, e.g. *BlastRun* objects. The information to generate these objects is retrieved via an SQL interface from the 3D-GENOMICS database, that uses MySQL as the database management system (http://www.mysql.com). For a *BlastRun* object, this contains the protein sequence (*the Pseq* Object) and the processing parameters (a *Params* object). The 3D-GENOMICS database server can be hosted on a remote machine, and the annotation script runs on a queue-server that manages a computer farm via the OpenPBS load sharing and queueing system (http://www.openpbs.org).

The generated *Run* objects are submitted to the queueing system via a special software module of the 3D-GENOMICS API (the *Workstations* module), that calls the *queue* method of each of the objects to be queued. The method creates a serialised version of the object which is inserted as text into the *PerlObject* table of the 3D-GENOMICS database, and in return gets a unique *ID* (identifier) for this object. The *queue* method creates an appropriate command for the queueing system. This command contains the name of an executable program (*runobject.pl*) and the *ID* of the persistent object as an argument. The object may also request special resources from the queueing system such as a minimum amount of memory (the resource management is implemented in OpenPBS).

The queue-server submits the command to one of the free computers that runs a queue client (a PBS-daemon). The *runobject.pl* script retrieves the persistent object via the unique *ID* from the *PerlObject* table of the database and recreates the object. The script then executes the *run* method of the recreated *Run* object, which first inserts some meta information about this run into the database, and then performs the particular type of analysis (for example the BLAST program is executed on the local machine). From the result (e.g. the BLAST program output) the special type of result objects are generated (e.g. *BlastHit* objects). These objects are then inserted into the database by calling their *sync* routine (object synchronisation with the database). Finally the *Run* object cleans up resources such as temporary files, updates the object status attribute with the final status and inserts the runtime of the analysis. The *runobject.pl* script removes the run object from the *PerlObject* table of the database (this is no longer required).

The growth of the data that has to be processed, and in particular the increasing number of completed genomes, challenge the development of distributed processing systems. It is sensible to re-run previous analyses on a regular basis, because new data may change existing annotations. The 3D-GENOMICS system is a prototype that is currently used in-house only, and substantial development and testing has to be done to distribute this system to other institutions. However, the system is suitable for the distribution of the run objects that perform the analyses over a large computer grid allowing for frequent annotation updates.

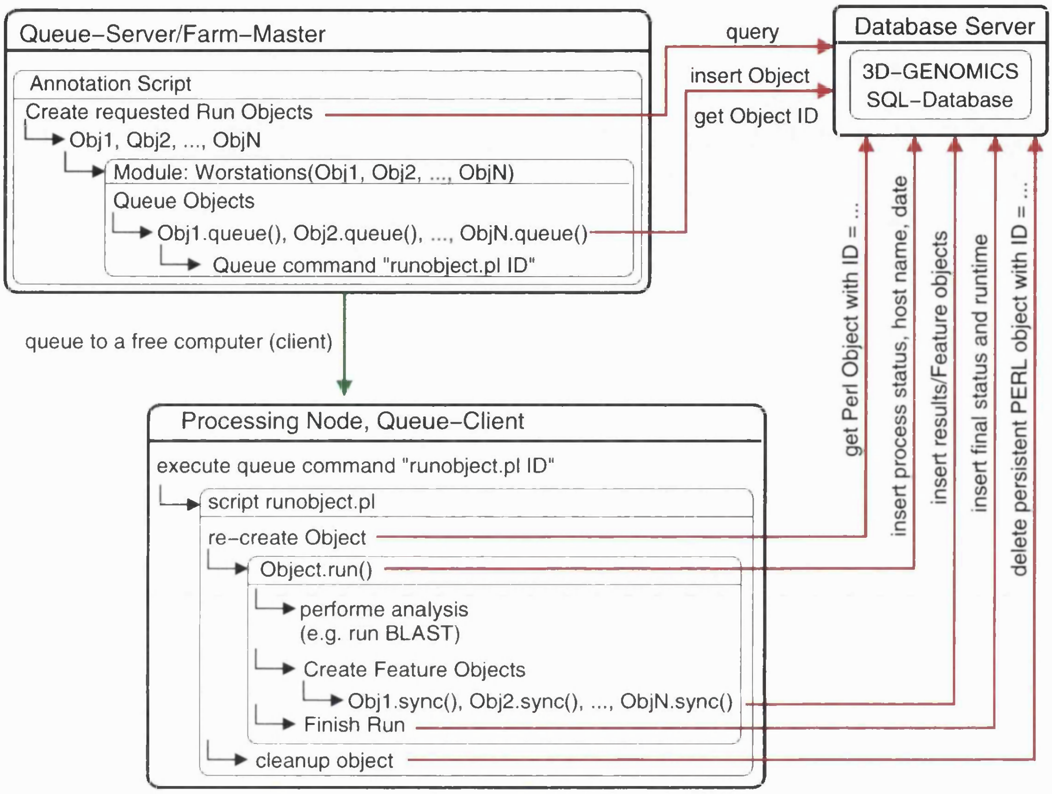

Queue–Server/Farm–Master
Annotation Script
Create requested Run Objects
Obj1, Qbj2, ..., ObjN
Module: Worstations(Obj1, Obj2, ..., ObjN)
Queue Objects
Obj1.queue(), Obj2.queue(), ..., ObjN.queue()
Queue command "runobject.pl ID"
query
insert Object
get Object ID
Database Server
3D–GENOMICS
SQL–Database
queue to a free computer (client)
Processing Node, Queue–Client
execute queue command "runobject.pl ID"
script runobject.pl
re–create Object
Object.run()
performe analysis
(e.g. run BLAST)
Create Feature Objects
Obj1.sync(), Obj2.sync(), ..., ObjN.sync()
Finish Run
cleanup object
get Perl Object with ID = ...
insert process status, host name, date
insert results/Feature objects
insert final status and runtime
delete persistent PERL object with ID = ...

Figure 3.9: Flow of the 3D-GENOMICS annotation pipeline. The three frames symbolise the main processing spaces, i.e. the physical location of computers and the execution space of programs and objects. Database requests (queries, updates and inserts) are symbolised with red arrows. The submission to a computer that executes the analysis is indicated by the green arrow. Arrows with a 90 degree angle indicate subsequent actions or a result of the previous step. Inner rectangles show the private execution spaces of scripts and objects. See text for details.

3.8 Discussion

The strength of the 3D-GENOMICS system has been discussed in the above sections. In particular the distribution of the *Run* objects for parallel processing is an important aspect. The straightforward implementation of new tools is certainly another strength. However, there are restrictions and problems with the current implementation, the more important of which are discussed below.

3.8.1 Restrictions of the current implementation

Although it is relatively simple to add new sequences to the 3D-GENOMICS database and to run these through the processing pipeline, regular updates are not yet supported. The main reason is that a single new protein sequence may change PSI-BLAST existing results for old query sequences (PSI-BLAST is a major component of the analysis pipeline), because it may provide intermediate sequence hits that are needed to detect for example a distant homology to a previously undetected family of proteins. Therefore on every database update one would have to re-run PSI-BLAST for all previously analysed proteins. This approach is time consuming and impractical, and in fact results may change for only a few proteins.

The effect of new sequences on PSI-BLAST PSSMs has to be studied to develop heuristics that will estimate the change of the path through evolution. Another probably simpler approach may be to compare the PSSM of an already processed sequence with new protein sequences. This is a relatively fast method that can be implemented via IMPALA or RPS-BLAST (part of the NCBI BLAST software).

The summary steps in the 3D-GENOMICS pipeline discussed in section 3.5 have to be re-run whenever the underlying 'atomic' data such as alignments changes. The genome wide summary does have a rather long runtime (several hours for the human proteome), and is mainly restricted by disk I/O of the database server, so that these runs cannot be distributed over a large number of clients to perform these runs in parallel. For the sake of speed, some parts of the database may have to be mirrored on different database servers, and the new concepts for fast *GenomeSummary* updates should be developed.

A version of the 3D-GENOMICS database that can be updated frequently may also need a history to keep track of changes. The definition of the gene of a processed sequence may change, and the old version of the gene should be marked as 'old', but should still be available to track changes.

There is a conceptual error in the 3D-GENOMICS database that can cause problems when a new sequence enters the database that is 100% identical to an existing sequence that has already been processed. The tag list of the protein description (see 3.1) is then updated by e.g. 'mouse' and may finally contain the keywords 'human' and 'mouse' (i.e. human and mouse have an identical sequence). Because of the relations between *Pseq, Pdesc* and *Tag* there is only one set of results from an analysis for this sequence (a protein sequence is stored only once, and several *PseqRuns* can refer to the same sequence). If one wants to delete all results for human, then the result for this sequence also get deleted for mouse. This systematic error has not yet affected the 3D-GENOMICS system because there are only very few 100% identical sequences between the processed genomes. Also, for the analysis described in chapter 4 all genomes have been processed with the same parameters and no data has been deleted. The problem also implies that only the non-redundant protein sequence set is stored, so that a few 100% protein duplications within a genome are ignored. This affects the analysis in chapter 4 because sequence features such as SCOP domains are only counted for each distinct protein sequence.

It is sensible to process identical protein sequences only once, even if these correspond to different genes. However, identical protein sequences from different genes have different accession numbers in the public databases, and the 3D-GENOMICS API should be modified so that the protein based analyses (*PseqRuns*) refer to an accession number rather than a distinct sequence. The API may handle cases for identical sequences, so that a requested analysis will not be run if it was already run with the same analysis parameters for another accession number referencing the same sequence. These 'virtual' sequence runs may be managed by reference counters, the results of an analysis only get physically deleted if the reference counter for the accession numbers to this run is zero.

3.8.2 Suggestions for future developments

Some technical and rather general enhancements should be considered for the future:

- Integration of InterPro (see 1.2.3). The collection of features that can be retrieved from 3D-GENOMICS via a run object for a protein sequence are very similar to the different descriptors for an InterPro entry. The InterPro Scan software is distributed from the EBI and contains all required programs and source databases. The baseline annotation can then be performed via InterPro, and 3D-GENOMICS can be focused on more specific tasks such as detection of remote homologues, structural characterised domains and proteome comparison.

- Export of all 3D-GENOMICS objects in XML format to provide the full repertoire of data in an state of the art format that can be distributed. The BLAST software from the NCBI can write its output in XML format. General handling of XML as an output format from the analysis programs and as a data source for sequence and annotation databases (InterPro and possibly GenBank in future) will ease the integration of other resources and data exchange.

- Management of free text information to enhance annotation. This can be initially approached by extracting text from different categories of the available source databases, and in particular the comment blocks of SwissProt entries which usually give manually curated detail about the biochemical and biological function of a protein. Abstracts from the scientific literature as well as a gene ontology may also be integrated to support annotation.

- Although the summary steps described in section 3.5 provide fast 'top-down' access (from an overview of the annotation down to more detail) to the results, it is useful to implement a non-normalised version of the database that can be generated from the normalised main database (the production database). Such a data warehouse may allow even faster access for research purposes and may be distributed to other bioinformatics sites.

- As mentioned in section 3.6, the data of an object is decomposed and stored in several tables of the database. On every level of inheritance for which data is stored in the database (e.g. for a *BlastHit* object the levels are *Feature* and *Alignment*) the data that belongs to a particular inheritance level is also exclusively managed on this level (generally by the particular class or baseclass). E.g. retrieval of a *BlastHit* requires three database requests: one to retrieve the feature data, one to retrieve the alignment data and one to retrieve the blast hit specific data. All three levels are logically linked by a common *FeatureId*. It may be much faster to create an object by using a single database request via a single join of the required tables. Each (base) class would have to contribute statements to the construction of an appropriate SQL statement that will join the required tables and to select the table attributes.

3.8.3 Other automated annotation systems

Automated annotation systems have been developed previously by others. In general these systems provide web based access and do not provide an external API that can be used for the development of specific research tools. However, these systems may be installed locally under special license agreements with the authors. The major goal of most public annotation systems is to support genome sequencing projects, and to provide up-to-date annotations, whereas the 3D-GENOMICS architecture is designed to provide consistent, but often not up-to-date, annotations that are easily accessible for large scale comparisons. In addition it should be pointed out that 3D-GENOMICS in its current version is maintained and developed by basically a single person mainly for the research described in this thesis, and the annotation systems described below are maintained by a team of authors often dedicated to maintenance and development of the system. Below a selection of popular annotation systems are introduced.

The *ENSEMBL* system (http://www.ensembl.org, Hubbard *et al.* (2002)) from which the protein data of the human genome is used within this work, provides an annotation system based on a MySQL database back-end with an object-oriented software interface written in Perl and C. ENSEMBL has been developed for the annotation of the human genome. Special versions for other ongoing metazoan genomes are also available. The ENSEMBL architecture is fully open and provides all data and software including a stable API. ENSEMBL is developed by a broad bioinformatics and biology community.

Despite the general management and dissemination of the human genome data, a special focus is the reliable identification of genes. On top of gene predictions with several levels of evidence, a baseline protein sequence annotation is performed. This includes the assignment of InterPro families and domains to human proteins. Some structure based analysis of human proteins (Gough & Chothia, 2002) is linked via DAS (Distributed Annotation System, Dowell *et al.* (2001)). Unlike the 3D-GENOMICS API that encapsulates the data processing within the biological objects (the *Run* objects), the data processing (for example BLAST sequence comparisons) in ENSEMBL is performed by mainly stand alone scripts that are separate from the biological objects (personal communication with Ewan Birney).

GeneQuiz (http://jura.ebi.ac.uk:8765/ext-genequiz/, Scharf *et al.* (1994); Andrade *et al.* (1999)) is one of the first published large scale annotation systems, that can be run remotely via the web. The input for GeneQuiz is a protein sequence or set of protein sequences for which the system runs several sequence analysis tools, including homology searches. A notable feature is the reasoning engine within GeneQuiz to accept or reject results contributing to an annotation. Different analysis tools and integrated source databases have different trust levels. Functional information from text descriptions is extracted for homologous sequences from the different source databases at different confidence levels, and together this information is used to place a protein into a functional category. GeneQuiz also provides structural models for proteins with homologues of known structure.

Magpie (Multipurpose Automated Genome Project Investigation Environment, http://genomes.rockefeller.edu/magpie/, Gaasterland & Sensen (1996)) is designed for (mainly prokaryotic) genome sequencing projects. The system takes DNA sequences such as DNA contigs (unassembled genomic DNA from cloning vectors) as input. Magpie guides the genome project from its beginning on, by performing gene predictions, detection of DNA frame shifts, homology searches on the protein and DNA level and suggests which pathways may exist in the genome. New tools can be integrated. The system is installed locally, and the analysis tools may be either installed locally or remote, in which case most data exchange is via an automated e-mail service. The Magpie system is configured and customised via a set of configuration files, so that no code editing is necessary.

Magpie stores the results of any analysis in flat files. Most of the infrastructure for data management is implemented in Perl. The results are then converted into Prolog facts that are digested and converted into 'deduced facts' from which HTML formated reports are generated. The Prolog rules for example to determine a coding region may be customised. Magpie also allows privileged users to manually edit and override automatically generated results

.

PEDANT (Protein Extraction, Description and ANalysis Tool, http://pedant.-mips.biochem.mpg.de, (Frishman *et al.*, 2001)) initially focused on protein based annotation. However, in version 2, many DNA based analysis tools such as those for gene prediction by homology to EST sequences or *ab initio* gene prediction have been integrated. PEDANT consists of three main parts: (i) the processing unit to access external databases and tools such as BLAST, (ii) the relational database (MySQL) for data storage and (iii) the user interface for user queries and data visualisation. The code for data management and processing is written in Perl and a part of the user interface is implemented in C++. All external databases such as the protein sequence databases and all tools are installed locally. Data processing may be performed in parallel by distributing tasks over a computer farm.

The database scheme of PEDANT is relatively simple, results are stored on two levels: the raw analysis output is kept as it is (e.g. the output from a BLAST run), and the parsed and disassembled output is stored, too (storing the e-value, the sequence identity etc. in different fields of a table). The results of an analysis are not mapped across several tables as in the 3D-GENOMICS database.

Since PEDANT is used for genome sequencing projects it implements a system to manage different versions of annotations and sequence data. The principle for genome annotation is to perform an automated analysis with relatively loose constraints to guarantee a great annotation extent over the whole genome, and then to allow manual correction of these annotations by accepting or rejecting annotations. PEDANT provides special user interfaces for manual data checking and correction.

PEDANT was used for SCOP superfamily assignments to more than 300,000 protein sequences.

A popular web based protein sequence annotation system is *PredictProtein* (http://-www.embl-heidelberg.de/predictprotein/predictprotein.html, Rost (1996)). The user can submit a protein sequence or a list of sequences to the server which runs a range of analysis and prediction software such as transmembrane predictions, homology, and motif searches. Many tools have been integrated in the PredictProtein system. The *meta server* facility in PredictProtein allows to submit a sequence automatically to several other servers that perform a specific analysis such as HMM based sequence comparisons. Results are formated as plain text or as HTML. PredictProtein is a service to provide biologists with as much information about a protein as possible, it is not intended for large scale comparative proteome projects.

Assignments of domains of known structure to proteins of fully sequenced genomes are provided by the *Gene3D* system (http://www.biochem.ucl.ac.uk/bsm/cath_new/-

Gene3D/, Buchan *et al.* (2002)), that is based on the CATH classification of protein structures introduced in section 1.4.4. Assignments are based on IMPALA (see section 1.3.6) and a set of specialised software to perform the actual delineation of domain boundaries within multi-domain proteins. Assignments can be browsed and downloaded over the web.

Chapter 4

Structural Characterisation of the Human Proteome

4.1 Summary

This chapter describes an analysis of the encoded proteins (the proteome) of the genomes of human, fly, worm, yeast and representatives of bacteria and archaea in terms of the three-dimensional structures of their globular domains together with a general sequence based study. This work shows that 39% of the human proteome can be assigned to homologues of known structure. The estimated extent of functional annotation for the human proteome is 77%, but only 26% of the proteome can be assigned to standard sequence motifs that characterise function. Of the human protein sequences, 13% are transmembrane proteins, but only 3% of the residues in the proteome form membrane-spanning regions. There are substantial differences in the superfamily composition of globular domains of transmembrane proteins between the proteomes that have been analysed. Commonly occurring structural superfamilies are identified within the proteome. The frequencies of these superfamilies enables one to estimate that 98% of the human proteome evolved by domain duplication, with four of the ten most duplicated superfamilies specific to multi-cellular organisms. The zinc-finger superfamily is massively duplicated in human compared to fly and worm, and occurrence of domains in repeats is more common in metazoa than in single-celled organisms. Structural superfamilies over- and under-represented in human

disease genes have been identified. Data and results can be downloaded and analysed via web based applications at http://www.sbg.bio.ic.ac.uk. This work has been accepted for publication by *Genome Research.*

4.2 Introduction

The interpretation and exploitation of the wealth of biological knowledge that can be derived from the human genome (Lander *et al.*, 2001; Venter *et al.*, 2001) requires an analysis of the three-dimensional structures and the functions of the encoded proteins (the proteome). Comparison of this analysis with those of other eukaryotic and prokaryotic proteomes will identify which structural and functional features are common and which confer species specificity. This work presents an integrated analysis of the proteomes of human and thirteen other species considering the folds of globular domains, the presence of transmembrane proteins, and the extent to which the proteomes can be functionally annotated. This integrated approach enables one to consider the relationship between these different aspects of annotation and thereby enhance previous analyses of the human and other proteomes (e.g. Frishman *et al.* (2001); Iliopoulos *et al.* (2001); Koonin *et al.* (2000), including the seminal papers reporting the human genome sequence from Lander *et al.* (2001) and Venter *et al.* (2001)).

A widely used first step in a bioinformatics based functional annotation is to identify known sequence motifs and domains from manually curated databases such as PFAM/InterPro (Bateman *et al.*, 2002; Apweiler *et al.*, 2001) and PANTHER (Venter *et al.*, 2001) . This strategy was used in the original analyses of the human proteome (Lander *et al.*, 2001; Venter *et al.*, 2001). These annotations tend to be reliable as these libraries have been carefully constructed to avoid false positives whilst maintaining a high coverage. In the absence of a match to these characterised motifs/domains, functional annotation is derived by homology to previously functionally annotated sequences. However, transfer of function by homology is problematic and the extent of the difficulty has been recently quantified (e.g. Devos & Valencia (2000); Todd *et al.* (2001); Wilson *et al.* (2000)). Below 30% pairwise sequence identity, two proteins often may have quite different functions even if their structures are similar. Because of this problem, global bioinformatics analyses of genomes generally do not use functional transfer from distant homologies for annotation. However, specific analyses by human experts still extensively employ this strategy, particularly as any suggestion of function can be refined from additional information or from further experiments.

A powerful source of additional information is available when the three-dimensional coordinates of the protein are known. The structure often provides information about the residues forming ligand-binding regions that can assist in evaluating the function and specificity of a protein. For example, recently it has been shown that spatial clustering of invariant residues can assist in assessing the validity of function transfer in this homology twilight zone (Aloy *et al.*, 2001). At higher levels of identity, knowledge of structure can assist in analysing ligand specificity and the effect of point mutations. Valuable tools in exploiting three-dimensional information are the databases of protein structure, in which domains with similar three-dimensional architecture are grouped together. Here the structural classification of proteins (SCOP) (Conte *et al.*, 2002) is used. SCOP is described in detail in section 1.4.4. In summary: in SCOP, protein domains of known structure that are likely to be homologous are grouped by an expert into a common superfamily based on their structural similarity together with functional and evolutionary considerations. SCOP is widely regarded as an accurate assessment of which domains are homologues. However, SCOP remains partially subjective and one cannot exclude the possibility that two domains placed within the same superfamily only share a common fold due to convergent evolution and therefore are not homologous.

The above considerations have led to focusing the analysis on the following three objectives:

- To estimate the extent to which the known proteomes can be annotated in terms of structure and function and how reliable these annotations are considered to be.

- To place the occurrence of particular SCOP structural superfamilies in terms of their biological and species-specific contexts.

- To derive evolutionary insights from frequency based analyses of homologous SCOP domains in terms of expansion in different species.

4.3 Strategy for structural and functional annotations

For details of materials and methods see section 4.6 on page 159.

Protein sequences from the human genome and from thirteen other species were analysed. The main strategy was to use the sensitive protein sequence similarity search program PSI-BLAST (Altschul *et al.*, 1997) to scan each protein sequence against a database composed of a non-redundant set of sequences, including sequences of SCOP domains and, to ensure up-to-date coverage, each protein entry of the PDB (Berman *et al.*, 2000).

A sequence match to an entry of the PFAM domain library Bateman *et al.* (2002) was considered as a functional annotation (excluding families of unknown function). In the absence of a match to these characterised motifs/domains, one needs to evaluate functional annotation via transfer from homology. To represent this approach computationally, functional annotation is simply considered if a homologue contains some textual description of function (see legend to figure 4.1, and section 3.5). Thus the total of the proteome that can be functionally annotated is the sections that are assigned to a PFAM domain or, if no assignment to PFAM, that are homologous to a protein with a text functional description.

4.4 Results

4.4.1 Status of structural and functional annotations

Figure 4.1 shows the annotation status of the proteomes expressed as the fraction of the total residues in each proteome. The residue fraction is used in order to include situations when only part of a protein sequence is annotated, since one cannot quantify this as a fraction of domains because one does not know the number of domains in un-annotated regions. 39% of the human proteome can be structurally annotated from either having a known protein structure or via a PSI-BLAST detectable homology to a known structure. This percentage is higher than that for yeast, fly and worm and is comparable to the coverage of many bacteria and archaea. A further 38% of the human proteome falls into the category of functional annotation without known structure. Since nearly every protein structure has some functional annotation, the total functional annotation of the human proteome is 77%. The remainder are (i) either homologous to another protein of unknown function or (ii) potentially globular orphan regions without any detectable homology or (iii) an un-annotated non-globular region (a region of low amino acid residue complexity, coiled-coil or a transmembrane segment).

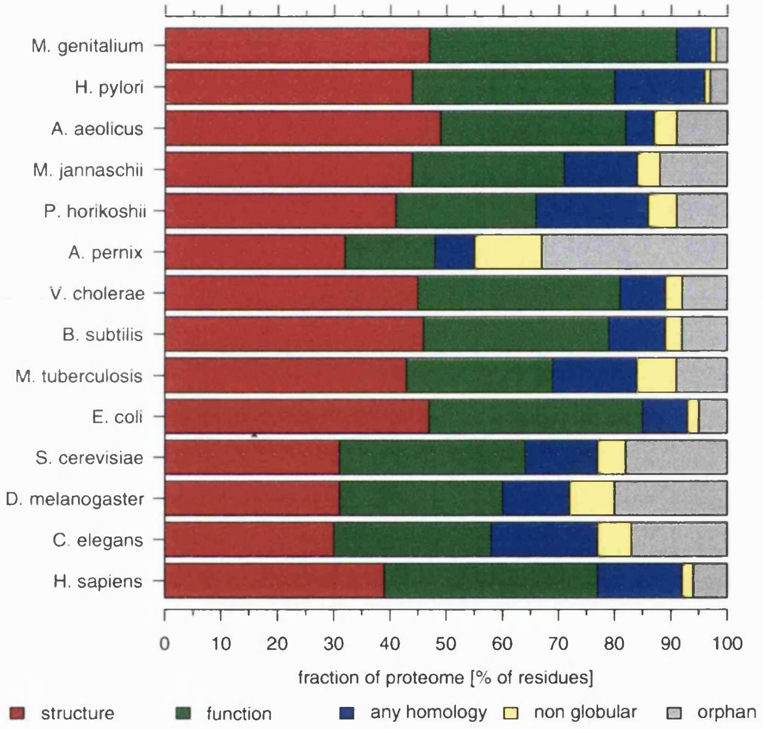


Figure 4.1: Annotation status of the proteomes. Coverage for each species is reported as the fraction of the residues in the proteome that are annotated. This allows for partial coverage of any sequence. Structural annotation is a homology to a sequence or domain of known structure. Functional annotation is when there is no structural annotation but there is an homology to an entry from SwissProt or PIR that has a description other than those that contain any of the following words: 'hypothetical', 'probable', 'putative', 'predicted'. Any homology denotes a sequence similarity to a structurally or functionally un-annotated protein, such as one described as hypothetical. See section 3.5 for a more detailed description of the classification of homologues. Non-globular denotes remaining sequence regions that were predicted as transmembrane, signal peptide, coiled-coils or low-complexity. Remaining residues are classified as orphans, i.e. unconserved potentially globular regions.

This work also considers how many protein sequences can be fully annotated. To allow for gaps >95% of a particular sequence are required to be covered without gaps of more than 30 residues (figure 4.2). The fraction of the human protein sequences that are fully annotated in terms of structure is only 15%. A further 14% of the human protein sequences are fully annotated in terms of function but not structure. The fraction of fully covered annotated sequences for human is much higher than for worm, fly and yeast. Another 8% of the human sequences are fully covered by hypothetical sequences or sequences of unknown function.

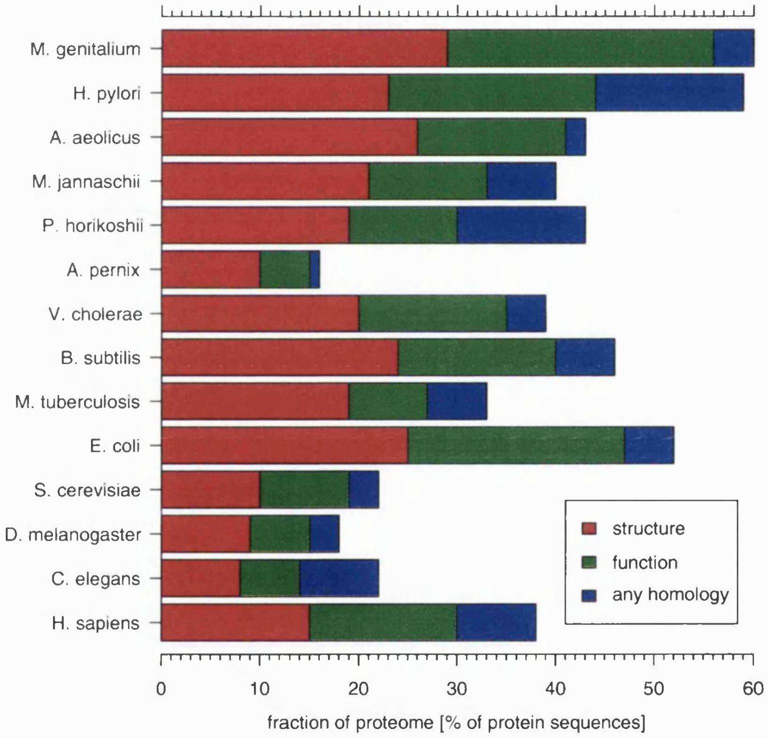


Figure 4.2: Structural and functional annotations that cover the entire protein sequence. For structural annotation >95% of the sequence is required to be structurally annotated, and there was no un-annotated segment of >30 residues. Functional annotation is evaluated after assigning structures and requires the same length constraints. Finally, any homologue (including those of unknown function) is assigned to the remainder (with the same sequence length constraints, also see figure 4.1 for a definition of any homology).

The accuracy of the above analysis is dependent on the quality of the gene prediction. For the eukaryotic genomes analysed, particularly for the human genome, this is problematic, and it is anticipated that new genes will be identified and some present assignments modified. The human proteome that is subject to the analysis described here is based on gene predictions that are confirmed by matches to ESTs or homologues in other species (see http://www.ensembl.org and Hubbard *et al.* (2002)). This use of homology would contribute to the high level of structural and functional annotation, and if additional genes were identified the values for coverage probably would be somewhat lower. An upper estimate of the magnitude of this problem can be obtained by noting that the human genome has 6% by residue of orphans. In worm this figure is 17%, and it is considered that most genes have been identified in this genome (Reboul *et al.*, 2001). Similar figures for orphans are found in yeast and fly. If one assumes that the true figure for orphan proteins in the human genome is 17%, then any other section of the annotation as shown in the bar-charts (e.g. of structural coverage) should be reduced to 83/94 (i.e. 0.88). Thus the structural coverage is reduced from 39% to 34%. In practise the true value is expected to lie between these two extremes.

