IMMERSIVE GRAPH-BASED VISUALIZATION AND EXPLORATION OF BIOLOGICAL DATA RELATIONSHIPS

N Férey, PE Gros, J Hérisson, R Gherbi*

**Bioinformatics team, Human-Computer Communication Department, LIMSI-CNRS / University Paris-Sud*
BP 133 F-91403 ORSAY CEDEX France
Email: genoteam@limsi.fr

ABSTRACT

Genomic information shows some characteristics that make them very difficult to interpret and to exploit. Such data constitute an important factual resource (GenBank, SwissProt, GeneOntology, or Decrypthon…), are heterogeneous, huge in quantity, and geographically distributed. This paper presents Genome3DExplorer, a new modeling and software solution to visualize textual and factual genomic data based on adapted federator description language. The exploration is based on a well-adapted graphical paradigm that automatically helps to build a graph-based representation, and allows biologist to highlight some global topological characteristics of data, which are uneasily visible using traditional exploration tools. Finally, we present results produced by Genome3DExplorer software on various sets of biological data.

Keywords: Virtual Reality, Immersive Exploration, Genomic Data, Graph-based Visualization.

1 INTRODUCTION

As in *ADN-Viewer* (Hérisson, Gros, Férey, Magneau & Gherbi, 2004) and *SequenceWord* (Rojdestvenski, Pettersson & Modjeska, 2000) our objective is to elaborate new solution in order to explore in virtual environment various kinds of genomic data. These data come from the many databases, such as *GenBank* (DDBJ/EMBL/Genbank n.d.), *SwissProt* (SIB/EBI n.d.), or *Decrypthon* (AFM, GENOMINING, IBM, 2000). Our approach is mainly based on the definition of a genomic data federator language, answering the requirements and specificities of genomic databases. Then we explain the representation methods to view these data within an immersive framework. Finally, we present some results produced by our *Genome3DExplorer* software on various sets of biological data.

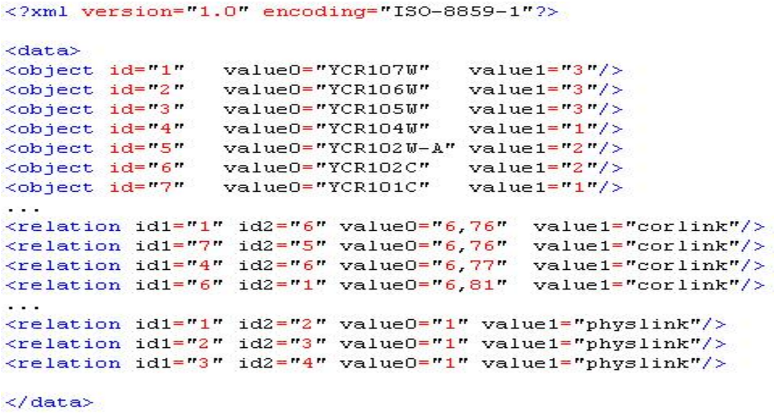
In order to visualize efficiently biological data, we need to define a common data description language that must accommodate and represent knowledge resulting from structured but heterogeneous databanks. We describe in this section how we used the specific characteristics of the genomic data to find an adapted description format for this kind of data.

2 BIOLOGICAL DATABANK SPECIFICITIES

Although the genomic databases are very heterogeneous (format or quality), they involve some specific characteristics. Indeed, they are often focused on biological object of interest (protein, gene...), described by an attribute set. Moreover, these objects are often compared one to another by a measurement (sequence alignment score, functional similarity…). For instance, *GenBank* contains annotated DNA sequences, and provides *BLAST* tools in order to compare these sequences, as *SwissProt*, which deals with annotated protein sequences.

