Chapter 10

BIOBITS A Study on Candidatus Glomeribacter Gigasporarum with a Data Warehouse

Francesca Cordero[1], Stefano Ghignone[2], Luisa Lanfranco[2], Giorgio Leonardi[3], Stefania Montani[3], Rosa Meo[4] and Luca Roversi[4]

[1] *Dipartimento di Scienze Cliniche e Biologiche, Università di Torino, Italy*
[2] *Dipartimento di Biologia Vegetale, Università degli Studi di Torino, Italy*
[3] *Dipartimento di Informatica, Università del Piemonte Orientale, Italy*
[4] *Dipartimento di Informatica, Università di Torino, Italy*

10.1 Introduction

Biotech is effectively changing the way the largest chemical companies do business, in which as demonstrated by pharmaceutical industries, DNA is used to create novel therapeutics. A crucial advance in this direction is coming from Metagenomics, the emerging science branch that has the potential to substantially impact industrial production, as shown by the company founded by Venter `http://www.tigr.org/`. Industries have different motivations to probe the enormous resource represented by uncultivated microbial diversity. Currently, there is a global political drive to promote biotechnologies as a central feature of the sustainable economic future of modern industrialized societies. This requires the development of novel enzymes, processes, products and applications.

Metagenomics is the branch of science that integrates biology and technology. Based on the genomic analysis of DNA, it has the power to solve problems in many different fields, from positively impacting human health to enabling a better understanding of the environment and agricultural systems as well as creating new biological sources of energy.

This chapter describes the on-going project BIOBITS[1] that aims to perform extensive comparative genomic studies in order to answer fundamental questions concerning the biology, ecology and evolutionary history.

The specific application field of these studies concerns the bacterial endosymbionts. These organisms are widespread in the animal kingdom, where they offer excellent models for investigating important biological events such as organelle evolution, genome reduction, and transfer of genetic information among host lineages [Moran *et al.* (2008)]. By contrast, examples of endobacteria living in fungi are limited [Lumini *et al.* (2006)] and those best investigated live in the cytoplasm of arbuscular mycorrhiza (AM) fungi [Bonfante and Anca (2009)]. AM fungi are themselves obligate symbionts since, to complete their life cycle, they must enter in association with the root of land plants. AM species, belonging to the family Gigasporaceae, have been grouped into a new taxon named Candidatus Glomeribacter gigasporarum [Bianciotto *et al.* (2003)]. The AM fungus and its endobacterium Ca. Glomeribacter gigasporarum are currently used as a model system to investigate endobacteria-AM fungi interactions.

The metagenomics focus of BIOBITS project is on studying the tripartite system: (i) endobacteria living in AM fungi, (ii) AM fungi living in plant roots, and (iii) plant roots, by the employment of a massive large-scale analysis and genomic comparison study of genomes belonging to phylogenetically related free-living bacteria. Moreover, the comparison with genomes of other endosymbionts species will provide insights about the reason of the strict endosymbiotic life style of this bacterium. Another aspect of metagenomics is the analysis of metabolic pathways. A strong reason of interest in this project is based on the assumption that the symbiotic consortia may lead to new metabolic pathways and to the appearance of molecules important for the development of novel therapies and other applications in biotech.

In this chapter we report specifically on a step of BIOBITS whose goal, roughly, is to develop a modular database which allows to import and store massive genomic data. Later in BIOBITS we will extensively develop computational genomic comparison (syntheny) focused on the above bacterium and fungi genomes. BIOBITS deploys a data warehouse that stores in a multi-dimensional model the interesting metagenomic components of the project. Such a metagenomic component should have the following char-

[1]BIOBITS is a project funded by Regione Piemonte under the Converging Technologies Call. BIOBITS involves Università di Torino, Università del Piemonte Orientale, CNR and the companies ISAGRO Ricerca s.r.l., GEOL Sas, Etica s.r.l.

acteristics: i) being able to store genomic data from multiple organisms, possibly taken from different public database sources; ii) annotating the genomic data making use of the alignment between the given sequences and the genomic sequences of other similar organisms; iii) annotating the genomic sequences and the protein transcript products by the full use of ontologies developed by the biology and bioinformatics communities; iv) comparing and visually presenting the results of the genomic alignment; iv) being able to cluster genomic or proteomic data coming from different organisms in order to easily find increasing levels of similarity and induce on one side the steps of the phylogenetic evolution and on the other side investigate on the metabolic pathways.