However even for prokaryotes, errors in gene prediction can affect the survey that is described here. For example, the proteome of the archaea *Aeropyrum pernix* contains the largest fraction of orphan regions. This result may be biased because the gene prediction in *Aeropyrum pernix* produced many very short questionable ORFs (Skovgaard *et al.*, 2001).

4.4.2 Reliability of annotation

The reliability of homology model-building depends on the level of sequence identity between the protein of known structure with that of the sequence for which one wants to build a model (Bates & Sternberg, 1999; Sanchez & Sali, 1998). Figure 4.3 shows the different level of reliability for structural modelling. Only 2% of the residues in the human proteome are from domains for which there is an actual crystal structure or which share >97% sequence identity with an experimental structure. However, 11% are within the identity range 97% to 40%, and homology models are likely to be of sufficient accuracy to place residues reasonably accurately. Between 40% and 30% sequence identity, modelling becomes error prone, but advances in modelling techniques may allow the inclusion of this homology band for reliable modelling in the future. Below 30%, modelling is likely to reveal only general features of the fold.

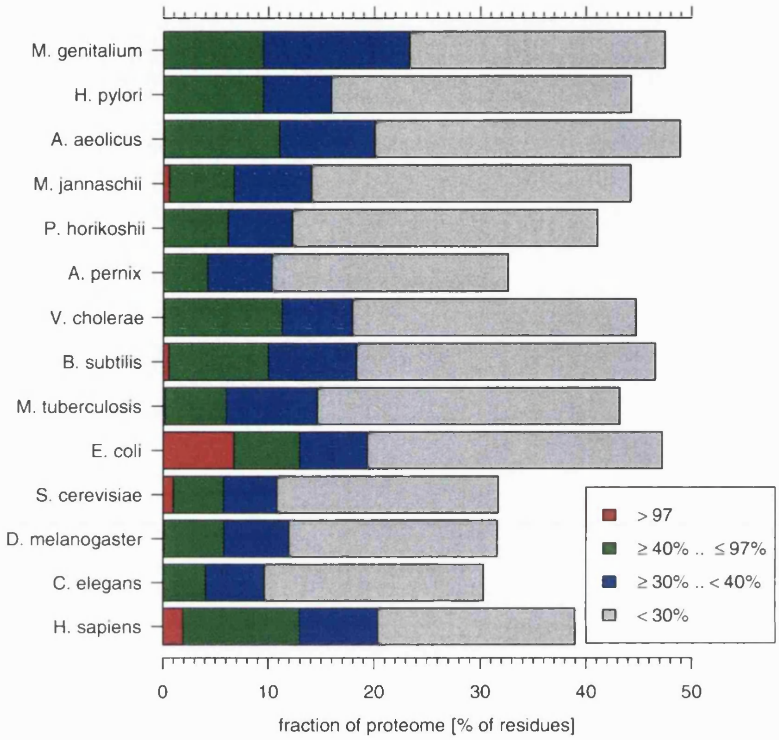


Figure 4.3: Reliability of structure assignments. Homologies are dissected into sequence similarity bands. The >97% identity effectively reports a match to an experimentally determined structure or to one that differs in only a few residues. Structures based on these annotations are accurate. The next band down to 40% sequence identity denotes annotations for which models can be constructed that are expected to be reasonably accurate (Bates & Sternberg, 1999; Sanchez & Sali, 1998). Between 40% and 30% sequence identity automated modelling is difficult. Below 30% identity, the sequence alignment suggested by the annotation is expected to have many errors and the structural annotation primarily provides an indication of the 3D fold.

Figure 4.4 provides an assessment of the reliability of functional annotation. A match to a PFAM domain (excluding domains of unknown function) is considered to constitute a reliable functional annotation. For the human proteome 26% of the residues can be assigned to PFAM domains (dark and light red bars in figure 4.4), this includes 19% for which a structural assignment can be made, which often will assist in functional annotation (dark red bars). Next, those proteins were identified for which the closest homologue that has a text functional description (see legend to figure 4.1) shares at least 30% sequence identity. This cut-off was chosen since studies have shown that below this value homologues often have diverged to radically different functions (Devos & Valencia, 2000; Todd *et al.*, 2001; Wilson *et al.*, 2000). A total of 41% of the proteome could potentially be functionally annotated based on a homology to a protein with at least 30% sequence identity (dark and light green bars). This 41% contains 15% without any match to PFAM but with an assigned structure (dark green bars) that could help to refine the proposed annotation. A further 8% of the proteome is below the 30% identity cut-off for functional annotation (blue bars). Of this fraction, 50% (4% of the total proteome, dark blue bars) has a structural homologue that may assist in assessing the validity of functional transfer. However the remaining 4% of the proteome with functional assignment below the 30% cut-off is without any structural information (light blue bars), and annotations for these sequence regions must be considered highly tentative.

4.4.3 SCOP superfamilies

Table 4.1 reports the commonly occurring SCOP superfamilies in human, fly, worm, yeast and average values for archaea and bacteria. Complete tables can be accessed from the following web-site: http://www.sbg.bio.ic.ac.uk.

First the commonly occurring superfamilies in the human proteome are considered. The most common domain in human is the C2H2 classic zinc finger, which occurs four times more often than the next most common domain, the immunoglobulin. The P-loop SCOP superfamily involved in nucleotide triphosphate hydrolysis is the fourth most common in human and second in fly, but the most common in the other analysed proteomes. In general, the commonly occurring superfamilies in the human proteome reflect the eukaryotic and multi-cellular organisation. Commonly observed superfamilies involved in or part of cell-surface receptors, protein-protein or cell-cell interaction, signalling or cytoskeleton structure are represented by superfamilies such as: immunoglobulin, EGF/laminin, fibronectin, cadherin, protein kinase, homeo-domain, tetratricopeptide repeat, spectrin repeat, PH-domain and SH3-domain.

In general, the fly and worm have similar rankings of the common superfamilies to those in human, reflecting the multi-cellular organisation. There are, however,

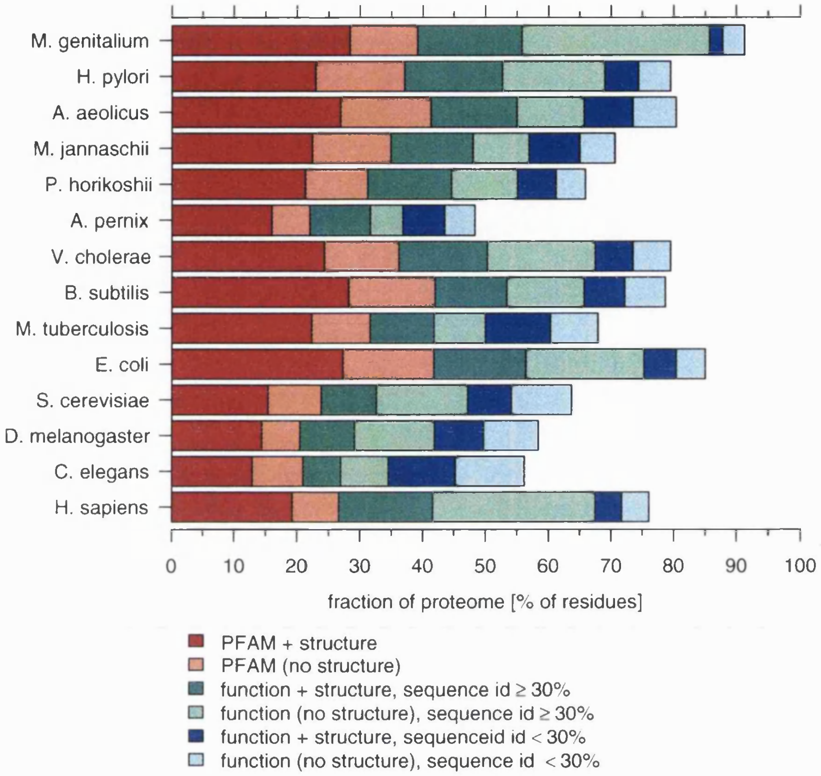


Figure 4.4: Reliability of functional annotation. Functional annotation is distinguished between reliable (30% sequence identity) and 'fuzzy' (< 30% sequence identity). The fractions are cumulative, i.e. regions that are assigned to a PFAM domain and a structure are counted first, then regions for which a PFAM domain could be assigned but no structural assignment can be obtained are counted. See text for details.

some differences. The c-type lectins are at rank 26 with 149 domains in human but at rank 5 with 310 domains in worm. C-type lectins have a wide spectrum of functions associated with carbohydrate binding and occur membrane bound and soluble. The high occurrence of c-type lectins has previously been noted by Koonin and co-workers (Koonin *et al.*, 2000). However, there has been no explanation for the abundance of this superfamily in worm. Similarly, the most common DNA binding domain in worm is the glucocorticoid receptor which is at rank 6 in worm (281 domains) but only at rank 27 (143 domains) in human and at rank 31 in the fly (69 domains). In contrast to the rank order, the domain frequencies of the top superfamilies in human are generally much higher than the corresponding frequencies

	Human		Fly		Worm		Yeast		Archaea		Bacteria	
SCOP superfamily	N	R	N	R	N	R	N	R	N	R	N	R
Classic zinc finger, C2H2	5092	1	1096	1	190	10	74	9	-	269	-	-
Immunoglobulin*	1214	2	483	3	457	2	8	91	1	135	4	94
EGF/Laminin	1192	3	320	4	413	4	-	-	-	-	-	-
P-loop containing nucleotide triphosphate hydrolases*	847	4	575	2	516	1	408	1	126	1	168	1
Fibronectin type III*	842	5	247	7	222	8	1	301	-	-	1	237
Cadherin	608	6	222	10	135	21	-	-	3	72	-	-
RNA-binding domain	587	7	282	5	199	9	128	3	-	-	-	420
Protein kinase-like (PK-like)*	557	8	271	6	434	3	142	2	3	72	5	82
Homeodomain-like	334	9	144	18	145	17	32	20	1	221	17	16
Spectrin repeat	327	10	227	9	150	13	-	-	-	-	-	-
PH domain-like*	327	10	140	19	100	31	23	29	-	-	-	-
SH3-domain	304	12	105	23	70	37	29	23	-	-	-	454
EF-hand*	284	13	163	14	120	26	23	29	-	-	-	420
Ankyrin repeat	278	14	120	21	128	24	31	22	-	-	1	342
Complement control module/SCR domain	277	15	57	38	52	43	-	-	-	-	-	-
PDZ domain-like	265	16	103	24	89	32	6	120	1	169	6	64
Ligand-binding domain of low-density lipoprotein receptor	247	17	196	12	143	18	3	194	-	-	-	-
Tetratricopeptide repeat (TPR)*	215	18	171	13	115	27	98	5	4	48	16	19
RING finger domain, C3HC4	207	19	108	22	122	25	33	19	-	-	-	-
Trp-Asp repeat (WD-repeat)	193	20	198	11	142	19	114	4	2	121	3	157
C2 domain (Calcium/lipid-binding* domain, CaLB)	186	21	68	32	89	32	32	20	-	-	-	-
NAD(P)-binding Rossmann-fold domains*	177	22	150	16	130	23	88	7	27	3	72	2
ARM repeat*	177	22	137	20	105	28	80	8	1	221	-	-
SH2 domain*	161	24	59	37	72	35	8	91	-	-	-	-
Thioredoxin-like*	152	25	148	17	148	14	50	12	8	21	18	13
C-type lectin-like*	149	26	40	53	310	5	-	-	-	-	-	454
Glucocorticoid receptor-like (DNA-binding domain)*	143	27	69	31	281	6	14	59	-	-	-	-
ConA-like lectins/glucanases*	136	28	66	34	105	28	8	91	1	169	3	157
Actin-like ATPase domain*	135	29	65	35	38	56	58	10	2	97	12	26
Numer of distinct proteins in proteome	28,913		13,922		16,323		6,237		2,176		2,789	
Numer of distinct superfamilies in proteome	546		518		482		434		328		499	

Table 4.1: Commonly occurring SCOP superfamilies in the proteomes. R is the rank of a superfamily within a proteome and N is the frequency of domains within this superfamily. * Denotes that several PFAM families (and hence several InterPro families) are included within the single SCOP superfamily (this association was evaluated by searching each SCOP superfamily against PFAM using the HMMer program, see 'Methods' section for details). The number of distinct proteins and the number of domains per superfamily (N) for archaea and bacteria are averages whereas the number of distinct superfamilies are totals over the species (including seven bacterial species and three for species from archea).

in fly and worm, whereas the frequencies in fly and worm are often similar. The human proteome is roughly double the size of that of fly or worm, but for several of the most common superfamilies in human (in particular within the first six ranks, except for the P-loop) a scaling factor of more than two is observed. At lower ranks the ratio is generally around two. The first superfamily that occurs with roughly the same frequency in human, fly and worm is the thioredoxin-like domain (152, 148, 148 domains respectively). Proceeding down the rank order of occurrence in human, the first superfamily with a lower frequency of domains in human than in another multi-cellular eukaryote is the c-type lectin (see above).

There are, however, major differences in rank order for the single-celled organisms. Several of the superfamilies in table 4.1 have similar ranks in human, fly and worm, whereas the rank in yeast often differs markedly (e.g. the immunoglobulin). Domains of superfamilies found in cell-cell interaction proteins and cell surface proteins such as the fibronectin and cadherin are not found or only occur infrequently in the proteomes of the single-cellular organisms. In bacteria, and especially in archaea, the top ranks are mainly occupied with superfamilies associated with enzymes. The most common DNA binding domain in bacteria and archaea is the winged helix-turn-helix motif (not included in table 4.1).

The abundance of several superfamilies in metazoans that are absent or have relatively low domain frequencies in yeast leads to conclusions different to those recently published for the *S. pombe* genome (Wood *et al.*, 2002). The work by Wood *et al.* (2002) shows that there are many new protein sequences in yeast (*S. pombe* and *S. cerevisieae*) compared to prokaryotes, but only a few new sequence families in metazoans compared to yeast (i.e. those proteins found in metazoans only). In this work 84 SCOP superfamilies present in metazoa and yeast that are not found in any of the processed prokaryota, and 113 new superfamilies in metazoa that are not found in yeast (data not shown) were identified. The analysis described in this work is based on the identification of structural domains rather than closely related full-length sequences which allows members of even diverse superfamilies to be found. These results suggest that in invention and expansion on the level of structural domains there may well be a bigger step from single-cellular eukaryotes to multi-cellular organisms than implied by Wood *et al.* (2002).

Domains forming a particular SCOP superfamily are identified on the basis of both their similar structure and function. In contrast PFAM, InterPro and PANTHER are primarily sequence and function based families. Because homologies can be recognised from structural conservation that are undetectable by sequence based methods, one SCOP superfamily can include several PFAM, InterPro or PANTHER families (also see the legend for table 4.1). In addition, SCOP is a structural domain database whereas PFAM identifies a single sequence motif that can be repeated to form a structural domain. For example, PFAM describes each of the β-sheet motifs of a WD-repeat by itself whereas SCOP considers the entire barrel of seven of these motifs as a domain. Thus there are several differences between the ranks of commonly occurring SCOP domains compared to the results from sequence based analyses (Lander *et al.*, 2001; Venter *et al.*, 2001).

The results of this work are in broad agreement with similar analyses by others (Frishman *et al.*, 2001; Iliopoulos *et al.*, 2001; Koonin *et al.*, 2000; Gough & Chothia, 2002; Lander *et al.*, 2001; Venter *et al.*, 2001), in particular with results from those describing the distribution of SCOP folds and superfamilies in different genomes. Differences in methodology, different confidence cut-offs and different sequence databases used for the analysis do not allow a direct comparison of domain frequencies and annotation coverage in proteomes. However, the relative rank order for folds and superfamilies within a proteome are suitable for a comparison between different work. Recent work from Gough & Chothia (2002) using hidden Markov models for SCOP superfamilies shows similar ranks for the top ten superfamilies in the processed genomes. The zinc-finger is the most abundant superfamily in human followed by the immunoglobulin. Although results from the HMM superfamily analysis by Gough & Chothia (2002) on a more recent version of the human genome (based on ENSEMBL-4.28.1, see http://stash.mrc-lmb.cam.ac.uk/SUPERFAMILY/) give different total numbers compared to this work, the general trend (i.e. ranks of superfamilies) is stable even for the different interpretations of the human genome. It should be noted that the analysis described here has a focus on the globular parts of the proteomes, and no PSI-BLAST homology assignments for the membrane all-α SCOP superfamily were obtained. However, BLAST assignments for close homologues of this superfamily are included in the analysis of this work (see section 4.6, Methods, for details). Therefore this superfamily is found far further down the list lower in the results described here compared to Gough & Chothia (2002), who constructed special HMMs for this superfamily.

Some of the top superfamily-rankings from this work are different to those in PartsList (Qian *et al.*, 2001b), which reports the EGF/laminin superfamily at rank one for *C. elegans* (rank four in this analysis) and the P-loop at rank eight, compared to rank one in the results of this work. The HMM superfamily analysis of the worm from Gough & Chothia (2002) ranks the P-loop at position two, following the membrane all-α superfamily.

Wolf *et al.* (1999) assigned SCOP-1.35 folds to several prokaryotes, yeast and *C. elegans* using an automated processing pipeline similar to the system used here (see section 3). Folds of coiled-coiled domains and immunoglobulins and those domains mainly found in viruses were omitted from their analysis. The top ranking SCOP folds for archaea are similar to the ranks from this analysis, but there is more variation in ranks for bacteria, possibly due to differences in the set of bacterial genomes that was chosen for this work. As shown by Wolf *et al.* (1999), the analysis described in this work also finds more agreement between archaea and bacterial folds compared to eukaryotic folds. The fold analysis by Wolf *et al.* (1999) was refined (Koonin *et al.*, 2000) by including the IMPALA program (Schaffer *et al.*, 1999) into the processing pipeline.

The results for *M. genitalium* (MG) and *M. tuberculosis* (TB) reported in chapter 2 differ from the results described in this chapter. Here, 46% and 43% of all residues in MG and TB respectively can be assigned to homologues of known structure compared to only 29% in both proteomes from the analysis reported in chapter 2. However, the analysis described here was carried out in 2001, and the analysis from chapter 2 was from 1998 and 1999. The main reason for the much higher coverage is the growth of the protein structure and sequence databases during this period. In 1999, there were 11,364 structures in the PDB (in less than 600 SCOP superfamilies) compared to 16,973 structures (in more than 1,000 SCOP superfamilies) in 2001 (see the database statistics at http://www.rcsb.org). The non-redundant protein sequence database grew from about 300,000 proteins to more than 600,000 proteins between the year 1999 and 2001. New protein folds have entered the database, and to some extent existing classifications have been revised.

The rank order of SCOP superfamilies based on SCOP version 1.37 from the analysis in chapter 2 are similar to those from the analysis based on SCOP 1.53 (the analysis described in this chapter), but the domain frequencies increased. Especially the number of identified P-loops increased from 20 to 69. Many new members of the P-loop containing nucleotide triphosphate hydrolases superfamily increased the coverage of this superfamily in the proteomes of MG and TB. In SCOP version 1.37 there were only five families within the P-loop superfamily. In SCOP version 1.53 there are fourteen P-loop families. The rank order in the TB proteome shows greater differences in superfamily rank orders. For example the P-loop changed rank from 10 (36 domain) to 1 (176 domains) when comparing the old with the new analysis. The NAD(P)-binding Rossmann-fold formerly the most popular superfamily in TB with 123 domains slipped to rank 2, but still with an increase in absolute frequency to 142 domains. Nevertheless, as mentioned above (page 130), the rank order of superfamilies in different versions of the human proteome has not changed markedly.

This brief comparison between versions of a similar analysis highlights the impact of data growth and the importance of the continuous increase in the experimentally determined repertoire of protein structures, including a refinement and diversification of already known folds with new family members. It is important to monitor and benchmark the changes of structural and functional coverage in genomes to refine existing results. The 3D-GENOMICS system described in chapter 3 is a step toward this goal.

4.4.4 SCOP superfamilies specific for phylogenetic branches

Table 4.2 presents SCOP superfamilies that occur within just one species or set of related species but not in any of the other organisms analysed. To identify species not included in the fourteen genomes that were analysed in this work, each member of a superfamily that is potentially unique to one of the analysed genomes was compared to the non-redundant sequence database using PSI-BLAST (with the parameters described in the method). This database contains more than 30,000 species. In table 4.2 any superfamily that occurs less than four times in a particular branch (human, fly, worm, yeast, bacteria, archaea) is excluded to prevent erroneous inferences due to the inherent difficulties of automated annotation. This information identifies biological functions potentially specific for one branch of life.

SCOP Superfamily	N	R	Functional description
Human			
MHC antigen-recognition domain	57	62	Immune system
Interleukin 8-like chemokines	48	71	Immune system, growth factors
4-helical cytokines	47	75	Immune system, diverse range of interferons and interleukins
β/γ Crystallins/protein S/yeast killer toxin	20	144	Eye lens component
Serum albumin	19	150	Major blood plasma component
Colipase-like	11	202	Enzyme regulation for pancreatic lipases, development
RNase A-like	8	237	Different ribonucleases found in pancreas, eosinophil granules and involved in angeogenesis
PKD domain	7	260	Possibly involved in extra-celluar protein-protein interaction
Defensin-like	7	260	Small anti bacterial, fungal and viral proteins
Uteroglobin-like	5	294	Binding of phospholipids, progesterone, inihibits phospholipase A2 (involved in metabolism of biomembranes)
Midkine	4	328	Growth factors
Fly			
Insect pheromon/odorant-binding proteins	26	81	Hormone related, sex recognition
Scorpion toxin-like	6	220	Drosomycin and defensin, antibiotic, fungicide
Worm			
Plant lectins/antimicrobial peptides	4	234	Anti microbial peptides, pathogen response, fungicides. Homologous to plant proteins.
Osmotin, thaumatin-like protein	4	234	Same description as for lectins above.
Yeast			
Zn2/Cys6 DNA-binding domain	53	11	Transcription factors
DNA-binding domain of Mlu1-box binding protein MBP1	4	155	Transcription factors
Bacteria			
TetR/NARL DNA-binding domain	112	19	Transcription factors
IIA domain of mannitol-specific and ntr phosphotransherase EII	28	99	Carbohydrate transport system: part of phosphoenolpyruvate-dependent sugar phosphotransferase system (PTS)
Prokaryotic DNA-bending protein	18	157	Bacterial histone like proteins
Zn2+ DD-carboxypeptidase, N-terminal domain	17	165	Found in enzymes involved in bacterial cell-wall degradation, possibly peptidoglucan binding domain
Glucose permease domain IIB	17	165	Part of PTS
Regulatory protein AraC	14	182	Part of the transcription regulation of the arabinose operon
LexA/Signal peptidase	11	211	1. Transcriptional regulation of SOS repair genes, protease domain of the LexA protein 2. Cleaves the N-terminal signal peptides of secreted or periplasmic proteins.
Histidine-containing phosphocarrier proteins (HPr)	11	211	Part of PTS
Periplasmic chaperone C-domain	11	211	Assembly of extra-cellular and periplasmic macromolecular structures
Duplicated hybrid motif	10	224	Part of PTS
Aspartate receptor, ligand-binding domain	10	224	Found in different membrane integral sensor and chemotaxis proteins, often associated with kinase domains.

Table 4.2: Superfamilies unique for one of the processed proteomes or group of proteomes. The functional description is taken from PFAM/InterPro and SwissProt homologues. N and R are the same as in table 4.1. For Human, fly, worm and yeast the superfamilies with N > 3 and for bacteria N > 9 are listed.

Human branch. The three most frequent domains are implicated with immunity, in particular the MHC antigen-recognition domain, interleukin 8-like chemokines and the 4-helical cytokines. Analysis of results that include the complete sequence database showed that in addition to mammals the interleukin 8-like superfamily is also found in sequences from birds and fish, and the MHC antigen-recognition domain is also found in amphibia. Several of the other domains specific to the mammalian branch are also involved in immunity - MHC class II-associated invariant chain ectoplasmic trimerization domain and p8-MTCP1 (mature T-cell proliferation). The mammalian defensin is involved in defense against a wide range of micro organisms, whereas the defensin-like superfamily is also found as neurotoxin in some cnidaria such as anemonae. At fifth in frequency in the human branch is serum albumin (19 domains in 19 sequences) that is a major protein component of blood.

Many of the superfamilies that appear potentially specific for human or other mammals (i.e. superfamilies that are not found in any of the other 13 processed genomes) are in fact also found in some viruses, amphibia, reptiles, fish and birds when considering sequences and species of the complete sequence database (>600,000 sequences and >30,000 species). These include the following frequently occurring superfamilies: colipase-like for enzyme regulation (particularly required by pancreatic lipases) and involved in development; RNaseA-like (also found in *Aspergillus*) with different ribonucleases involved in endonuclease function in pancreas, blood (eosinophil granules) and in angiogenesis; the PKD domain which is possibly involved in extra-cellular protein-protein interaction.

Fly. Insect pheromone/odorant-binding proteins are the most common SCOP superfamily (which occurs 26 times). The next most common are the scorpion toxin-like domains which occur as parts of the fungicide drosomycin, and the anti-bacterial defensin. Thus the insect form of immunity/defense leads to a commonly occurring branch-specific SCOP superfamily. However, in addition to arthropods, the scorpion like-toxin and the anti-bacterial defensin are also found in plants.

Worm. Two superfamilies occur with a frequency four (the osmotin, thaumatin-like proteins and the plant lectins/antimicrobial peptides). These superfamilies are not found in any of the other 13 proteomes. Both superfamilies are involved in pathogen response. However, further comparison of these superfamilies with the complete sequence database identified close homologues in plants.

Yeast (*S. cerevisiae*). This is dominated by the Zn-Cys DNA-binding domain of transcription factors. This family is also found in the recently sequenced genome of the yeast *S. pombe* (Wood *et al.*, 2002).

Bacteria. Given the smaller size of bacterial genomes, the superfamilies and their frequencies from the seven organisms that were annotated in this work were pooled (i.e. the reported frequencies are the sums of domains in superfamilies from all seven bacterial proteomes, and not averages). Here, the higher ranking superfamilies are discussed. The most frequent domain is a transcription factor - the tetR/NARL DNA-binding domain (also found in some archaea and algae when considering the complete sequence database). This is followed by the dimerisation domain of the AraC protein that is involved in the transcription regulation of that operon. Third is the superfamily of the DNA-bending protein. Other potentially specific superfamilies are involved in transport (especially the phosphate transferase system, possibly also present in fungi). There is one superfamily involved in the phosphate transferase system, the duplicated hybrid motif, that is also found in mouse (but not human) as previously noted (Nakamura *et al.*, 1994). In addition there are superfamilies specific for the cell wall synthesis, with one superfamily, the Zn2+ DD-carboxypeptidase, that is also found in plants.

Archaea. There are only three species of archaea in the set of organisms that are included in the analysis described here, and no frequently occurring archaea specific SCOP superfamilies could be identified.

The general conclusion from this analysis is that three general classes of biological activity lead to commonly occurring branch-specific superfamilies. These functions are defense (e.g. immunity), transcriptional regulation and hormone-related signalling.

4.4.5 Gene duplication

The presence of multiple copies of any particular SCOP domains within the proteome is the result of domain duplication and divergence during evolution, both within and between proteins. The extent of this duplication can be quantified:

$$duplication = \frac{\sum_i (N_i - 1)}{\sum_i (N_i)} \tag{4.1}$$

where N_i is the number of occurrence of domains in SCOP superfamily type i (Teichmann *et al.*, 1998). This can be estimated from the frequencies of the SCOP superfamilies in a proteome, using these domains as a sample of the entire proteome. Note that the value is for domain duplication and is not necessarily a value for the fraction of the proteome residues that arose from duplication. Figure 4.5 shows that 98% of the human proteome is estimated to arose via duplication. There are 28,913 different peptide sequences in the data set of human proteome, and 23,573 SCOP domains were identified within these sequences, which belong to only 546 different SCOP superfamilies with 23,027 duplication events. The figure shows that as the number of proteins in the genome increases, there is an increase in the extent of domain duplication from the 55% observed in the smallest proteome (*M. genitalium*) to 98% in the biggest proteome (human). There is a very rapid increase in the extent of domain duplication in the bacteria and archaea until the smallest eukaryote included in this analysis (yeast) is reached. However, one does not observe a marked difference in the extent of duplication between the largest prokaryote (*E. coli*, 4257 peptide sequences) and the smallest eukaryote (yeast, 6237 peptide sequences) despite the major differences in the organisation of their genes (in terms of the presence of introns/exons and of chromosomes). Importantly, since several different PFAM families are homologues that belong to the same SCOP superfamily, when the same estimate is made using PFAM one obtains a lower estimate of the extent of domain duplication in each species.

