

Granularity of genomics data in genome visualisation

Asia Jakubowska
Department of Computing
Science
University of Glasgow
Glasgow, G12 8QQ
Scotland
asia@dcs.gla.ac.uk

Ela Hunt
Database Technology
Research Group
Department of Informatics
University of Zürich
CH-8057 Zürich
ela@dcs.gla.ac.uk

Matthew Chalmers
Department of Computing
Science
University of Glasgow
Glasgow, G12 8QQ
Scotland
matthew@dcs.gla.ac.uk

ABSTRACT

Biologists collect genomic data of increasing complexity. New technologies give rise to new data types and the volume of both raw and processed data is growing fast. Biomedical researchers would like to analyze the data using user-friendly interfaces, however, the tools, computer monitors and machines have their limitations and do not always precisely show the data under investigation. We survey different genomics visualisation software including AceDB, SyntenyVista, DerBrowser, Apollo, Artemis, BugView, Ensembl, Sockeye, K-BROWSER, GBrowse, NCBIMapView, eQTL Explorer, and Expressionview.

This paper presents a short survey of genomic browsers and visualisation effects which were used or can be used in such applications. It presents a classification of genome browsers according to three dimensions and argues the need for a new browser which offers improved zooming functions. This leads us to introduce a new version of SyntenyVista, VisGenome, which allows the user to visualise single and comparative representations of the rat, the mouse, and the human genome at different levels of detail.

Author Keywords

genomics visualisation

INTRODUCTION

Current genomics visualisations are inadequate in many respects, as they do not allow for flexible view adjustment. We are aiming to derive general principles of data representation and visualisation usability for genomics. We would like to find a solution which will clearly present the information, including all relevant information the biologists wish to see. We study existing visualisation solutions in order to find out what features they offer, which of those correctly support data analysis, and which are not helpful. Our study will allow us to find a better solution for data analysis which overcomes

cognitive problems. We would like to discover how best to compare data coming from various sources and experiments in a biological setting.

Our work focuses on the use of visualisation to support the understanding of very large data sets. With an eye to create a universal solution, we are collaborating with biologists who use genome browsing tools, such as that we create, in their everyday work. We are aiming to solve in VisGenome both the visualisation problems and some of the database integration problems. We would like to offer a clear presentation of the data the biologists wish to see.

A BIOLOGICAL INTRODUCTION

In this section we motivate our work and introduce the concepts used in this paper. Biomedical and agricultural research is motivated in two ways. One is to acquire new knowledge and understand how living organisms function, and the other is to improve our lives. Improvement is the treatment or prevention of diseases, better diagnosis, new medications, new crops, and better understanding of the environmental impact our technologies have.

Genomics is the study of genomes and of the relationship between genomes and the way an organism functions. Each living organism has a genome which encodes information passed down from generation to generation. A bacterial genome consists of several million DNA molecules (tuberculosis genome is about 5 million long). A human or mouse has a genome of around 3 billion letters of DNA code. A genome is encoded in DNA or RNA molecules of four types (A, C, G, T for DNA). It encodes all proteins and signalling molecules needed by an organism. Only 1.5% of the human or mouse genome is translated into proteins which are the building blocks of our bodies. Chemically, they are strings of amino-acids, where each three letters of DNA correspond to one amino-acid. Proteins use an alphabet of 21 letters, and in our bodies they fold into *3-D structures* (see Fig. 2H) which may change conformation as they perform their various functions. We do not know exactly how many genes the humans have, with the current estimate being between 20 and 30 thousand. Those give rise to probably around 1 million proteins. The process of translation from DNA to protein is complex, and it is important to remember that a stretch of DNA of some 30 thousand letters gives rise to a protein of some 300 letters. The parts of DNA which translate

into protein are called *exons* while the parts which control the process are called *introns* or untranslated regions. Biologists want to know for each protein what gene produced it, which parts of the gene were used in this particular protein and which control regions were activated during the production process. The process of protein production is dependent on the type of cell, developmental stage, the environment, and many other factors which altogether influence the health of an organism.

Genomes of a very large number of animals are known relatively well, and are publicly available, along with genome maps which show how genes are arranged and structured. Mammalian genomes are split into around 20 chromosomes, and the set of chromosomes forms a *karyotype* (see Fig. 2A), while bacterial genomes form a circle. Groups of genes that are shared between related organisms are often collocated in the so-called *synteny groups*, and biologists study such gene groups, as there is proof for synchronised activity over groups of genes, and for similar gene functions shared between related organisms. Similar gene functions arise from similar DNA and protein sequences, and the biologists *align* (see Fig. 2I) genomes and genes to understand what sequences are shared, and what functions are common to a group of organisms.

Genes and the resulting proteins interact and form *pathways*. Such pathways stand for chemical and structural reactions which orchestrate all the processes which keep us alive. Pathways may be shared by groups of organisms but there are known cases where they diverge. Pathway visualisation tools include [32].

Very large numbers of genes have no known function, and genes responsible for common diseases like hypertension are not known. It is assumed that such diseases are controlled by a number of genes, and are under strong influence of the environment (diet, smoking, exercise). The search for disease genes uses the techniques of gene mapping, where populations of subjects are tested and a statistical correlation between a part of a chromosome, containing a number of genes, and the disease is expressed as a quantitative trait locus (QTL) [21]. The study of QTLs leads to the identification of genes which are candidate genes first, until it is proven that they are the cause of a disease. The study of QTLs is easier in animals (rat, mouse) because they are bred to be genetically identical, while human genomes have a variant DNA letter every few thousand letters for any two humans, and that is why statistical correlations are harder to make. Diseases are often studied in animals, and then the candidate human gene will be sequenced (from the blood samples gathered from patients), and subjected to further analysis which may uncover the biochemical causes of disease.