As a matter of fact, we wish to take advantage from the possibilities offered by computer science technology and its methodologies to analyse the genomic data the project will produce. The analysis of genomic data requires computing tools that allow to "navigate" flexibly data from arbitrary (at least in principle) user defined perspectives and under different degrees of approximation.

10.2 State of the art of metagenomics for genomic comparison

There is a wide variety of approaches in designing tools to analyze biological data. Experience suggests the best way to data analysis is to set up a database. An 'historical' example is ACeDB (A C.elegans Database), one of the first hierarchical, rather than relational, model organism databases. Another example is ArkDB [Hu J (2001)], a schema that was created to serve the needs for the subset of the model organism community interested in agriculturally important animals. ArkDB has been successfully used across different species by different communities, but is rarely used outside the agricultural community.

On "top" of databases a great variety of applications is available, from those ones for the annotation community to molecular pathway visualization, or from the workflow management to the comparative genome visualization.

Currently, there is a rich community and many available software tools built around MAGE (http://scgap.systemsbiology.net/standards/mage_miame.php) and GMOD(http://gmod.org/wiki/Main_Page). GMOD stands for Generic

Model Organism Database project (`http://gmod.org`), which brought to the development of a whole collection of software tools for creating and managing genome-scale biological databases, in the forthcoming description. In the BIOBITS project GMOD and its database Chado have been selected as the data elaboration and management center.

10.2.1 GMOD *and* Chado *database*

The BIOBITS software architecture is built upon a layer provided by GMOD system. The design and implementation of database applications is time consuming and labor-intensive. When database applications are constructed to work with a particular schema, changes to the database schema require in turn changes to the software. Unfortunately, these changes are frequent in real projects due to changes in requirements. In particular they are frequent in bioinformatics. Most critical are the changes in the nature of the underlying data, which follow the current understanding of the natural world. Additional requirements are placed by the rapid technological changes in experimental methods and materials. Finally, the wide variety of biological properties in the organisms species always has made difficult to create a unique model schema valid for all the species.

All the above outlined motivations led to the design of Chado database model which is a generic and extensible model, whose software is available under an open source delivery policy. Chado schema can be employed as the core schema of any model organism data repository. This common schema increases interoperability between software modules that operate on it.

Chado data population is driven by *ontologies*, also known as *controlled vocabularies*. Ontologies give a typing to the entities with the result of partitioning the whole schema into *subschemas*, called *modules*. Each *module* encapsulates a different biological domain and uses an appropriate ontology. An ontology characterizes the different types of entities that exist in a world under consideration by means of primitive relations. These primitives are easy to understand and to use, are expressive and consitently allow the reasoning about the concepts under representation. Typical examples of ontological relations are: (i) *is_a* which expresses when a class of entities is a subclass of another class, and (ii) *part_of* which expresses when a component constitutes a composite. Many others are discussed in [Eilbeck and Lewis (2004)].

Concerning the schema of Chado it is worth remarking *feature* and *sequence* entities. *feature* allow both data and meta data. *feature* can be

populated by instances each determining the type of every other instance in the schema, in accordance with the ontology SO [Eilbeck and Lewis (2004)]. *sequence* contains biological sequence features, that include genetically encoded entities like genes, their products, exons, regulatory regions, *etc.* *feature* and *sequence* are further described by properties.

10.3 BIOBITS system architecture

Here we deepen the description of the system which is designed to manage all the information and all the in-silico activities in the context of the project BIOBITS. This system is implemented through a modular architecture, described in detail in Section 10.3.2. The system architecture permits (1) to store and access locally all the information regarding the organisms to be studied, and (2) to provide algorithms and user interfaces to support the researchers' activities like: (i) searching and retrieving genomes, (ii) comparing and aligning with a genome of reference, (iii) investigating syntenies and (iv) locally storing potentially new annotations.