This estimate of domain duplication relies on two assumptions. First is that the duplication frequency of structurally characterised domains (i.e. SCOP) is a representative sample of all proteins in the genomes. This has been analysed for proteins in the *M. genetalium* genome by Teichmann *et al.* (1998) who concluded that the SCOP superfamilies are representative for the proteins in the genome. However, a study by Gerstein (1998a) on eight microbial genomes suggested that there are several differences between the proteins in the PDB and those in the genomes, including differences in the lengths of the sequences. Nevertheless, the trend of increasing domain duplication with the size of the proteome is the same for the SCOP

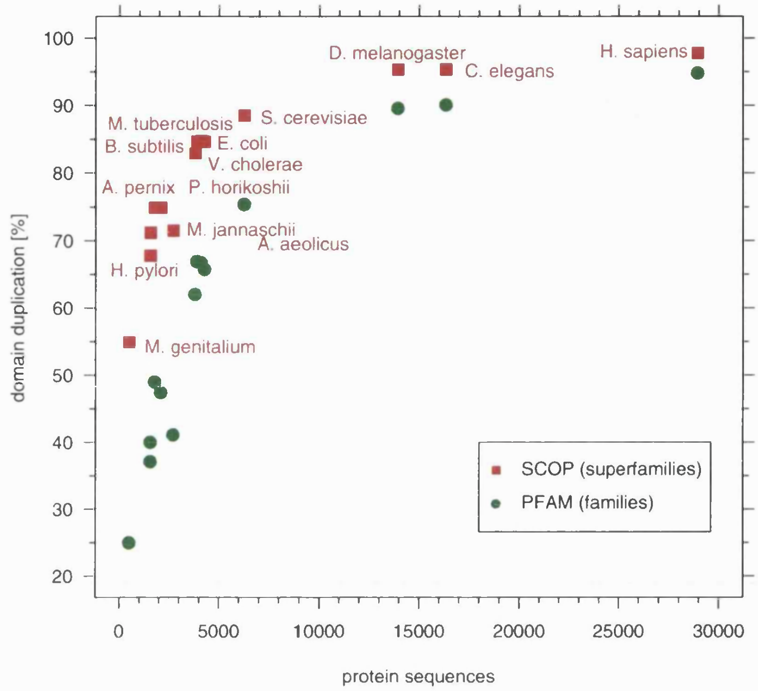


Figure 4.5: Extent of domain duplication in different proteomes. The extent of duplication is estimated from the frequencies of observing domains in the different SCOP superfamilies is shown as the fraction of total assigned domains for each proteome. The size of the human proteome is estimated at the number of protein sequences in the ENSEMBL dataset (2̃9,000). Comparable results from frequencies of PFAM families are reported

and PFAM based analysis, suggesting that any bias from using SCOP alone is not marked. The second assumption is that all the proteins have been identified in the genome, and one has to estimate the effect of uncharacterised proteins. However, the worm, where gene prediction is more accurate than in human, and therefore even rare and orphan protein families are more likely to be identified (Reboul *et al.*, 2001), yields a value for domain duplication of 95% which is probably a lower estimate of the extent in human.

The values for domain duplication are without a time scale and substantial further work is required to estimate the extent of duplications since divergence of the different phylogenetic branches. Recently Qian *et al.* (2001a) have developed an evolutionary model and estimated the extent of fold acquisition within a species. Here the extent of duplication in the different species of the ten most frequently occurring SCOP superfamilies found in the human proteome is considered (figures 4.6 to 4.8). Taking the frequency in humans as 100%, figure 4.6 shows that all of these ten SCOP superfamilies have been expanded in human compared to all other species. The greatest expansion from worm and fly to human is for the classic zinc finger. This suggests the major increase in importance of transcriptional regulation in humans via zinc-fingers compared to fly and worm. In contrast, the smallest extent of expansion from prokaryotes to human is for the P-loop that has a central role in housekeeping metabolism. This smaller rate of expansion is also observed for another housekeeping superfamily, the RNA-binding domain found at rank three in yeast. The protein kinase-like superfamily has a markedly bigger expansion in worm than in fly, and corresponds to 80% of the expansion in human. This may account for the expansion of certain types of signalling in worm. Note that three of the superfamilies shown are not found in yeast (EGF/laminin, cadherin and the spectrin repeat), and one, the fibronectin, is only found once.

These results can be contrasted with an analysis of the top superfamilies in bacteria. Of the top ten, seven are expanded in bacteria between 150% and 350% relative to human (data not shown). The two superfamilies that are reduced in bacteria compared to human are the periplasmic binding protein-like II (extra-cellular receptor domains in human and mainly extra-cellular solute binding domains in bacteria) with 70% and the thiolase-like domain (84%). In human Chey-like transcription factors could not be found at all.

Figure 4.7 shows the relative domain frequencies (number of observed domains in a superfamily normalised by the total number of domains in all superfamilies in the proteome) of the top ten human superfamilies for the processed proteomes. The 5092 zinc-finger domains that were identified for human comprise more than 20% of the identified domains. Zinc-finger domains have an average length of just 27 residues, and together this corresponds to only 1.5% of the residues in the human proteome. Compared to the majority of the top ten human superfamilies, the P-loop decreases its relative abundance from prokaryotes to human. Although the domain fraction comprised by P-loops is much lower than for the zinc-finger, because of its average length of 217 residues in human, the P-loop accounts for 2% of all residues. In yeast and worm the protein kinase-like superfamily seems to have more

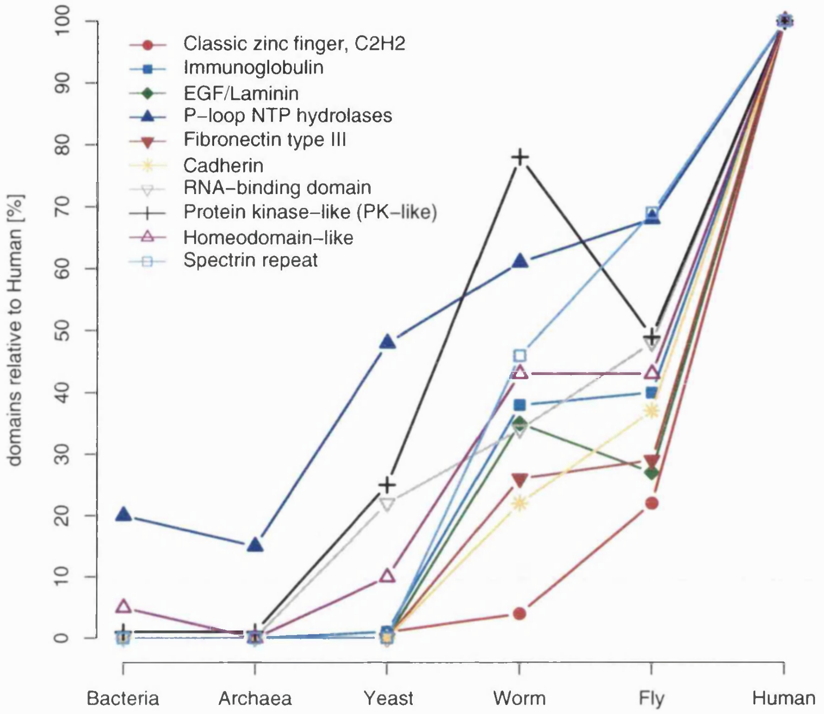


Figure 4.6: Superfamily expansion relative to the human proteome. For the ten most abundant human superfamilies the superfamily expansion within the other proteomes relative to the human proteome is plotted as the number of domains in superfamily X in proteome Y divided by the number of domains in superfamily X in human (times 100). All superfamilies are 100% in human.

importance than in fly and human. In addition the RNA-binding domain, involved in a range of functions, is more abundant in yeast than in the metazoan proteomes where this superfamily accounts for roughly the same fraction of domains. The worm proteome contains relatively more EGF/laminins compared to fly. In general the relative abundance of the top ten superfamilies in human, except for the zinc-finger, is similar between the metazoan proteomes. Plotting the top ten superfamilies for yeast shows a similar trend (data not shown); there are only slight changes in the relative domain abundance for most superfamilies between the eukaryotic proteomes. These results imply that in general the most popular superfamilies in a particular proteome do not comprise a substantially different fraction of the domain repertoire in other proteomes. Given an increasing number of domains for larger proteomes, it may not be a change in relative domain abundance of a set of superfamilies that leads to specialisation.

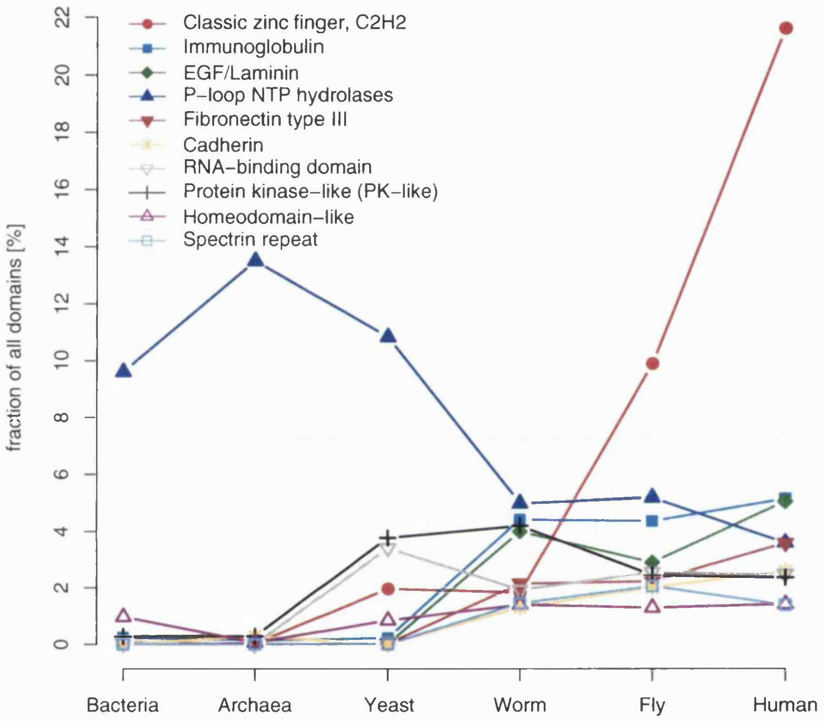


Figure 4.7: Relative expansion of the ten most abundant human superfamilies. For all proteomes the number of domains in a superfamily is normalised by the number of domains in all superfamilies for a proteome (multiplied by 100).

In general, domains of superfamilies found at a high rank are often found in repeats. Here a repeat is defined as at least two domains of the same superfamily that are found within the same peptide sequence irrespective of the sequence distance between these domains. Indeed, the zinc finger is the most repeated domain in human. The average numbers of repeats for the zinc-finger are 7 (max. 36), 4 (max. 17), 2 (max. 5) and 2 (max. 5) per zinc finger containing sequence for human, fly, worm and yeast respectively. In fly and worm the most repeated domain is the cadherin with on average twelve repeats in fly and eight in worm. The most repeated superfamily in yeast is the KH-domain (probably involved in RNA-binding) with four repeats on average, and in prokaryotes this is the thiolase-like superfamily (found in proteins of degradative pathways such as fatty acid β-oxidation) with two repeats on average.

Considering only the existence (and not the frequency) of a superfamily in a sequence to exclude the effect of repeats overall just slightly changes the order of the top ranks of superfamilies. The domain based top ten ranks in human are still present in the top 22 list that excludes repeats (except for the spectrin repeat at rank 43). The immunoglobulin, the EGF/laminin and the fibronectin are still within the top ten (data not shown). Figure 4.8 plots the average number of repeats within a protein for each of these ten SCOP superfamilies in human. The most notable feature is that the fly has far more duplicated copies per protein for cadherins (cell surface) and spectrin repeats (e.g. associated with the cytoskeleton) compared to human. Both, worm and fly have more repeated copies per protein of fibronectin and immunoglobulin than human. Overall five of the ten superfamilies are repeated on average at least twice per sequence in human. The most abundant superfamilies in yeast and especially in bacteria are not as frequently found in repeats as the most popular superfamilies in metazoa (data not shown).

In general this implies that repetitiveness on the domain level may play an important role in the divergence of the metazoan branch from single-cell eukaryotes. As mentioned above, several of the popular superfamilies in human are associated with cell-surface functions such as cell adhesion, for which long proteins with regular structure may be required.

Another analysis of this work considers the number of different domain-domain associations for the commonly occurring SCOP superfamilies. An association is taken when two different SCOP superfamilies occur within the same sequence (including self association) irrespective of the sequence separation betwwen these domains. For a detailed analysis of pairs of adjacent domains and their phylogenetic distributions see Apic *et al.* (2001). Figure 4.9a plots the number of partners for the ten most common superfamilies in human, figure 4.9b for those in yeast and figure 4.9c for bacteria (note, that for better scaling of the plots, in 4.9b and 4.9c only superfamilies are shown that are not already plotted in 4.9a). The general trend is that the numbers of different associations is roughly similar for the three multi-cellular eukaryotes.

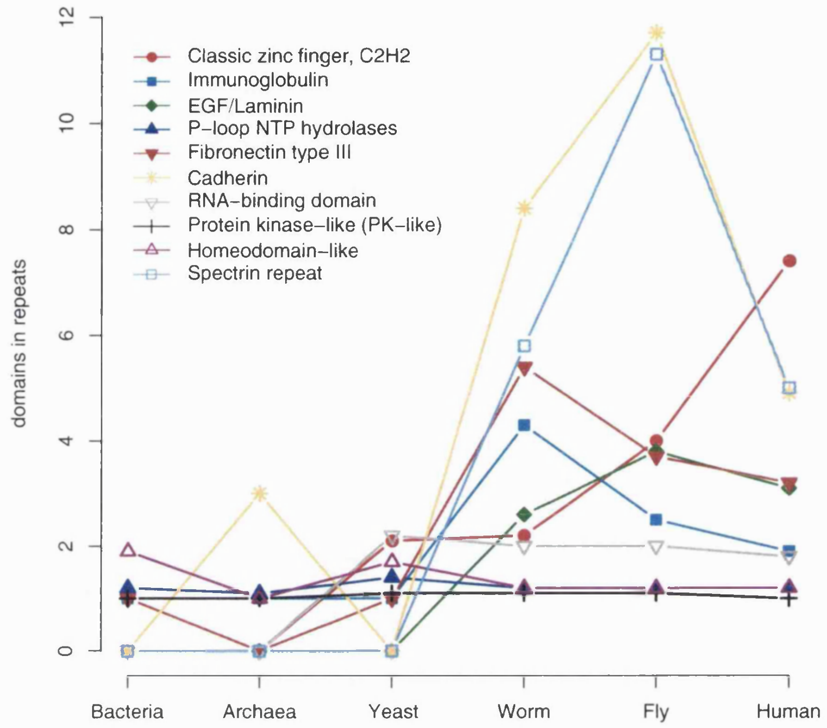


Figure 4.8: Average repetitiveness of the ten most abundant human SCOP superfamilies. For each superfamily the number of domains divided by the number of sequences this superfamily was found in is plotted for each of the each proteome.

An interesting feature is that there tends to be somewhat more domain pairings in fly compared to worm. Although the protein kinase-like superfamily is more popular in worm than in fly, and also more than in human when normalised by the number of domains in the proteome as in figure 4.7, the worm has fewer partners for this superfamily. In addition the most popular partner for the protein kinase-like superfamily in human and fly is the SH3 domain with 43 occurrences in human and 14 in fly (partner data not shown); in worm there are only seven such co-occurrences. The most popular protein kinase-like partner in worm is the adenylyl and guanylyl cyclase catalytic domain with a frequency of 24, and 5 in human. In all three metazoan proteomes the SH2 domain is a frequent partner for the protein-kinase like superfamily.

The number of partners for EGF/laminin domains decreases from worm to fly, but in human there are more partners for this superfamily than in worm. A frequent domain partner for EGF/laminin domains in worm is the c-type lectin (found 22 times) that has been mentioned above (see section 'SCOP superfamilies'), which is not a partner for EGF/laminin domains in the fly but is found as an EGF/laminin partner 25 times in human.

The immunoglobulin superfamily has more co-occurrences in fly than in worm and human. In fly this superfamily combines for example with di-copper-centre-containing domains that are also found in human (but not as a partner of immunoglobulins). Also the hemocyanin N-terminal domain, absent in human and worm, is found in combination with immunoglobulins. In fly the hemocyanin N-terminal domain, the di-copper centre-containing domain and the immunoglobulin are in fact found together in sequences that belong to the invertebrate copper containing oxygen transport proteins and larval storage proteins (InterPro family IPR000896). In human a popular partner for immunoglobulins is the MHC antigen-recognition domain which is not found at all in fly and worm. However, in human, fly and worm the fibronectin type III is the most common partner for the immunoglobulin (and vice versa) which may be the reason why these two superfamilies follow a similar trend in figures 4.6 to 4.8 (relative domain abundance and repetitiveness).

Figure 4.9b shows the top ten superfamilies in yeast. Only the tetratricoidpeptide repeat, a domain probably involved in a wide range of protein-protein interactions, expands its domain partner repertoire in a step from yeast and worm to fly and to human. The other superfamilies have similar frequencies in the three metazoans.

Figure 4.9c shows that all the popular superfamilies in bacteria have markedly fewer co-occurrence partners in archaea, although seven of these superfamilies are also found in the top ten superfamilies in archaea (data not shown). With 27 partners the Rossmann-fold, involved in a range of enzyme activities, has more partners in bacteria than in any of the other processed proteomes. However, the most frequent superfamily partners for the Rossmann-fold are similar between bacteria and metazoans (data not shown). In worm five of the popular bacterial superfamilies have an increased number of partners compared to yeast, fly and human, possibly reflecting a closer phylogenetic relationship between worm and bacteria.

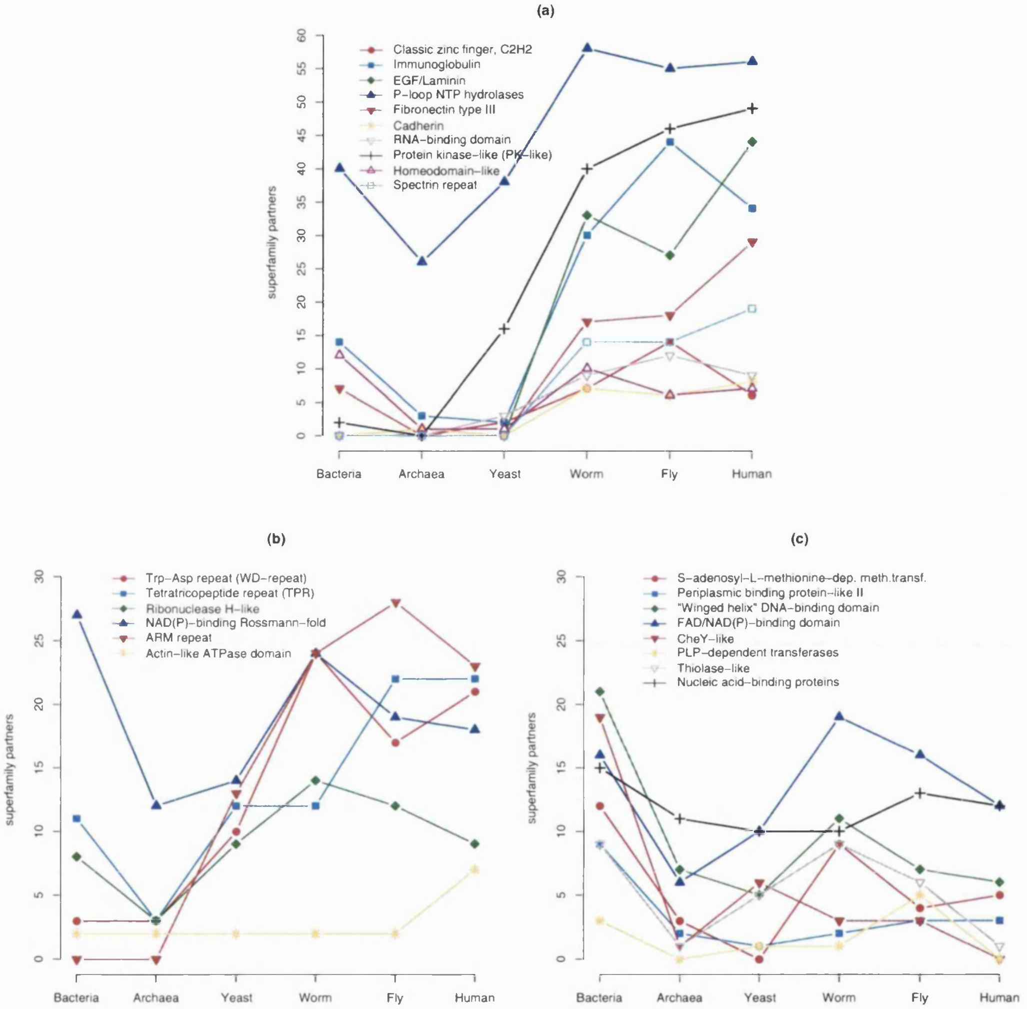


Figure 4.9: SCOP superfamily partners. The plots show the number of different SCOP superfamilies that are found together in the same sequence with a given superfamily, including the superfamily itself and irrespective of the order or the sequence space between domains. This implies that at least two domains have to be identified in a sequence. Superfamily partners for the ten most abundant superfamilies in human (a), in yeast (b) and bacteria (c) are plotted. Only those superfamilies not found within the first ten ranks in human are shown in b (P-loop, protein kinase-like, tetratricopeptide repeat and the classic zinc finger), and only those are shown in c that are not shown in a or b (P-loop and NAD(P) binding Rossmann-fold).

The plots in figure 4.9 only show the number of different superfamily partners. However even if the number of partners is similar, the actual frequencies and composition of these partnerships often shows great variation. Hegyi & Gerstein (2001) demonstrated that there is less functional conservation in multi-domain than in single-domain proteins except if they have exactly the same domain combination, so that a superfamily can have different functional contexts. This observation from Hegyi and Gerstein suggests a higher degree of functional variation than expected for a superfamily in different proteomes even if the number of domain partners is similar. For example, fifteen partners for the c-type lectin are found in human and worm, but some of the frequently found partners are different. In worm, many spermadhesin and integrin A domains are found together with c-type lectins, whereas the integrin A is not found at all as a partner for c-type lectins in human, although the overall integrin domain frequency in human is more than twice as high than in worm. In human more complement control modules (SRC domain) and immunoglobulins are found in combination with c-type lectins (the immunoglobulin is not found at all in the list of lectin partners in worm). In addition, it has been shown that in many cases of adjacent domains the domain order is an important functional aspect (Apic *et al.*, 2001; Bashton & Chothia, 2002).

In summary, the analysis described here suggests that for most superfamilies, as the organism increases in complexity, specialisation and diversity does not arise from an increasing number of domain combinations, but rather from refinement and diversification of the superfamily repertoire itself (for example, the immunoglobulins belong to a diverse superfamily with many members and possibly different functions in different proteomes) and probably by changing the repertoire of domain partners.

The web-site mentioned in the methods section provides a link to an application that allows generic ranking of selected proteomes according to selected properties such as domain frequencies, superfamily partners or domain repetitiveness of superfamilies. The results can be displayed as a table and as a plot similar to those shown in this work.

4.4.6 SCOP superfamiles in disease genes

The OMIM database (Antonarakis & McKusick, 2000) (Online Mendelian Inheritance in Man) identifies genes that have been associated with human disease. Human proteins were associated with OMIM identifiers via the *genelink* table from

ENSEMBL. 6656 different OMIM entries are linked to 5856 human proteins, indicating that a human protein can be associated with several OMIM entries. The frequency of each SCOP superfamily in the proteome assigned to disease genes versus the non-disease genes is then evaluated. 7,621 SCOP domains in 481 different superfamilies could be assigned to disease genes.

This analysis directly associates SCOP superfamilies with disease and non-disease genes. However, the cause of the disease state could be the result of one (or a combination) of effects not directly involving the protein, for example alteration of regulation or deletion of the entire gene. In addition, any point mutation or deletion within a protein may not be within a particular SCOP domain. However, for many genes in OMIM the location of the alteration (e.g. point mutation) is not known. Thus to analyse the entire OMIM database one can only gain an overview of the distribution of SCOP superfamilies between disease and non-disease genes. A more focused analysis would consider only those genes where the location of the alteration has been identified (see Sreekumar *et al.* (2001) for a review of computational analysis of disease genes).

The analysis of the superfamilies in disease genes was performed on the protein sequence level rather than on the domain level, so that only one domain per superfamily per protein sequence was counted. The aim of the analysis is to describe general trends for superfamilies and their biological function in association with disease, and therefore superfamilies with low sequence frequencies but significantly high domain frequency due to repeats, which confuse a trend analysis, were excluded. For example the extra-cellular domain of the cation-dependant mannose 6-phosphate receptor has fifteen domains in only two proteins that are associated with a disease (one domain in the small mannose 6-phosphate receptor and fourteen repeated domains in the big receptor) and only two domains in non-disease proteins. This receptor plays an important role in targeting lysosomal enzymes to the lysosome. This superfamily is strongly over-represented in the domain based analysis but not in the sequence based analysis.

The overall frequencies of SCOP superfamilies in the two sets of genes are significantly different at >99.9% confidence. Table 4.3 reports the SCOP superfamilies that are significantly over- and under-represented in the disease genes at >95% confidence as confirmed by a χ^2 test.

SCOP superfamily	R	ND	NnD	f	Description
Interleukin 8-like Chemokines (**V**)	62	36	12	3	Mainly small inducible cytokines (single domain proteins), immuno-regulatory and inflammatory processes, homoeostasis, development. Secreted proteins, activity via GPCRs.
Nuclear receptor ligand-binding domain (**M**)	56	40	15	2.67	Growth factor inducible intra-cellular steroid/thyroid receptors coupled with a DNA binding domain (glucocortocoid-receptor like) such as estrogen receptor (breast cancer associated). Transcription factors and enhancers.
Cystine-knot cytokines (**E**)	49	42	17	2.47	Growth factors belonging to TGF-b, cell determination, differentiation and growth. Neurotrophins, differentiation and function of neurons.
Periplasmic binding & protein-like I	96	21	9	2.33	Glutamate receptors, ionotropic (ion channels) and metabotropic (GPCRs with activity via a second messenger), also found in receptors involved in regulation of blood pressure.
Serpins (**M**)	76	26	12	2.17	Serine protease inhibitors of the blood clotting cascade.
4-helical cytokines (**V**)	66	32	15	2.13	Different interferons and interleukins (extra-cellular single domain proteins), regulatory in differentiation and proliferation, antiviral, immune and inflammatory response.
Winged helix DNA-binding domain	21	70	57	1.23	Associated with at least 25 disease entries. Transcription factors (activation and repression). Dominated by forkhead family members, important in embryogenesis of the nervous system in mammals, associated with different leukemia; ETS family of oncogene products; histones (chromatin remodelling) and others.
Helix-loop-helix DNA-binding domain (**E**)	28	54	45	1.2	Transcriptional control for cell type determination during development, also transcriptional control of histone acetyltransferases (preparation of chromatin for transcription).

continued on next page

continued from previous page

SCOP superfamily	R	ND	NnD	f	Description
Glucocorticoid receptor-like (DNA-binding domain) (**E**)	25	62	52	1.19	Together with nuclear receptor ligand-binding domains (see above). Frequently found in proteins of developmental genes. LIM domain proteins de-regulated in cancer cell-lines.
Homeodomain-like	8	131	142	0.92	Different homoebox proteins (transcription factors), particularly important in early embryogenesis. Some homeobox genes are oncogenes.
Protein kinase-like (PK-like)	4	246	291	0.85	About 100 different associated disease entries (e.g. different cancers). Range of kinases such as MAP or PKC (signal transduction).
RNA-binding domain	6	76	255	0.3	RNA splice factors (alternative splicing), rapid degradation of mRNAs in particular from cytokines and proto-oncogenes. Involved in e.g. spermatogenesis related to male infertility.
RING finger domain, C3HC4 (**E**)	13	43	163	0.26	Zinc-finger like domain associated with protein-protein interaction, often found in transcription regulatory proteins. Linked to e.g. apoptosis inhibitors, breast cancer gene BRACA1, acute leukemia.
Classic zinc finger, C2H2	2	135	549	0.25	Nucleic acid binding, range of transcription factors, cell proliferation and differentiation, early development, some are proto-oncogenes.
Tetratricopeptide repeat (TPR)	19	25	121	0.21	Interaction partner of regulatory proteins, subunit of G-proteins. Involved in a range of biological functions such as cell-cycle, activation of apoptosis, chromatin assembly, actin binding, cancer.
Ankyrin repeat	12	33	187	0.18	Protein-protein interaction domain. Found at least 17 different OMIM entries describing e.g. inhibitor of NFkB and cyclin-dep. kinase inhibitors, interaction with p53 in apoptosis. Co-occurrence with other interaction and regulatory domains such as DEATH and SH3.

continued on next page

continued from previous page

SCOP superfamily	R	ND	NnD	f	Description
eL30-like	58	5	45	0.11	Ribosomal protein L30, translation termination.
Pyk2-associated protein β ARF-GAP domain (**E**)	91	1	31	0.03	RIP protein that assists HIV in replication by facilitating the nuclear export of mRNA. Corresponds to the putative GTP-ase activating protein for Arf in PFAM. Non-disease proteins are often associated with PH-domains or ankyrin repeats and may have a range of biological functions.

Table 4.3: Over- and under-represented SCOP superfamilies in OMIM disease genes. For each SCOP superfamily, the rank order R of superfamily occurrences in sequences of the human proteome is given (see text for details), followed by the sequence frequency in disease genes (ND) and the frequency in non-disease genes (NnD) . The ratio (f) of these occurrences is then given as ND/NnD, the double horizontal line separates over-represented from under-represented superfamilies. Taking all SCOP domains together, the two populations (disease and non-disease) are significantly different (>99.9% confidence) as calculated by a χ^2 test. For each SCOP superfamily, the frequency ratio compared to the others was significant at > 95% confidence, after allowing for the number of SCOP domains tested (testing domains of each superfamily against all remaining domains). Bold letters in braces in the superfamily field indicate that this superfamily is specific for eukaryotes (**E**), metazoans (**M**) or vertebrates (**V**). The Description field gives an overview over the broad biological functions associated with the disease genes.