Biologists are faced with very large data sets. A QTL may contain around a hundred genes, or a few million letters of DNA code. On the other hand, single nucleotide polymorphisms (SNPs), which are individual DNA differences, are one letter long, and need to be shown along QTLs. Visualisation is the only viable way of making this data availa-

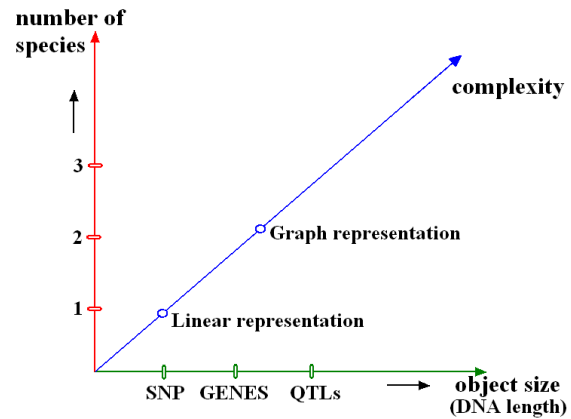


Figure 1. Genome browser classification schema.

ble, as close reading of thousands of letters is not a solution. That is why genetic databases visualise data in the form of maps which show linear arrangement of genetic features. To our knowledge, the resulting visualisations have not been subjected to much scientific scrutiny, so far. They are used by thousands of scientists daily, but it is not clear how they should be designed and how well they support scientific activity. It is our aim to study this, and to deliver better visualisations which can enhance the process of scientific discovery.

USER SCENARIO

We cooperate with a number of biological research groups who work in the areas of cardiology, metabolic diseases, schizophrenia and cancer. Those researchers conduct large scale experiments using *micro array* technology. In a micro array experiment the activity of all genes is examined simultaneously. What is measured is gene expression, that is the amount of the intermediate product, produced by the DNA, and leading to the production of a protein or a gene control element. The interpretation of such experiments requires simultaneous visualisation of chromosomes, genes, micro array probes, markers, and QTLs in three species: the mouse, the human, and the rat. Additionally, SNPs which may harbour DNA mutations causing a disease need also to be shown, along blocks of SNPs shared by population groups, and called haplotypes (www.hapmap.org).

CLASSIFICATION SYSTEM

We classify genome browsers according to tree dimensions, see Fig. 1. In the first dimension (number of species) we find that genome browsers represent between one and many species. Ensembl can be used to view one species at a time but other species information can be superimposed (see Fig. 5C). K-BROWSER can show a number of species and the number is limited by the size of the web page (see Fig. 7). Multiple alignment tools can show a number of aligned sequences from different species and those sequences can also be shown as a tree, see Fig. 2I (alignment) and Fig. 2J (phylogeny).

The second dimension represents the size of the objects shown. The smallest objects are one DNA letter long (SNPs).

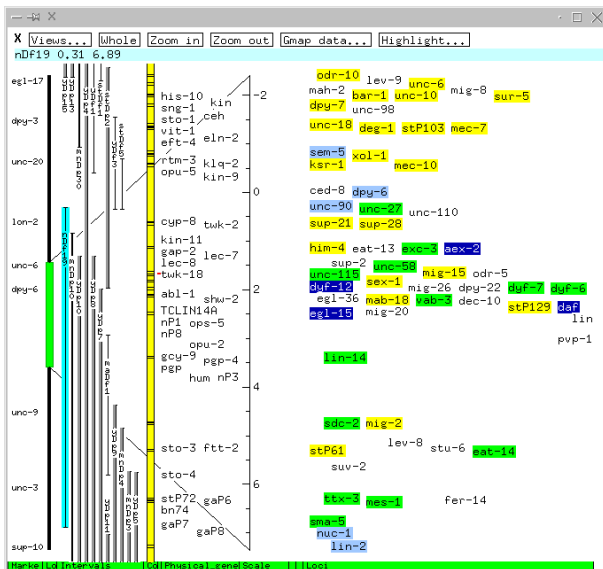


Figure 3. AceDB-representation of worm chromosome X.

In the order of increasing size one can show gene promoters, exons and introns, and other constituent parts of genes. Genes are about 20-30 thousands of DNA letters long and QTLs may contain thousands of genes. QTLs may approximate chromosome bands in size. Finally, human chromosomes are between 50 and 300 million letters of DNA long.

From the point of view of representation complexity, we classify browsers into linear and graph representations. In graph representations we can distinguish trees, networks (pathways), and 3D structures (proteins). JMol [25], see Fig. 2H, is one of the viewers showing protein 3D structure, while Treeview, Fig. 2J, offers a tree representation of a phylogeny. BugView (Fig. 2F), Ensembl (Fig. 2B) and SyntenyVista (Fig. 2D) show genome comparisons as bipartite graphs.

SURVEY OF GENOME BROWSERS

In this section we survey genomics visualisation software such as AceDB [29], SyntenyVista [1], DerBrowser [2], Apollo [3], Artemis [4], BugView [5], Ensembl [6], Sockeye [9], K-BROWSER [10], GBrowse [11], NCBIMapViewer [12], eQTL Explorer [14] and Expressionview [22]. We compare the systems in order to understand the problems and possible solutions to data visualisation.

