UNIVERSITY OF OSLO
Department of Informatics

Merging the physical properties of DNA with genomic annotations in Ensembl

Master thesis

Geir Ivar Jerstad

24th April 2006

Table of contents

Abstract

On a DNA sequence, we attach information about its features and attributes, and this kind of information is called annotations. Over the past few years there has been a development to gather and group annotations to a central service, so that scientists will be able to compare all kinds of annotations. Comparisons are performed with the aim of identifying related biological features. Ensembl is such an annotation centre, and this thesis addresses the issue of integrating an annotation made by the stitch profile algorithm into Ensembl. This stitch profile algorithm is a novel way of calculating the different conformations corresponding to a DNA melting profile, i.e. modeling of the physical attributes of the DNA double helix, so that it becomes easier to see what state the DNA molecule is in. We then analyze the how accurately the stitch profiles correlate to the annotations in Ensembl.

1 Introduction

Today we have three major centralized web services to gather and provide genomic annotation for most genomes. These are the NCBI map viewer[18], the UCSC golden path[17] and the EMBL/Sanger Ensembl [2], [14]. With these services, scientists can browse different kinds of annotations to discover new genes or functionality of the genes. All three services serve several types of genomic annotation browsing, but they differ in functionality and layout. Ensembl is the only one which is open-source and is available for downloading. They are all operating on the same genome sequence for a given organism, and this makes it possible to cross reference genomic annotations.

There exists an algorithm that calculates the physical properties of a DNA sequence and plots the results into something called a “stitch profile” [4]. The stitch profile represents an analyzed melting map, and draws the several possible conformations which a melting DNA might have for a given temperature.
These conformations show the probability of different parts of the genomic sequence to be in a closed or open conformation. The physical properties of the DNA are shown to be connected to coding and non-coding regions of a sequence [6], and it is therefore interesting to investigate how we can use the melting information in order to find out more of the sequence.

Prior to the thesis work, an online stitch profile analysis service was created where the scientist could submit the sequence of interest for analysis and the results would be shown in a plotted image.
We wish to use this profile as an annotation in the Ensembl framework in order to make it easier to make comparisons between the many annotations in Ensembl. This integration will use the same code for stitch profile calculations as the online version, but with modifications to make it useable with Ensembl.

Ensembl was chosen as the framework solution for this integration because it was the only open-source annotation service and it is comprehensive enough to make comparisons.

1.1 Readers’ guide

This thesis is basically divided into three main parts, where the first one provides background information of the concepts and technology used in the implementation. The next main part is the design and implementation, and the last part investigates the biological correlation between the stitch profile against known coding and non-coding regions annotated in Ensembl. The biological part tries to find patterns between the annotations and also investigate how accurate the profiles are in their predictions.

Background information is covered in the chapters 2-8, while design and implementation are described in chapter 9 and 10 and the biological comparison is in chapter 11. At the end of the thesis, there is a short summary and conclusion.