2.1 Definition of a genomic data representation language

In the most commonly cases, different kinds of biological objects (protein sequences, DNA sequences, biological terms…) are often connected by binary relationships. For example, using text corpora, biologists could extract co-occurrence relationship between two biological terms, as in *BioBiblioMetrics* (Stapley & Benoit, 2000) or more specific semantic relationships, coming from text information extraction processing (Pustejovsky, Castano & Zhang, 2002). In databank case, biologist can extract alignment measurements between two DNA or protein sequences. These binary numerical relationships can be computed, by alignments or co-occurrence measurements (numerical values), or by semantic relations extracted from texts (symbolic values). Biological objects can also be characterized by their biological properties. The properties can be of string type (label, sequence…), numerical one (like co-occurrence score, alignment measurements), or symbolic one (type of interaction, like positive retroaction…). Taking account into these characteristics, we define a XML-based data representation language, based on the concept of multi-valuated objects and relationships, which is particularly adapted to describe biological data. In this uncompleted example, the studied biological objects are yeast genes. Two values characterize these genes. The name (*value0* of object tag) and the number of co-regulator factors (*value1* of object tag). Two values characterize the relationships between two genes, the kind of relationship (*value1* of relation tag), and a numerical value (*value1* of relation tag).



```
<?xml version="1.0" encoding="ISO-8859-1"?>

<data>
<object id="1"   value0="YCR107W"   value1="3"/>
<object id="2"   value0="YCR106W"   value1="3"/>
<object id="3"   value0="YCR105W"   value1="3"/>
<object id="4"   value0="YCR104W"   value1="1"/>
<object id="5"   value0="YCR102W-A" value1="2"/>
<object id="6"   value0="YCR102C"   value1="2"/>
<object id="7"   value0="YCR101C"   value1="1"/>
...
<relation id1="1" id2="6" value0="6,76"  value1="corlink"/>
<relation id1="7" id2="5" value0="6,76"  value1="corlink"/>
<relation id1="4" id2="6" value0="6,77"  value1="corlink"/>
<relation id1="6" id2="1" value0="6,81"  value1="corlink"/>
...
<relation id1="1" id2="2" value0="1" value1="physlink"/>
<relation id1="2" id2="3" value0="1" value1="physlink"/>
<relation id1="3" id2="4" value0="1" value1="physlink"/>

</data>
```

3 REPRESENTATION MODALITIES IN AN IMMERSIVE FRAMEWORK

The characteristics of the data representation language allow us to describe biological data with a generic way, but it remains to define a visualization paradigm. This paradigm must be adapted at the same time to this language and to the user's needs. We present in this section how we map the data representation language defined in the second section, to a visual representation.

3.1 Graph visualization

The selected federator language describes a list of valued objects with their binary valued relationships. We choice to consider this data as a multi-valued graph, where biological objects are nodes and relationships between us are edges. Visualizing and exploring data with 3D graph has no reality references (on the contrary to metaphoric representation) and is independent from data.

3.2 Visualization description format

Visualizing data by a 3D graph results from the following motivation: invent a system that ensures independence between semantic of data and their visual representation. This motivation transformed into requirements because of the format heterogeneity. Indeed, we did not want to import this heterogeneity in the visualization system. However it remains to choice how graphically represents each value (both object and relationship values) within a 3D graph. In order to map data to their 3D graph, we defined and XML-based data visualization language. To do that, we started to build a short inventory of the graphic 3D object characteristics for 3D graph nodes and edges: Taking into account the description data format, these graphic characteristics can be classified in the three following groups: symbolic, numeric, or both. So within 3D graph visualization, numerical object values may be represented by numerical graphic properties, like node position (x, y, z), node size, node color (r, g, b components) or node transparency (alpha component). Numerical relationship values may be visualized by edge length, edge weight, edge color, or edge transparency. Each symbolic values may be represented by a predefined shape (cube, sphere for nodes, and cylinder, line for edges) or predefined color (red, pink, blue), according to the kind of value (object or relationship value). Finally, string values may be visualized by a 3D text label.

3.3 Node placement problem with 3D weight graph visualization

However, we are faced of the following problem: mapping numerical value (correlation) to a distance (edge length) in the 3D space between graph nodes has often no graphic solution, due mainly to 3D Euclidian space constraints.

We use an approach, proposed by Eades (Eades, 1984) simulating two kinds of force between each node. To place two nodes that are in relations respecting distance constraints, he proposed to apply them an attraction force, in order to minimize the global energy E of this spring system,

$$E = \sum_{\le i<j\le|Node|} k\left(\left|p_i - p_j\right| - l_{ij}\right)^2$$

where p_i is the position of a node i, l_{ij} the optimal distance between the node i and the node j, and k a constant factor. Moreover, a repulsion force is applied on two close nodes, which are not connected. After several iterations (*nbiter*), this dynamic property allows system to converge into a satisfactory solution where all the distances are as closed more possible than desired edges length. The main disadvantage of above approach is its complexity about:

$$\theta\left(\left|Node\right|^2 + \left|Edge\right|\right) \times nbiter$$

Indeed, each node reacts to the presence of all its connected neighbors by an attraction force, and moves according to the presence of all the other nodes per repulsion. The complexity strongly decreases by applying a visibility threshold on the not connected nodes. The nodes too much far according to this fixed threshold do not repulse. Moreover, we can remark that, the superposition of node problem on 2D space is not a problem in space, because user cans easy turn around his data. We just approximate distance between connected nodes, but not repulsion between not-connected nodes, decrease complexity to :

$$\theta\left(\left|Edge\right|\right) \times nbiter$$

On very huge graph (million of edges), it however was necessary to use segmentation process in order to make some partition of the graph, using MCL clustering (Enright, Van Dongen & Ouzounis, 2002), (Karypis & Kumar, 2004) or Fiduccia algorithm (Fiduccia & Mattheyses, 2004).