AceDB

AceDB [29], see Fig. 3, is one of the first tools for genome visualisation. It offers a graphic representation which contains many objects in various colours. Colours help the researcher to identify the objects. For example, when a marker is coloured in yellow, it means that this marker has been cloned. The users can view textual details by double clicking on an object. AceDB offers simple zooming activated via zoom buttons. The viewer offers three types of sequence view: a genetic map, a physical map, and a sequence window which shows the DNA or AA letters. All views offer pop-up menus.

SyntenyVista

SyntenyVista [1], see Fig.2D, is the first interactive representation of synteny data designed for large genomes. It shows information about the human, rat and mouse genomes, and allows us to see the relationships between genes and chromosomes in two species at a time.

SyntenyVista focuses on the visualisation of gene comparisons. The tool shows relationships between genes, syntenic groups, chromosomes and QTLs. It has features which make it more usable than other existing genome browsers. SyntenyVista shows the whole chromosome with detail and supports choosing the part which will be investigated. The view uses colour and chromosome numbering to support understanding at the starting point of the visualisation. The users can manipulate the view by using both mouse and keyboard interaction. The application (SV1) offers the option to invert the chromosomes, which was found to be useful. It offers smooth zooming which supports the visual exploration of the chromosome space. The users can keep an area of interest in focus during the zooming process. The developers have also enabled panning. The users can move the chromosome with the mouse on the gene panel, or drag the box enclosing the region of interest. The display of the genes can be scaled by using a mouse action. The second version of SyntenyVista (SV2) has the cartoon scaling feature.

SyntenyVista includes also a top panel allowing additional user interaction and presenting information. The panel displays information on genes or QTLs in response to mouse movement in the gene area. QTLs are displayed as thin lines along the chromosome axes. The panel offers options to search for a gene name or a chromosome position. A gene is then highlighted on the whole chromosome image and the gene and its counterpart in the other species blink for a few seconds.

DerBrowser

DerBrowser [2], see Fig.2E, was designed at the time of the human genome sequencing project. It is a Java applet which supports interactive visualisation of one chromosome, or of a chromosome part. It can use a local database to produce web pages showing all the information describing a given map object.

It can be used to display genes, chromosome bands (chromosome parts coloured light or dark in the karyotype pictures), markers, and can represent any object on a map [15]. It provides an illusion of smooth zooming (a slider), and supports the hiding of objects, based on object type. It can also perform search functions.

Apollo

Apollo [3], see. Fig.4, is a sequence annotation viewer and editor. It allows the biologist to improve on the genomic feature descriptions derived from automated analyses and computational pipelines. It facilitates connecting to various databases and the comparison of existing annotations with other biological data. The tool offers researchers the ability to probe, manipulate and alter the interpretation of the underlying

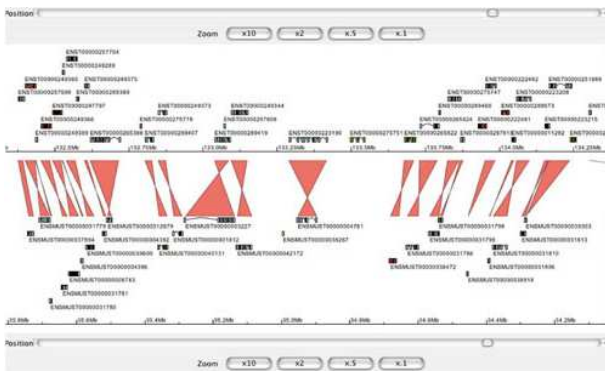


Figure 4. Chromosome comparison represented in Apollo. We can see human chromosome 20 at the top and part of mouse chromosome 2 at the bottom.

data. Within the various views offered by the package, annotations can be created, deleted, merged, split, classified and commented upon. The tool allows the view to be scaled using zoom buttons and provides a degree of semantic zooming. Some features are not displayed at low zoom levels and appear more precisely only when the user zooms in on them. The users can move to a specific position by specifying a coordinate, gene name, or short sequence string, or by using the horizontal scroll bar. Apollo can display features on two genomes at the same time. The view offers zooming and panning but it still does not present the data clearly, and the users cannot see all the relevant details.

Artemis

Artemis [4], see Fig. 2G, is a genome viewer and annotation tool that visualises sequence features and the results of analyses within the context of the sequence, and its translation from DNA to protein. Artemis can be used as a sequence viewer and is suitable for smaller genomes. Properties of the sequence can be plotted. Each plot allows dynamic modification of the window size used for the calculation. The sequence and plots can be zoomed together into the single base level or out for the complete genome. Artemis provides two sequence windows to view the same sequence at different zoom levels simultaneously. The tool can be run as an applet within a web browser.

BugView

BugView [5], see Fig.2F, is a comparative genome browser. It allows one to compare the arrangement of genes in two genomes, and can also be used to view individual genomes. It was written to enable comparative study of bacteria, including the comparison of bacterial strains.

The view presented by BugView is restricted to the genes, showing gene overlaps, and, where relevant, intron-exon structure, including alternative splicing. The users can scroll and zoom smoothly, and search for gene names. BugView includes support for the comparison of genes (sequence analysis) and analysis of gene alignments and other sequence features. For instance, one can filter sequence alignments by specifying percentage similarity.