The system architecture has been engineered exploiting the standard modules and interfaces offered by the GMOD project (http://gmod.org/wiki/Main_Page), and completed with custom modules to provide new functionalities. The main module of the system contains the database which provides all the data needed to perform the in-silico activities related to the project.

Thanks to the adoption of Chado database schema we take advantage for its support in controlled vocabularies and ontologies. Furthermore, Chado is the standard database for most of the GMOD modules; therefore we can reuse these modules to support the main activities of the project and extend the system incrementally as the researchers needs evolve. An example, is the possibility to use BioMart Chado's module which helps the user to identify the relevant dimensions of the problem, their hierarchies and to transform and import input data in the data warehouse conforming them in a typical star schema.

The database stores and provides in a multidimensional schema all the information about the organisms to be studied (i.e. bacteria), their genomes, their known annotations, their proteins and metabolic pathways, and the newly discovered annotations, which can be stored and managed locally until they are confirmed and published.

10.3.1 *Star schema in* BIOBITS *Data Mart*

Essential in the data warehouse is the logical star schema of the stored data. The star schema defines the dimensions of the problem. Often, each dimension of the star schema can be viewed at different abstraction levels. The levels are organized in a hierarchy. Finally, the central entity in the star schema collects the main facts or events of interest. In the case of BIOBITS project, there are two star schemas.

(1) The star built around the *genome composition* facts. It represents the composition of each genome in terms of genes and chromosoms and with reference to the belonging organism.
(2) The star schema around *protein* facts. It describes the proteins in terms of PROSITE domains and with respect to the dimensions of phylogenetic classification and metabolic pathways.

The genes and proteins facts are linked by the relationship representing the enconding.

For most of the dimensions, such as genes and phylogenetic classification, the scientific literature already has provided an ontology (e.g. GO) and controlled vocabularies (e.g. COG) that are available in public domain databases and are imported in the system. Another example of available hierarchy on the genes and proteins are the family organizations.

In the following we describe the BIOBITS Data Mart schema (shown in Figure 10.1) in detail.

Genome composition: It includes all the relevant information about a genome fragment. Considering a fragment view of the genome, genome composition includes all the known fragments composing a genome: it reports the precise boundaries of the fragments (which depend on the user experience and discoveries), the start position and the fragment order w.r.t to the genome, its nucleotide sequence and strand.

Chromosome/Plasmid DNA: It specifies the localization of the fragment expressed by the number or the name of the corresponding chromosome/plasmid location. Indeed, the genome could be inserted either in a chromosome sequence or in a plasmid sequence.

Organism: It specifies both endosymbiotic and ectosymbiotic bacteria. Extensive comparative genomics studies of many organisms are needed. It represents the identifier, the organism scientific name and its classifications in the taxonomy database.