Superfamilies over-represented in proteins of disease genes are mainly associated with regulation, having biological functions in development, differentiation and proliferation, and not being directly involved in metabolism. Overall the over-represented superfamilies can be put into the following categories, immune-response, immune-regulation, growth factors and transcription factors (helix-loop-helix domains, winged helix domains, DNA-binding domain of the glucocorticoid receptors). The main biological relevance of the under-represented superfamilies may be summarised as transcription factors (homeodomain and classic zing fingers), protein-protein interaction domains involved in signalling and transcription (other than transcription factors) and translation. However, many of the superfamilies are involved in a wide range of biological functions and may be placed in more than one category, e.g. the interleukin 8-like chemokines are not only involved in immune-response but also play a regulatory role during development.

The most over-represented superfamilies (with a ratio >2) are biased towards small mainly extra-cellular single or two domain messenger proteins (interleukin, cyctine-knot cytokines and 4-helical cytokines), whereas three of the seven strongly under-represented superfamilies (with a ratio $\leq$0.3) are involved in regulation via protein-protein interaction, and another three superfamilies in are involved in transcription and translation. Further, the five most over-represented superfamilies are specific for human, metazoa or at least eukaryota, whereas in the set of under-represented superfamilies only two eukaryotic specific superfamilies are found. On the other hand, eight of the nine under-represented superfamilies are in the list of the top twenty superfamilies in human sequences, four within the top ten. None of the over-represented superfamilies is found within the top twenty ranks. The over-represented superfamily with lowest rank (highest frequency) in human is the 'winged helix' DNA-binding domain (rank 21)[1].

Taking the above observations together, the most over-represented superfamilies in disease genes are those likely to have evolved within the metazoan branch of evolution and that are moderately expanded in human (the average sequence rank is 65 of 463 ranks in total). The homeodomain-like and protein kinase-like superfamilies are just slightly but significantly under-represented, and are found with high overall frequencies in both categories. These two superfamilies are associated with biological key functions in many regulatory pathways (see table 4.3 for details). The results of the analysis of the association of SCOP superfamilies with disease genes suggest that it is in general unlikely to find abundant superfamilies with a massive bias towards disease associated proteins, possibly because the disruption of key functions may often be lethal. However, despite this general suggestion, the analysis described here does not have any explanation why certain superfamilies are over- or under-represented in disease genes. These observations may encourage future work to formulate hypotheses that may lead to deeper insight into the relationship between disease and protein folds.

4.4.7 Transmembrane proteins

Transmembrane regions in the proteomes were identified using the hidden Markov approach implemented in TMHMM-2 (Sonnhammer *et al.*, 1998). Figure 4.10 shows

[1] Note that the ranks and frequencies are based sequence frequencies rather than domain frequencies as in table 4.1

the fraction of the proteomes that were predicted to occur as membrane spanning regions. Within this work at least 13% of the human protein sequences are predicted to be membrane proteins (data not shown). However, for the human proteome only 3% of residues are predicted to be in transmembrane regions (the membrane spanning parts of the protein) which is a similar percentage as for yeast and fly but less than in worm and the average values for bacteria and archaea. The Figure also shows that 13% of the proteome consists of globular regions (regions excluding coiled-coils, low complexity regions or signal peptides) that are part of a protein chain that spans a membrane (yellow bars, 'TM/globular'). In human, only about 1% of the residues either form short loops (<30 continuous residues) linking two membrane spanning regions or appear at a chain terminus of a membrane protein. The ratio between the globular part of transmembrane proteins and the membrane spanning part is smaller in bacteria and archaea than in the four eukaryotes. This may be due to a larger fraction of proteins in bacteria and archaea that are completely membrane integral (i.e. proteins mainly built by membrane helices and connecting loops such as bacteriorhodopsin and probably those of membrane integral redox-cascades). The proteome of *C. elegans* contains both the largest fraction and the largest absolute number of transmembrane proteins (4559 membrane proteins, 28% of the proteome). The high number of transmembrane proteins is mainly due to an expansion of the family of seven helix transmembrane G-protein coupled receptors (Bargmann, 1998).

Figure 4.11 shows the ratio of residues in globular domains to residues in transmembrane regions for different membrane proteins as determined by the number of predicted membrane spanning helices. The ratios are substantially different between species for proteins with one to three transmembrane regions and become more similar as the number of transmembrane regions increases. This shows that the full sequence of transmembrane proteins with only one to three membrane-spanning regions differ in length between the proteomes of the analysed organisms reflecting a higher number of potential globular domains, with the fly having longer protein sequences for transmembrane proteins than the other organisms. In bacteria and archaea the ratio drops below one (e.g. the majority of the protein is membrane integral) at about six to seven membrane segments. In contrast eukaryotes have the majority of the residues of their proteins in potential globular domains, suggesting additional functionality such as protein-protein interaction or receptor capabilities of these membrane proteins.

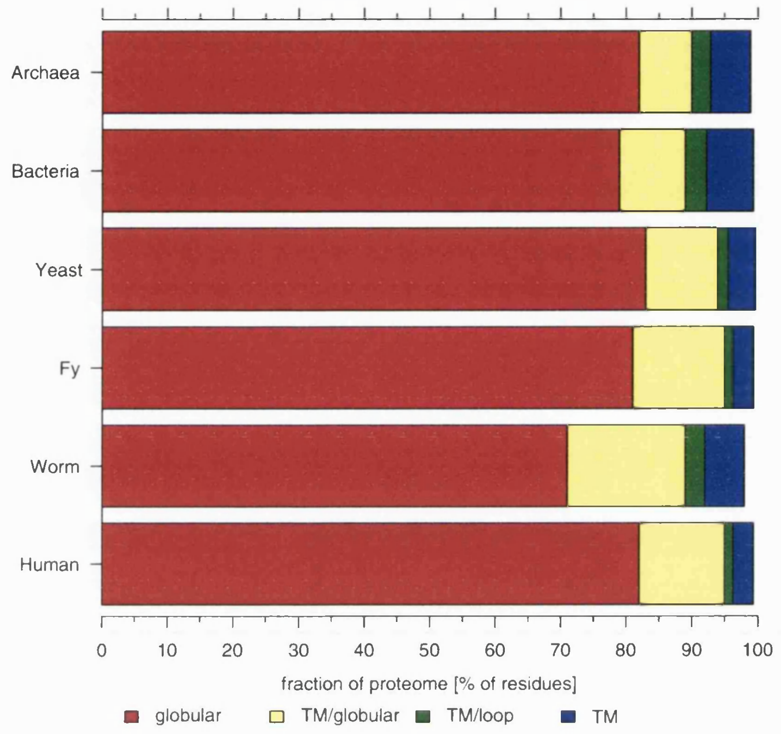


Figure 4.10: Fractions in residues of globular and non-globular parts in membrane proteins. Globular denotes globular domains in non-transmembrane proteins, TM/Globular are globular regions within membrane spanning proteins (those protein with at least one transmembrane helix domain), TM/Loop are short loops in transmembrane proteins and TM are the residues in actual membrane integral helices. See text for details.

Table 4.4 reports the frequencies of SCOP superfamilies that occur in protein chains that span the membrane. This analysis has a focus on the globular domains associated with transmembrane proteins and accordingly excludes completely membrane integral proteins of the analysed proteomes and does not consider the SCOP class of membrane proteins. The four superfamilies of highest rank are domains that can be found in cell surface proteins involved in cell-cell interaction and receptor molecules. In human, the most common SCOP domain associated with membrane-spanning chains is the immunoglobulin superfamily, whereas in fly and worm this superfamily is at rank four and five, respectively. The cadherin is the most common SCOP superfamily in fly, and in worm the EGF/laminin is the most popular membrane associated superfamily. The relative importance of superfamilies involved in

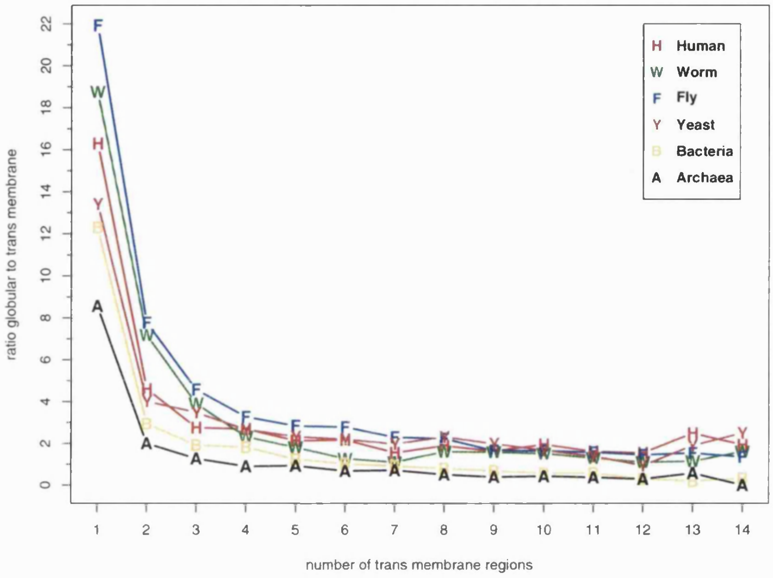


Figure 4.11: Ratio of globular regions to transmembrane regions in membrane sequences classified according to the number of transmembrane regions. The diagram only shows ratios for which at least nine transmembrane proteins were found. See text for details.

cell-cell interaction and cell surface proteins is also pointed out by the absence of these superfamilies in yeast (also see table 4.1). All eight immunoglobulin domains found in yeast are located in soluble, probably intra-cellular, proteins (no signal peptides could be found via prediction).

In conclusion, the results of the transmembrane analysis reflects the multi-cellular environment of human, fly and worm, where specialised systems for cell-cell communication and recognition are required in, for example, tissue formation.

	Human			Fly			Worm			Yeast		
SCOP superfamiliy	N	%	R	N	%	R	N	%	R	N	%	R
Immunoglobulin	463	38	1	126	26	4	74	16	5	-	-	-
Cadherin	440	72	2	206	93	1	114	84	2	-	-	-
Fibronectin type III	359	43	3	134	54	3	66	30	7	-	-	-
EGF/Laminin	216	18	4	139	43	2	163	39	1	-	-	-
Ligand-binding domain of low-density lipoprotein receptor	126	51	5	106	54	5	79	55	4	-	-	-

continued on next page

continued from previous page

	Human			Fly			Worm			Yeast		
SCOP superfamiliy	N	%	R	N	%	R	N	%	R	N	%	R
P-loop containing nucleotide triphosphate hydrolases	87	10	6	89	15	6	91	18	3	41	10	1
Protein kinase-like (PK-like)	65	12	7	27	10	12	72	17	6	-	-	-
Complement control module/SCR domain	56	20	8	25	44	13	3	6	65	-	-	-
C-type lectin-like	53	36	9	3	8	54	34	11	8	-	-	-
MHC antigen-recognition domain	47	82	10	-	-	-	-	-	-	-	-	-
TNF receptor-like	38	73	11	2	100	67	-	-	-	-	-	-
RNI-like	34	35	12	31	35	8	14	38	23	-	-	-
Serine proterase inhibitors	32	25	13	17	41	19	18	21	19	-	-	-
Periplasmic binding protein-like I	28	93	14	16	73	22	30	88	11	-	-	-
ConA-like lectins/glucanases	27	20	15	28	42	10	27	26	12	5	63	10
RING finger domain, C3HC4	25	12	16	17	16	19	20	16	18	5	15	10
L domain-like	25	21	16	25	26	13	23	16	15	1	8	38
Spermadhesin, CUB domain	24	19	18	42	50	7	23	13	15	-	-	-
(Phosphotyrosine protein) phosphatases II	23	21	19	7	17	30	14	14	23	-	-	-
EF-hand	23	8	19	15	9	24	10	8	29	-	-	-
Metalloproteases ('zincins'), catalytic domain	22	33	21	4	15	43	8	16	37	-	-	-
POZ domain	22	18	21	5	5	37	22	15	17	-	-	-
C2 domain (Calcium/lipid-binding domain, CaLB)	21	11	23	17	25	19	32	36	10	16	50	2
Ankyrin repeat	21	8	23	18	15	18	34	27	8	5	16	10
Extracytoplasmic domain of cation-dependent mannose 6-phosphate receptor	15	88	32	-	-	-	-	-	-	1	100	38
SpoIIaa	5	83	63	4	100	43	5	100	53	2	100	25
Adenylyl and guanylyl cyclase catalytic domain	16	67	29	28	76	10	26	70	13	-	-	-

continued on next page

continued from previous page

	Human			Fly			Worm			Yeast		
SCOP superfamiliy	N	%	R	N	%	R	N	%	R	N	%	R
Blood coagulation inhibitor (disintegrin)	18	67	26	3	43	54	4	67	58	-	-	-
Periplasmic binding protein-like II	16	62	29	30	77	9	12	92	25	-	-	-
Syntaxin 1A N-terminal domain	8	62	47	5	56	37	8	62	37	7	88	5
L-2-Haloacid dehalogenase	11	61	34	2	10	67	5	28	53	2	13	25
Snake toxin-like	5	56	63	2	100	67	1	50	98	-	-	-
Metal-binding domain	6	55	58	4	80	43	4	80	58	5	71	10
Transferrin receptor ectodomain, apical domain	7	54	53	-	-	-	2	50	79	3	75	20

Table 4.4: SCOP superfamilies associated with transmembrane proteins. The table gives the number (N) of domains in each superfamily that are found in sequences that have a transmembrane section. The list of superfamilies is ordered by the most abundant superfamilies in human membrane sequences. The '%' is percentage of the total occurrence of each superfamily in the proteome (the total is the sum of domains in a superfamily in transmembrane and non-transmembrane chains, this is the same as in table 4.1). R denotes the rank of N. The lower part of the table (separated by a double horizontal line) details superfamilies with highest percentages in membrane proteins and with a frequency of at least five domains in human that are not reported in the upper part.

Table 4.4 also presents the fraction of the total domain frequency for each superfamily that is associated with membrane spanning chains. Of the superfamilies with at least five domains in transmembrane proteins, only the MHC antigen-recognition domain and the periplasmic binding protein-like I have more than 80% of their representative domains in transmembrane proteins. Further down the list (bottom part of table 4.4), several other superfamilies are found with more than 50% of their domains in transmembrane proteins. However, in worm all six scavenger receptor cystein-rich domains (not shown in table 4.4) are found in membrane glycoproteins, and all five spoIIa domains (involved in sulphate transports) are found in membrane proteins.

SCOP superfamilies that are frequently associated with transmembrane regions are also common in chains that do not span the membrane. This supports the view that domains are mobile elements that are not restricted to co-evolve either always in association with a transmembrane section or always in a chain that does not span the membrane.

The top ranking superfamilies in bacteria are different from those found in eukaryotes (table 4.5). These superfamilies are mainly associated with bacterial signalling (ATPase domain and homodimeric domain of signal transduction histidine kinase, PYP-like sensor domain, CBS-domain) or with small molecule binding probably as membrane bound receptors or enzymes (P-loop containing nucelotide hydrolases, nucleotide-diphospho-sugar transferases of glycosyltransferases, NAD(P)-binding Rossmann-fold, L-2-Haloacid dehalogenase of heavy metal transporters). In bacteria no globular superfamily with more than two representatives (an average over the seven processed bacterial proteomes) could be identified that is exclusively found in membrane proteins. The list of the most popular superfamilies found in transmembrane proteins for archaea is similar to those for bacteria (data not shown), but the frequencies of which domains are found are much lower, e.g. the top ranking superfamily is the P-loop with only eight domains in the three archaea proteomes. In addition, domains that may belong to the immunoglobulin (three domains in *P. horikoschii*) and the cadherin (three domains in *M. jannaschii*) superfamilies were found in two archaea sequences.

	Metazoa			Yeast			Bacteria			Archaea		
SCOP superfamiliy	N	%	R	N	%	R	N	%	R	N	%	R
ATPase domain of HSP90 chaperone/DNA topoisomerase II/histidine kinase	-	-	-	-	-	-	13	59	1	-	-	-
P-loop containing nucleotide triphosphate hydrolases	89	14	6	41	10	1	11	7	2	3	2	1
Homodimeric domain of signal transducing histidine kinase	-	-	-	2	67	25	9	64	3	-	-	-
PYP-like sensor domain	2	8	88	-	-	-	7	41	4	-	-	-
CBS-domain	8	44	38	2	20	25	5	42	5	1	4	4
Nucleotide-diphospho-sugar transferases	6	29	48	4	80	16	3	43	6	2	40	2
NAD(P)-binding Rossmann-fold domains	13	9	29	2	2	25	3	4	8	1	4	4
L-2-Haloacid dehalogenase	6	32	47	2	13	25	3	33	7	-	-	15

Table 4.5: SCOP superfamilies associated with transmembrane proteins in bacteria. The table is ordered by the most abundant superfamilies in bacterial membrane proteins (with at least three domains associated with membrane proteins). Averages are given for Metazoa (human, fly and worm), the processed bacterial and archaea proteomes. Otherwise the legend for table 4.4 applies.

Figure 4.12 shows the frequencies of the overall top ten human superfamilies (the same superfamilies as in figure 4.6) with the number of domains in membrane proteins compared to the other processed proteomes (4.12a) and the same for the top ranking bacterial superfamilies (4.12b, the P-loop is not shown as it is already shown in 4.12a). As expected, the immunoglobulin, cadherin, fibronectin and EGF/laminin are most expanded in human compared to fly and worm. Interestingly the P-loop is found with very similar absolute numbers in membrane proteins in all metazoan proteomes, compared to the overall expansion shown in figures 4.6. This suggests that, although there are more P-loops in human than in fly and worm, the additional duplications are associated with soluble proteins only.

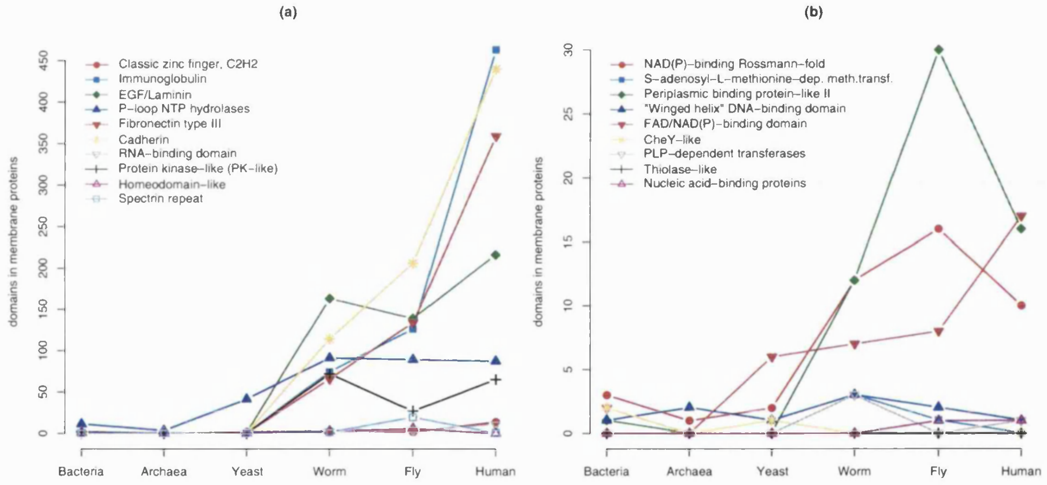


Figure 4.12: Expansion of SCOP superfamilies in membrane proteins. The number of domains in a superfamily that are found in proteins that have at least one transmembrane helix is shown for the different proteomes. The ten overall most abundant superfamilies in human (a), as in figure 4.6, and bacteria (b) are plotted. For better scaling the P-loop is excluded from b as it is already shown in a.

The top ranking superfamilies in bacteria (figure 4.12b) are rarely associated with membrane proteins in prokaryotes and yeast, and this trend also remains across the metazoans for six of the ten superfamilies (no Chey-like domains could be identified in human). Note that the total numbers in 4.12b are much lower than in figure 4.12a. Only one periplasmic binding protein-like II domain is found on average in membrane proteins in bacteria, and although the total number of domains in this superfamily is higher than for the other proteomes (data not shown), membrane association has only been expanded in metazoa. However, the periplasmic binding protein-like II is a diverse superfamily that contains at least ten different PFAM families, and in bacteria there seem to be many soluble extra-cellular members of this superfamily (suggested by signal peptide prediction). Most of the metazoan domains of this superfamily are associated with ligand-gated ion channel proteins and receptor family ligand binding proteins, and both of these families are membrane proteins. In yeast four of the five domains of this superfamily are part of presumably intra-cellular soluble proteins involved in pyrimidine biosynthesis. The divergence of the periplasmic binding protein-like II superfamily to produce different functional families in bacteria and metazoa seems to be coupled to some extent with different sub-cellular locations (soluble and membrane bound).

4.5 Concluding remarks

This work describes an integrated analysis of the human proteome and compared the results to those of other proteomes. The key aspect of this study is the integration in the context of the different species of the following features: the extent and reliability of structural and functional annotations of the proteomes; the extent of domain duplication; change and expansion of the structural superfamily repertoire between different proteomes; the relationship between human disease genes and structural superfamilies; and the relationship between transmembrane proteins and their globular regions. The study included a structure based analysis from which it was possible to make evolutionary insights that could not be obtained from sequence based methods alone.

These general bioinformatics analyses require simplifications and are also subject to errors in the predictive methods. In particular, a simplified strategy to estimate the extent to which there is some functional information derivable by homology had to be employed. However, this reflects the standard practice in obtaining an initial suggestion of protein function in the absence of characterised motifs as found in PFAM. Automated proteome annotation, particularly in eukaryotes, is complex and the exact numbers reported here will need to be refined as the bioinformatics tools improve and more experimental data becomes available.

This study and related work by others (e.g. Frishman *et al.* (2001); Iliopoulos *et al.* (2001); Koonin *et al.* (2000)) have highlighted the extent to which we still need structural information as a step towards understanding the function and evolution of the human and other proteomes. The experimental determination of the protein structures of these proteomes is the goal of structural genomics initiatives. Sander and coworkers have suggested that within 10 years we can have representatives of most protein families (Vitkup *et al.*, 2001). However, today some structural information for about 40% of the human proteome is available that can be used to provide functional insights.

4.6 Methods

The analysis described in this chapter is based on the 3D-GENOMICS system that was developed during this work (see chapter 3). This section describes the programs, parameters and special rules used for the processing.

4.6.1 Protein sequences from complete genomes

Eukaryota: *Saccharomyces cerevisiae* (No authors listed, 1997), *Caenorhabditis elegans* (The C. elegans Sequencing Consortium, 1998), *Drosophila melanogaster* (Adams *et al.*, 2000), *Homo sapiens* (Lander *et al.*, 2001). **Bacteria:** *Mycobacterium tuberculosis* (Cole *et al.*, 1998), *Escherichia coli* (Blattner *et al.*, 1997), *Bacillus subtilis* (Kunst *et al.*, 1997) , *Mycoplasma genitalium* (Fraser *et al.*, 1995), *Helicobacter pylori* (Tomb *et al.*, 1997), *Aquifex aeolicus* (Deckert *et al.*, 1998), *Vibrio cholerae* (Heidelberg *et al.*, 2000). **Archaea:** *Aeropyrum pernix* (Kawarabayasi *et al.*, 1999), *Pyrococcus horikoshii citep* (Kawarabayasi *et al.*, 1998), *Methanococcus jannaschii* (Bult *et al.*, 1996). See table 1.1 for the size of each of the genomes. The *H. sapiens* proteome is the ENSEMBL-0.8.0 confirmed peptide data set (http://www.-ensembl.org). Other sequences were taken from the NCBI (ftp://ftp.ncbi.nlm.nih.-gov/genomes/). See also table 3.2.

4.6.2 Sequence analysis

Sequences, annotations and results are stored in a relational database (MySQL, http://www.mysql.com), which serves as the back-end for an automated processing pipeline running on a Linux computer farm. The software and database system developed within this work allows for updates of the data and results as well as comparisons across proteomes. See section 3 for details.

The sequences were first scanned for: signal peptides (SignalP-1); transmembrane helices (THMM-2); coiled-coils (Coils2); low complexity regions (SEG); and repeats (Prospero V1.3). See table 3.1 for web resources (URLs) and references. The default parameters were used.

Protein sequence database searches were performed using PSI-BLAST version 2.0.14 (Altschul *et al.*, 1997), based on the experience from the work described in chapter 2. Sequences were masked for low complexity regions, transmembrane regions, coiled-coils and repeats. The h-value and e-value cut-offs both were 5×10^{-4} (the h-value is the e-value cut-off for sequences to be included in the next PSSM), and the maximum number of iterations was 20. The sequence database used contained 634,179 different protein sequences from the NCBI NRPROT (all non-redundant GenBank CDS translations, PDB, SwissProt and PIR, the protein sequences of the genomes processed in this work and the sequences from the SCOP-1.53 database). SCOP sequences were taken from the ASTRAL database, a supplement for SCOP, Chandonia *et al.* (2002), see section 1.2.3 for a description of these databases. Low complexity regions of sequences from this database were masked by 'X' (the 'X' character is ignore by the sequence comparison programs).

It has been shown (Park *et al.*, 1998) that PSI-BLAST detects relationships that are not symmetric, i.e. a query with sequence A might not have a significant match to B whilst searching with B could have a significant match to A. To address this problem, each SCOP sequence was run against the protein sequence database via PSI-BLAST to construct a position specific scoring matrix (PSSM) that was used with the IMPALA program (Schaffer *et al.*, 1999) to assign SCOP domains to each of the genome sequences. This procedure increases the sensitivity without introducing many new false positives (this was confirmed by manual investigation of SCOP domain assignments). The e-value cut-off for IMPALA was 5×10^{-3} (this cut-off is higher than for PSI-BLAST because of a different scoring scheme, see sections 1.3.5 and 1.3.6 for details).

In addition, for all sequences BLAST was run against a sequence database that contains only the SCOP sequences to ensure that close homologues not identified by PSI-BLAST because of the masking described above are found by BLAST. Query sequences were not masked (not even for low complexity regions).

BLAST (Altschul *et al.*, 1997) was run for those sequences that contain a transmembrane region, coiled-coil region or a repeat but without removing (masking) these regions. Only low complexity regions were masked. This ensures that at least close homologues of membrane integral proteins, coiled-coils and proteins that mainly consist of repeats, are identified. These close homologues may not be detected by PSI-BLAST because there may not be enough valid residue signal left after the masking. The masking, as described above, is necessary for PSI-BLAST to avoid the corruption of the PSI-BLAST PSSM and the aggregation of false positive alignments. Repeats were masked for PSI-BLAST runs because these tend to increase the number of significant HSPs (alignments) dramatically without providing much additional information (a protein A with three domains of the same family could in theory produce 3^2 alignments with another protein B that contains three homologous domains of the same family). For PSI-BLAST and BLAST the same database was used. The e-value cut-off was 5×10^{-4}.

Examination of initial results from this work showed that there was a problem in PSI-BLAST detecting very short SCOP domains (less than 50 residues) because BLAST/PSI-BLAST e-values may not be significant for short alignments, yet manual investigation of the region strongly suggested that it should be assigned to a particular SCOP domain (for example by a PROSITE pattern). Within this work a heuristic method was developed to address this problem: An assignment to a SCOP domain was accepted if the e-value is <10 for an IMPALA or BLAST hit and <1.0 for a PSI-BLAST hit and if the domain is shorter than 50 residues and the sequence identity of the alignment satisfies the identity cut-off described by Rost (1999). This identity cut-off requires a much higher sequence identity for shorter than for longer alignments (see also equation 2.1 in chapter 2). Overall, this procedure weights sequence identity more than e-values for alignments between short domains. If the identity condition was not satisfied, a SCOP domain was still accepted if the alignment shares a common PROSITE pattern (Falquet *et al.*, 2002) between query and subject.

All accepted SCOP domains must be present with at least 65% of their domain in the alignment, to avoid partial domain assignments that are in many cases false positives. The analysis described in chapter 2 showed that a 50% coverage of SCOP domains is a sensible choice to avoid false positive alignments while maintaining a relatively high coverage of true positives. However, manual investigation of a subset of alignments between protein sequences from the analysed proteomes and SCOP domains showed that many of these alignments that represent just a fraction of the actual domain are likely to be false positives. To find a sensible cut-off for the fraction of a SCOP domain that has to be present in the alignment, the highest scoring alignments (those with the lowest e-value) were taken from each query region of the proteomes (see below for a definition of the term *region*) to analyse the distribution of the fractions represented by the alignments.

Figure 4.13a shows that most of the alignments between protein sequences from the proteomes and SCOP domains represent between 90% and 105% of the SCOP domains (insertions may contribute to a coverage $>$ 100%). The dataset shown in blue in figure 4.13 shows the distribution of alignments between SCOP domains as queries and as subjects, and is shown to validate the analysis of alignment fractions. The number of alignments start to increase at about 65% (the domain length fraction present in the alignment) in both distributions. However, the proteome dataset shows a smoother increase in the number of alignments between 10% and 80% domain length fraction (i.e. there are more of these alignments than in the SCOP/SCOP). This may be because the SCOP single domain sequences are not good approximation of the real situation for protein sequences from the proteomes which are often multi-domain proteins. An alternative explanation is that some domains may be more flexible in length with only a conserved core that comprises on average 65% of the existing SCOP domains. Also, some of the domain definitions in SCOP may be wrong, when considering a huge and diverse protein dataset as in this analysis. In addition, wrong gene predictions may account for truncated domain alignments.

However, the assumption is that the distributions of alignment fractions are comparable, and the SCOP single domain dataset is to some extent representative for the protein sequences from the genomes. Figure 4.13b analyses the distribution of the domain fractions of false positive alignments between SCOP domains as the ratio between false positive and true positive domain alignments. This is basically a simplified version of the benchmark described in chapter 2. True positive alignments are alignments between domains of the same superfamily. Between 60% and 70% domain fraction the number of false positive alignments decreases. The assumption is that the SCOP domain assignments in the proteomes have a similar distribution of false positives to the SCOP/SCOP benchmark in figure 4.13b. Alignments that represent less than 20% or more than 105% of their domain are most likely to be false positives (the smallest fraction is 6% and the biggest fraction is 173%, only the fractions between 10% and 110% are shown).