4 RESULTS

4.1 Factual data: Yeast gene block duplications

This system was used firstly in order to gene duplications in the yeast chromosomes. In this experiment, each object is one of the yeast chromosome arms. For each object, the values are chromosome name, chromosome size, chromosome side (right or left arm). In each relationship between chromosomes, the values are the number of same gene shared by two chromosomes.

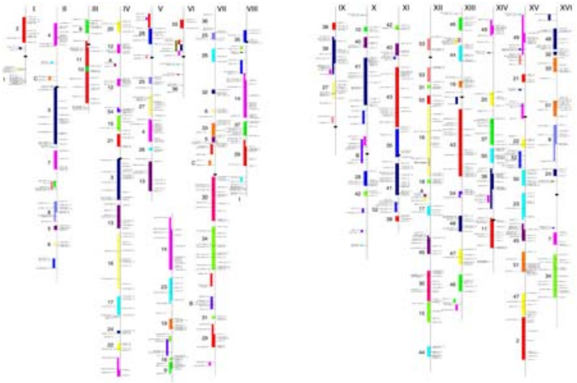


Figure 1. Traditional 2D visualization (16 Yeast chromosomes)

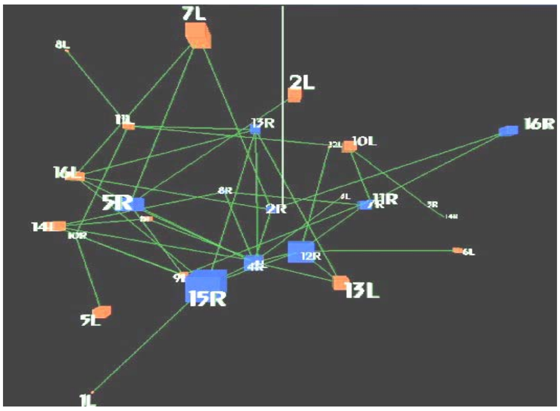


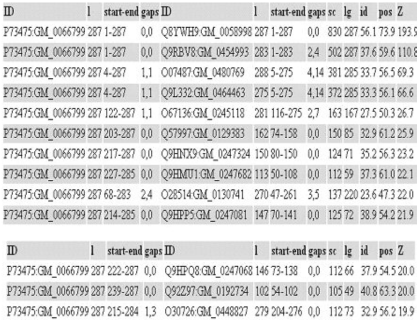
Figure 2. Immersive synthetic visualization (32 Yeast chromosome arms)

In the synthetic representation showed in Figure 2, we can directly see that several chromosome arms, like *4R*, are in the centre of the global gene duplications, whereas other chromosome arms take placed in periphery. This representation helps biologist to launch a work on correlation between chromosome placement in the cells and the gene duplication between Yeast chromosomes during evolution.

4.2 *Decrypthon*: A huge protein-to-protein sequences alignment dataset

The *Decrypthon* project leaded by AFM, GENOMINING, IBM, (2000) is a databank which contains the results of an exhaustive comparison of all known proteins from living organisms (animals, plants and humans), including the coding sequences from 76 completely sequenced genomes. There are currently two ways for *Decrypthon* analyzing: biologists can use the *Decrypthon* browser to query which proteins are homologous to a targeted protein. However, Decrypthon browser does not allow request on a set of proteins. In order to request *Decrypthon* on a set of protein, users must follow the second way: download raw data.

Unfortunately these data are too huge to be easily explored (39 Go, about 500000 proteins, 300 millions of alignment results). Therefore, a segmentation process was performed on this graph (see section 3.3) in order to visualize and explore this kind of data.