2 Genomes

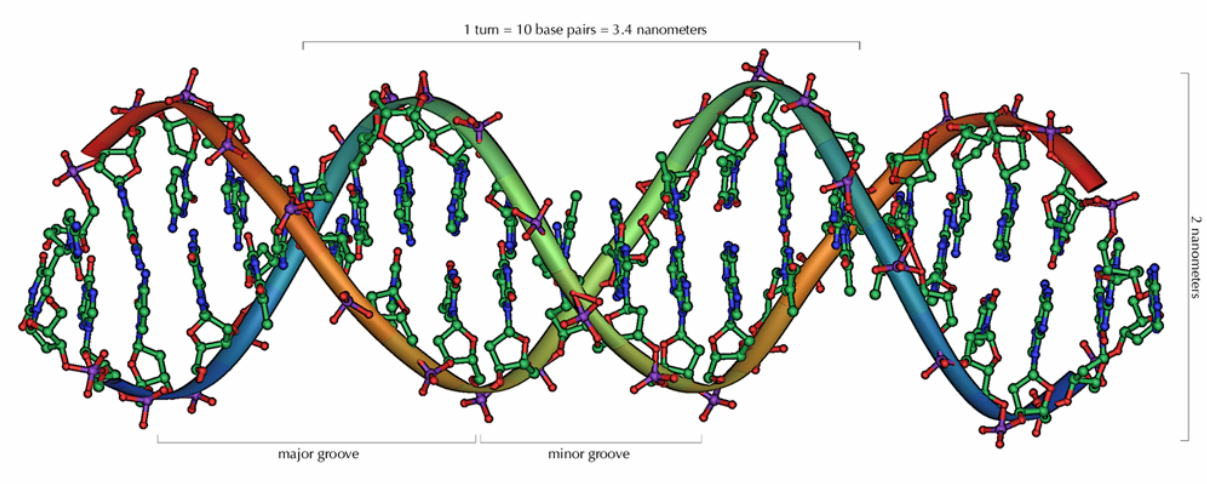


Figure 1 - A piece of the DNA showing the structure

2.1 The discovery of the genome

After Watson and Crick [23] discovered the structure of the Deoxyribonucleic acid (DNA) back in the year 1953, there has been a tremendous effort to find out more of the function to this structure. Almost everything within a cell and outside is controlled by the cell nucleus, where the DNA gets transcribed and exported out of the nucleus to the ribosome where it becomes a protein, which are the functional parts of the cell.

The DNA is the description of all the genetic information or hereditary material within a cell, and consists of the nucleotides adenine (A), thymine (T), cytosine(C) and guanine (G) to form a chemically linked chain. This chain is connected to a complementary chain of nucleotides and together these forms a helix like illustrated in Figure 1. This genomic information can be extracted from an organism by the procedure of sequencing.

2.2 Sequencing

The procedure of turning the inheritance material inside the cell nucleus into human readable form is complex and takes a long time. The humane genome [22] contains about 3 billion nucleotides in length and is stored as 24 chromosomes (22 autosomal chromosome pairs and 2 sex chromosomes). Before sequencing can start, the genome data gets split up into 150k bp chunks and inserted into a bacterial artificial chromosome (BAC). This BAC is then put into bacterial culture to grow and to make clones of the inserted DNA. The cloned sequence gets extracted and ‘shotgunned’ into pieces with about 1500 bp in length. The shotgunned sequences are then read by a sequencing machine, which translates the DNA molecule into human readable form. These sequences have many regions which overlap with each other and this makes it possible to find the correct order of the clones to be put together and thus reconstructing the genome.

This process can be compared to the activity of solving a puzzle where the correct bits and pieces need to fit in. The end strands on each sequence is checked against the other sequences in order to find overlap which will determine the order of the sequence.

Putting the human genome together has been done with several sequencing centers around the world collaborating in the project called the Human Genome Project (HGP), which lasted for 13 years until it was completed in April 2003. At this point the project had basically sequenced most of the human DNA, but there were still some missing pieces, and gaps still exist today (see recent chromosome 15 reference [24])

Since the publication of the human genome sequence, there have been many new assemblies like the elephant, armadillo and rabbit genomes, and in the near future we have others like the cat and guinea pig genome to be fully sequenced.

Depending on the requirements of the project (i.e. the sequencing and assembly progress), each assembly gets an update on a regular basis and then all annotations should also be updated. As the number of new genomes grows, so does the workload to keep the annotations updated.

2.3 Representation and storage

The genetic information is stored in units symbolized by the “letters” A,T,C and G, where each letter represents a nucleic acid unit component called a base. The genome is found by sequencing the DNA and putting together the pieces so that we get one long and searchable text string. These strings can become very large, e.g. chromosome I of the human genome with 247 million basepairs, which means a string with that many ‘letters’.

There are many ways to store a genome, but the representation is basically the same. In addition to [A,T,C,G] there may be other letters to represent, for example, the presence of uncharacterized base positions and SNPs (Single nucleotide polymorphism)[21]. The genome also has markers mapped, such as genetically variable loci that work like landmarks and several coordinate systems exist for positioning the annotation to a specific area.

A widely used format to represent the nucleic acids is the FASTA format. This format includes symbols for the nucleotides and degenerately mapped nucleotides, and stores sequences as plain text files separated by sequence description lines starting with the symbol “>”. Every annotation center can serve every sequence in this format and many more formats. Internally, the annotation centers are not using this format, but instead have their own database schemes. Ensembl store their genomes in a mySQL database [5] along with many other attributes like versioning, attributes and history.

2.4 Expression

In the human genome, only a few percent (3%) of the sequence comprises what we call the genes, i.e. subsequently being converted to proteins or functional RNA molecules. The rest of the regions have often been considered ”junk” or to some extent have other functions. A eukaryotic cell has less density of genes than a bacterium because of the size and the structure of the DNA molecule, thus making gene finding more difficult because of all the ‘noise’. Also, eukaryotic genes are generally not continuous,but contain exons and introns.