The above analysis leads to the choice of a 65% cut-off for the domain fraction to accept assignments to SCOP domains. On average this reduces the total number of SCOP regions assigned to the proteomes by about 10%. However, many of these alignments can be considered tentative, and are often without any supporting evidence by PFAM domains and/or PROSITE patterns. In this study it is critical to avoid false positive domain assignments. The SCOP domain partner analysis described in section 4.4.5 (figure 4.9) is especially prone to errors, because a domain partnership requires only one observation per superfamily, so that a single false domain assignment would bias the analysis of domain co-occurrence.

The analysis of domain fractions does not distinguish between superfamilies. Further detailed analysis considering specific superfamily cut-offs and domain length variability within a superfamily may lead to a better discrimination between true and false positive alignments and a good description of the domain core. However, in this study only the very simple approach of treating all domains and superfamilies as a whole was considered, for the purpose of choosing a fraction cut-off that lies outside the main population of domain fraction. Nevertheless, some true positive alignments may be missed due to the 65% cut-off.

It should be noted that residue based calculations rely on the accuracy of the sequence comparison heuristics that were employed. For the BLAST (and derivatives) based assignments this means that ends of domains may not be correctly identified during the extension step of the algorithm. Also potential inter-domain regions were not considered, so that even in theory 100% residue based assignment may not be reached. This affects the results represented in the bar-plots shown this chapter. However, this is a systematic error on the algorithm level of the employed methods, and one has to assume that this affects the results of all the processed sequences equally, so that as a first approximation a comparison of residue based fractions is

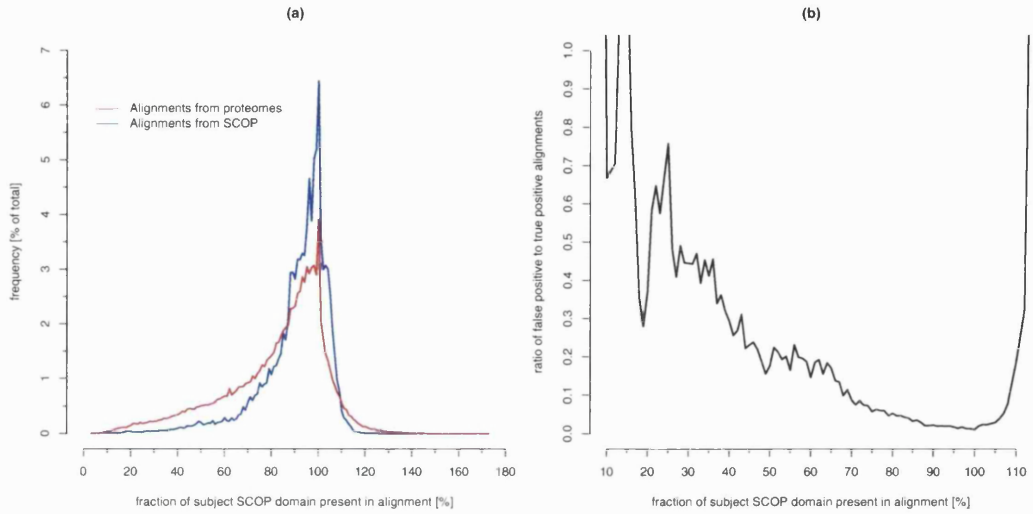


Figure 4.13: Fractions of SCOP domains present in alignments generated by PSI-BLAST. (a) Shows the distribution of the fractions of SCOP domains in alignments between proteins from the processed genomes (the queries) and SCOP domains (the subject); and SCOP domains as queries and SCOP domains as subjects (blue dataset). (b) Shows the distribution of the different domain fractions of false positive SCOP/SCOP domain alignments as the ratio of false positive alignments to true positive alignments. Alignments between domains of different superfamilies are considered to be false positives. Alignments between identical sequences are ignored. Insertions in the alignment are counted and may give fractions bigger than 100%. See text for details.

still valid.

As described in section 3.5 the 3D-GENOMICS system (chapter 3) clusters alignments within the same region of a query sequence. These clusters are referred to as *regions*. For reasons of data retrieval performance, alignments produced by BLAST, PSI-BLAST and IMPALA are clustered as described in section 3.5, and only the representative sequence for a region (the one with the lowest e-value) is taken for the annotation described in this chapter. For SCOP domains the criteria to be allowed to enter the region clustering is described above. All SCOP domains of the same cluster overlap by at least 50% (with respect to the shorter domain). All other sequence types described in section 3.5 on page 98 have to be at least 50 residues long or must represent 50% of their sequence to be accepted for the clustering. These regions are single linkage clusters, and sequences only have to overlap by one residue (the main purpose of these regions is to reduce the amount of data).

PFAM domains were assigned via HMMer (Eddy, 1998) and the PFAM hidden Markov model library version 6.2. The e-value cut-off to accept a hit was 0.1 and a domain had to be present in the reported alignment with at least 75% of its entire length.

For the analysis of transmembrane proteins, sequences were truncated if the SignalP program (Nielsen *et al.*, 1997) could identify a potential signal peptide. This avoids false positive predictions of transmembrane regions at the N-terminus of a sequence.

4.6.3 Availability of annotation

The results of the analysis are available as 3D-GENOMICS via the web at http://-www.sbg.bio.ic.ac.uk. This includes query forms for database searches and the display of tables and alignments. The web-site provides a special section with results from comparative analyses, including an application to list different domain properties such as repetitiveness, association with transmembrane proteins or domain partners ranked by frequency in a selected 'master' proteome.

Chapter 5

Summary, Conclusions and Outlook

This chapter summarises the work described in the previous chapters. Problems, limitations and possible future developments are discussed.

5.1 Summary and conclusions

This thesis described the development of an automated system for the structural and functional annotation of proteomes and its application to fourteen proteomes including the proteins from the human genome. The main parts of this work are summarised and briefly discussed below.

5.1.1 Benchmarking PSI-BLAST in genome annotation

An important step in structural and functional annotation of proteins is the identification of homologous proteins of already known structure and/or function. In chapter 2 the performance of the commonly used sequence comparison method PSI-BLAST (Altschul *et al.*, 1997) for the structural and functional annotation of proteins of completely sequenced genomes was evaluated.

In previous work by others (e.g. Park *et al.* (1998)) the performance of sequence comparison methods was evaluated based on the assumption that a perfect comparison method is able to identify all homologues of a query protein (in a one-to-one relationship, i.e. all pairwise relationships should be identified). This one-to-one procedure describes the overall performance of a method and may be used to compare different methods. However, for the functional and structural annotation of genomes only one homologue has to be identified to transfer the information from the homologue to the un-annotated query sequence (this is a one-to-many relationship, i.e. many homologues provide the same information that is used to annotate a query sequence). If several homologues can be identified these can be used as supporting evidence for the annotation. This means that previous benchmarks underestimated the performance of sequence comparison methods in genome annotation.

In this work the success rate based one the one-to-many relationship was evaluated for the PSI-BLAST method. An artificial query proteome assembled from SCOP domains (Murzin *et al.*, 1995) and a database of remotely related SCOP domains serving as targets were constructed. The homologous relationships between SCOP domains based are known. For the benchmark the superfamily level was considered. The benchmark also takes into account the multi-domain character of proteins, and the performance is evaluated on the domain level.

With the assumption that close homologues relationships can easily be identified, the benchmark concentrates on the identification of a remote homologues only. For about 40% of the domains of the SCOP test proteome the correct superfamily can be assigned via a remote homologue of the test SCOP database. This coverage is about three times as much as for a one-to-one based approach. Only 1% of the assignments are wrong (where the superfamily of the query is different from the superfamily of the alignment subject). The sources of common errors were identified. A set of sensible parameters for PSI-BLAST was extracted to minimise the number of false assignments (error rate) and to maximise the number of true assignments (coverage).

The proteins from two completely sequenced genomes (*M. genitalium* and *M. tuberculosis*) were analysed in terms of their homology to SCOP domains and proteins of known function using PSI-BLAST with the evaluated set of parameters. From the success rate of the benchmark the expected fraction of the proteomes with new folds and function was calculated.

The work carried out in chapter 2 demonstrated the importance of systematic evaluation of the performance of the sequence comparison methods to highlight limitations and to estimate the extent of what is still unknown. The evaluation described in this work is different from the *classical* approach (one-to-one versus one-to-many relationship) and shows markable differences in the results. This work also highlights the importance of structural information via structural classification of proteins, that is necessary to identify homologous relationships in the absence of detectable sequence homology.

5.1.2 3D-GENOMICS: A proteome annotation pipeline

Based on the experience of the benchmark described in chapter2, a system for automated large scale structural and functional annotation of proteins from completely sequenced genomes was developed to provide a research platform for comparative proteome analysis. The analysis of the two genomes described in chapter 2 demonstrated the requirements for an automated analysis pipeline that is able to processes large amounts of sequence data, to store the results and to allow for further analysis of these results such as cross comparisons between genomes.

Chapter 3 describes a software and database system to analyse protein sequence data and to manage the result from different analyses. The developed system is able to manage different versions of data and can be, to some extent, updated. An important feature of the 3D-GENOMICS system is the decomposition of the output from an analysis software into several descriptive fields. For example PSI-BLAST output is not stored as a single raw text field, instead the informative parts of the output such as hits (homologues sequences), e-values, scores, sequence identities and alignments are extracted and stored as indexed fields in the 3D-GENOMICS database. Relational queries can then be performed on these data-fields, allowing to link and relate results from different analyses.

The database is encapsulated by an object oriented software interface that manages the data stored in the database as well as performing sequence and proteome based analysis (for example running PSI-BLAST for a sequence). Analysis objects have special properties that allow the distribution of these objects over a computer farm for parallel processing. The software interface also allows transparent access to the database without requiring the user to know the structure of the underlying database.

The developed system is generic and allows to integrate new analysis methods and source data. The system has been used for different projects carried out by other members of the group. These projects include an analysis of enzymatic pathways, analysis of protein-protein interaction and an analysis of protein function via automated processing of the scientific literature. Several web based applications allow users to query the database, to export data and to perform analyses such as comparing distributions of SCOP domains between proteomes.

Existing annotation systems developed by others may serve a similar purpose. However, research such as the large scale comparative analyses of proteomes as described in this thesis require an open and expandable architecture to allow for easy integration of new methods and data as well as for the distribution of the analyses for parallel processing. The integration of a processing pipeline capable for large scale processing as an open architecture together with the decomposition of the results for storage and relational retrieval was not provided by the existing systems at the time this project was started (1999).

The 3D-GENOMICS system was applied for a comparative analysis of proteomes described in chapter 4 and is summarised below.

5.1.3 Structural Characterisation of the Human Proteome

Chapter 4 described the extent of structural and functional annotation of fourteen proteomes including the human proteome. In particular the distribution of SCOP superfamilies (Murzin *et al.*, 1995) across proteomes was analysed.

For about 40% of the human proteome homologues of known structure could be identified, this is comparable with the structural annotation for most prokaryotes but is more than for the other eukaryotes that were analysed in this work. For about 13% of the human proteome a homologue of known structure was identified where the sequence alignments provide sufficient sequence identify for reliable homology modelling. For about 40% of the human proteome reliable functional annotation can be obtained via homology to an already annotated proteins.

From the analysis of domains in SCOP superfamilies within the processed proteomes the extent of domain duplication was calculated (all domains within the same superfamily are assumed to be homologues and are therefore the result of duplication events of a common ancestor). About 98% of the domains in the human proteome is estimated to have arisen via domain duplication, compared to only 55% of the smallest organism that was analysed (*M. genitalium*).

The extent of domain duplication was further analysed. Superfamilies expanded in the human and other proteomes were identified and compared. Several superfamilies were found that are abundant in metazoans only, these are dominated by cell-surface proteins. The results suggest that more superfamilies were invented during evolution between yeast and metazoans than between prokaryotes and yeast.

Combinations of co-occurring SCOP superfamilies within the same protein sequence were analysed and compared between proteomes showing that the number of superfamily partners generally remains stable between proteomes. Nevertheless, the composition of the set of partners for a given superfamily differs between proteomes. In addition the organisation of domains in repeats may play an important role in the development from single- to multi-cellular life.

The distribution of SCOP superfamilies associated with inherited disease in human was analysed. Superfamilies significantly over-represented and under-represented in proteins of disease genes were identified. Those superfamilies that are over-represented in disease genes are dominated by rare eukaryotic, metazoan or even vertebrate specific superfamilies compared to more abundant superfamilies that are generally under-represented in disease genes.

In some proteomes nearly 30% of the proteins are predicted to be membrane proteins. However, only a small fraction of membrane proteins are completely membrane integral (i.e. with no globular domains inside or outside the cell), and most of the residues in membrane proteins are in fact found in globular domains. The distribution of SCOP superfamilies in membrane proteins was analysed, showing that most SCOP domains are mobile elements that are associated with both types of sub-cellular location: soluble and membrane standing. Metazoan proteomes show greater expansion of their abundant superfamilies in membrane proteins compared to the abundant superfamilies in prokaryotes for which membrane association is rather rare.

5.2 Outlook

The scientific and technical work carried out in this work may be subject to more detailed and specific future analyses. Bioinformatics research is mainly driven by the available data such as protein sequences, structures or expression data. New technologies provide new data sources and usually trigger the development of new methods to analyse these new data types. An important aspect in bioinformatics will therefore be the integration of these data types and associated methods to discover parameters and rules that ideally lead to the successful simulation of complex biological processes. On a small scale this work gathers the basic requirements to understand complex biological processes. However, the work described here is limited by concentrating on protein sequences and structures.

Between 1998 and 1999 when the number of fully sequenced genomes started to increased due to the establishment of automated large scale sequencing technologies, genome annotation became an important aspect. The rigorous evaluation of automated annotation such as described in chapter 2 was a requirement to show limits and expectations as well as leading to enhancements of methodologies.

The processing of eukaryotic proteomes using PSI-BLAST described in chapter 4 highlighted additional problems such as the existence of repeats which often lead to an explosion of the resulting data (the number of significant alignments reported by PSI-BLAST). Short domains often remain undetected because alignments do not produce significant scores due to insufficient length. These short domains, also often found in repeats, are frequently found in eukaryotic proteomes. These additional problems were undetected by the benchmark described in chapter 2, and only the extensive processing of the eukaryotic proteomes highlighted these problems. Therefore some parameters for the protein processing described in chapter 4 were re-evaluated and adjusted, and new rules were added.

The additional experience for large scale protein annotation gathered during the analysis of the eukaryotic proteomes showed that additional benchmarking of proteome annotation is required taking into accounts the enormous problems within eukaryotic genomes (some of the origins of problems are associated directly with the genome such as gene prediction). The 3D-GENOMICS architecture can be used for a continuous benchmark, because different versions of an analysis can be managed and compared.

Different processing pipelines and information retrieval systems such as 3D-GENOMICS to perform a fully automated annotation of sequences were developed (see section 3.8.3). It is now important to extend these systems to integrate different sources of information such as expression profiles, protein-protein interaction networks, pathways and protein structures to discover complex relationships between these biological entities. An important step to integrate several heterogenous protein sequence-, domain- and motif databases was the development of InterPro (Apweiler *et al.*, 2001).

The 3D-GENOMICS system will have to be adjusted to cope with extensive data integration. However, it will be generally difficult to gather the required expert knowledge and resources for extensive data integration. Therefore it may be more feasible to connect different domains of expertise (i.g. specialised databases and analysis software) via specialised and distributed warehouses, each maintained by a specialised research group. To guarantee transparent queries to relate biological entities located in different warehouses hosted at different sites, communication standards and protocols have to be developed. The DAS project (Dowell *et al.*, 2001) and XML in general are promising steps towards distributed data management. Nevertheless, biological data integration goes beyond linking annotations from different sources in the users web-browser (see Stein (2002) for a recent commentary on web based bioinformatics resources). Such a system has to be fully open (i.e. the source code must be available) as well as allowing for large scale queries. There will be many technical challenges such as version management (e.g. managing different revised versions of a genome taking into account dependencies of the downstream analyses).

The analysis described in chapter 4 is a *top-down* approach to classify and compare proteomes. Based on SCOP superfamilies the comparison of the protein domain repertoire of different proteomes includes very distant relationships and provides a rather general view. On the superfamily level it is difficult to perform functional comparisons. It is now important to choose a finer granularity for the analysis of protein function by identifying families and sub-families within a superfamily. Functional specificities may be encoded by just a few different residues between highly related sequence families. For example, this work showed that there are more nuclear receptor-binding domains in *C. elegans* than in human. However, different functional families have been expanded in worm compared to human (these data were not shown or discussed in chapter 4 because they are beyond the scope of this work).

The functional context of these families and sub-families (for example the pathways these proteins and domains are found in) will show the extent of functional flexibility of a superfamily and will provide evolutionary insights into the structure-function relationship.

In the past the collection of experimental data was often the bottleneck in biological research. With the rapid development of high throughput technologies, the computational data analysis becomes more a bottleneck. It will be interesting to observe how bioinformatics will keep up with these challenges, but it will even more exciting to participate.

Acknowledgements

First, I would like to thank Dr. Robert M. MacCallum for support and discussion. In particular I am grateful for his suggestions and participation in the software development.

I thank Prof. Michael J.E. Sternberg for supervision. I acknowledge the members of the Biomolecular Modelling Laboratory at Cancer Research UK (former Imperial Cancer Research Fund) and the Structural Bioinformatics Group at Imperial College for support and discussion. Thanks to Drs. Lawrence A. Kelley, Graham R. Smith, Adrian P. Cootes and David N. Perkins for discussion and corrections. Also, thanks very much to Paul Fitzjohn who reliably maintained the Linux computer farm at Cancer Research UK.

Finally, I thank Cancer Research UK for funding and providing an interesting research environment.

Appendix A

Supplementary material for 3D-GENOMICS

A.1 Database tables

Attribute	Type	Description
Alignment (Alignment): Stores information common to all kinds of alignments		
FeatureId	int	ref. to a Feature
Sbj	int	ref. to a Pseq (subject of alignment)
SbjStart	smallint	start of alignment in subjects
SbjStop	smallint	stop of alignment in subject
Identity	tinyint	percent sequence identity
AutoAnnot (-): Dump of text information from other tables, generated by a script for fast annotation search via the web		
Tags	varchar	space separated list of genome names
PseqId	int	ref. to a Pseq of the genomes described by Tags
Descrip	text	a text description
Type	varchar	type of annotation (e.g. scop or pfam)
BlastHit (BlastHit): BLAST specific hit information		
FeatureId	int	ref. to an Alignment
Evalue	double	e-value of bit score
Score	float	bit score
BlastRun (BlastRun): BLAST run information		
RunId	int	ref. to a PseqRun
DbSize	int	size of sequence database in sequences
Status	enum('crash', 'void', 'empty', 'drifted', 'limited', 'blast', 'collecting', 'converged', 'impala', 'rps')	final status of BLAST analysis (or run inheriting from a BlastRun), 'drifted' existing confident hits got lost due to possibly corrupted PSSM (PSI-BLAST), 'collecting' not converged (PSI-BLAST only) and 'converged' (PSI-BLAST only), 'blast' means the hit was produced by BLAST (otherwise PSI-BLAST), 'rps' means produced by 'RPS-BLAST' and 'impala' produced by IMPALA
ClassName (Feature, Run): Class names to reconstruct API objects		
ClassNameId	tinyint	identifier

continued on next page

continued from previous page

Attribute	Type	Description
Name	varchar	class name
Coil (Coil): Coiled-coil description		
FeatureId	int	ref. to a Feature
Score	float	confidence score of this coil
CoilRun (CoilRun): Description of a Coils2 analysis		
RunId	int	ref. to a PseqRun
NumHits	int	number of identified coiled-coils
DomainPartner (ScopStatRun): Domain partners (combinations) for SCOP domains		
RunId	int	ref. to a DomainStat
AC	varchar	SCOP code for superfamily in DomainStat
AC2	varchar	code/accession for other domain
Name	varchar	Name of other domain
Type	enum('scop', 'pfam')	type of other domain
Freq	smallint	total frequency of co-occurrence
DomainStat (ScopStatRun): Genome specific SCOP superfamily information		
RunId	int	ref. to a GenomeRun
AC	varchar	family/superfamily code
Name	varchar	family/superfamily name
Type	enum('scop', 'pfam')	type of domain
FreqDom	smallint	number of domains in family/superfamily
RankDom	smallint	rank of FreqDom
FracDom	float	FreqDom normalised by number of all domains
FreqSeq	smallint	number of sequences with domain type AC
RankSeq	smallint	rank of FreqSeq
FracSeq	float	FreqSeq normalised by number of sequences with domains
FreqTM	smallint	number of domains in transmembrane proteins
RankTM	smallint	rank of FreqTM
AvgSeqId	tinyint	average sequence identity of domain type
StdevSeqId	tinyint	standard deviation of AvgSeqId
ScopPartners	smallint	number of co-occurring SCOP superfamilies
PfamPartners	smallint	number of co-occurring PFAM entries
Feature (Feature): Describes any kind of sequence feature with its location in the sequence		
FeatureId	int	identifier
ClassNameId	tinyint	ref. to ClassName of feature object
Start	smallint	start (within sequence)
Stop	smallint	stop (within sequence)
RunId	int	ref. to Run that produced this feature
GSCount (GenomeSummary): Genome wide frequencies of different annotation features		
GSCountId	int	identifier
RunId	int	ref. to a GenomeRun
Name	varchar	name of annotation feature
Number	int	number of observations for this annotation
Type	enum('Sequences', 'Regions', 'Residues')	'Number' refers to sequences, regions or residues
GSMember (GenomeSummary): Members of a GSCount entry		
GSCountId	int	ref. to a GSCount
MemberId	int	ref. to a PseqId or FeatureId (depends on the 'Type' of the GSCount)

continued on next page

continued from previous page

Attribute	Type	Description
Gaps (Alignment): Helper table for Alignment		
FeatureId	int	ref. to an Alignment
QryGaps	blob	compressed list of gap positions and extent in query
SbjGaps	blob	compressed list of gap positions and extent in subject
GenomeRun (GenomeRun): Genome wide analysis or data summary		
RunId	int	ref. to a Run
Tags	varchar	space separated list of genome names/tags
HMM (HMM): HMM associated information (currently not the HMM itself!). Stores annotation of PFAM HMMs.		
Acc	varchar	identifier
Name	varchar	short name
Description	varchar	annotation
Leng	int	length of HMM
HMMHit (HMMHit): Match to an HMM (from PFAM)		
FeatureId	int	ref. to a Feature
Evalue	double	e-value of bit score
Score	float	bit score
HMMStart	smallint	start of hit within HMM
HMMStop	smallint	stop of hit within HMM
Acc	char	ref. to HMM
Host (Run): Client that executed a run		
HostId	smallint	identifier
Name	varchar	name of host (or IP-address)
LCR (LCR): Low complexity region		
FeatureId	int	ref. to a Feature
Score	float	confidence score of assignment
LCRun (LCRun): Run information of SEG (detection of low complexity regions		
RunId	int	ref. to a PseqRun
NumHits	int	number of LCR features produced
MakeMat (PsiBlastRun): Binary checkpoint file of last PSI-BLAST iteration		
RunId	int	ref. to a PsiblastRun
Checkpoint	mediumblob	checkpoint data (platform dependant!)
OMIMgenmap (OMIM): cytogenetic locations and other information for OMIM entries, see http://www.ncbi.nlm.nih.gov/omim/ for details		
ChrMap	varchar	numbering system
EntryDate	date	OMIM entry date
Loc	varchar	cytogenetic location (locus)
Symbols	varchar	gene symbols (short names)
Status	enum('C', 'P', 'I', 'I', 'L')	certainty of locus assignment
Title	text	title of disease or gene
MIM	int	MIM number (should be unique)
Method	varchar	method for genetic mapping
Comments	text	list of comments
Disorders	varchar	list of disorders
Mouse	varchar	mouse correlate
Ref	varchar	list of literature references
Params (Params): Analysis/Run specific parameters		
ParamsId	smallint	identifies the set of parameters that belong together
Pkey	varchar	name of parameter (key)
Pvalue	varchar	value of parameter (may be NULL)

continued on next page

continued from previous page

Attribute	Type	Description
Pdesc (Pdesc): Protein description		
PdescId	int	identifier
Acc	varchar	accession number of source database (usually a GI-number)
Name	varchar	list of all known names and identifiers, NCBI-style
Description	text	description line
PseqId	int	ref. to a Pseq
Date	timestamp	entry of modification date
TaxId	int	ref. to a node in taxon database
PdescTag (Pdesc): Linker for Pdesc Tag relationship		
TagId	int	ref. to a Tag
PdescId	int	ref. to a Pdesc
PerlObject (PerlObject): Storage for a persistent perl object (serialised objects)		
PerlObjectId	int	identifier
Class	varchar	class name of object
Perl	mediumblob	compressed object
PrositeMatch (PrositeMatch): A match of a PROSITE pattern		
FeatureId	int	ref. to a Feature
AC	char	accession code of pattern
PrositeRun (PrositeRun): PROSITE pattern database scan		
RunId	int	ref. to a PseqRun
NumHits	int	number of matches produced by this run
ProsperoHit (ProsperoHit): Hit from the prospero program (self alignment to find repeats)		
FeatureId	int	ref. to an Alignment
Evalue	double	e-value of bit score
Score	int	bit score
ProsperoRun (ProsperoRun): Repeat analysis with prospero		
RunId	int	ref. to a PseqRun
k	float	calculated k of scoring scheme
lambda	float	calculated lambda of scoring system
Pseq (Pseq): Protein sequence		
PseqId	int	identifier
Seq	text	amino acid sequence as a string
md5	varchar	hexadecimal 16 byte MD5 checksum of Seq
Date	timestamp	entry date
QuickBits	int unsigned	annotation bitmask, precompiled from Pdesc list
Len	smallint unsigned	length of Seq
PseqMask (GenomeSummary): Bitmask for generated annotation for each sequence residue position		
RunId	int	ref. to a GenomeRun
PseqId	int	ref. to a Pseq
Mask	blob	compressed list of integers for sequence, each position is a bitmask for a residue
PseqOMIM (Pseq, OMIM): Relationship between Pseq and OMIM		
PseqId	int	ref. to a Pseq
MIM	int	OMIM identifier, ref. to OMIMgenmap
PseqRun (PseqRun): Protein sequence based analysis		
RunId	int	ref. to a Run
PseqId	int	ref. to a Pseq
Start	smallint unsigned	start of analysed region

continued on next page

continued from previous page

Attribute	Type	Description
Stop	smallint unsigned	stop of analysed region
PsiBlastHit (PsiBlastHit): A PSI-BLAST hit		
FeatureId	int	ref. to a BlastHit
Iteration	tinyint	iteration of this hit
Flag	set('firstPS', 'last', 'lastSeen', 'best')	description of iteration, 'firstPS' is the 1st position specific iteration this hit was found in (at least iter. 2), 'last' iteration, 'best' iter. is where the hit has the best e-value, 'lastSeen' is the iter. after which this hit disappeared)
PsiBlastRun (PsiBlastRun): PSI-BLAST analysis		
RunId	int	ref. to a BlastRun
ItersRequest	tinyint	maximum number of requested iterations
ItersDone	tinyint	number of performed iterations
PSSM	blob	compressed text PSSM of last iteration
Region (Region): Cluster of alignments within a region produced by a SummaryRegionRun of the API		
FeatureId	int	ref. to a Feature
RepFeatureId	int	ref. to a Feature/Alignment
RegionFeature (Region): Member of a region		
RegionId	int	ref. to a Region
FeatureId	int	member (ref. to a Feature/Alignment)
Run (Run): Superclass for any kind of analysis		
RunId	int	identifier
ClassNameId	tinyint	ref. to a ClassName
Date	datetime	date when analysis was carried out
RunTime	mediumint	runtime of analysis
HostId	smallint	ref. to Host
ParamsId	smallint	ref. to Params
Error	varchar	optional error or status string
SecStr (SecStr): A secondary structure element		
FeatureId	int	ref. to a Feature
State	enum('C', 'T', 'H', 'E')	Coil, Turn, Helix, Strand
Score	blob	compressed list of scores at each position from Feature.Start to Feature.Stop
SigPep (SigPep): Signal peptide		
FeatureId	int	ref. to a Feature
Model	enum('gram+', 'gram-', 'euk')	best model (gram positive or negative or eukaryotic)
Score	float	confidence score of prediction
TMH (TMH): Transmembrane helix		
FeatureId	int	ref. to a Feature
Ori	enum('in', 'out')	topology, N-terminus of first helix is inside or outside the cell
TMRun (TMRun): Transmembrane analysis		
RunId	int	ref. to a PseqRun
Score	float	overall confidence of prediction
NumHits	int	number of predicted membrane helices

continued on next page

continued from previous page

Attribute	Type	Description
Tag (Pdesc): A descriptive keyword or label for sequences, e.g. 'Ecoli' to label all sequences of the genome.		
TagId	int	identifier
Name	varchar	keyword
Type	enum('user', 'static', 'db')	keyword was set by a user, automatically or is a database name

Table A.1: Tables of the 3D-GENOMICS database. For a detailed description of the data-types see the MySQL manual (http://www.mysql.com). For many data-types MySQL allows a size definition in digits or characters (for char, varchar, text and blob), these are not shown in the Type column. The table name is given in **bold font** with the managing class of the API in braces. Primary key, non-unique keys and unique keys are shown. 'ref.' is 'reference', 'iter.' is 'iteration'.