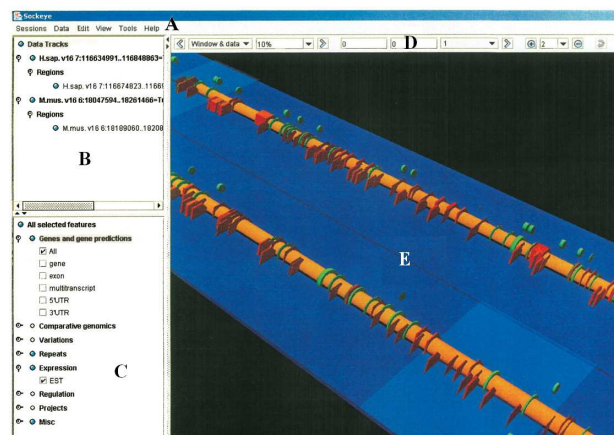


Figure 6. Sockeye chromosome visualisation in 3D. We see the menu (A), the sequence track selection tree (B), the feature selection tree (C), the navigation toolbar (D), and the 3D viewport (E). The application allows the users to show/hide and obtain detailed information for loaded sequence track annotation types. In 3D viewport the users can perform analysis and annotations.

Ensembl

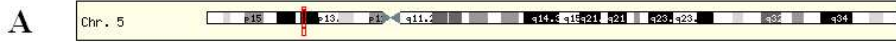
Ensembl [6], see Fig. 5, is probably one of the most popular systems for genome analysis. Ensembl database organizes biological information around the sequences of large genomes. It is an interactive Web site, a set of downloadable flat files, and a complete, portable open source software system for handling genomes. The Ensembl browser displays assembled sequences, cross-species synteny, genes, transcripts, proteins, supporting evidence, dot-plots, protein domains and gene/protein families.

The users can find 17 different views for data offered by Ensembl such as: AlignView, AnchorView, ChromoView, ContigView, CytoView, DomainView, ExonView, FastaView, GeneView, KaryoView, MapView, MarkerView, MultiContigView, ProteinView, SNPView, SyntenyView, and TransView. Different views are used to represent different kind of data. In our experiment, a number of genomic data was represented by ContigView, MultiContigView and SyntenyView. In SyntenyView a diagram of chromosomes with blocks of conserved synteny and homology matches between individual genes with syntenic blocks are shown. In ContigView, Fig. 5, a set of different views of a gene is shown, from broad chromosome context to fine nucleotide detail. These views are in separate horizontal frames, one below the other. The data presented in Ensembl is supported by labelling and searching. MultiContigView is an extension of ContigView. It allows display of genome annotation for several species. We find that because of the size of the data set, it is difficult to show all requested details on one screen. The users need to use scrolling and very often get lost in the information space.

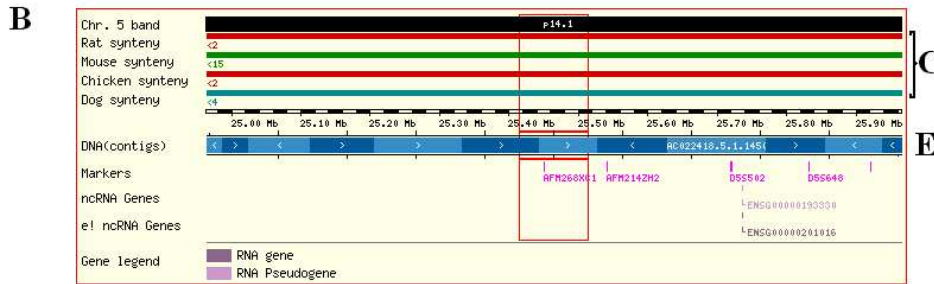
Sockeye

Sockeye [9], Fig. 6, uses 3D graphics and data from the Ensembl database project. A user can also import custom se-