Transcripting the DNA first require that the DNA opens up in that particular region and a RNA polymerase produces the complementary RNA. To break up the DNA, the polymerase needs to ‘open’ the region in order to bind and copy the sequence.

3 Annotation

An annotation is a description of an area within the genome. An annotation might describe a coding area, a feature of the area or other comment. This annotation will be mapped against the DNA sequence and must use a coordinate system to position it.

Figure 2 – Example from Ensembl showing horizontal annotations

The example in Figure 2 is an edited screenshot from Ensembl and shows the DNA contigs as the main ruler that represents the DNA thread. Above and below the thick blue line are the annotations shown in little boxes that have a start and an end point, and annotations above the blue line is the annotations positioned on the forward strand of the sequence while the annotations below is positioned on the reverse strand. All boxes are clickable and link to more information about the annotation.

Annotations are used for storing information concerning an area of interest in the genome. These annotations can be ‘novel’, which means that they are predicted rather than experimentally found, and where ‘known’ means that they have been mapped from a protein.

Annotations can be made in several ways such as “The Ensembl Automatic Gene Annotation System” which is a system that performs several analyses on a genome and automatically produces annotations. One of these analyses performs similarity matching to other known sequences in other genomes to find similar functionality, and annotate the sequence with this information.

There are annotations on the gene and protein levels. It depends on the context. In this thesis, annotations are on the gene level, because a stitch profile is based on the string of nucleotides. An annotation can contain any information about the area of interest. Examples of this may be a unique gene ID, description of the gene and links to other annotations that are relevant.

3.1 Annotation services

Centralized annotation centers began to emerge as the amount of annotation information kept growing. These online services are of great value to those that are trying to discover the functionality of a gene and the fundamental dynamics within the genome. The services are continuing to grow in both functionality and quality.

National Center for Biotechnology Information (NCBI) - USA
This center was established on November 4, 1988, as a division of the National Library of Medicine (NLM) at the National Institutes of Health (NIH).
The bioinformatics within this center provide the biologists with tools like Entrez, Map viewer, Blast and many other tools.
Map viewer is a tool which resembles Ensembl, but it displays the sequence map vertically. This Map viewer also has some different features like several ways of searching for a known gene and even more ways to analyze this gene.
Map viewer does not have the support of aligning manual annotation alongside onsite material.

University of California, Santa Cruz (UCSC) - Genome Bioinformatics
This university has an annotation service called “The genome browser” which resembles Ensembl in many ways. It can be discussed which of these two visualization methods are the better one, but both services can represent each others annotations to a certain degree. For example the “genome browser” can show Ensembl’s gene annotation that links into Ensembl’s webpage.
This browser also support custom made annotations, but they are limited to start and stop box annotations with links. Ensembl also have this limitation, but Ensembl is open-source and therefore it is possible to add functionality.

Ensembl - EMBL - European Bioinformatics Institute
“Ensembl is a joint project between EMBL - EBI and the Sanger Institute to develop a software system which produces and maintains automatic annotation on selected eukaryotic genomes. Ensembl is primarily funded by the Wellcome Trust.” [2]

This service is based on Open source and is available under the GNU license. Because of this, everyone can download and alter it to meet their goals. This application has a way of importing external annotation through a system called DAS (Distributed Annotation System) [7] and this is a way of uploading annotation into the genome and displaying it in the Ensembl browser window. UCSC supports DAS aswell. More information on DAS in the section called DAS in this thesis.

Ensembl is not enganged in sequencing DNA, but it assembles sequenced DNA from other institutes such as NCBI and Saccharomyces Genome Database (SGD). Each month, Ensembl releases a new version with the newest builds from the sequencing institutes with updated annotations that are connected to those genomes that were updated.

Ensembl is discussed in chapter 7.

3.2 Types of annotations

Ensembl supports several standards for making annotations. The user can upload the annotation into Ensembl through the DAS system and then view the results together with Ensembl's annotations. There are several standards such as the GFF (General Feature Format)[1] -format which is a simple annotation-format with the ability to align a feature to a sequence.
A simple annotation can be made with few parameters like:

- GenomeID = Genome identification
- Start = Specifies the start of the annotation on the sequence
- Stop = Specifies the stop of the annotation
- GeneFeature = This is the feature to the selected sequence

GFF and some other standards have more attributes to add more additional information to the annotation such as Score, Strand, Version handling and more. There are still scientists that use these formats even though they are to be considered out-of-date, but since there are no other standard available there is still a need for them.