Attribute	Type	Description
Classif: The SCOP classification		
DomainCode	varchar	e.g. d3sdha_
Release	smallint	e.g. 1.53
FullCode	varchar	numerical code, e.g. 1.001.001.001.001.001
ClassDescRef	int	ref. to Descrip (class name)
FoldDescRef	int	ref. to a Descrip (fold name)
SfamDescRef	int	ref. to a Descrip (superfamily name)
FamilyDescRef	int	ref. to a Descrip (family name)
ProteinDescRef	int	ref. to Descrip (protein name)
SpeciesDescRef	int	ref. to a Descrip (species name)
PDBCode	varchar	the PDB code, e.g. 3sdh
Region	varchar	the domain definition within the PDB entry, e.g. 'a:' or 'a:143-283'
Descrip: Names		
Id	int	identity
Txt	varchar	text description, e.g. protein name

Table A.2: Tables of the SCOP database. The table name is given in **bold font**. Primary keys and non-unique keys. 'ref.' is 'reference'. The FullCode defines the root (1), class, fold, superfamily, family and protein+species accession number separated by a '.' (e.g. '1.002.012.033.004.008'). The classification is taken from ASTRAL flat files (Chandonia *et al.* (2002), http://astral.stanford.edu/). Sequences and structures are not stored in the tables, but in flat files. The identifier system has changed starting with release 1.55, and is not compatible with the Classif table, which stores SCOP releases 1.48, 1.50 and 1.53.

A.2 Classes and modules of the API

Method/Function	Description
Alignment (Feature): Baseclass for alignment based classes such as BlastHit.	
get	redef. of baseclass method
getPairwise	returns a pairwise alignment as an array
sprintPairwise	returns a pairwise alignment as a string
fullSbj	alignment with terminal gaps and gaps removed from the query.
calcIdentity	recalculates sequence identity in percent
hssp	scores an alignment by length and percent identity (Rost, 1999)
swapQrySbj	swaps query and subject of an alignment
coverage	returns alignment coverage in query and subject sequence
AnnotRegion (HomolRegion): A functionally annotated sequence region.	
isAnnot	true for this type of region
BlastHit (Alignment): A BLAST HSP (hit). No special methods.	
BlastRun (PseqRun): A complete BLAST run.	
makeRuns	non oop funtion to generate a list of BlastRun objects for a sequence (extension of baseclass function)
queueResourceOpt	required computing resources
queueCommand	redef. of baseclass method
getQryMaskedFeatures	get feature objects that were used to mask query
getQrySeqString	get query string as it was passed to BLAST
run	redef. of baseclass method
getHits	return list of BlastHits (or other hit types for classes inheriting from BlastHit)
getSummary	extension of baseclass method
seaview	display a list of hits as a multiple alignment using the 'seaview' program
clustalx	display a list of hits as a multiple alignment using 'clustalx'
Coil (Feature): A coiled-coil sequence region. No special methods.	
CoilRun (PseqRun): A coiled-coil analysis of a sequence.	
run	redef. of baseclass method
DbConnection (-): A database connection object. Provides methods to retrieve data from and to insert data into the database. It is the baseclass for most of the other classes, because most objects are stored in the database. Database connections are managed via the Perl DataBase Interface (DBI).	
new	object constructor
sync	synchronises the object with the database (reads from or writes to database)
get	takes a list of object attribute names and returns their values
modify	modifies an object (call sync afterwards!)
set	sets value for an attribute (call sync afterwards!)
readOnly	makes the object read only (changes do not get written to the database)
dbConnect	connects to database
isConnected	true if object is connected to database
refresh	refreshes a database connection (if it was lost)
RaiseError	makes connections verbose on errors (called by dbConnect)
dbLogging	makes data modifying actions logged by the sql-server(called by dbConnect)
dbDisconnect	disconnects from database
prepareForDump	prepares object for PerlObject table (disconnects form database)
reconnect	opposite to prepareForDump (reconnects persistent object to database)
dbHandle	returns a Perl DBI database handle
dbSource	returns a Perl DBI database source string
dbName	returns the name of the database
dbHost	returns the database host
dbUser	returns the database user name
dbPasswd	returns the database password for the user

continued on next page

continued from previous page

Method/Function	Description
lastInsertId	returns the last insert ID from AUTOINCREMENT tables.
selectRow	executes an SQL SELECT statement and returns one row
doSQL	executes any SQL statement, does not return a value
dbQuote	quotes a string to be SQL compatible
now	current date and time in a format readable by the SQL-server
DomDbRegion (Region): A Region that is a domain.	
isDomain	true for this type of domain
maxStoredFeatures	returns the maximum number of stored members for this region
DomainStat (DbConnection): Objects of this class store high level information about a domain type such as a particular SCOP superfamily. This class was mainly developed for web-purposes.	
normalise	normalises data by the number of genomes that were used for the analysis (the number of Tags from the GenomeRun object)
getPartners	gets a list of domain partners for a domain type
getLink	gets the URL for an attribute to link to a script that gives more information
Feature (DbConnection): The baseclass for all feature types that describe a location within an object (currently only Pseq objects).	
remove	removes feature object
overlaps	returns the overlap (in residues) between two features
within	returns true if the other feature is contained within the feature
len	returns the length of the feature
getStringRep	string representation of the feature (extended by subclasses)
getStringRepChar	a single character representation (extended by subclasses)
getSummary	summary information about a feature (implemented by inheriting classes)
clone	returns a copy of the database synchronised object blessed to its correct class
cloneCopy	returns a copy of the current object as it is (including modifications)
insertWebFeature	inserts the web-representation of the feature into a string
webFeature	the actual web-feature (extended by subclasses)
webLinkText	the text of the URL (extended by subclasses)
webLinkUrl	the URL itself (extended by subclasses)
webLeftEndChar	left terminal character of the web-presentation (extended by subclasses)
webRightEndChar	right terminal character of the web-representation (extended by subclasses)
webPadChar	characters outside this feature (extended by subclasses)
webColour	colour of the feature (extended by subclasses)
webLinkMouseOver-Text	text to appear in browser on mouse-over (extended by subclasses)
name	the name of the feature (class name, can be overwritten by other classes)
GapCoder (-): Not a class, helper module to manage gaps of alignments.	
encode	encodes an alignment or a list of gaps into a compact form
decode	decodes an encoded alignment into a list of gap positions and gap-extensions
Genome (DbConnection): Simple representation of a genome.	
getPseqs	returns all Pseq object for this genome
makePseqRuns	generates a particular run type for the genome
writeTable	writes an SQL table with every PseqId of the genome, and returns the table name
writeFeatureTable	writes an SQL-table that contains all requested features for all Pseq objects of the genome, returns the table name
linkByBlast	returns an SQL-table name with all homologues between the genome and a given other genome.
GenomeRun (Run): Baseclass for all analysis that treat a genome or proteome as a whole.	
alreadyRun	implementation of baseclass object, returns true if the object is was already processed before with the same parameters
queueStderrId	where the stderr of the analysis is copied to

continued on next page

continued from previous page

Method/Function	Description
queueStdoutId	where stdout of the analysis is copied to
GenomeSummary (GenomeRun): Genome wide annotation summary and statistics.	
getBitTemplates	returns a hash with annotation types as keys and their corresponding bits in the residue wise description of a sequence
getNextFreeBit	returns the next bit to be used for a new annotation type
run	redef. of baseclass method
writeCount	writes a generated counts to the database
readCount	reads the count for a particular annotation type from the database
getGSCountId	returns a GSCountId for the requested 'Name'/'Type' pair of annotation
getMemberIds	returns a list of IDs that are members of this annotation type
getPfamRegionMembers	returns a hash of PFAM entries found for within this genome
getScopRegionMembers	returns a hash of SCOP domains found in this genome
getBitMask	returns the residue wise bit mask of a Pseq object that is part of the genome
queueResourceOpt	redef. of baseclass method
HMM (DbConnection): Simple representation of a hidden Markov model, currently contains PFAM annotation information only.	
noopGetDesciption	fast non oop funtion, returns the HMMs description (annotation)
HMMHit (Feature): A high scoring match of a protein sequence to an HMM.	
isAnnot	true if the HMMHit is to a functional annotated HMM (should be moved to the HMM class)
coverage	returns length coverage of the query by the HMM and the HMM length coverage by the query as two real numbers
HMMRun (PseqRun): Run class for 'hmmpfam' of the HMMer package.	
run	implementation of baseclass method
queueCommand	redef. of baseclass method
getDomains	returns HMMHit objects, temporarily modified to be non-overlapping
HomolRegion (Region): A sequence region with homologous sequences.	
getRepPdesc	returns the Pdesc object of the subject sequence the representative alignment
IMPALAHit (BlastHit): A Hit and HSP produced by IMPALA (subject is a sequence that is representative for the PSSM).	
getPsiBlastRun	returns the PsiBlastRun object that produced the checkpoint file used to generate the IMPALA matrix
IMPALARun (BlastRun): Run class for IMPALA program.	
run	redef. of baseclass method
queueResourceOpt	redef. of baseclass method
queueCommand	redef. of baseclass method
LCR (Feature): A Low Complexity Region produced by an LCRun. No special methods.	
LCRun (PseqRun): Run class for the SEG program.	
run	implementation of baseclass method
MultiRun (Run): Objects of this class contain several other Run objects that will all be executed on the same client computer. This avoids overloading the queueing system if the runtime time for the actual Run object that performs an analysis is short.	
getRuns	returns the Run objects to be executed
run	redef. of baseclass method
alreadyRun	returns the object if it was already run before
queueResourceOpt	redef. of baseclass method
queueStderrId	where stderr is copied to
queueStdoutId	where stdout is copied to

continued on next page

continued from previous page

Method/Function	Description
Names (-): Not a class. Contains hashes and arrays of organism names and tags (abbreviations) to group genomes. Does not provide any functions.	
Nobody (-): This package overwrites some routines of the DbConnection package (but it does not inherit from DbConnection), and may be used for anonymous read only database access.	
dbPasswd	redef. of baseclass method (no password)
dbUser	redef. of baseclass method ('nobody')
dbName	redef. of baseclass method
OMIM (DbConnection): Representation of an OMIM entry.	
remove	raises and error (object cannot be removed)
getPseqs	get Pseq objects linked to this OMIM object
setPseq	link a Pseq object to this OMIM object
removePseq	remove link between Pseq object and OMIM object
getByPseq	non oop funtion to search a OMIM objects by PseqId
getByTextField	non oop function to search OMIM objects by text
webLinkText	same as Feature.webLinkText
webLinkUrl	same as Feature.webLinkUrl
webLink	same as Feature.webLink
PSSM3dHit (Alignment): A hit produced by a PSSM3dRun, a (usually remote) homologue of known structure.	
confidence	returns a confidence measure in percent
PSSM3dRun (PseqRun): Run class to perform the 3D-PSSM analysis.	
makeRuns	non oop function to generate a list of PSSM3dRun objects for a sequence
queueResourceOpt	redef. of baseclass method
queueCommand	redef. of baseclass method
run	implementation of baseclass method
Params (DbConnection): Parameter sets used by an analysis (Run object).	
remove	remove this object from the database
getAll	get all parameter key/value pairs
hasKey	true if the parameter key exists
get	redef. of the baseclass method that does not raise an error if called with a non existing attribute (makes 'hasKey' obsolete)
PdbRegion (Region): A Region defined by homology to sequences of known structure (PDB chains).	
isStructure	true for this kind of region
maxStoredFeatures	redef. of baseclass method
myTag	'pdb'
Pdesc (DbConnection): A description of a protein sequence, contains free text and different tags (keywords).	
remove	removes the object from the database
getDb	returns the tag of the source database this object come from
getTaxon	returns the corresponding Taxon object if it exists
webLinkText	same as Feature.webLinkText
webLinkUrl	same as Feature.webLinkUrl
webLink	same as Feature.webLink
webLinkMouseOverText	same as Feature.webLinkMouseOverText
PerlObject (DbConnection): Helper class to distribute Run objects over a computer farm, stores uncomposed PerlObjects as Perl code.	
remove	removed the object from the database
PrositeMatch (Feature): A match to a PROSITE pattern. No special methods.	
PrositeRun (PseqRun): Finding PROSITE patterns within a query sequence.	
run	implementation of baseclass method

continued on next page

continued from previous page

Method/Function	Description
getPrositePatterns	non oop function to retrieve the patterns from a flat file
ProsperoHit (Alignment): Alignment produced by the 'prospero' program. No special methods.	
ProsperoRun (PseqRun): Runs the prospero program for a protein sequence.	
run	implementation of baseclass method
Pseq (DbConnection): A protein sequence.	
remove	removes object from the database (including all objects depending on this object)
getBioSeq	generate a BioPerl object from this object
fseq	write object in fasta format
getFeatures	returns the list of features for this sequences (from all run objects)
getPsiBlastHits	returns PsiBlastHit objects
getBlastHits	returns BlastHit objects
getHits	wrapper for the two methods above
getXBlastHits	returns both, PsiBlast and Blast hit objects
getSeqsHittingMe	returns Pseq objects for which a PsiBlastHit object has this Pseq object as subject of the alignment
getHitsToMe	similar to the above method, but it returns PsiBlastHit objects
getSummaryRegions	returns the list of Region objects for this sequence
xmapFeatures	uses a list of features and replaces the corresponding sequence positions with the 'X' character (sequence masking)
getRuns	returns a list of Run objects
getDbs	returns a hash of source database tags for this sequence
getPdesc	returns a requested Pdesc object
getPdescs	returns all Pdesc objects for this object
getTaxIds	returns a hash of corresponding TaxIds (for Taxon objects)
makePseqRuns	non oop function to generate a list of Run objects
getPSSMs	returns a list of PSI-BLAST PSSMs for all sequence fragments of this sequence
getPSSM	returns one PSI-BLAST PSSM that covers the whole sequence
getPSSMerror	returns an error message if there was any while calling one of the two above methods
getOMIM	returns OMIM objects linked to this object
seaview	launched the 'seaview' multiple sequence alignment viewer and displays homologues
clustalx	same as 'seaview', but using the 'clustalx' program
makeSCOPdom	generates a SCOPdom object if this object corresponds to a SCOP domain
makeSCOPdoms	if the object corresponds to a PDB chain, a list of corresponding SCOPdom objects is generated
getBits	returns a hash with source database tags and some other tags and corresponding bits (shortcut to get the annotation status and the source databases for the object, this bypasses the slow request of Pdesc objects)
SQLsetBits	non oop function to set the bits described above for the object (should be run by the administrator on Pseq or Pdesc table updates)
PseqFrag (Pseq): A region within a protein sequence.	
set	redef. of baseclass method, raises and error
remove	redef. of baseclass method, raises and error
modify	redef. of baseclass method, raises and error
getFull	returns the full-length Pseq object
getBioSeq	same as for a Pseq, but on a fragment
getOverlappingFeatures	returns feature objects that overlap with this sequence region
getWithinFeatures	feature objects that are contained within this region
getFeatures	same as getOverlappingFeatures
getBlastHits	same as baseclass method, but filters to non-overlapping hits
getPsiBlastHits	same as method above, but for PsiBlastHits

continued on next page

continued from previous page

Method/Function	Description
getHits	same as baseclass method, but filters non-overlapping hits
getSummaryRegions	same as baseclass method, but filters non-overlapping regions
getHitsToMe	same as baseclass method, but filters non-overlapping hits
getRuns	same as getOverlappingRuns
getExactOrFullRuns	gets all runs that are exactly mapped to this fragment or the whole Pseq object
getWithinRuns	run objects contained within this region
getOverlappingRuns	run objects overlapping with this region
getPSSM	extends baseclass method, sub-matrix of the full length PSI-BLAST PSSM
xmapFeatures	extends baseclass method
overlaps	same as Feature.overlaps but for a Pseq fragment
within	same as Feature.within but for a Pseq fragment
PseqRun (Run): A Run that is performed on a protein sequence, baseclass for many other Run objects.	
alreadyRun	implementation of baseclass method, returns object if it was already run with the same Params object and Start/Stop definitions
queueStderrId	redef. baseclass method
queueStdoutId	redef. of baseclass method
makeRuns	non object oriented function, implementation of baseclass function
overlaps	similar to PseqFrag method
within	similar to PseqFrag method
getPseqId	fast method to retrieve the PseqId of the object
PsiBlastHit (BlastHit): A hit produced by the PSI-BLAST program. No special methods.	
PsiBlastRun (BlastRun): Runs the PSI-BLAST program.	
writeCheckpointFile	retrieves a checkpoint and writes it to a file.
getCheckpoint	retrieves a checkpoint
drifted	tries to determine if the run drifted (use with caution!)
PsiPredRun (PseqRun): Runs the PSI-Pred secondary structure prediction program (requires a PsiBlastRun).	
run	implementation of baseclass method
Region (Feature): A cluster of Features (currently alignments only), that define a region within a sequence, base class for many specialised region types.	
getFeatures	list of features that are a member of this region
countsAs	this region is only a fraction of a domain (e.g. a region from a discontinuous domain)
isDomain	true if the region is a domain
isStructure	true if region has known 3D-structure
isAnnot	true if region is annotated
maxStoredFeatures	default of maximum number of members to store
myTag	a tag/keyword for this region (to be implemented by other classes)
getRepPdesc	Pdesc object of the subject of the representative alignment
Run (DbConnection): The basic analysis object, to manage execution of the actual analysis. Baseclass for all other runs.	
makeRuns	generate one or more run objects (to be implemented by other run classes)
remove	remove this object from the database
getFeatures	list of Feature object from this Run object
getSummary	descriptive information about the object (to be implemented by other run classes)
run	execute the analysis (to be implemented by other Run classes)
queue	submit object to the queueing system
queueCommand	the command submitted to the queueing system
queueName	the name of the queue
queueStdoutDir	directory to which stdout gets copied to
queueStderrDir	directory to which stderr gets copied to
queueStderr	filename of stderr

continued on next page

continued from previous page

Method/Function	Description
queueStdout	filename of stdout
queueStderrId	unique name for object stderr
queueStdoutId	unique name for object stdout
queueSleep	pause between subsequent submissions to the queue
queuedMax	maximum number of objects in the execution queue
queueResourceOpt	required computing resources to execute the analysis
alreadyRun	true if the analysis was already run before, e.g. if the object already exists in the database (to be implemented by specialised classes)
clone	copies and returns the database synchronised object blessed with the correct class (similar to Feature.clone)
countFeatures	number of Feature objects from this run
makeNonOverlappingFeatures	returns a list of read only non overlapping Feature objects (temporarily modifies the Features Start/Stop)
getRunIdsByParams	non oop function that returns a list of cloned Run objects that satisfy given parameter key/value pairs of Params objects.
SCOPdom (Pseq): A sequence that is a SCOP domain. Links the 3D-GENOMICS main database to the scop helper database. Currently provides attributes only (also see A.2)	
ScopRegion (DomDbRegion): A sequence region defined by SCOP homologues.	
isStructure	true for this region type
getRepScopDom	returns a representative SCOPdom object
getSuperfamilies	the superfamily of the region
myTag	'scop'
countsAs	1, or the fraction of a discontinuous SCOP domain
ScopStatRun (GenomeRun): High level analysis of SCOP superfamilies, requires many other analysis to be done before (e.g. GenomeSummary).	
run	implementation of baseclass method
getHash	a hash of DomainStat objects
getGS	get corresponding GenomeSummary object
getDomainPairs	gets all domain pairs for this run with one request
ScratchDb (DbConnection): Database connection to a user writable database (even 'nobody' is allowed to write to the scratch database). Stores temporary user specific objects.	
dbName	redef. of baseclass method ('scratch')
dbUser	redef. of baseclass method
dbPasswd	redef. of baseclass method
SecStr (Feature): A Secondary structure element (produced by a PsiPredRun).	
getResidueScore	score for the secondary structure state at a residue position
SigPep (Feature): An N-terminal signal peptide. No special methods.	
SigPepRun (PseqRun): Searches for signal peptides.	
queueResourceOpt	redef. of baseclass method
run	implementation of baseclass method
SummaryRegionRun (PseqRun): Clusters alignments into different types of regions (specialised Region objects).	
run	implementation of baseclass method
TMH (Feature): A transmembrane helix. No special methods.	
TMRun (PseqRun): A transmembrane helix prediction for a sequence.	
makeRuns	redef. of baseclass method
run	implementation of baseclass method
Taxon (DbConnection): Taxonomy object, interface to the taxon helper database.	
remove	redef. of baseclass method, raises and error
getParent	get the Taxon object of parent node of the taxonomic tree
getChildren	get all Taxon objects that have this Taxon as parent

continued on next page

continued from previous page

Method/Function	Description
getRank	the name of the rank of the object (e.g., 'kingdom', 'genus')
inSubTree	true if the object is in a tree rooted by a given other node
isRoot	reverse of 'inSubTree', true if object is root of a given other node
webLinkText	same as Feature.webLinkText
webLinkUrl	same as Feature.webLinkUrl
webLink	same as Feature.webLink
Workstations (-): Not a class. Helper module to submit Run objects to a computer farm.	
run	submit several Run objects to the queueing system
fastaDB (-): Not a class. Inserts sequences and sequence descriptions together with specific user information as objects into the database. Used for large scale database insertions and updates.	
insertEntries	insert a Pseq and with several Pdesc entries into the database
insert	insert all entries of an annotated (description line) fasta formated sequence file into the database
nextSeq	returns the next fasta entry of the sequence file
pbPSSM (Feature): A PSSM generated by PSI-BLAST.	
remove	redef. of baseclass, raises and error
getResidue	the amino acid at a given position
getScore	the score for a given amino acid type at a given position
getScores	all amino acid scores for a given position
getSubMatrix	a sub-matrix that describes a given region of the sequence

Table A.3: Overview over modules and classes of the 3D-GENOMICS API. The class or module name is given in **bold font** above each subtable, and the base class is given in braces. Only methods and functions are described. Class attributes are usually the same as the attributes of the corresponding SQL table (see A.1). Some specialised modules, classes or methods of the API are not shown. For simplification the returned data types and the list of possible arguments for methods are not explicitely shown. 'redef'. means redefinition, 'def.' means definition. If a class does not provide any special methods it may still redefine or extend baseclass methods. For example most classes that inherit from Feature redefine some of the `web*` and `getString*` methods as well as the `getSummary` method. Classes inheriting from Run redefine the `getSummary` method.

Appendix B

Internet resources

URL	Description
ftp://ftp.ebi.ac.uk/pub/software/unix/coils-2.2/	software to predict coiled-coils in protein sequences
ftp://ftp.ncbi.nih.gov/blast	BLAST, PSI-BLAST and IMPALA executable programs
ftp://ftp.ncbi.nih.gov/blast/db/nr.Z	non-redundant protein sequence database
ftp://ftp.ncbi.nih.gov/genomes/	Nucleic acid and protein sequences from completely sequenced genomes (or nearly finished genome projects)
ftp://ftp.ncbi.nlm.nih.gov/genbank/genomes/	the old site for genome sequences
ftp://ftp.ncbi.nih.gov/pub/seg/	software to detect low complexity regions in protein sequences
ftp://ftp.ncbi.nih.gov/pub/taxonomy/	tables of the NCBI taxonomy database
ftp://ftp.ncbi.nlm.nih.gov/blast/db/	sequence databases for BLAST and PSI-BLAST (nucleotide and protein)
http://astral.stanford.edu/	protein sequences for SCOP domains
http://bioinf.cs.ucl.ac.uk/psipred/	secondary structure prediction of protein sequences
http://genomes.rockefeller.edu/magpie/	Magpie, genome annotation software package
http://hmmer.wustl.edu/	HMMer software package for hidden Markov models
http://jura.ebi.ac.uk:8765/ext-genequiz/	GeneQuiz software for web based for protein annotation
http://pedant.mips.biochem.mpg.de	genome and proteome annotation database
http://presage.berkeley.edu/	database for structural genomics projects
http://prodes.toulouse.inra.fr/prodom/doc/prodom.html	ProDom, protein domain database
http://scop.mrc-lmb.cam.ac.uk/scop/	Structural Classification Of Proteins
http://smart.embl-heidelberg.de	domain database
http://stash.mrc-lmb.cam.ac.uk/SUPERFAMILY/	HMMs for SCOP and proteome assignments of SCOP domains
http://wit.integratedgenomics.com/GOLD	list and status of completed and ongoing genome sequencing projects
http://www.biochem.ucl.ac.uk/bsm/cath/	another structural classification of proteins
http://www.biochem.ucl.ac.uk/bsm/cath_new/Gene3D/	CATH domain assignments to genomes
http://www.bioinf.man.ac.uk/dbbrowser/PRINTS	database of protein domains and motifs
http://www.bioperl.org	BioPerl software project
http://www.blocks.fhcrc.org	BLOCKS domain and motif database
http://www.bmm.icnet.uk	Biomolecular Modelling site at Cancer Research UK
http://www.cbs.dtu.dk/services/SignalP-2.0/	signal peptide prediction of protein sequences
http://www.cbs.dtu.dk/services/TMHMM/	transmembrane helix prediction of protein sequences
http://www.ebi.ac.uk	European Bioinformatics Institute, general bioinformatics resource
http://www.ebi.ac.uk/interpro	combined database of domains, motifs and protein sequences
http://www.ebi.ac.uk/proteome	proteome annotation site
http://www.embl-heidelberg.de/ rost/	B. Rost homepage with supplementary material for alignment accuracy
http://www.embl-heidelberg.de/predictprotein/predictprotein.html)	Predict Protein, protein sequence annotation and structure prediction
http://www.ensembl.org	human genome annotation
http://www.enzim.hu/hmmtop/	transmembrane helix prediction in protein sequences
http://www.expasy.ch/swissmod/SM_3DCrunch.html	results from large homology modelling of protein sequences
http://www.expasy.org/prosite	PROSITE patterns for functional motifs
http://www.geneontology.org	Gene Ontology project
http://www.integratedgenomics.com/	bioinformatics company
http://www.mysql.com	relational database system
http://www.ncbi.nlm.nih.gov	general bioinformatics resource, National Center for Biotechnology Information
http://www.ncbi.nlm.nih.gov/BLAST/	interactive BLAST and PSI-BLAST

continued on next page

continued from previous page

URL	Description
http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=OMIM	resource for inherited human disease (OMIM)
http://www.ncbi.nlm.nih.gov/omim/	the old OMIM page (different interface)
http://www.openpbs.org	load sharing system for distributed processing
http://www.rcsb.org/	database of protein structures
http://www.sanger.ac.uk/Projects/M_tuberculosis	*M. tuberculosis* sequence and annotation resource
http://www.sanger.ac.uk/Software/Pfam	PFAM, protein family and domain database
http://www.sbg.bio.ic.ac.uk	Structural Bioinformatics Group at Imperial College
http://www.sbg.bio.ic.ac.uk/3dpssm/	remote homology detection of protein of known structure
http://www.structuralgenomics.org/	resource for structural genomics
http://www.tigr.org/	TIGR genome sequencing centre
http://www.well.ox.ac.uk/rmott/ARIADNE/	protein sequence comparison software including repeat detection in proteins

Table B.1: URLs for Internet resources mentioned or used within this work.

Appendix C

Abbreviations

C.1 Amino acids

A	ALA	Alanine
C	CYS	Cysteine
D	ASP	Aspartate
E	GLU	Glutamate
F	PHE	Phenylalanine
G	GLY	Glycine
H	HIS	Histidine
I	ILE	Isoleucine
K	LYS	Lysine
L	LEU	Leucine
M	MET	Methionine
N	ASN	Asparagine
P	PRO	Proline
Q	GLN	Glutamine
R	ARG	Arginine
S	SER	Serine
T	THR	Threonine
V	VAL	Valine
W	TRP	Tryptophane
Y	TYR	Tyrosine
X	-	'ignored' residue position

Table C.1: One letter and three letter codes for amino acids.

C.2 Proteins, domains and other biomolecules

ARF-GAP	Adenyl-Ribosylation-Factor, GTPase Activated Protein
ARM repeat	Armadillo Repeat
ATP	Adenosin Tri-Phosphate
BRACA1	Breast Carcinoma 1 gene product
CBS domain	named after a protein (Cystathionine-β-Synthase) that contains this domain
CUB	probably named after the first proteins this domain was found in (human complement components C1r and C2r, sea urchin uEGF and human bone morphogenic protein)
CaLB	Calcium/lipid-binding domain, CaLB)
DD-carboxypeptidase	D-alanyl-D-alanine-cleaving carboxypeptidase
DEATH domain	first described in TNF-mediated cell death signalling
EF-hand	Protein domain named after two important helices E and F
EGF	Epidermal Growth Factor
ERK2	Extracellular Signal-Regulated Kinase 2
EST	Expressed Sequence Tag
ETS domain	Erythroblast Transformation Specific
GPCR	G-Protein Coupled Receptor
GSK-3β	Glycogen Synthase Kinase 3-β
GTP	Guanosin Tri-Phosphate
HPr	Histidine-containing phosphocarrier proteins
HSP90	Heat-Shock Protein 90
KH domain	K (ribonucleo protein) homology domain
LIM domain	zinc finger domain named after the proteins containing this domain (Lin-11 from *C. elegans* vertebrate Isl-1 and Mec-3 *C. elegans*)
MAP	Mitogen-Activated Protein kinase
MBP1	Mlu1-box binding protein
MHC	Major Histo-Compatibility Complex
NAD(P)	Nicotinamide Adenine Dinucleotide (Phosphate)
NFkB	Nuclear Factor κ-B
PDZ domain	signalling domain also known as DHR or GLGF (named after ZO-1 a zonula occludent protein)
PH domain	Pleckstrin homology domain
PK-like	protein kinase-like
PKC	Protein Kinase C
PKD domain	first identified in the PKD1 protein (Polycystic Kidney Disease)
PMS1	Post Meiotic Segregation Protein 1
POZ domain	Pox virus and Zinc finger
Pyk2	Protein Tyrosine kinase

continued on next page

continued from previous page

PYP domain	domain found in Photoactive Yellow Protein
RING domain	'Really Interesting' (zinc finger) domain
RIP	REV protein (RNA binding protein) Interacting Protein
RMS	Root Mean Square
RNI-like domain	Ribonuclease Inhibitor
RNaseA	Ribonuclease A
RNaseH	Ribonuclease H
SH2	SRC (Scavenger Receptor) homology-2 domain
SH3	SRC (Scavenger Receptor) homology-3 domain
SpoIIaa	stage II Sporulation Protein AA
SRCR	Scavenger Receptor, Cysteine-Rich
TFIIA	Transcription Factor IIA
TGF-b	Transforming Growth Factor β
TIM	Triose-phosphate Isomerase
TNF	Tumour Necrosis Factor
TetR/NARL	Tetracycline Resistance regulator and Nitrate/Nitrite metabolism regulatory protein
TPR	Tetratricopeptide repeat
WD repeat	the motif of the repeat is defined by the C-terminal amino acids tryptophan and aspartate
aa-tRNA	Amino-Acyl transfer RNA (Ribonucleic Acid)
aaRS	Amino-acyl-tRNA Synthetase
mRNA	messenger RNA (Ribonucleic Acid)
p8-MTCP1	Mature T-Cell Proliferation-1 protein

Table C.2: Abbreviations of proteins, protein domains and other biomolecules. Capitalisation of the explanations may give a hint how the abbreviation was derived. Some capitalised names are nouns rather than abbreviations, and explanations are given where the origin of the name is not clear.