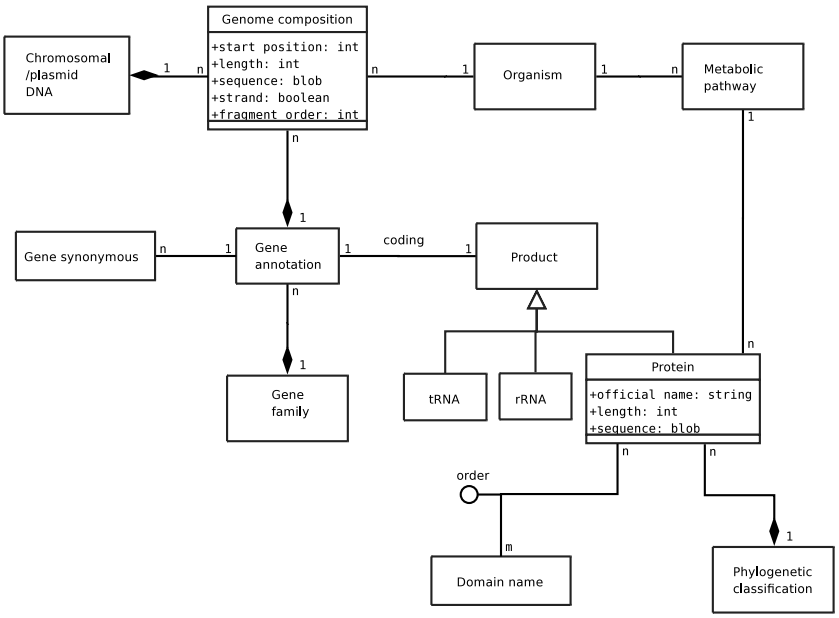

Fig. 10.1: Star schema of BIOBITS Data Mart

Gene annotation: It consists in a short report of gene-specific information (identifier and name), comprehensive of a brief description of gene products using both the information reported in Gene Ontology and the main reference stored in `Pubmed`.

Gene synonymous: It contains all the synonymous names associated to each gene. Genes and proteins are often associated to multiple names; additional names are included as new functional or structural information is discovered. Since authors often alternate between synonyms, computational analysis benefits from collecting synonymous names.

Gene family: Following the gene classification into families, consistent to the genes biochemical similarity, it reports the family identifiers.

Product: It is a class of the products that genes codify. Products are categorized into in three classes: transfer RNA (tRNA), ribosomal RNA (rRNA) and proteins. Moreover it reports a pseudogene indication if the gene has lost its coding ability.

tRNA: Transfer RNA is a small RNA molecule that transfers a specific active amino acid to a growing polypeptide chain.

rRNA: Ribosomal RNA is the central component of the ribosome. The

ribosome is a complex of ribosomal RNA and ribonucleoproteins.

Metabolic Pathways: It represents pathways which are composed by a set of biochemical reactions. Each pathway represents the knowledge on the molecular interactions and reactions network.

Protein: It refers to protein-specific information (protein identifier and name). A protein is a set of organic compounds (polypeptides) obtained by transcription and translation of a DNA sequence.

Phylogenetic classification: It consists of Cluster of Ortologous Groups (COG) of protein sequences encoded in a complete genome.

Domain Name: It reports the domains extracted from PROSITE database [Hulo *et al.* (2006)], characterizing the protein sequence. PROSITE consists of documentation entries describing protein domains, families and functional sites.

The relationship among proteins and domains is characterized by the attribute *order* describing how the domains that compose a specific protein are sorted.

10.3.2 *System architecture*

Figure 10.2 summarizes the main architecture of BIOBITS system. In the following we focus on objectives and features of BIOBITS system.

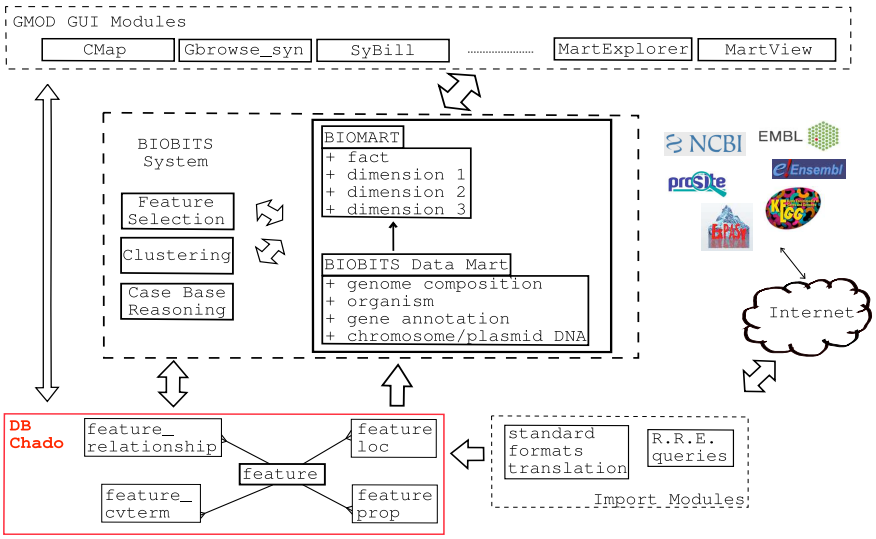

Fig. 10.2: The architecture of BIOBITS system.