ID	l	start-end	gaps	ID	l	start-end	gaps	sc	lg	id	pos	Z
P73475:GM_0066799	287	1-287	0,0	Q8YWH9:GM_0058998	287	1-287	0,0	830	287	56.1	73.9	193.5
P73475:GM_0066799	287	1-287	0,0	Q9RBV8:GM_0454993	283	1-283	2,4	502	287	37.6	59.6	110.8
P73475:GM_0066799	287	4-287	1,1	O07487:GM_0480769	288	5-275	4,14	381	285	33.7	56.5	69.3
P73475:GM_0066799	287	4-287	1,1	Q9L332:GM_0464463	275	5-275	4,14	372	285	33.3	56.1	66.6
P73475:GM_0066799	287	122-287	1,1	O67136:GM_0245118	281	116-275	2,7	163	167	27.5	50.3	26.7
P73475:GM_0066799	287	203-287	0,0	Q57997:GM_0129383	162	74-158	0,0	150	85	32.9	61.2	25.9
P73475:GM_0066799	287	217-287	0,0	Q9HNX9:GM_0247324	150	80-150	0,0	124	71	35.2	56.3	23.2
P73475:GM_0066799	287	227-285	0,0	Q9HMU1:GM_0247682	113	50-108	0,0	112	59	37.3	61.0	22.1
P73475:GM_0066799	287	68-283	2,4	O28514:GM_0130741	270	47-261	3,5	137	220	23.6	47.3	22.0
P73475:GM_0066799	287	214-285	0,0	Q9HPP5:GM_0247081	147	70-141	0,0	125	72	38.9	54.2	21.9

ID	l	start-end	gaps	ID	l	start-end	gaps	sc	lg	id	pos	Z
P73475:GM_0066799	287	222-287	0,0	Q9HPQ8:GM_0247068	146	73-138	0,0	112	66	37.9	54.5	20.0
P73475:GM_0066799	287	239-287	0,0	Q92Z97:GM_0192734	102	54-102	0,0	105	49	40.8	63.3	20.0
P73475:GM_0066799	287	215-284	1,3	O30726:GM_0448827	279	204-276	0,0	112	73	32.9	56.2	19.9

Figure 3. Textual sample result on a *Decrypthon* request on P73475 protein

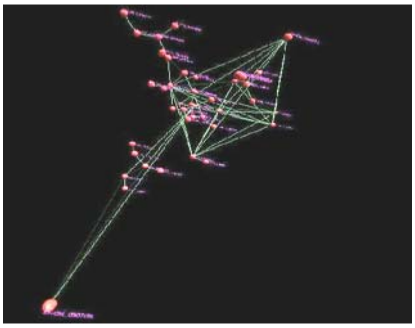


Figure 4. Global proteins sequences alignment of a biconnexe component of *Decrypthon*

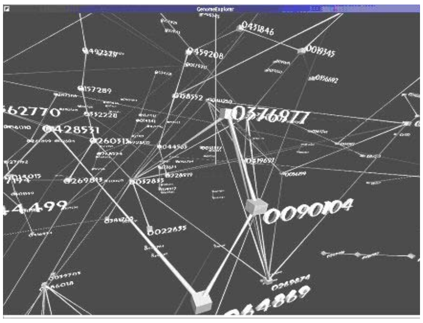


Figure 5. *Decrypthon* data exploration (thousands nodes)

We can see in Figure 4 an output of first results, compared to the textual result offered by the *Decrypthon* browser in Figure 3. In this view, the distance between two proteins nodes are inversely proportional to the Z-score (alignment score) between these protein sequences. In this kind of visualization, the problem of proteins set query is partially solved, because the result is not only centered on a targeted protein. User can directly see the vicinity of a targeted protein, but he can also explore *Decrypthon* result without any a priori searching criteria. In Figure 5, we will show the advantage of the immersive visualization system on huge amount of data. The fact to move in an immersive way (stereovision) into the data allows biologists to build a spatial representation using their natural perception and spatial clustering skills.

4.3 Microarray data

We present in this section two examples about results obtained with *Genome3DExplorer* for exploring microarray data. We used two sets of public microarray data: a partial dataset coming from yeast gene expressions data (public data) during elutriation phase and plasmodium gene expression coming from DiRiSi Lab (Bozdech, Llinas, Pulliam, Wong, Zhu & DeRisi, 2003)

A DNA microarray data set consists of expression levels of N genes in M different experimental conditions. We are interested in patterns of co-expression, namely groups of genes with parallel or anti-parallel profiles. We measure co-expression between genes *k* and *l* by the (Pearson) correlation of their profiles X_k:

$$cor(k,l) = \frac{1}{N}\frac{\sum_{j=1}^{M}\left(X_{k,j} - \mu_k\right)\left(X_{l,j} - \mu_l\right)}{\sigma_k \sigma_l}$$

In these experiments, biologist choices to represent gene profiles by cubes, green cube for gene which function is known grey for the others. *Pearson* correlation between two genes profiles are represented by edge: blue edge for parallel ones and green edge for anti-parallel ones, making the following biological hypothesis: anti-parallel profiles are as important as parallel ones to identify genes in the same pathway. In order to position gene profile nodes, the edge length between is inversely proportional to their correlation score.