C.3 Other abbreviations including tools, databases and programs

1D	one dimensional
3D	three dimensional
3D-PSSM	three dimensional PSSM (Position Specific Scoring Matrix)
API	Application Program Interface
ASTRAL	Sequence and structure database, supplement to SCOP
ASV	avian sarcoma virus

continued on next page

continued from previous page

BLAST	Basic Local Alignment Search Tool
BLIMPS	BLocks IMProved Searcher
BLOCKS	'alignment Blocks' (no abbreviation)
BLOSUM	Blocks Substitution Matrix
CASP	Critical Assessment of Structure Prediction
CATH	Class(C), Architecture(A), Topology(T) and Homologous superfamily (H) (a structural classification of proteins)
CDS	Coding Sequence
CGI	Common Gateway Interface
DAS	Distributed Annotation System
DBI	Database Interface
EBI	European Bioinformatics Institute
EMBL	European Molecular Biology Laboratory
ENSEMBL	Human genome resource (not an abbreviation)
ERGO	Genome annotation system from Integrated Genomics, Inc. (not an abbreviation)
ETS domain	Erythroblast Transformation Specific
FASTA	Fast Alignment Search Tool
GO	Gene Ontology
HIV	Human Immune-deficiency Virus
HMM	Hidden Markov Model
HMMer	Hidden Markov Model software package
HSP	High-scoring Segment Pair
HTML	Hypertext Markup Language
ID	Identifier
IMPALA	Integrating Matrix Profiles And Local Alignments
IP-address	Internet Protocol (-address)
Kb	Kilo bases (1000 bases)
KEGG	Kyoto Encyclopedia of Genes and Genomes (enzyme pathway database)
Mb	Mega bases (million bases)
MD5	Message-Digest Algorithm
MG	*Mycoplasma genitalium*
MULTICOIL	Multiple Coiled-Coil (prediction)
MySQL	Product name for a relational database management system
NCBI	Natioanl Center for Biotechnology Information
NMR	Nuclear Magnetic Resonance
NRPROT	Non-Redundant Protein Database
OMIM	Online Mendelian Inheritance in Man
ORF	Open Reading Frame
OpenPBS	Open Portable Batch System
PAM	Point Accepted Mutation

continued on next page

continued from previous page

PANTHER	A protein classification database
PDB	Protein Databank
PEDANT	Protein Extraction, Description and ANalysis Tool
PFAM	Protein Family database of alignments and HMMs
PIR	Protein Information Resource
PRINTS, PRINTS-S	finger Prints
PRODOM, ProDom	Protein Domain (database)
PROSITE	not an abbreviation (protein sequence pattern database)
PSI-BLAST	Position Specific Iterated BLAST
PSI-Pred	Position Specific Iterated Prediction
PSSM	Position Specific Scoring Matrix
ProDom-GC	ProDom for genome wide domain assignments
RMS	Root Mean Square
RMSD	Root Mean Square Deviation
RPS-BLAST	Reversed Position Specific Blast
SAMT98	Sequence Alignment and Modelling software (using HMMs)
SAMT99	Sequence Alignment and Modelling software (using HMMs)
SCOP	Structural Classification Of Proteins
SEG	not an abbreviation, detection of composition biased segments in protein sequences
SMART	Simple Modular Architecture Research Tool
SQL	Structured Query Language
TB	*Mycobacterium tuberculosis*
TIGR	The Institute of Genome Research
TIGRFAM	TIGR Family (protein family database)
TM	Transmembrane
TMHMM	Transmembrane Hidden Markov Model
TOPPRED	Topology Prediction (of transmembrane proteins)
TrEMBL	Translated EMBL (protein database)
TrEMBL-NEW,	new entries in Translated EMBL
URL	Unified Resource Locator
WIT	Genome annotation database from Integrated Genomics, Inc.
XML	eXtended Markup Language
def.	defined
iter.	iteration
max.	Maximum
redef.	redefined

Table C.3: Abbreviations of programs, databases, non standard abbreviated organism names and commonly used abbreviations. Capitalisation of the explanations may give a hint how the abbreviation was derived. Some capitalised names are nouns rather than not be abbreviations, and explanations are given where the origin of the name is not clear.

References

Abola, E. E., J. L. Sussman, J. Prilusky & N. O. Manning (**1997**). Protein Data Bank archives of three-dimensional macromolecular structures. *Methods Enzymol*, 277:556–571.

Acharya, K. R., D. I. Stuart, N. P. Walker, M. Lewis & D. C. Phillips (**1989**). Refined structure of baboon alpha-lactalbumin at 1.7 A resolution. Comparison with C-type lysozyme. *J Mol Biol*, 208(1):99–127.

Adams, M. D., S. E. Celniker, R. A. Holt, C. A. Evans, J. D. Gocayne *et al.* (**2000**). The genome sequence of Drosophila melanogaster. *Science*, 287(5461):2185–2195.

Alm, R. A., L. S. Ling, D. T. Moir, B. L. King, E. D. Brown *et al.* (**1999**). Genomic-sequence comparison of two unrelated isolates of the human gastric pathogen Helicobacter pylori. *Nature*, 397(6715):176–180.

Aloy, P., E. Querol, F. X. Aviles & M. J. Sternberg (**2001**). Automated structure-based prediction of functional sites in proteins: applications to assessing the validity of inheriting protein function from homology in genome annotation and to protein docking. *J Mol Biol*, 311(2):395–408.

Altschul, S. F., R. Bundschuh, R. Olsen & T. Hwa (**2001**). The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Res*, 29(2):351–361.

Altschul, S. F. & W. Gish (**1996**). Local alignment statistics. *Methods Enzymol*, 266:460–480.

Altschul, S. F., W. Gish, W. Miller, E. W. Myers & D. J. Lipman (**1990**). Basic local alignment search tool. *J Mol Biol*, 215(3):403–410.

Altschul, S. F. & E. V. Koonin (**1998**). Iterated profile searches with PSI-BLAST–a tool for discovery in protein databases. *Trends Biochem Sci*, 23(11):444–447.

Altschul, S. F., T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang *et al.* (**1997**). Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. *Nucleic Acids Res*, 25(17):3389–3402.

Andersson, S. G., A. Zomorodipour, J. O. Andersson, T. Sicheritz-Ponten, U. C. Alsmark *et al.* (**1998**). The genome sequence of Rickettsia prowazekii and the origin of mitochondria. *Nature*, 396(6707):133–140.

Andrade, M. A., N. P. Brown, C. Leroy, S. Hoersch, A. de Daruvar *et al.* (**1999**). Automated genome sequence analysis and annotation. *Bioinformatics*, 15(5):391–412.

Antonarakis, S. E. & V. A. McKusick (**2000**). OMIM passes the 1,000-disease-gene mark. *Nat Genet*, 25(1):11. Letter.

Apic, G., J. Gough & S. A. Teichmann (**2001**). Domain combinations in archaeal, eubacterial and eukaryotic proteomes. *J Mol Biol*, 310(2):311–325.

Apweiler, R., T. K. Attwood, A. Bairoch, A. Bateman, E. Birney *et al.* (**2001**). The InterPro database, an integrated documentation resource for protein families, domains and functional sites. *Nucleic Acids Res*, 29(1):37–40.

Aravind, L. & E. V. Koonin (**1999**). Gleaning non-trivial structural, functional and evolutionary information about proteins by iterative database searches. *J Mol Biol*, 287(5):1023–1040.

Ashburner, M., C. A. Ball, J. A. Blake, D. Botstein, H. Butler *et al.* (**2000**). Gene ontology: tool for the unification of biology. The Gene Ontology Consortium. *Nat Genet*, 25(1):25–29.

Attwood, T. K., M. J. Blythe, D. R. Flower, A. Gaulton, J. E. Mabey *et al.* (**2002**). PRINTS and PRINTS-S shed light on protein ancestry. *Nucleic Acids Res*, 30(1):239–241.

Bairoch, A. & R. Apweiler (**2000**). The SWISS-PROT protein sequence database and its supplement TrEMBL in 2000. *Nucleic Acids Res*, 28(1):45–48.

Bargmann, C. I. (**1998**). Neurobiology of the Caenorhabditis elegans genome. *Science*, 282(5396):2028–2033.

Barker, W. C., J. S. Garavelli, H. Huang, P. B. McGarvey, B. C. Orcutt *et al.* (**2000**). The protein information resource (PIR). *Nucleic Acids Res*, 28(1):41–44.

Bashton, M. & C. Chothia (**2002**). The geometry of domain combination in proteins. *J Mol Biol*, 315(4):927–939.

Bateman, A., E. Birney, L. Cerruti, R. Durbin, L. Etwiller *et al.* (**2002**). The Pfam protein families database. *Nucleic Acids Res*, 30(1):276–280.

Bateman, A., E. Birney, R. Durbin, S. R. Eddy, R. D. Finn *et al.* (**1999**). Pfam 3.1: 1313 multiple alignments and profile HMMs match the majority of proteins. *Nucleic Acids Res*, 27(1):260–262.

Bates, P. A., R. M. Jackson & M. J. Sternberg (**1997**). Model building by comparison: a combination of expert knowledge and computer automation. *Proteins*, Suppl 1:59–67.

Bates, P. A. & M. J. Sternberg (**1999**). Model building by comparison at CASP3: Using expert knowledge and computer automation. *Proteins*, 37(S3):47–54.

Bax, B., P. S. Carter, C. Lewis, A. R. Guy, A. Bridges *et al.* (**2001**). The structure of phosphorylated GSK-3beta complexed with a peptide, FRATtide, that inhibits beta-catenin phosphorylation. *Structure*, 9(12):1143–1152.

Bellman, R. (**1957**). *Dynamic Programming*. Princeton University Press, Princeton, New Jersey, USA.

Benson, D. A., I. Karsch-Mizrachi, D. J. Lipman, J. Ostell, B. A. Rapp *et al.* (**2002**). GenBank. *Nucleic Acids Res*, 30(1):17–20.

Berman, H. M., J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat *et al.* (**2000**). The Protein Data Bank. *Nucleic Acids Res*, 28(1):235–242.

Bernal, A., U. Ear & N. Kyrpides (**2001**). Genomes OnLine Database (GOLD): a monitor of genome projects world-wide. *Nucleic Acids Res*, 29(1):126–127.

Blake, C. C., D. F. Koenig, G. A. Mair, A. C. North, D. C. Phillips *et al.* (**1965**). Structure of hen egg-white lysozyme. A three-dimensional Fourier synthesis at 2 Angstrom resolution. *Nature*, 206(986):757–761.

Blattner, F. R., G. r. Plunkett, C. A. Bloch, N. T. Perna, V. Burland *et al.* (**1997**). The complete genome sequence of Escherichia coli K-12. *Science*, 277(5331):1453–1474. Comment.

Bolotin, A., P. Wincker, S. Mauger, O. Jaillon, K. Malarme *et al.* (**2001**). The complete genome sequence of the lactic acid bacterium Lactococcus lactis ssp. lactis IL1403. *Genome Res*, 11(5):731–753.

Bork, P., T. Dandekar, Y. Diaz-Lazcoz, F. Eisenhaber, M. Huynen *et al.* (**1998**). Predicting function: from genes to genomes and back. *J Mol Biol*, 283(4):707–725.

Bork, P. & E. V. Koonin (**1998**). Predicting functions from protein sequences–where are the bottlenecks. *Nat Genet*, 18(4):313–318.

Bowie, J. U., R. Luthy & D. Eisenberg (**1991**). A method to identify protein sequences that fold into a known three-dimensional structure. *Science*, 253(5016):164–170.

Bowman, S., D. Lawson, D. Basham, D. Brown, T. Chillingworth *et al.* (**1999**). The complete nucleotide sequence of chromosome 3 of Plasmodium falciparum. *Nature*, 400(6744):532–538.

Brenner, S. E., D. Barken & M. Levitt (**1999**). The PRESAGE database for structural genomics. *Nucleic Acids Res*, 27(1):251–253.

Brenner, S. E., C. Chothia & T. J. Hubbard (**1997**). Population statistics of protein structures: lessons from structural classifications. *Curr Opin Struct Biol*, 7(3):369–376.

Brenner, S. E., C. Chothia & T. J. Hubbard (**1998**). Assessing sequence comparison methods with reliable structurally identified distant evolutionary relationships. *Proc Natl Acad Sci U S A*, 95(11):6073–6078.

Buchan, D. W. A., A. J. Shepherd, D. Lee, F. M. G. Pearl, S. C. G. Rison *et al.* (**2002**). Gene3D: structural assignment for whole genes and genomes using the CATH domain structure database. *Genome Res*, 12(3):503–514.

Bujacz, G., M. Jaskolski, J. Alexandratos, A. Wlodawer, G. Merkel *et al.* (**1996**). The catalytic domain of avian sarcoma virus integrase: conformation of the active-site residues in the presence of divalent cations. *Structure*, 4(1):89–96.

Bult, C. J., O. White, G. J. Olsen, L. Zhou, R. D. Fleischmann *et al.* (**1996**). Complete genome sequence of the methanogenic archaeon, Methanococcus jannaschii. *Science*, 273(5278):1058–1073.

Burge, C. & S. Karlin (**1997**). Prediction of complete gene structures in human genomic DNA. *J Mol Biol*, 268(1):78–94.

Burley, S. K. (**2000**). An overview of structural genomics. *Nat Struct Biol*, 7 Suppl:932–934.

Chambaud, I., R. Heilig, S. Ferris, V. Barbe, D. Samson *et al.* (**2001**). The complete genome sequence of the murine respiratory pathogen Mycoplasma pulmonis. *Nucleic Acids Res*, 29(10):2145–2153.

Chandonia, J. M., N. S. Walker, L. Lo Conte, P. Koehl, M. Levitt *et al.* (**2002**). ASTRAL compendium enhancements. *Nucleic Acids Res*, 30(1):260–263.

Chen, P. P.-S. (**1976**). The Entity-Relationship Model - Toward a Unified View of Data. *ACM Transaction of Database Systems*, 1(1):9–36.

Chothia, C. & A. M. Lesk (**1986**). The relation between the divergence of sequence and structure in proteins. *EMBO J*, 5(4):823–826.

Cole, S. T., R. Brosch, J. Parkhill, T. Garnier, C. Churcher *et al.* (**1998**). Deciphering the biology of Mycobacterium tuberculosis from the complete genome sequence. *Nature*, 393(6685):537–544.

Cole, S. T., K. Eiglmeier, J. Parkhill, K. D. James, N. R. Thomson *et al.* (**2001**). Massive gene decay in the leprosy bacillus. *Nature*, 409(6823):1007–1011.

Connolly, T., C. Begg & A. Strachan (**1998**). *Database Systems*. Addison-Wesley, Harlow, England, 2nd edition. ISBN 0-201-34287-1.

Conte, L. L., S. E. Brenner, T. J. P. Hubbard, C. Chothia & A. G. Murzin (**2002**). SCOP database in 2002: refinements accommodate structural genomics. *Nucleic Acids Res*, 30(1):264–267.

Corpet, F., F. Servant, J. Gouzy & D. Kahn (**2000**). ProDom and ProDom-CG: tools for protein domain analysis and whole genome comparisons. *Nucleic Acids Res*, 28(1):267–269.

Cross, D. A., D. R. Alessi, P. Cohen, M. Andjelkovich & B. A. Hemmings (**1995**). Inhibition of glycogen synthase kinase-3 by insulin mediated by protein kinase B. *Nature*, 378(6559):785–789.

Dajani, R., E. Fraser, S. M. Roe, N. Young, V. Good *et al.* (**2001**). Crystal structure of glycogen synthase kinase 3 beta: structural basis for phosphate-primed substrate specificity and autoinhibition. *Cell*, 105(6):721–732.

Davies, J. F., Z. Hostomska, Z. Hostomsky, S. R. Jordan & D. A. Matthews (**1991**). Crystal structure of the ribonuclease H domain of HIV-1 reverse transcriptase. *Science*, 252(5002):88–95.

Dayhoff, M. O., R. M. Schwartz & B. C. Orcutt (**1978**). *Atlas of Protein Sequence and Structure*, volume 5 of *3*, pages 345–352. Natl. Biomed. Res. Found., Washington, DC.

Deckert, G., P. V. Warren, T. Gaasterland, W. G. Young, A. L. Lenox *et al.* (**1998**). The complete genome of the hyperthermophilic bacterium Aquifex aeolicus. *Nature*, 392(6674):353–358.

Devos, D. & A. Valencia (**2000**). Practical limits of function prediction. *Proteins*, 41(1):98–107.

Diamond, R. (**1974**). Real-space refinement of the structure of hen egg-white lysozyme. *J Mol Biol*, 82(3):371–391.

Douglas, S., S. Zauner, M. Fraunholz, M. Beaton, S. Penny *et al.* (**2001**). The highly reduced genome of an enslaved algal nucleus. *Nature*, 410(6832):1091–1096.

Dowell, R. D., R. M. Jokerst, A. Day, S. R. Eddy & S. L. (**2001**). The Distributed Annotation System. *BMC Bioinformatics*, 2(1):7.

Dyda, F., A. B. Hickman, T. M. Jenkins, A. Engelman, R. Craigie *et al.* (**1994**). Crystal structure of the catalytic domain of HIV-1 integrase: similarity to other polynucleotidyl transferases. *Science*, 266(5193):1981–1986.

Eddy, S. R. (**1998**). Profile hidden Markov models. *Bioinformatics*, 14(9):755–763.

Falquet, L., M. Pagni, P. Bucher, N. Hulo, C. J. A. Sigrist *et al.* (**2002**). The PROSITE database, its status in 2002. *Nucleic Acids Res*, 30(1):235–238.

Ferretti, J. J., W. M. McShan, D. Ajdic, D. J. Savic, G. Savic *et al.* (**2001**). Complete genome sequence of an M1 strain of Streptococcus pyogenes. *Proc Natl Acad Sci U S A*, 98(8):4658–4663.

Fiol, C. J., J. S. Williams, C. H. Chou, Q. M. Wang, P. J. Roach *et al.* (**1994**). A secondary phosphorylation of CREB341 at Ser129 is required for the cAMP-mediated control of gene expression. A role for glycogen synthase kinase-3 in the control of gene expression. *J Biol Chem*, 269(51):32187–32193.

Fischer, D., C. Barret, K. Bryson, A. Elofsson, A. Godzik *et al.* (**1999**). CAFASP-1: critical assessment of fully automated structure prediction methods. *Proteins*, Suppl 3:209–217.

Fischer, D. & D. Eisenberg (**1997**). Assigning folds to the proteins encoded by the genome of Mycoplasma genitalium. *Proc Natl Acad Sci U S A*, 94(22):11929–11934.

Fleischmann, R. D., M. D. Adams, O. White, R. A. Clayton, E. F. Kirkness *et al.* (**1995**). Whole-genome random sequencing and assembly of Haemophilus influenzae Rd. *Science*, 269(5223):496–512.

Fraser, C. M., S. Casjens, W. M. Huang, G. G. Sutton, R. Clayton *et al.* (**1997**). Genomic sequence of a Lyme disease spirochaete, Borrelia burgdorferi. *Nature*, 390(6660):580–586.

Fraser, C. M., J. D. Gocayne, O. White, M. D. Adams, R. A. Clayton *et al.* (**1995**). The minimal gene complement of Mycoplasma genitalium. *Science*, 270(5235):397–403.

Fraser, C. M., S. J. Norris, G. M. Weinstock, O. White, G. G. Sutton *et al.* (**1998**). Complete genome sequence of Treponema pallidum, the syphilis spirochete. *Science*, 281(5375):375–388.

Frishman, D., K. Albermann, J. Hani, K. Heumann, A. Metanomski *et al.* (**2001**). Functional and structural genomics using PEDANT. *Bioinformatics*, 17(1):44–57.

Gaasterland, T. & C. W. Sensen (**1996**). Fully automated genome analysis that reflects user needs and preferences. A detailed introduction to the MAGPIE system architecture. *Biochimie*, 78(5):302–310.

Galibert, F., T. M. Finan, S. R. Long, A. Puhler, P. Abola *et al.* (**2001**). The composite genome of the legume symbiont Sinorhizobium meliloti. *Science*, 293(5530):668–672.

Gardner, M. J., H. Tettelin, D. J. Carucci, L. M. Cummings, L. Aravind *et al.* (**1998**). Chromosome 2 sequence of the human malaria parasite Plasmodium falciparum. *Science*, 282(5391):1126–1132.

Gerstein, M. (**1997**). A structural census of genomes: comparing bacterial, eukaryotic, and archaeal genomes in terms of protein structure. *J Mol Biol*, 274(4):562–576.

Gerstein, M. (**1998**a). How representative are the known structures of the proteins in a complete genome? A comprehensive structural census. *Fold Des*, 3(6):497–512.

Gerstein, M. (**1998**b). Patterns of protein-fold usage in eight microbial genomes: a comprehensive structural census. *Proteins*, 33(4):518–534.

Gerstein, M. & M. Levitt (**1997**). A structural census of the current population of protein sequences. *Proc Natl Acad Sci U S A*, 94(22):11911–11916.

Glaser, P., L. Frangeul, C. Buchrieser, C. Rusniok, A. Amend *et al.* (**2001**). Comparative genomics of Listeria species. *Science*, 294(5543):849–852.

Glass, J. I., E. J. Lefkowitz, J. S. Glass, C. R. Heiner, E. Y. Chen *et al.* (**2000**). The complete sequence of the mucosal pathogen Ureaplasma urealyticum. *Nature*, 407(6805):757–762.

Gough, J. & C. Chothia (**2002**). SUPERFAMILY: HMMs representing all proteins of known structure. SCOP sequence searches, alignments and genome assignments. *Nucleic Acids Res*, 30(1):268–272.

Govindarajan, S., R. Recabarren & R. A. Goldstein (**1999**). Estimating the total number of protein folds. *Proteins*, 35(4):408–414.

Gracy, J. & P. Argos (**1998**). DOMO: a new database of aligned protein domains. *Trends Biochem Sci*, 23(12):495–497.

Gribskov, M., A. D. McLachlan & D. Eisenberg (**1987**). Profile analysis: detection of distantly related proteins. *Proc Natl Acad Sci U S A*, 84(13):4355–4358.

Hayashi, T., K. Makino, M. Ohnishi, K. Kurokawa, K. Ishii *et al.* (**2001**). Complete genome sequence of enterohemorrhagic Escherichia coli O157:H7 and genomic comparison with a laboratory strain K-12. *DNA Res*, 8(1):11–22.

Hegyi, H. & M. Gerstein (**1999**). The relationship between protein structure and function: a comprehensive survey with application to the yeast genome. *J Mol Biol*, 288(1):147–164.

Hegyi, H. & M. Gerstein (**2001**). Annotation transfer for genomics: measuring functional divergence in multi-domain proteins. *Genome Res*, 11(10):1632–1640.

Heidelberg, J. F., J. A. Eisen, W. C. Nelson, R. A. Clayton, M. L. Gwinn *et al.* (**2000**). DNA sequence of both chromosomes of the cholera pathogen Vibrio cholerae. *Nature*, 406(6795):477–483.

Hendlich, M., P. Lackner, S. Weitckus, H. Floeckner, R. Froschauer *et al.* (**1990**). Identification of native protein folds amongst a large number of incorrect models. The calculation of low energy conformations from potentials of mean force. *J Mol Biol*, 216(1):167–180.

Henikoff, J. G., E. A. Greene, S. Pietrokovski & S. Henikoff (**2000**). Increased coverage of protein families with the blocks database servers. *Nucleic Acids Res*, 28(1):228–230.

Henikoff, S. & J. G. Henikoff (**1992**). Amino acid substitution matrices from protein blocks. *Proc Natl Acad Sci U S A*, 89(22):10915–10919.

Henikoff, S. & J. G. Henikoff (**1993**). Performance evaluation of amino acid substitution matrices. *Proteins*, 17(1):49–61.

Henikoff, S., J. G. Henikoff, W. J. Alford & S. Pietrokovski (**1995**). Automated construction and graphical presentation of protein blocks from unaligned sequences. *Gene*, 163(2):GC17–26.

Henikoff, S., J. G. Henikoff & S. Pietrokovski (**1999**). Blocks+: a non-redundant database of protein alignment blocks derived from multiple compilations. *Bioinformatics*, 15(6):471–479.

Himmelreich, R., H. Hilbert, H. Plagens, E. Pirkl, B. C. Li *et al.* (**1996**). Complete sequence analysis of the genome of the bacterium Mycoplasma pneumoniae. *Nucleic Acids Res*, 24(22):4420–4449.

Holm, L. & C. Sander (**1997**). An evolutionary treasure: unification of a broad set of amidohydrolases related to urease. *Proteins*, 28(1):72–82.

Hoskins, J., W. E. J. Alborn, J. Arnold, L. C. Blaszczak, S. Burgett *et al.* (**2001**). Genome of the bacterium Streptococcus pneumoniae strain R6. *J Bacteriol*, 183(19):5709–5717.

Hubbard, T., D. Barker, E. Birney, G. Cameron, Y. Chen *et al.* (**2002**). The Ensembl genome database project. *Nucleic Acids Res*, 30(1):38–41.

Huynen, M., T. Doerks, F. Eisenhaber, C. Orengo, S. Sunyaev *et al.* (**1998**). Homology-based fold predictions for Mycoplasma genitalium proteins. *J Mol Biol*, 280(3):323–326.

Iliopoulos, I., S. Tsoka, M. A. Andrade, P. Janssen, B. Audit *et al.* (**2001**). Genome sequences and great expectations. *Genome Biol*, 2(1):INTERACTIONS0001. Letter.

Jolles, P., F. Schoentgen, J. Jolles, D. E. Dobson, E. M. Prager *et al.* (**1984**). Stomach lysozymes of ruminants. II. Amino acid sequence of cow lysozyme 2 and immunological comparisons with other lysozymes. *J Biol Chem*, 259(18):11617–11625.

Jonassen, I., I. Eidhammer & W. R. Taylor (**1999**). Discovery of local packing motifs in protein structures. *Proteins*, 34(2):206–219.

Jones, D. T. (**1997**). Progress in protein structure prediction. *Curr Opin Struct Biol*, 7(3):377–387.

Jones, D. T. (**1999**a). GenTHREADER: an efficient and reliable protein fold recognition method for genomic sequences. *J Mol Biol*, 287(4):797–815.

Jones, D. T. (**1999**b). Protein secondary structure prediction based on position-specific scoring matrices. *J Mol Biol*, 292(2):195–202.

Jones, D. T., W. R. Taylor & J. M. Thornton (**1992**). A new approach to protein fold recognition. *Nature*, 358(6381):86–89.

Kalman, S., W. Mitchell, R. Marathe, C. Lammel, J. Fan *et al.* (**1999**). Comparative genomes of Chlamydia pneumoniae and C. trachomatis. *Nat Genet*, 21(4):385–389.

Kanehisa, M., S. Goto, S. Kawashima & A. Nakaya (**2002**). The KEGG databases at GenomeNet. *Nucleic Acids Res*, 30(1):42–46.

Kaneko, T., Y. Nakamura, S. Sato, E. Asamizu, T. Kato *et al.* (**2000**). Complete genome structure of the nitrogen-fixing symbiotic bacterium Mesorhizobium loti. *DNA Res*, 7(6):331–338.

Kaneko, T., S. Sato, H. Kotani, A. Tanaka, E. Asamizu *et al.* (**1996**). Sequence analysis of the genome of the unicellular cyanobacterium Synechocystis sp. strain PCC6803. II. Sequence determination of the entire genome and assignment of potential protein-coding regions. *DNA Res*, 3(3):109–136.

Karlin, S. & S. F. Altschul (**1990**). Methods for assessing the statistical significance of molecular sequence features by using general scoring schemes. *Proc Natl Acad Sci U S A*, 87(6):2264–2268.

Karlin, S. & S. F. Altschul (**1993**). Applications and statistics for multiple high-scoring segments in molecular sequences. *Proc Natl Acad Sci U S A*, 90(12):5873–5877.

Karp, P. D. (**1998**). What we do not know about sequence analysis and sequence databases. *Bioinformatics*, 14(9):753–754. Editorial.

Karp, P. D., M. Riley, M. Saier, I. T. Paulsen, J. Collado-Vides *et al.* (**2002**). The EcoCyc Database. *Nucleic Acids Res*, 30(1):56–58.

Karplus, K., C. Barrett & R. Hughey (**1998**). Hidden Markov models for detecting remote protein homologies. *Bioinformatics*, 14(10):846–856.

Katayanagi, K., M. Miyagawa, M. Matsushima, M. Ishikawa, S. Kanaya *et al.* (**1990**). Three-dimensional structure of ribonuclease H from E. coli. *Nature*, 347(6290):306–309.