Ensembl creates and store annotations in its own standard which is similar to the simple annotation form mentioned earlier, where the only differences are some extra parameters. The annotations are linked to the genomes stored in Ensembl through these parameters. When the annotation webpage (ContigView) is created, the annotations gets printed out to the webbrowser and presented to the user. An example of this is the annotation boxes in Figure 1.

3.3 Scalability - Comparing annotations

Disregarding the quality of the information being annotated, it is important that the viewer understands what is being presented. One of the aspects is scalability. To present an annotation which is only 100 bp long and the current window is 2 Mbp wide then the annotation becomes too short to be noticed. With the annotations in the original Ensembl, these will be scaled down to a 'dot' in the window which is barely noticeable, but still clickable.

The annotations in Ensembl are horizontally aligned with the contigs which again is mapped to a coordinate system representing all the bases in the specific window.

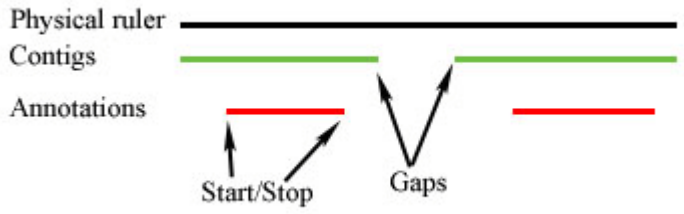


Figure 3 - Describing the annotation mapping with a sequence coordinate system

An example on how to use the comparability is to check several gene finding programs to see if one or more predicts the same gene. These programs will annotate the sequence and the annotations will be presented to the scientist in such a way that it becomes apparent where the genes are. In Ensembl these annotations are displayed as in Figure1 and 2 where each line represents one source of annotations aligned with sequence. In Ensembl, the scientist can click the annotation to get more information about that annotation with the exact source, sequence, correct interval and other features that this piece of sequence might have stored in Ensembl.

[1] http://www.sanger.ac.uk/Software/formats/GFF/

3.4 Coordinate systems

A genome browser needs a way to uniquely identify the positions of genes and to display these to the scientist in a coordinate based window.

Even though it is said that the human genome is completed, there are still holes in the sequencing. This leads to problems when we are adding annotations that might stretch over these holes, because the sequence is unknown. The process of sequencing begins with determining which contigs to sequence, then to use BACs to create the sequence, and then to assemble the bits and pieces afterwards into the correct order. This sequence then gets fit into a coordinate system.

Consensus Coding sequence (CCDS)[20]
The three genome browsers (Ensembl, NCBI, UCSC) still have some differences on various annotations, due to different methods in use when creating the annotations. This means that a gene/annotation may be on slightly different coordinates depending on the genome browser.
In order to make the browsers more consistent, they have started to collaborate into making another standard called Consensus CDS (CCDS). This will make a core genome set after identifying identical regions from the genome builds.

Reference Sequence (RefSeq)[19]
NCBI distributes a set of reference sequences which all three browsers include. These known genes will have the same coordinates in all browsers. Ensembl takes the monthly RefSeq build from NCBI and then adds that to their annotations. RefSeq is based on Genbank and is designed to be less redundant.

The coordinate systems exist so that the scientist will be able to orient themselves in the genome jungle and to extract correct pieces of the sequence. A gene can be located by different means, e.g. a segment of raw data, a primary or secondary accession number, a similarity search, a gene product name or a set of coordinates which comes from the genome builder.

Ensembl also supports other coordinate systems such as 'contigs', 'clones', 'supercontigs', 'scaffolds' and 'chromosomes'.

Clone coordinates
Clones are used when sequencing and will be analyzed into contigs afterwards. A clone can be complete, meaning it is one big contig, while incomplete means that the clone have several contigs. Incomplete also means that there are holes in this clone that have not been identified.

Contigs
Contigs are complete clones that are contiguous on the genome. These sequences are the fundamental building blocks in reproducing the genome structure. In genome browsers each contig have been labeled and represents a region in the genome.