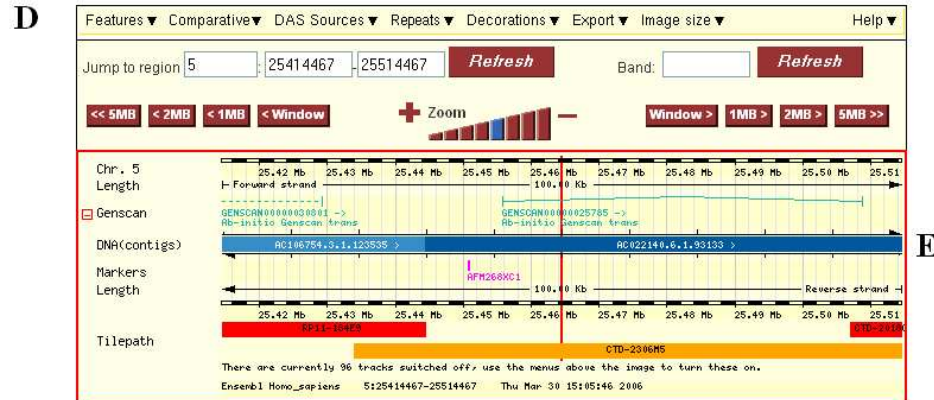
Chromosome 5



Overview



Detailed view



Basepair view

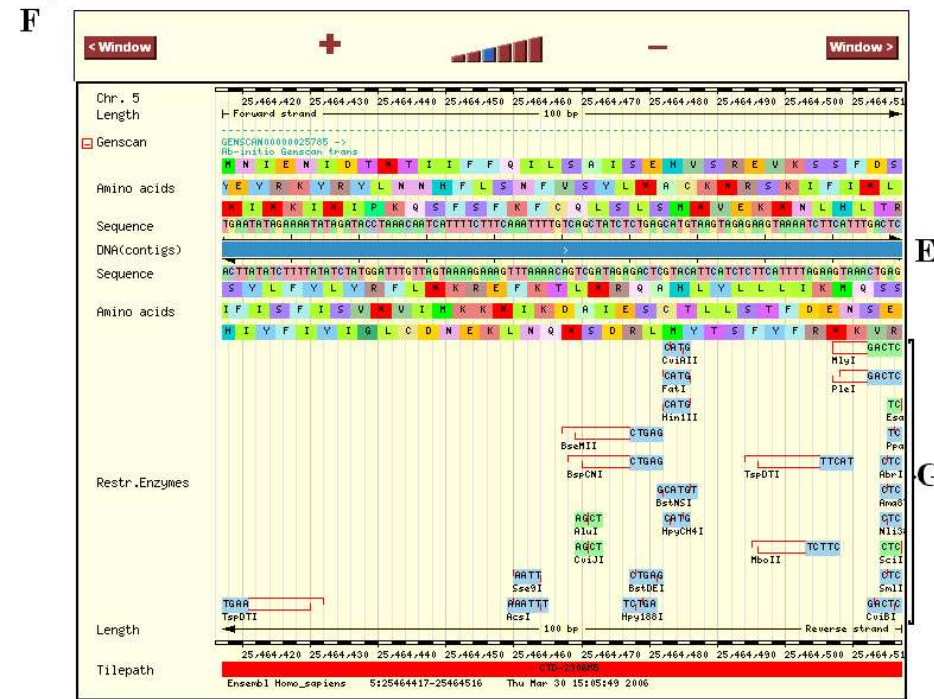


Figure 5. Ensembl - ContigView - human chromosome 5. ContigView provides a high level view of the contig sequences (E) that form the genome sequence assembly, and of genes and other features that have been placed on it. The figure shows the entire chromosome (human chromosome 5, see A), an 'Overview' (B) panel displaying a chromosome region of up to 1 Mb, the 'Detailed View' (D) panel showing genes and markers, and a 'Basepair View' (F) panel showing within a small assembly region of up to 500 bases the actual sequence, translations and restriction enzyme recognition sites (G). C shows syntenic chromosome fragments in other species.



Figure 7. K-BROWSER showing the cystic fibrosis gene region (CFTR). Human, mouse and rat annotations are presented (from the top to the bottom panel). The grey bars indicate gaps (arising from insertions or deletions) in the sequences. The user can navigate using zoom buttons, gene name searching, and position jumping.

quences and annotation data. Large sets of functionally linked sequences containing genes that are coexpressed, and orthologous across multiple species, can be analysed. The difference between Sockeye and other existing browsers is in the 3D environment. Each 3D model is specified in a user configurable XML format file. Sockeye integrates the process of obtaining sequence and annotation data. The application also allows a user to simultaneously visualise several different alignments and easily view their gaps. Montgomery et al. [9] stress that the 3D environment has a lot of advantages and disadvantages but only a few researchers decided to use it in their work. 3D visualisation is uncommon in genomics and researchers find it difficult to use. The developers find Sockeye to be user-friendly, but the users cannot easily see all interesting objects. The interface shows the sequence track selection tree, the feature selection tree, several navigation controls, and the 3D viewport. The users can also compare the extensive information contained across multiple genomics sequences, and zoom, pan and rotate the position of the sequence track.

K-BROWSER

K-BROWSER [10], see Fig. 7, is a comparative browser which visualises biological information at a higher level of resolution than is the case in most other tools. Its novelty is the representation of sequence similarity histograms along sequence features on several genomes. K-BROWSER was built on the foundation of the UCSC Genome Browser [24]. It can display a number of genomes overlaid with annotations and predictions, and shows the multiple alignments that describe global sequence relationships.

K-BROWSER takes as input a specific region in a genome and produces a set of images that succinctly represent the requested region and all orthologous regions in other genomes. The two critical components of the application are track

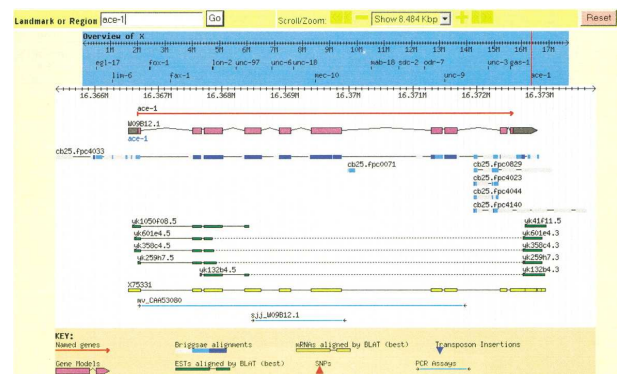


Figure 8. GBrowse. The users can type a landmark name into the text field at top. Landmarks can be gene names, clone names, accession numbers, or any other identifier configured by the administrator. Once a region is selected, it is displayed in a detailed view that summarizes annotations and other genomic features.

realignment and image generation. Track realignment is responsible for the necessary scaling of DNA lengths in the comparative views to make it consistent with the multiple alignment. Image generation takes as input a genomic region query and produces an image for every corresponding region in the multiple alignment. The tool displays a sequence conservation plot above the tracks. It allows the users to select a track according to which the conservation plot is to be coloured. The K-BROWSER can compute the percentage similarity between the root sequence and the leaf sequence in a window centred on a specified position. It allows one not only to determine if a genomic region is conserved within other genomes, but also to infer the rate at which it is evolving.