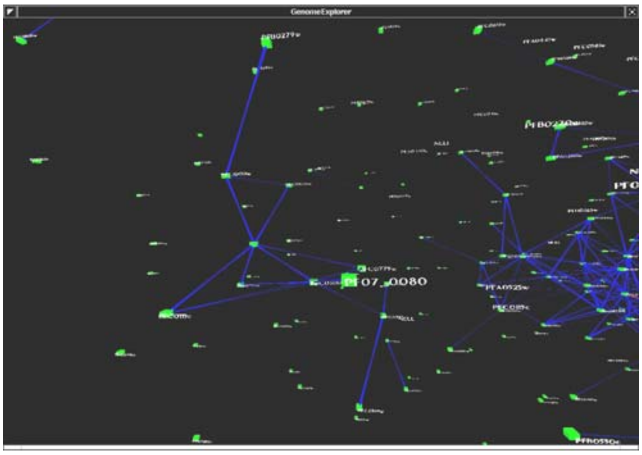


Figure 6. Plasmodium falciparum correlation expression profile network (1)

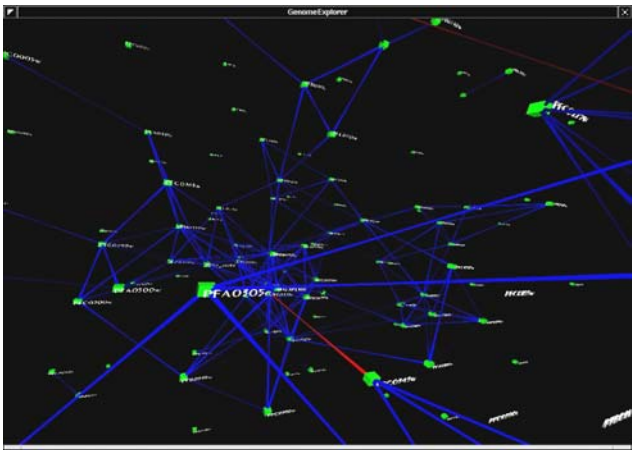


Figure 7. Plasmodium falciparum correlation expression profile network (2)

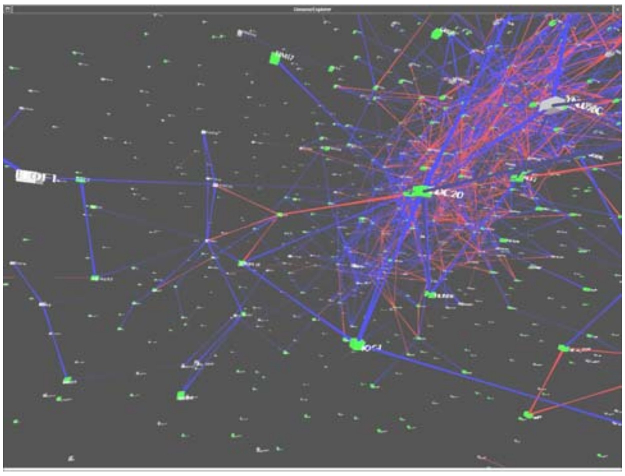


Figure 8. Yeast correlation expression profile network during elutriation phase (1)

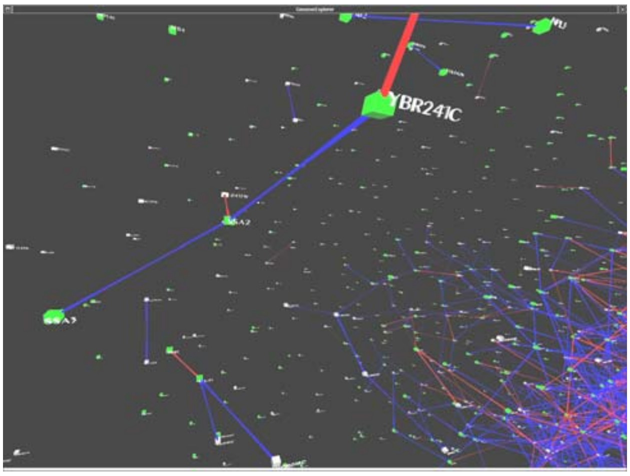


Figure 9. Yeast correlation expression profile network during elutriation phase (2)

Biologists validate this exploration method partially, because many known genes in the same sub network are the same known pathway, for example SSA2 and SSA3 in Figure 9.