Supercontigs / Scaffolds
This is a structure containing several contigs that have been sorted and oriented to make sure the correct contigs will be assembled together.

Chromosomal coordinates
The genome/chromosomal coordinates are of the highest interest when using raw DNA sequence to create annotations, because coordinates is the primary method used when navigating in the genome. If a scientist wants to compare annotations against known genes then both accession numbers and coordinates will be used. This way the scientist will be able to extract the gene exactly and then comparing that to the sequence dependent annotation.

In some cases ‘genome coordinates’ have the same functionality as ‘chromosomal coordinates’, and the rest of this document will be using ‘chromosomal coordinates’. The coordinates resets to ‘1’ on each chromosome. So for an example, chromosome 11 which is 134 452 384bp long will have coordinates from 1 to 134 452 384.

4 Melting

A DNA sequence consists of a thread of nucleic acid bases and the composition of the bases forms the function of that sequence. When the sequence containing a gene begins the translation process, the double helix opens and forms a bubble so that a polymerase can bind. The opening and closing of the DNA is essential for the replication of the DNA.

In vitro thermodynamic analyses of the DNA show that the DNA conforms into different open and closed areas when heat is applied. The binding energy of the DNA sequence depends on the composition of the DNA. Bubbles mostly form in weakly bound regions. At different temperatures the DNA opens and closes and these conformations relate to the coding and non-coding segments of the DNA sequence [10]. These areas can be identified with several algorithms that takes sequence and temperature into account.

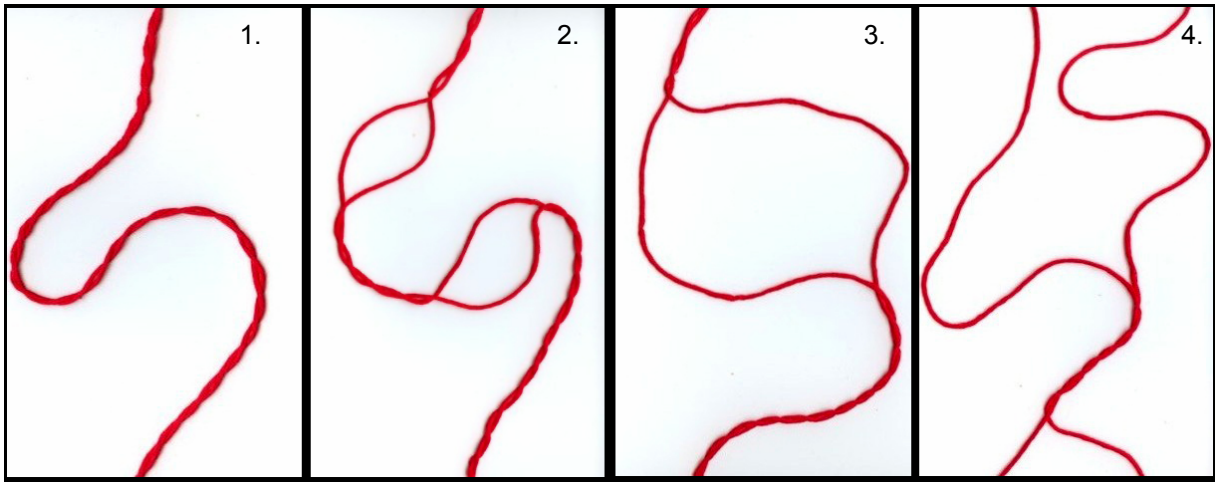


Figure 4 – Example of DNA melting when increasing heat is applied.

The left window (frame 1) of Figure 4 shows the DNA before heat is applied. In the following windows increasing heat is applied. In the upper right window (frame 2) the DNA has started to melt and we see some stable areas between the bubbles of the DNA. In frame 3, the two bubbles from frame 2 have collapsed into a single large bubble. This bubble continues to expand as more heat is applied in frame 4 until we have open bubbles at both ends.

Areas of the DNA sequence with high GC-content will have greater stability than other areas because of the extra hydrogen bond of the G:C pair [11]. Coding areas of the DNA are typically more GC-rich than non-coding areas. In eukaryotic genomes there are structural differences of the GC-content between introns and exons in which the GC-rich areas are typically found in the coding parts of the gene.