GBrowse

The Generic Genome Browser [11], see Fig. 8, is a combination of a database and interactive web pages for the manipulation and display of genome annotations. GBrowse can display an arbitrary set of features on a nucleotide or protein sequence, and can accommodate genome-scale sequences. GBrowse provides most of the features available in other browsers but was designed from the outset to be portable and extensible. It provides multiple configurable levels of zoom and two scroll speeds, and it also offers semantic zooming. The users can customize the view, the track, and the width of the image. The application allows for adding annotations to the genome. GBrowse supports also a plug-in architecture that allows third party modules to extend it.

NCBIMapViewer

The NCBI Map Viewer [12] is a Web interface used to view and search an organism's complete genome. The users can also view maps of individual chromosomes and zoom into specific regions within chromosomes to explore the genome at the sequence level. They have access to several different types of maps for different organisms. Map Viewer allows the user to view these maps graphically or in a table format. NCBI Map Viewer's graphic display is limited to features

Genome browsers	Technology
AceDB	initially in C, later connectivity via Perl, Java or CORBA [18]
SyntenyVista	Java - Piccolo [13] and Swing [27]
DerBrowser	Java 1.02, java applet
Apollo	Java 1.2 or 1.3
Artemis	Java Application, but can be run as an applet
BugView	Java 1.1, java applet
Ensembl	MySQL [31], Perl API, and Java API, images are generated dynamically using Ensembl drawing code [7]
Sockeye	standalone application in Java, using JDK 1.4.x and Java 3D 1.3.x
UCSC GenomeBrowser	MySQL, BLAST-like Alignment Tool (BLAT) [19]
K-BROWSER	image generation component borrowed from UCSC GenomeBrowser
GBrowse	MySQL, DAS, Perl, and Apache
NCBI	Entrez System
eQTL Explorer	Java
Expressionview	Perl script derived from the Ensembl program blastview

Table 1. Technologies used to implement genome browsers.

related to gene identification, although there are text links to other pages. Zooming and other visualisation features are not as sophisticated, in our opinion, as those offered in Ensembl.

eQTL Explorer and Expressionview

eQTL Explorer [14] visualises QTL data on the background of each chromosome. The chromosomes are drawn as vertical bars, and the QTLs are shown as coloured triangles. The application can display individual chromosomes in a separate view, with options to browse, zoom and export data. The tool has also a pop-up menu which provides access to annotations and cross-references to external data sources. The tool represents only a small subset of genome data.

Expressionview [22], see Fig. 2C, and eQTL Explorer, which is similar in appearance, are two applications designed specifically for the analysis of micro array experiments. Both applications show entire karyotypes and draw QTLs and genes identified in a micro array experiment alongside the chromosomes. Both applications are single-purpose, in that they do not show other biologically relevant information at the same time, for instance all the genes or SNPs.

SUPPORTING TECHNOLOGIES

The genome browsers we discuss use different technologies which offer differing levels of support for visualisation and user interaction. The newer viewers, such as SyntenyVista [1], use Piccolo [13] which allows for smooth zooming and panning. Piccolo toolkit supports the development of 2D structured graphics programs. It implements a hierarchical structure of objects and cameras, allowing the developers to manipulate objects, and the users more options in the presentation of data. SyntenyVista also uses Swing [27], which is a GUI toolkit for Java. Swing graphical user interface offers text, boxes, buttons, split-panels, and tables. The technology allows the developer to add ready-made and sometimes complex components to an application. At the other end of the spectrum we have clickable graphics generated by a server, such as Ensembl [7]. The developers define a clickable

area for graphics, and then, after user interaction, images are generated.

Visualisation software often requires the user to modify her software environment. The users need to have a specific version of Java and adjust security settings if they want to use an application based on Piccolo or Swing. There is also a very important limitation because of available memory and CPU speed on the users' machine. Some genome browsers, especially the ones which use the newest technology, expect a lot of memory. There is a trade-off currently between portability and visualisation. Most portable viewers use simple, server-side technology, and offer little in terms of view adjustment. On the other hand, powerful browsers written in Java need better hardware and need to be set-up but offer improved data analysis support. Most of the browsers we describe connect to a database, while some rely on flat files. Ensembl and GBrowse, for instance, support the addition of new data sources via the Distributed Annotation Service (DAS) protocol [30]. DAS is an open source standard supporting the sharing of genomic annotations on the web.

TESTING

In cooperation with the biologist groups, we tested all the described genome browsers in order to find which one supports the interpretation of their experiments (see <http://www.dcs.gla.ac.uk/~asia/work.html>). We found that none of the browsers fully supports user requirements we identified. Using AceDB with Ensembl data is not feasible, and would make us inherit the limited zooming support offered by the AceDB maps. On the other hand, it would have been possible to add new data from the lab easily, and add new data types. We found that DerBrowser's functionality does not fulfill users' expectations. We expected that the user should be able to move around the data columns and to zoom smoothly and precisely. Beyond a certain point, we could not zoom in any further and we could not see the genes and micro array probes in detail. We could not compare two genomes either. It was also impossible to add new features because of the old version of Java the application uses.