Katayanagi, K., M. Okumura & K. Morikawa (**1993**). Crystal structure of Escherichia coli RNase HI in complex with Mg2+ at 2.8 A resolution: proof for a single Mg(2+)-binding site. *Proteins*, 17(4):337–346.

Kawarabayasi, Y., Y. Hino, H. Horikawa, K. Jin-no, M. Takahashi *et al.* (**2001**). Complete genome sequence of an aerobic thermoacidophilic crenarchaeon, Sulfolobus tokodaii strain7. *DNA Res*, 8(4):123–140.

Kawarabayasi, Y., Y. Hino, H. Horikawa, S. Yamazaki, Y. Haikawa *et al.* (**1999**). Complete genome sequence of an aerobic hyper-thermophilic crenarchaeon, Aeropyrum pernix K1. *DNA Res*, 6(2):83–101, 145–52.

Kawarabayasi, Y., M. Sawada, H. Horikawa, Y. Haikawa, Y. Hino *et al.* (**1998**). Complete sequence and gene organization of the genome of a hyper-thermophilic archaebacterium, Pyrococcus horikoshii OT3. *DNA Res*, 5(2):55–76.

Kawashima, T., N. Amano, H. Koike, S. Makino, S. Higuchi *et al.* (**2000**). Archaeal adaptation to higher temperatures revealed by genomic sequence of Thermoplasma volcanium. *Proc Natl Acad Sci U S A*, 97(26):14257–14262.

Kelley, L. A., R. M. MacCallum & M. J. Sternberg (**2000**). Enhanced genome annotation using structural profiles in the program 3D-PSSM. *J Mol Biol*, 299(2):499–520.

Klenk, H. P., R. A. Clayton, J. F. Tomb, O. White, K. E. Nelson *et al.* (**1997**). The complete genome sequence of the hyperthermophilic, sulphate-reducing archaeon Archaeoglobus fulgidus. *Nature*, 390(6658):364–370.

Koehl, P. & M. Levitt (**1999**). A brighter future for protein structure prediction. *Nat Struct Biol*, 6(2):108–111. Congresses.

Koonin, E. V., Y. I. Wolf & L. Aravind (**2000**). Protein fold recognition using sequence profiles and its application in structural genomics. *Adv Protein Chem*, 54:245–275.

Krogh, A., M. Brown, I. S. Mian, K. Sjolander & D. Haussler (**1994**). Hidden Markov models in computational biology. Applications to protein modeling. *J Mol Biol*, 235(5):1501–1531.

Krogh, A., B. Larsson, G. von Heijne & E. L. Sonnhammer (**2001**). Predicting transmembrane protein topology with a hidden Markov model: application to complete genomes. *J Mol Biol*, 305(3):567–580.

Kulkosky, J., K. S. Jones, R. A. Katz, J. P. Mack & A. M. Skalka (**1992**). Residues critical for retroviral integrative recombination in a region that is highly conserved among retroviral/retrotransposon integrases and bacterial insertion sequence transposases. *Mol Cell Biol*, 12(5):2331–2338.

Kunst, F., N. Ogasawara, I. Moszer, A. M. Albertini, G. Alloni *et al.* (**1997**). The complete genome sequence of the gram-positive bacterium Bacillus subtilis. *Nature*, 390(6657):249–256.

Kuroda, M., T. Ohta, I. Uchiyama, T. Baba, H. Yuzawa *et al.* (**2001**). Whole genome sequencing of meticillin-resistant Staphylococcus aureus. *Lancet*, 357(9264):1225–1240.

Lander, E. S., L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody *et al.* (**2001**). Initial sequencing and analysis of the human genome. *Nature*, 409(6822):860–921.

Letunic, I., L. Goodstadt, N. J. Dickens, T. Doerks, J. Schultz *et al.* (**2002**). Recent improvements to the SMART domain-based sequence annotation resource. *Nucleic Acids Res*, 30(1):242–244.

Lewis, S., M. Ashburner & M. G. Reese (**2000**). Annotating eukaryote genomes. *Curr Opin Struct Biol*, 10(3):349–354.

Lupas, A., M. Van Dyke & J. Stock (**1991**). Predicting coiled coils from protein sequences. *Science*, 252(5010):1162–1164.

Marchler-Bauer, A., A. R. Panchenko, B. A. Shoemaker, P. A. Thiessen, L. Y. Geer *et al.* (**2002**). CDD: a database of conserved domain alignments with links to domain three-dimensional structure. *Nucleic Acids Res*, 30(1):281–283.

Martin, A. C., C. A. Orengo, E. G. Hutchinson, S. Jones, M. Karmirantzou *et al.* (**1998**). Protein folds and functions. *Structure*, 6(7):875–884.

May, B. J., Q. Zhang, L. L. Li, M. L. Paustian, T. S. Whittam *et al.* (**2001**). Complete genomic sequence of Pasteurella multocida, Pm70. *Proc Natl Acad Sci U S A*, 98(6):3460–3465.

McClelland, M., K. E. Sanderson, J. Spieth, S. W. Clifton, P. Latreille *et al.* (**2001**). Complete genome sequence of Salmonella enterica serovar Typhimurium LT2. *Nature*, 413(6858):852–856.

McGuffin, L. J., K. Bryson & D. T. Jones (**2000**). The PSIPRED protein structure prediction server. *Bioinformatics*, 16(4):404–405.

McKenzie, H. A. (**1996**). alpha-Lactalbumins and lysozymes. *EXS*, 75:365–409.

McKenzie, H. A. & F. H. J. White (**1991**). Lysozyme and alpha-lactalbumin: structure, function, and interrelationships. *Adv Protein Chem*, 41:173–315.

Mott, R. (**2000**). Accurate formula for P-values of gapped local sequence and profile alignments. *J Mol Biol*, 300(3):649–659.

Moult, J. (**1999**). Predicting protein three-dimensional structure. *Curr Opin Biotechnol*, 10(6):583–588.

Muller, A., R. M. MacCallum & M. J. Sternberg (**1999**). Benchmarking PSI-BLAST in genome annotation. *J Mol Biol*, 293(5):1257–1271.

Murzin, A. G., S. E. Brenner, T. Hubbard & C. Chothia (**1995**). SCOP: a structural classification of proteins database for the investigation of sequences and structures. *J Mol Biol*, 247(4):536–540.

Mushegian, A. R. & E. V. Koonin (**1996**). A minimal gene set for cellular life derived by comparison of complete bacterial genomes. *Proc Natl Acad Sci U S A*, 93(19):10268–10273.

Myler, P. J., L. Audleman, T. deVos, G. Hixson, P. Kiser *et al.* (**1999**). Leishmania major Friedlin chromosome 1 has an unusual distribution of protein-coding genes. *Proc Natl Acad Sci U S A*, 96(6):2902–2906.

Nakamura, S., Y. Tsuji, Y. Nakata, S. Komori & K. Koyama (**1994**). Identification and characterization of a sperm peptide antigen recognized by a monoclonal antisperm autoantibody derived from a vasectomized mouse. *Biochem Biophys Res Commun*, 205(3):1503–1509.

Needleman, S. B. & C. D. Wunsch (**1970**). A general method applicable to the search for similarities in the amino acid sequence of two proteins. *J Mol Biol*, 48(3):443–453.

Nelson, K. E., R. A. Clayton, S. R. Gill, M. L. Gwinn, R. J. Dodson *et al.* (**1999**). Evidence for lateral gene transfer between Archaea and bacteria from genome sequence of Thermotoga maritima. *Nature*, 399(6734):323–329.

Ng, W. V., S. P. Kennedy, G. G. Mahairas, B. Berquist, M. Pan *et al.* (**2000**). Genome sequence of Halobacterium species NRC-1. *Proc Natl Acad Sci U S A*, 97(22):12176–12181.

Nielsen, H., J. Engelbrecht, S. Brunak & G. von Heijne (**1997**). A neural network method for identification of prokaryotic and eukaryotic signal peptides and prediction of their cleavage sites. *Int J Neural Syst*, 8(5-6):581–599.

Nierman, W. C., T. V. Feldblyum, M. T. Laub, I. T. Paulsen, K. E. Nelson *et al.* (**2001**). Complete genome sequence of Caulobacter crescentus. *Proc Natl Acad Sci U S A*, 98(7):4136–4141.

Nitta, K. (**2002**). Alpha-lactalbumin and (calcium-binding) lysozyme. *Methods Mol Biol*, 172:211–224.

Nitta, K., H. Tsuge, K. Shimazaki & S. Sugai (**1988**). Calcium-binding lysozymes. *Biol Chem Hoppe Seyler*, 369(8):671–675.

No authors listed (**1997**). The yeast genome directory. *Nature*, 387(6632 Suppl). Directory.

Nolling, J., G. Breton, M. V. Omelchenko, K. S. Makarova, Q. Zeng *et al.* (**2001**). Genome sequence and comparative analysis of the solvent-producing bacterium Clostridium acetobutylicum. *J Bacteriol*, 183(16):4823–4838.

Ogata, H., S. Audic, P. Renesto-Audiffren, P. E. Fournier, V. Barbe *et al.* (**2001**). Mechanisms of evolution in Rickettsia conorii and R. prowazekii. *Science*, 293(5537):2093–2098.

Orengo, C. A., N. P. Brown & W. R. Taylor (**1992**). Fast structure alignment for protein databank searching. *Proteins*, 14(2):139–167.

Orengo, C. A., A. D. Michie, S. Jones, D. T. Jones, M. B. Swindells *et al.* (**1997**). CATH–a hierarchic classification of protein domain structures. *Structure*, 5(8):1093–1108.

Orengo, C. A., A. E. Todd & J. M. Thornton (**1999**). From protein structure to function. *Curr Opin Struct Biol*, 9(3):374–382.

Park, J., K. Karplus, C. Barrett, R. Hughey, D. Haussler *et al.* (**1998**). Sequence comparisons using multiple sequences detect three times as many remote homologues as pairwise methods. *J Mol Biol*, 284(4):1201–1210.

Parkhill, J., M. Achtman, K. D. James, S. D. Bentley, C. Churcher *et al.* (**2000**a). Complete DNA sequence of a serogroup A strain of Neisseria meningitidis Z2491. *Nature*, 404(6777):502–506.

Parkhill, J., G. Dougan, K. D. James, N. R. Thomson, D. Pickard *et al.* (**2001**a). Complete genome sequence of a multiple drug resistant Salmonella enterica serovar Typhi CT18. *Nature*, 413(6858):848–852.

Parkhill, J., B. W. Wren, K. Mungall, J. M. Ketley, C. Churcher *et al.* (**2000**b). The genome sequence of the food-borne pathogen Campylobacter jejuni reveals hypervariable sequences. *Nature*, 403(6770):665–668.

Parkhill, J., B. W. Wren, N. R. Thomson, R. W. Titball, M. T. Holden *et al.* (**2001**b). Genome sequence of Yersinia pestis, the causative agent of plague. *Nature*, 413(6855):523–527.

Patthy, L. (**1987**). Detecting homology of distantly related proteins with consensus sequences. *J Mol Biol*, 198(4):567–577.

Pearl, F. M., N. Martin, J. E. Bray, D. W. Buchan, A. P. Harrison *et al.* (**2001**). A rapid classification protocol for the CATH Domain Database to support structural genomics. *Nucleic Acids Res*, 29(1):223–227.

Pearson, H. (**2001**). Biology's name game. *Nature*, 411(6838):631–632.

Pearson, W. R. (**1990**). Rapid and sensitive sequence comparison with FASTP and FASTA. *Methods Enzymol*, 183:63–98.

Pearson, W. R. & D. J. Lipman (**1988**). Improved tools for biological sequence comparison. *Proc Natl Acad Sci U S A*, 85(8):2444–2448.

Perna, N. T., G. r. Plunkett, V. Burland, B. Mau, J. D. Glasner *et al.* (**2001**). Genome sequence of enterohaemorrhagic Escherichia coli O157:H7. *Nature*, 409(6819):529–533.

Prager, E. M. & A. C. Wilson (**1988**). Ancient origin of lactalbumin from lysozyme: analysis of DNA and amino acid sequences. *J Mol Evol*, 27(4):326–335.

Qian, J., N. M. Luscombe & M. Gerstein (**2001**a). Protein family and fold occurrence in genomes: power-law behaviour and evolutionary model. *J Mol Biol*, 313(4):673–681.

Qian, J., B. Stenger, C. A. Wilson, J. Lin, R. Jansen *et al.* (**2001**b). PartsList: a web-based system for dynamically ranking protein folds based on disparate attributes, including whole-genome expression and interaction information. *Nucleic Acids Res*, 29(8):1750–1764.

Read, T. D., R. C. Brunham, C. Shen, S. R. Gill, J. F. Heidelberg *et al.* (**2000**). Genome sequences of Chlamydia trachomatis MoPn and Chlamydia pneumoniae AR39. *Nucleic Acids Res*, 28(6):1397–1406.

Reboul, J., P. Vaglio, N. Tzellas, N. Thierry-Mieg, T. Moore *et al.* (**2001**). Open-reading-frame sequence tags (OSTs) support the existence of at least 17,300 genes in C. elegans. *Nat Genet*, 27(3):332–336.

Reese, M. G., G. Hartzell, N. L. Harris, U. Ohler, J. F. Abril *et al.* (**2000**). Genome annotation assessment in Drosophila melanogaster. *Genome Res*, 10(4):483–501.

Roberts, L. (**1991**). GRAIL seeks out genes buried in DNA sequence. *Science*, 254(5033):805. News.

Robinson, A. B. & L. R. Robinson (**1991**). Distribution of glutamine and asparagine residues and their near neighbors in peptides and proteins. *Proc Natl Acad Sci U S A*, 88(20):8880–8884.

Rost, B. (**1996**). PHD: predicting one-dimensional protein structure by profile-based neural networks. *Methods Enzymol*, 266:525–539.

Rost, B. (**1999**). Twilight zone of protein sequence alignments. *Protein Eng*, 12(2):85–94.

Rost, B. & C. Sander (**1993**a). Improved prediction of protein secondary structure by use of sequence profiles and neural networks. *Proc Natl Acad Sci U S A*, 90(16):7558–7562.

Rost, B. & C. Sander (**1993**b). Prediction of protein secondary structure at better than 70accuracy. *J Mol Biol*, 232(2):584–599.

Rubin, G. M., M. D. Yandell, J. R. Wortman, G. L. Gabor Miklos., C. R. Nelson *et al.* (**2000**). Comparative genomics of the eukaryotes. *Science*, 287(5461):2204–2215.

Ruepp, A., W. Graml, M. L. Santos-Martinez, K. K. Koretke, C. Volker *et al.* (**2000**). The genome sequence of the thermoacidophilic scavenger Thermoplasma acidophilum. *Nature*, 407(6803):508–513.

Russell, R. B. (**1998**). Detection of protein three-dimensional side-chain patterns: new examples of convergent evolution. *J Mol Biol*, 279(5):1211–1227.

Russell, R. B., M. A. Saqi, P. A. Bates, R. A. Sayle & M. J. Sternberg (**1998**a). Recognition of analogous and homologous protein folds–assessment of prediction success and associated alignment accuracy using empirical substitution matrices. *Protein Eng*, 11(1):1–9.

Russell, R. B., P. D. Sasieni & M. J. Sternberg (**1998**b). Supersites within superfolds. Binding site similarity in the absence of homology. *J Mol Biol*, 282(4):903–918.

Rychlewski, L., B. Zhang & A. Godzik (**1998**). Fold and function predictions for Mycoplasma genitalium proteins. *Fold Des*, 3(4):229–238.

Salamov, A. A., M. Suwa, C. A. Orengo & M. B. Swindells (**1999**). Genome analysis: Assigning protein coding regions to three-dimensional structures. *Protein Sci*, 8(4):771–777.

Sali, A. & T. L. Blundell (**1993**). Comparative protein modelling by satisfaction of spatial restraints. *J Mol Biol*, 234(3):779–815.

Sanchez, R., U. Pieper, F. Melo, N. Eswar, M. A. Marti-Renom *et al.* (**2000**). Protein structure modeling for structural genomics. *Nat Struct Biol*, 7 Suppl:986–990.

Sanchez, R. & A. Sali (**1997**a). Advances in comparative protein-structure modelling. *Curr Opin Struct Biol*, 7(2):206–214.

Sanchez, R. & A. Sali (**1997**b). Evaluation of comparative protein structure modeling by MODELLER-3. *Proteins*, Suppl 1:50–58.

Sanchez, R. & A. Sali (**1998**). Large-scale protein structure modeling of the Saccharomyces cerevisiae genome. *Proc Natl Acad Sci U S A*, 95(23):13597–13602.

Sander, C. & R. Schneider (**1991**). Database of homology-derived protein structures and the structural meaning of sequence alignment. *Proteins*, 9(1):56–68.

Schaffer, A. A., L. Aravind, T. L. Madden, S. Shavirin, J. L. Spouge *et al.* (**2001**). Improving the accuracy of PSI-BLAST protein database searches with composition-based statistics and other refinements. *Nucleic Acids Res*, 29(14):2994–3005.

Schaffer, A. A., Y. I. Wolf, C. P. Ponting, E. V. Koonin, L. Aravind *et al.* (**1999**). IMPALA: matching a protein sequence against a collection of PSI-BLAST-constructed position-specific score matrices. *Bioinformatics*, 15(12):1000–1011.

Scharf, M., R. Schneider, G. Casari, P. Bork, A. Valencia *et al.* (**1994**). GeneQuiz: a workbench for sequence analysis. *Proc Int Conf Intell Syst Mol Biol*, 2:348–353.

She, Q., R. K. Singh, F. Confalonieri, Y. Zivanovic, G. Allard *et al.* (**2001**). The complete genome of the crenarchaeon Sulfolobus solfataricus P2. *Proc Natl Acad Sci U S A*, 98(14):7835–7840.

Shewale, J. G., S. K. Sinha & K. Brew (**1984**). Evolution of alpha-lactalbumins. The complete amino acid sequence of the alpha-lactalbumin from a marsupial (Macropus rufogriseus) and corrections to regions of sequence in bovine and goat alpha-lactalbumins. *J Biol Chem*, 259(8):4947–4956.

Shigenobu, S., H. Watanabe, M. Hattori, Y. Sakaki & H. Ishikawa (**2000**). Genome sequence of the endocellular bacterial symbiont of aphids Buchnera sp. APS. *Nature*, 407(6800):81–86.

Shirai, M., H. Hirakawa, M. Kimoto, M. Tabuchi, F. Kishi *et al.* (**2000**). Comparison of whole genome sequences of Chlamydia pneumoniae J138 from Japan and CWL029 from USA. *Nucleic Acids Res*, 28(12):2311–2314.

Simpson, A. J., F. C. Reinach, P. Arruda, F. A. Abreu, M. Acencio *et al.* (**2000**). The genome sequence of the plant pathogen Xylella fastidiosa. The Xylella fastidiosa Consortium of the Organization for Nucleotide Sequencing and Analysis. *Nature*, 406(6792):151–157.

Sippl, M. J. (**1990**). Calculation of conformational ensembles from potentials of mean force. An approach to the knowledge-based prediction of local structures in globular proteins. *J Mol Biol*, 213(4):859–883.

Sippl, M. J. (**1999**). An attempt to analyse progress in fold recognition from CASP1 to CASP3. *Proteins*, 37(S3):226–230.

Skovgaard, M., L. J. Jensen, S. Brunak, D. Ussery & A. Krogh (**2001**). On the total number of genes and their length distribution in complete microbial genomes. *Trends Genet*, 17(8):425–428.

Smith, D. R., L. A. Doucette-Stamm, C. Deloughery, H. Lee, J. Dubois *et al.* (**1997**). Complete genome sequence of Methanobacterium thermoautotrophicum deltaH: functional analysis and comparative genomics. *J Bacteriol*, 179(22):7135–7155.

Smith, S. G., M. Lewis, R. Aschaffenburg, R. E. Fenna, I. A. Wilson *et al.* (**1987**). Crystallographic analysis of the three-dimensional structure of baboon alpha-lactalbumin at low resolution. Homology with lysozyme. *Biochem J*, 242(2):353–360.

Smith, T. F. & M. S. Waterman (**1981**). Identification of common molecular subsequences. *J Mol Biol*, 147(1):195–197.

Sonnhammer, E. L. & D. Kahn (**1994**). Modular arrangement of proteins as inferred from analysis of homology. *Protein Sci*, 3(3):482–492.

Sonnhammer, E. L., G. von Heijne & A. Krogh (**1998**). A hidden Markov model for predicting transmembrane helices in protein sequences. *Proc Int Conf Intell Syst Mol Biol*, 6:175–182.

Sreekumar, K. R., L. Aravind & E. V. Koonin (**2001**). Computational analysis of human disease-associated genes and their protein products. *Curr Opin Genet Dev*, 11(3):247–257.

Stein, L. (**2001**). Genome annotation: from sequence to biology. *Nat Rev Genet*, 2(7):493–503.

Stein, L. (**2002**). Creating a bioinformatics nation. *Nature*, 417(6885):119–210.

Stephens, R. S., S. Kalman, C. Lammel, J. Fan, R. Marathe *et al.* (**1998**). Genome sequence of an obligate intracellular pathogen of humans: Chlamydia trachomatis. *Science*, 282(5389):754–759.

Sternberg, M. J., P. A. Bates, L. A. Kelley & R. M. MacCallum (**1999**). Progress in protein structure prediction: assessment of CASP3. *Curr Opin Struct Biol*, 9(3):368–373. Congresses.

Stoesser, G., W. Baker, A. van Den. Broek., E. Camon, M. Garcia-Pastor *et al.* (**2002**). The EMBL Nucleotide Sequence Database. *Nucleic Acids Res*, 30(1):21–26.

Stover, C. K., X. Q. Pham, A. L. Erwin, S. D. Mizoguchi, P. Warrener *et al.* (**2000**). Complete genome sequence of Pseudomonas aeruginosa PA01, an opportunistic pathogen. *Nature*, 406(6799):959–964.

Takami, H. & K. Horikoshi (**2000**). Analysis of the genome of an alkaliphilic Bacillus strain from an industrial point of view. *Extremophiles*, 4(2):99–108.

Tatusov, R. L., S. F. Altschul & E. V. Koonin (**1994**). Detection of conserved segments in proteins: iterative scanning of sequence databases with alignment blocks. *Proc Natl Acad Sci U S A*, 91(25):12091–12095.

Tatusov, R. L., E. V. Koonin & D. J. Lipman (**1997**). A genomic perspective on protein families. *Science*, 278(5338):631–637.

Tatusov, R. L., D. A. Natale, I. V. Garkavtsev, T. A. Tatusova, U. T. Shankavaram *et al.* (**2001**). The COG database: new developments in phylogenetic classification of proteins from complete genomes. *Nucleic Acids Res*, 29(1):22–28.

Taylor, W. R. (**1986**). Identification of protein sequence homology by consensus template alignment. *J Mol Biol*, 188(2):233–258.

Teichmann, S. A., C. Chothia & M. Gerstein (**1999**). Advances in structural genomics. *Curr Opin Struct Biol*, 9(3):390–399.

Teichmann, S. A., J. Park & C. Chothia (**1998**). Structural assignments to the Mycoplasma genitalium proteins show extensive gene duplications and domain rearrangements. *Proc Natl Acad Sci U S A*, 95(25):14658–14663.

Tettelin, H., K. E. Nelson, I. T. Paulsen, J. A. Eisen, T. D. Read *et al.* (**2001**). Complete genome sequence of a virulent isolate of Streptococcus pneumoniae. *Science*, 293(5529):498–506.

Tettelin, H., N. J. Saunders, J. Heidelberg, A. C. Jeffries, K. E. Nelson *et al.* (**2000**). Complete genome sequence of Neisseria meningitidis serogroup B strain MC58. *Science*, 287(5459):1809–1815.

The Arabidopsis Genome Initiative (**2000**). Analysis of the genome sequence of the flowering plant Arabidopsis thaliana. *Nature*, 408(6814):796–815.

The C. elegans Sequencing Consortium (**1998**). Genome sequence of the nematode C. elegans: a platform for investigating biology. *Science*, 282(5396):2012–2018.

The Gene Ontology Consortium (**2001**). Creating the gene ontology resource: design and implementation. *Genome Res*, 11(8):1425–1433.

Thornton, J. M., C. A. Orengo, A. E. Todd & F. M. Pearl (**1999**). Protein folds, functions and evolution. *J Mol Biol*, 293(2):333–342.

Thornton, J. M., A. E. Todd, D. Milburn, N. Borkakoti & C. A. Orengo (**2000**). From structure to function: approaches and limitations. *Nat Struct Biol*, 7 Suppl:991–994.

Todd, A. E., C. A. Orengo & J. M. Thornton (**2001**). Evolution of function in protein superfamilies, from a structural perspective. *J Mol Biol*, 307(4):1113–1143.

Tomb, J. F., O. White, A. R. Kerlavage, R. A. Clayton, G. G. Sutton *et al.* (**1997**). The complete genome sequence of the gastric pathogen Helicobacter pylori. *Nature*, 388(6642):539–547.

Tusnady, G. E. & I. Simon (**2001**). The HMMTOP transmembrane topology prediction server. *Bioinformatics*, 17(9):849–850.

Uberbacher, E. C. & R. J. Mural (**1991**). Locating protein-coding regions in human DNA sequences by a multiple sensor-neural network approach. *Proc Natl Acad Sci U S A*, 88(24):11261–11265.

Ursing, B. M., F. H. J. van Enckevort, J. A. M. Leunissen & R. J. Siezen (**2002**). EXProt: a database for proteins with an experimentally verified function. *Nucleic Acids Res*, 30(1):50–51.

Venter, J. C., M. D. Adams, E. W. Myers, P. W. Li, R. J. Mural *et al.* (**2001**). The sequence of the human genome. *Science*, 291(5507):1304–1351.

Vitkup, D., E. Melamud, J. Moult & C. Sander (**2001**). Completeness in structural genomics. *Nat Struct Biol*, 8(6):559–566.

von Heijne, G. (**1992**). Membrane protein structure prediction. Hydrophobicity analysis and the positive-inside rule. *J Mol Biol*, 225(2):487–494.

Wallace, A. C., N. Borkakoti & J. M. Thornton (**1997**). TESS: a geometric hashing algorithm for deriving 3D coordinate templates for searching structural databases. Application to enzyme active sites. *Protein Sci*, 6(11):2308–2323.

Waterman, M. S. & M. Vingron (**1994**). Rapid and accurate estimates of statistical significance for sequence data base searches. *Proc Natl Acad Sci U S A*, 91(11):4625–4628.

Wetlaufer, D. B. (**1973**). Nucleation, rapid folding, and globular intrachain regions in proteins. *Proc Natl Acad Sci U S A*, 70(3):697–701.

White, O., J. A. Eisen, J. F. Heidelberg, E. K. Hickey, J. D. Peterson *et al.* (**1999**). Genome sequence of the radioresistant bacterium Deinococcus radiodurans R1. *Science*, 286(5444):1571–1577.

Wilbur, W. J. & D. J. Lipman (**1983**). Rapid similarity searches of nucleic acid and protein data banks. *Proc Natl Acad Sci U S A*, 80(3):726–730.

Wilson, C. A., J. Kreychman & M. Gerstein (**2000**). Assessing annotation transfer for genomics: quantifying the relations between protein sequence, structure and function through traditional and probabilistic scores. *J Mol Biol*, 297(1):233–249.

Wolf, E., P. S. Kim & B. Berger (**1997**). MultiCoil: a program for predicting two- and three-stranded coiled coils. *Protein Sci*, 6(6):1179–1189.

Wolf, Y. I., S. E. Brenner, P. A. Bash & E. V. Koonin (**1999**). Distribution of protein folds in the three superkingdoms of life. *Genome Res*, 9(1):17–26.

Wolf, Y. I., N. V. Grishin & E. V. Koonin (**2000**). Estimating the number of protein folds and families from complete genome data. *J Mol Biol*, 299(4):897–905.

Wood, D. W., J. C. Setubal, R. Kaul, D. E. Monks, J. P. Kitajima *et al.* (**2001**). The genome of the natural genetic engineer Agrobacterium tumefaciens C58. *Science*, 294(5550):2317–2323.

Wood, T. C. & W. R. Pearson (**1999**). Evolution of protein sequences and structures. *J Mol Biol*, 291(4):977–995.

Wood, V., R. Gwilliam, M.-A. Rajandream, M. Lyne, R. Lyne *et al.* (**2002**). The genome sequence of Schizosaccharomyces pombe. *Nature*, 415(6874):871–880.

Wootton, J. C. (**1994**). Non-globular domains in protein sequences: automated segmentation using complexity measures. *Comput Chem*, 18(3):269–285.

Wootton, J. C. & S. Federhen (**1996**). Analysis of compositionally biased regions in sequence databases. *Methods Enzymol*, 266:554–571.

Xu, Y., J. R. Einstein, R. J. Mural, M. Shah & E. C. Uberbacher (**1994**). An improved system for exon recognition and gene modeling in human DNA sequences. *Proc Int Conf Intell Syst Mol Biol*, 2:376–384.

Yang, W. & T. A. Steitz (**1995**). Recombining the structures of HIV integrase, RuvC and RNase H. *Structure*, 3(2):131–134.

Yi, T. M. & E. S. Lander (**1994**). Recognition of related proteins by iterative template refinement (ITR). *Protein Sci*, 3(8):1315–1328.

Zdobnov, E. M. & R. Apweiler (**2001**). InterProScan–an integration platform for the signature-recognition methods in InterPro. *Bioinformatics*, 17(9):847–848.

Zhang, C. & C. DeLisi (**1998**). Estimating the number of protein folds. *J Mol Biol*, 284(5):1301–1305.

Zhang, J., F. Zhang, D. Ebert, M. H. Cobb & E. J. Goldsmith (**1995**). Activity of the MAP kinase ERK2 is controlled by a flexible surface loop. *Structure*, 3(3):299–307.