Local and global access to data. The instance of Chado we want to set up will contain both data on genome we shall explicitely produce as part of the project BIOBITS and data retrieved from the biological databases accessible through the Internet. The `Import modules` in Figure 10.2 will accomplish such a requirements. Concerning the retrival from Internet, *RRE - Queries* is a GUI wizard, built on the basis of a previously published tool [Lazzarato *et al.* (2004)], able to query different biological databases like, for example `GenBank` (http://www.ncbi.nlm.nih.gov/Genbank/), converting the results of the queries into standard formats. Alternatively, we can convert the format of data retrieved from Internet thanks to the scripts available as part of the GMOD project. A remarkable example are those scripts that convert `GenBank` genes annotations into the Generic Feature Format (`GFF`), adopted as a standard in the GMOD project. Of course, once data have been retrieved, *Import Modules* update Chado, either on-demand, or automatically, possibly on a regular basis.

An On Line Architecture Mining architecture. One of the advantages of a data warehouse is the ready availability of clean, integrated and consolidated data represented by a multiplicity of dimensions. Once that data are stored in the data warehouse, elementary statistics can be computed on the available facts and aggregation of measures and frequencies of facts can be immediately computed. The results can be browsed and compared by OLAP primitives and tools. Finally, on these statistics the power of data mining algorithms can be further exploited. This is the On Line Architecture Mining (OLAM) view of a software architecture [Han and Kamber (2001)]. OLAM is composed by a suite of data mining algorithms that receive from the client a query for a knowledge discovery task. The request can be answered by the predictive and semi-automatic capabilities of data mining algorithms that work on the results of an underlying OLAP server that receives the input data from the underlying data warehouse.

For the transformation of the data stored in Chado into the star schema of Figure 10.1 we exploit BioMart (http://www.biomart.org/), which is a software package available inside GMOD.

10.3.2.1 *Services on Chado and the star schema*

In Figure 10.2, associated to both the Chado instance and to the BIOBITS data mart we plan to offer two types of services. One type is implemented on the basis of existing modules of GMOD. Figure 10.2 highlights them in the uppermost dashed box, named GMOD `GUI Modules`. The second

type of services are internal to the real BIOBITS system: they are shown in Figure 10.2 inside the central dashed box, named BIOBITS system. Now, we discuss the latter components in more detail, putting much emphasis, of course, on the features of the software modules that we specifically develop to support the realization of the goals of the BIOBITS project.

GMOD Graphical User Interface Modules. These modules exploit the available GMOD modules using Chado database to provide the researchers with the tools for comparative genomics needed by the BIOBITS project. GUI modules have also a graphical user interface and allow the user to interact with the system. In particular,

- `CMap` allows users to explore comparisons of genetic and physical maps. The package also includes tools for maintaining map data;
- `GBrowse` is a genome viewer, and also permits the manipulation and the display of annotations on genomes;
- `GBrowse_syn` is a GBrowse-based synteny browser designed to display multiple genomes, with a central reference species compared to two or more additional species;
- `Sybil` is a system for comparative genomics visualizations;
- `MartExplorer` and `MartView` are two user interfaces allowing the user to explore and visualize the stored experimental results and the database content.

BIOBITS system specific modules. Their goal is to allow data analysis under two perspectives that should each other complement and validate.

The first perspective is the one offered by the *Case Base Reasoning* module. It supports efficient retrieval strategies in the context of the search for genomic similarity and syntenies, directly operating on our implementation of the star schema inside BioMart.

The other perspective will exploit tools from Data Mining. We shall use them to perform advanced elaboration on the genomic data. Among the data mining modules we foresee modules for classification, for feature selection and clustering. The latter will be discussed in more detail in this chapter, since it will be the first to be integrated into the BIOBITS system. Indeed, one of the main goal of the whole BIOBITS project is to provide the results of fragment alignment tools (syntheny). Since, clustering provides a specifically useful service for the exploration and elaboration of the similarities among genes and proteins, its results could provide to the syntheny tools additional information that would enhance the fragment

elaboration.