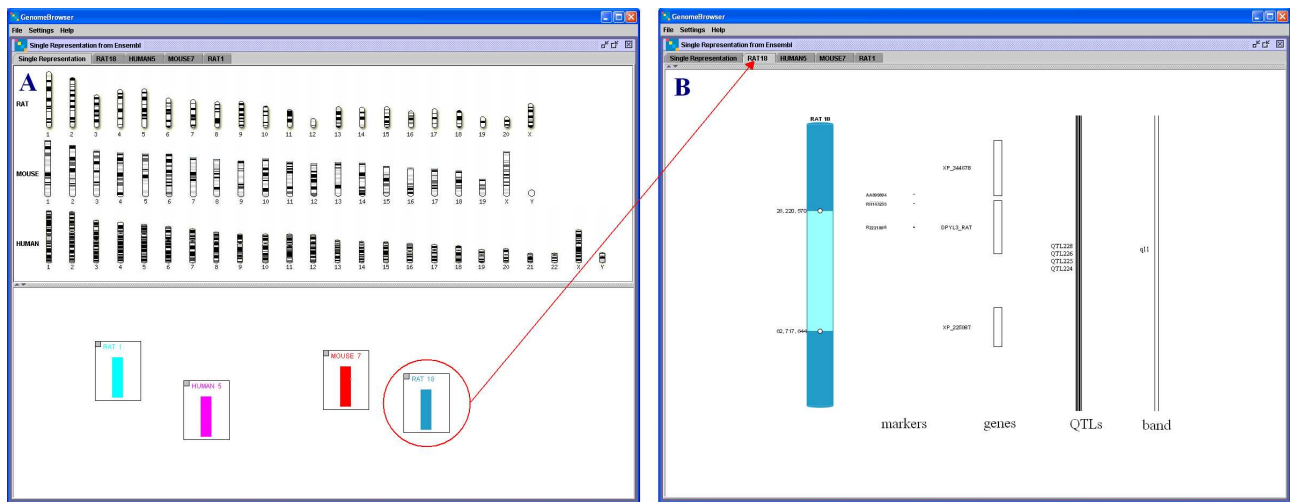


Figure 9. VisGenome offers two new views. View A shows chromosomes from three species (the mouse, the rat and the human) in the upper part of the display and the chromosomes for which data has been retrieved from Ensembl in the lower part. View B contains an overview and detail for the rat chromosome 18.

We found that Ensembl is not appropriate for our users, as it does not support the comparison of QTLs and their gene content, due to the limited flexibility of view manipulation. The same is true for the NCBI Entrez system, Gbrowse and K-BROWSER. We also looked at the maps offered by the Rat Genome Database [8] and saw that the data required by the biologists was not shown. SyntenyVista solves only some of the visualisation problems. It supports the comparison of two genomes, but it does not display all the relevant data (micro array probes, SNPs, markers, etc.). We experimented also with Apollo's user interface which is meant to be intuitive. We found it harder to use than other interfaces, and found that the display was not clear and zooming was not satisfactory. We also found that BugView was easy to use, but, unfortunately, shows only a subset of data that the biologists want to visualise. This is similar to the situation we encountered in SyntenyVista.

eQTL explorer and Expressionview show only a subset of data, and the users cannot compare known genes or SNPs with data represented by the visualisation. Completely different are Ensembl and NCBI Map Viewer, which read data directly from a huge database and show as much data as it is possible. The users easily get lost in such interfaces, as the data is shown in several screens which do not fit simultaneously on the computer screen. This limitation is the result of the lack of support for image manipulation within web browsers. The images shown by Ensembl and NCBI never fit on one screen and we found that disorientating. In Ensembl the display of synteny, see Fig. 2B, does not present much detail and can not be used for micro array data analysis. The MultiContigView is much less legible than SyntenyVista, and does not offer smooth navigation. We also examined Sockeye and found the 3D view to be confusing. This was mostly due to poor labelling and possibly visual occlusion.

DESIGN OF VISGENOME

We developed a new version of SyntenyVista, VisGenome. The software extends SyntenyVista with new features, and allows for the addition of new data types to the display, and will be able to satisfy user requirements fully. The data are presented vertically. The application loads the data from Ensembl. It welcomes the user with a view of all rat, mouse and human chromosomes. Then, after choosing a chromosome of interest, the user sees it in the bottom window. After the user selects the chromosome by clicking on it, a new view with detailed data about the chromosome is created. This solution allows the users to see in one place what data was downloaded from Ensembl as well as the detailed information on the chromosome, including bands, markers, QTLs and genes. After choosing a chromosome the users can manipulate the view by mouse and keyboard interaction. We offer smooth zooming which supports the visual exploration of the chromosome space. The users, in the same way as in SyntenyVista, can keep an area of interest in focus during the zooming process. We implemented zooming and panning using Piccolo [13]. The users can choose the chromosome region of the interest by dragging the box enclosing the region or typing in the coordinates in the top info panel. Then only the data in the selected area is displayed. The solution allows us to keep the context, the users can navigate the data and all the time they know exactly in which region of the chromosome the data is situated.

The new genome browser, see Fig. 9, shows bands, markers, QTLs and genes in a single representation. It can show any data types specified by a query sent to the Ensembl database. We are currently adding the display of SNPs, and will also add haplotype blocks and protein expression results, and allow the user to adjust the display to suit their information needs.