5 CONCLUSION AND FUTURE WORK

In this paper, our objective was to elaborate new solutions in order to explore both biological genomic data. This approach is mainly based on the definition of a genomic data representation language, answering the requirements and specificities of biological databases. The representation methods to view these data within an immersive environment, like in *SequenceWorld* (Pustejovsky, Castano & Zhang, 2002) were presented and approved successively on various sets of biological data. We try to represent these biological data by 3D graph, using force directed placement algorithm, like *BioBiblioMetrics* (Stapley & Benoit, 2000) Compared to this work, the data description language offers biologist to precise the semantic of the edge in the representation, and the semantic of all the others graphic characteristics. Moreover, we can represent any biological objects (like protein sequence, biological terms, chromosome arm…), on the contrary to the *Sequence World*, which only deals with genetic sequences. The immersive aspect gives the possibility of exploring huge data in a synthetic way and so constitute the strong points of our system, because it offers a global point of view of the data subjacent structure. These characteristics are particularly interesting when biologists wish to explore a mass of data without precisely knowing what they seek. For example, the partial analysis of *Decrypthon* data shows directly several clusters within the representation. This study was concretized by a software development, named *Genome3DExplorer*, which was used to generate the results presented in this paper.

6 REFERENCES

AFM, GENOMINING, IBM (2000) Homepage of Decrypthon: first exhaustive comparison of known proteins from living organisms. Available from: http://www.infobiogen.fr/services/decrypthon/index.html

Bozdech, Z. Llinas, M. Pulliam, B.L. Wong, E.D. Zhu, J. DeRisi, J. (2003) The Transcriptome of the Intraerythrocytic *Developmental Cycle of Plasmodium Falciparum. PLoS Biol .1*(1): e5.

DDBJ/EMBL/Genbank (n.d.) Homepage of GenBank Database. Available from: http://www.psc.edu/general/software/packages/genbank/genbank.html

Eades, P. (1984) A Heuristic for Graph Drawing. *Congressus Nutnerantiunt*, *42*, 149–160.

Enright A.J., Van Dongen S., Ouzounis C.A. (2002) An Efficient Algorithm for Large-Scale Detection of Protein Families. *Nucleic Acids Research 30*(7), 1575-1584.

Fiduccia, C.M. & Mattheyses, R.M. (1982) A Linear-Time Heuristic for Improving Network Partitions. *In Proceedings of the 19th ACM/IEEE Design Automation Conference* (pp. 175–181). Las Vegas, Nevada.

Hérisson, J. Gros, P.-E. Férey, N. Magneau, N. and Gherbi, R. (2004) DNA in Virtuo: Visualization and Exploration of 3D Genomic Structures. *3rd ACM International Conference on Virtual Reality, Computer Graphics, Visualization and Interaction* (pp. 35-40). Stellenbosh, South Africa.

Karypis, G. & Kumar, V. (1998) Multilevel Algorithms for Multi-Constraint Graph Partitioning. *In Proceedings of the IEEE/ACM SC98 Conference* (pp 28). Orlando, Florida.

Pustejovsky, J. Castano, J. & Zhang J. (2002) Robust Relational Parsing over Biomedical Literatures: Extracting Inhibit Relations. *Proceedings of Pacific Symposium on Biocomputing*. Lihue, Hawaii.

Rojdestvenski, I. Pettersson, F. Modjeska, D. (2000) Sequence World: A Genetics Database in Virtual Reality. *Proceedings of the International Conference on Information Visualization* (pp. 513-517). Dearborn, Michigan, USA.

SIB/EBI (n.d.) Homepage of SwissProt Protein Knowledgebase and TrEMBL Computer-annotated supplement to Swiss-Prot: Available from: http://us.expasy.org/sprot/

Stapley, B.J. & Benoit, G. (2000) BioBibliometrics: Information Retrieval and Visualization from Co-occurrences of Genes Names in Medline Abstracts. *Proceedings of Pacific Symposium on Biocomputing*, *5*, 526-537. Oahu, Hawaii.