As a concluding remark, the plan is to develop BIOBITS system specific modules as web-based GUI to gain user-friendliness, similar to current GMOD modules and will be able to connect to other modules by standard interfaces. Moreover, we do not exclude some work will be required to customize current GMOD modules to the usage requirements of biologists involved in the BIOBITS project. Of course we shall adhere to the open source philosophy. So, any BIOBITS system specific module will be available as part of the whole project GMOD.

10.4 Software modules to support researchers' activities

The following section will describe the details of the new modules introduced in the BIOBITS system.

10.4.1 *Case-Based Reasoning*

Within the BIOBITS architecture, we are currently working at the design and implementation of an *intelligent retrieval* module, which implements the *retrieval* step of the Case-Based Reasoning (CBR) [Aamodt and Plaza (1994)] cycle. CBR is a reasoning paradigm that exploits the knowledge collected on previously experienced situations, known as *cases*. The CBR cycle operates by (1) *retrieving* past cases that are similar to the current one and by (2) *reusing* past successful solutions; (3) if necessary, past solutions are properly *adapted* to the new context in which are used; (4) the current case can then be *retained* and put into the system knowledge base, called the *case base*. *Purely retrieval* systems, leaving to the user the completion of the reasoning cycle (steps 2 to 4), are very valuable decision support tools [Watson (1997)], especially when automated adaptation strategies can hardly be identified, as in biology and medicine [Montani (2008)].

Our retrieval module is meant to support comparative genomic studies that represent a key instrument to: (i) discover or validate phylogenetic relationships, (ii) give insights on genome evolution, and (iii) infer metabolic functions of a particular organism.

10.4.1.1 *Multiple abstractions on the genome*

In our module, cases are genomes as sequences of nucleotides, each one taken from a different organism, and properly aligned with the same ref-

erence organism. For each nucleotide, a percentage of similarity with the aligned nucleotide in the reference organism is also provided. However, depending on the type of analysis which is required, a "view" of the genomes at the nucleotide level may not always be the most appropriate: sometimes, a "higher level" view, abstracting the available data at the level of genes, regions, or even complete chromosomes, would be more helpful. Our tool supports this need, by allowing the retrieval of the available cases at any level of detail, according to a taxonomy of granularities, which is depicted in figure 10.3.

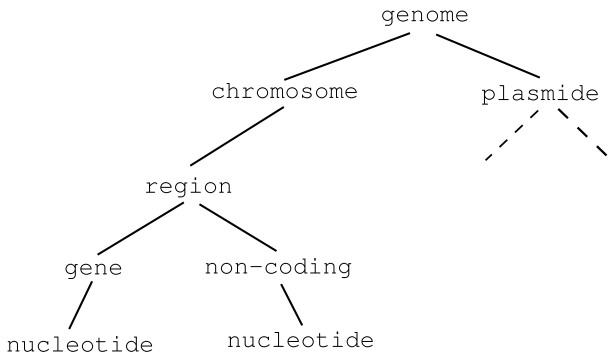

Fig. 10.3: A taxonomy of sequence granularities.

Moreover, a sequence of consecutive granules, sharing the same *qualitative level* (e.g. low, medium, high) of similarity with respect to the reference organism, can be abstracted into a single interval, labeled with the qualitative level of similarity itself: such an abstraction process is very similar to the Temporal Abstractions (TA) methodology, described in [Shahar (1997); Bellazzi *et al.* (1998)], even if in our domain the independent variable is the granules sequence instead of time. As in TA, in fact, we move from a *point-based* to an *interval-based* representation of the data, where the input points are the granules, and the output intervals (*episodes*) aggregate adjacent points sharing a common behavior, persistent over the sequence. In particular, we rely on *state* abstractions [Bellazzi *et al.* (1998)], to extract episodes associated with qualitative levels of similarity with the reference organism, where the mapping between qualitative abstractions and quantitative values (percentages) of similarity can be parametrized on the basis of domain knowledge.