USER TEST

We will carry out a user test with 10 users, in two settings, cardiovascular research and schizophrenia. The users will be performing the following tasks. First they will read in the data from their latest micro array experiment, stored in a spreadsheet. The visualisation system will show an overview of chromosomes to which the new results relate, similar to that seen in Expressionview (Fig. 2C), where both the QTLs and differentially expressed genes will be shown superimposed on a karyotype picture. Then the biologists will select the longest of the QTLs in which they are interested, and verify that they can see the QTL, the genes, and the micro array results. Micro array results will be coloured in two colours, one for genes showing increased expression, the other for the genes showing reduced expression. The view will also display results imported from another micro array experiment, from external published data selected by the biologist, for comparison.

The following will be measured: total time required to perform the visual assessment of the new experiment; time to examine one QTL in detail; and number of mouse and keyboard actions executed. Additionally, a survey will be used to get user impressions on the legibility of the display, aesthetic appeal, and the subjective ease of use.

CONCLUSION

Visualisation of genome comparisons is an important research tool in biology and medicine. There are a variety of genome browsers which in practice should perform the same function - show the chromosomes of some species in detail. The differences in the view and also in functionality of the tools for genome browsing motivated us to create a classification of genome browsers.

Our future plans include more experiments with the users to check which of the tools' properties are welcome and which are less user-friendly. We would like to test not only VisGenome but also different tools, with biologist groups we cooperate with, to find the most intuitive visualisation technique. We are going to continue our work with VisGenome, which shows genome data at different level of details. We believe, that biologists still require new methods to visualise genomic data.

ACKNOWLEDGMENTS

We thank Prof. A. Dominiczak, BHF Glasgow Cardiovascular Research Centre and the Wellcome Trust Cardiovascular Functional Genomics Consortium for their collaboration and Prof. K. Dittrich at the University of Zurich for hospitality.

REFERENCES

1. Hunt, E. et al. The visual language of synteny. *OMICS* 8(4), (2004), 289–305.
2. Leser, U. et al. IXDB, an X chromosome integrated database. *NAR* 26(1), (1997), 108–111.
3. Lewis, S. E. Apollo: a sequence annotation editor. *Genome Biology*, (2002).
4. Rutherford, K. et al. Artemis: sequence visualization and annotation. *Bioinformatics* 16(10), (2000), 944–945.
5. Leader, D. P. BugView: a browser for comparing genomes. *Bioinformatics* 20, (2004), 129–130.
6. Ensemble database. <http://www.ensembl.org>.
7. Hubbard, T. et al. Ensembl 2005. *Nucleic Acids Res.* 2005 Jan 1;33 Database issue:D447-D453.
8. Rat Genome Database (RGD). <http://rgd.mcw.edu>.
9. Montgomery, S. B. et al. Sockeye: A 3D Environment for Comparative Genomics. *Submitted Genome Research*, (2003).
10. Chakrabarti, K. and Pachter, L. Visualization of multiple genome annotations and alignments with the K-BROWSER. *Genome Research*, (2004).
11. Stein, L. D. et al. The genetic genome browser: a building block for a model organism system database. *Genome Research* 12(10), (2002).
12. Online Mendelian Inheritance in Man. <http://www.ncbi.nlm.nih.gov/>.
13. Piccolo Toolkit. <http://www.cs.umd.edu/hcil/piccolo/>.
14. Mueller, M. et al. eQTL Explorer: integrated mining of combined genetic linkage and experiments. *Bioinformatics* 22(4), (2006), 509–511.
15. The human chromosome 21 database. <http://chr21.molgen.mpg.de/>.
16. NCBI Entrez. <http://www.ncbi.nlm.nih.gov/Entrez>.
17. British Heart Foundation Blood Pressure Group. <http://www.medther.gla.ac.uk/bhf/index.htm>.
18. CORBA. <http://www.corba.com/>.
19. Kent, W. J. BLATthe BLAST-like alignment tool. *Genome Res.*, 12, (2002), 656–664.
20. Cinema. <http://umber.sbs.man.ac.uk/dbbrowser/CINEMA2.1/>.
21. Hubner, N. et al. Integrated transcriptional profiling and linkage analysis for identification of genes underlying disease. *Nat. Genet.*, 37, (2005), 243–253.
22. Fischer, G. et al. Expressionview: visualization of quantitative trait loci and gene-expression data in Ensembl. *Genome Biology*, 4, (2003).
23. MGI. <http://www.informatics.jax.org>.
24. Karolchik, D. et al. The UCSC Genome Browser Database. *Nucleic Acids Res.* 31: , (2003), 51–54.
25. Jmol. <http://jmol.sourceforge.net>.
26. Medical Research Council. <http://www.mrc.ac.uk>.
27. Swing. <http://java.sun.com/products/jfc/>.
28. Page, R. D. M. TREEVIEW: An application to display phylogenetic trees on personal computers. *Computer Applications in the Biosciences* 12, (1996), 357–358.
29. Durbin, R. and Mieg, J. T. A C. elegans Database. Documentation, code and data available from anonymous FTP servers at lirmm.lirmm.fr, cele.mrc-lmb.cam.ac.uk and ncbi.nlm.nih.gov, (1991-).
30. Dowell, R. D. et al. The distributed annotation system. *BMC Bioinformatics*, 2:7, (2001).
31. MySQL. <http://www.mysql.com>.
32. Metabolic Pathways. <http://www.lirmm.fr/~fjourdan/mainE.html>.