Different levels of abstraction can be further exploited both to reduce space occupancy in the database and to allow the user to focus on a different level of detail.

In synthesis, our retrieval framework allows for *multi-level abstractions*, according to two *dimensions*, namely a taxonomy of state abstraction symbols, and a variety of sequence granularities. In particular, we allow for *flexible querying*, where queries can be expressed at any level of detail in both dimensions, also in an *interactive* fashion, progressively refining the retrieval set.

10.4.1.2 *Multi-dimensional index structures*

Moreover, our framework takes advantage of *multi-dimensional orthogonal index structures*, which make retrieval faster, allowing for early pruning and focusing. Technical details of the query answering algorithms can be found in [Montani *et al.* (2009)]. Indexes can be defined and grow on demand, depending on the types of queries issued so far. An example multi-dimensional index, rooted in the H symbol, is represented in figure 10.4.

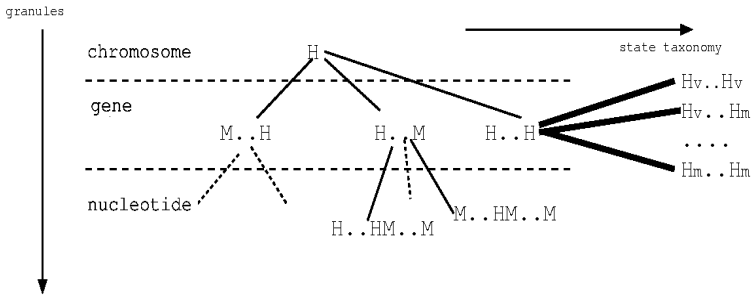

Fig. 10.4: An example multi-dimensional orthogonal index. Note that indexes may be incomplete with respect to the taxonomies: here, for instance, the *region* level is missing in the granularity dimension.

10.4.2 *Clustering modules*

In this chapter we do not go in detail in describing all the predictive and exploratory capabilities offered by data mining algorithms. The aim of

this section is to depict a portray built on a single example: *clustering*. It offers the possibility to show the benefits in terms of interoperability, extendability and flexibility offered by a modular system built upon a data warehouse in which a multi-dimensional representation of a ground set of facts is stored. On these data, whenever it is needed, a query can be issued by the user in order to retrieve from the data warehouse the values of the interesting subset of dimensions. On this initial set of values multi-level reasoning is possible exploiting the relationships between facts in the knowledge network.

One of the classical aims of clustering is to provide a description of the data by means of an abstraction process. In many applications, the end-user is used to study natural phenomena by the relative proximity relationships existing among the analyzed objects. For instance, he/she compares organisms by means of the relative similarity in terms of the common features with respect to a same referential example. Many Hierarchical Clustering (HC) algorithms have the advantage that are able to produce a dendrogram which stores the history of the merge operations (or split) between clusters. Moreover, the dendrogram produced by a hierarchical clustering algorithm constitutes a useful, immediate and semantic-rich conceptual organization of the object space. As a result HC algorithms produce a hierarchy of clusters and the relative position of clusters in this hierarchy is meaningful because it implicitly tells the user about the relative similarity between the cluster elements. HC approaches help the experts to explore and understand a new problem domain. As regards the exploitation of object distances, clustering algorithms offer immediate and valuable tools to the end-user for the biological analysis.

10.4.2.1 *Co-clustering*

A clustering algorithm useful in biological domain is *co-clustering* [Dhillon *et al.* (2003)] whose solution provides contemporaneously a clustering of the objects and a clustering of the attributes. Further, often co-clustering algorithms exploit similarity measures on the clusters in the other dimension of the problem: that is, clusters of objects are evaluated by means of the clusters on the features and vice versa. They simultaneously produce a hierarchical organization in two of the problem dimensions: the objects and the features that describe the objects themselves. In many applications both hierarchies are extremely useful and are searched for.

An appealing algorithm based on co-clustering has been obtained by

the introduction of *constraints*. Constraints are very effective in many applications, including gene expression analysis [Pensa *et al.* (2010)] and sequence analysis [Cordero *et al.* (2009)], since the user can express which type of biological knowledge leads to the association among gene clusters and biological condition clusters. The sequence analysis was directed into the discovery of gene motifs or protein domains. It relies on two main ideas: exhaustive search and automatic association of motifs with protein subfamilies. The same idea can be also applied on genome sequence in order to discover regularity associated to specific portions of the nucleotide sequence.

Sebat and co-authors report in [Sebat *et al.* (2003)] the importance of expression profile data in metagenomics. Indeed, the use of microarrays to profile metagenome libraries offers an effective approach to characterizing many organisms rapidly. As a consequence, in the future, microarray data could be integrated into the proposed star schema making possible this kind of analysis in our system. Constrained co-clustering algorithms will allow us to identify groups of genes that show similar activity patterns under a specific subset of the experimental conditions by measuring the similarity in expression within these groups. Under this view point, constraints are effective because take into account the similarity/dissimilarity between pairs of genes.

10.4.2.2 *Proximity measures*

The majority of clustering algorithms are driven by distances between objects. In our multi-dimensional problem distances may be computed in different ways according to the dimension. Moreover distances can take into account the hierarchy on the dimension itself.

Notice in fact, that for the main dimensions in the star schema of Section 10.3.1 (gene annotations, proteins, etc) a hierarchical ontology is defined (GO and orthology) starting from the available knowledge on the biological domain.

Therefore, the hierarchies can be exploited to define a distance between two facts. Later these distances could be used for distance-based classification and clustering algorithms.

A distance measure d_i on a single dimension X_i can be integrated in the standard distance function (like the euclidean one) in an m-dimensional space. In the computation of the distance between two objects o_k and o_j, we can combine the distances between the values of any dimension in the

two objects. The result of this combination is the following:

$$dist(o_k, o_j) = \sqrt{\sum_i d_i(o_k[X_i], o_j[X_i])^2}$$

where o_k and o_j represent two facts stored in the data warehouse fact table, X_i denotes i-th dimension in the star schema and finally $o_k[X_i]$ and $o_j[X_i]$ are the values of the dimension i for the two facts. $d_i(o_k[X_i], o_j[X_i])$ represents the computed distance measure between two facts with respect to the i-th dimension. Whenever, $o_k[X_i]$ and $o_j[X_i]$ represent two values of the dimension X_i in two different positions of the X_i hierarchy, a measure that takes into account the different position in the hierarchy can be adopted. We mention here [Wang *et al.* (2007)] in which such a measure has been proposed and frequently adopted for the gene ontology.

10.5 Conclusion

In this chapter we reported on BIOBITS project whose goal is to extensively develop a computational genomic comparison (known as syntheny) focused on the Ca. Glomeribacter gigasporarum bacterium and arbuscular mycorrhiza (AM) fungi genome.

We presented the software architecture essentially developed over an existing software layer provided by GMOD Community. GMOD system offers powerful data visualization and analysis tools, data warehouse modules, such as BioMart and the possibility to exploit import modules for the inclusion of data from the external, public resources. Furthermore, it contains the Chado database which presents an extensible and flexible model for any organism species built upon the generic concept of feature which can be customized by the use of types and ontologies.

We presented the logical data representation of the genomic and proteomic components of the biological problem: it has the form of a double star schema - the first one centered around the genetic fragments composing the genome and the second one on the proteins encoded by the genes.

Finally we presented the main software blocks of BIOBITS system: a Case-Based Reasoning module and a clustering module - first of a series of data mining modules that will be integrated in the future - which allow the user to retrieve and analyse in a flexible and intelligent way the data coming from the multidimensional star schema.

Both these modules complement each other. Case-Based Reasoning and

temporal analysis retrieve the information at different abstraction levels, as needed by the analyst. Clustering provides a novel annotation to genetic sequences based on computational data mining algorithms. The annotation occurs on the basis of proximity measures defined by the analyst. Notice, however, that proximity could be defined on the genome fragments viewed at the desired abstraction level by the former Case-Based Reasoning module.