4.1 Temperature

A DNA sequence will have several different conformations possible at a given temperature, but some conformations have higher probability to occur than others. A “stitch profile”[2] shows all the possible different conformations that may occur in a hierarchical presentation. The “stitch profile”-algorithm can for example find the temperature that corresponds to 50% helicity which is when 50% of the DNA is in the helical (closed) state.

A melting profile is useful for discovering interesting areas in the genome. The stitch profile shows more information than a melting map. While the melting map shows the average probability at the given temperature, it does not show the different conformations that may exist.

[2] Read more about stitch profiles in the “Stitch profiles” chapter.

Yeramian [12] describes a way to calculate the physical stability of the DNA structure. The stability translates into areas that are robust to temperature increase and also to different ion and salt concentrations.

When comparing gene annotations to the stability map, we will find structural comparable properties. In other words, we find that coding areas are the most stable ones. Note that the physical analysis of the DNA will only find stable/unstable areas that correspond to coding/non-coding areas of the DNA. To be able to find genes, we will need to include other gene finding methods in order to find the whole gene if it is divided by introns.

A melting profile shows the average melting probability at a given temperature, but does not show the different conformations that the sequence can have. The stitch profile creates a plot showing possible bubbles and also closed areas and each conformation has a labeled probability.

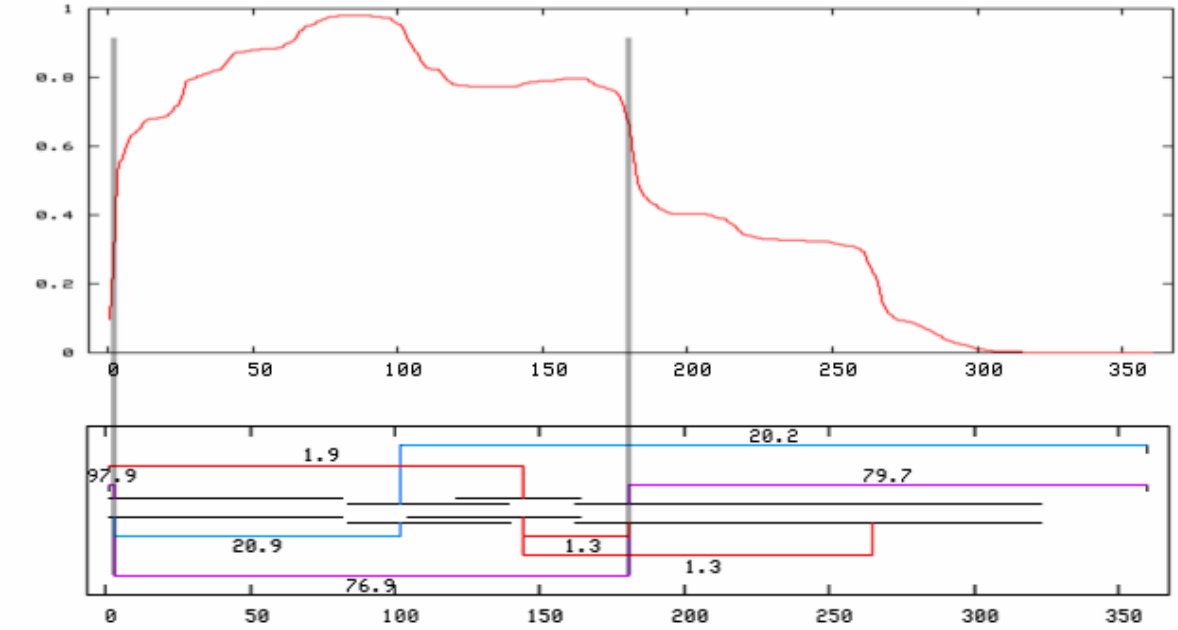


Figure 5 - Comparing Probability profile(top) and stitch profile(bottom). Both are calculated with 50% helicity (50% of the sequence still in helix state).

This comparison shows a connection between the probability profile and the stitch profile.
Regions with increase or decrease in probability in the meting profile correlates to the stitches in the stitch profile.

4.2 Gene discovery

Since the DNA sequence is built from segments that fit together it will also have a structure which corresponds to the way the sequence functions. This structure can be calculated and we will then have created the physical melting map of the sequence. We will compare this map to genomic annotations in order to find the connections between melting and coding/non-coding areas.

5 Stitch profile

The stich profile algorithm created by Eivind Tøstesen [4] constructs a profile which shows the probability of melted and non-melted segments of a sequence at a given temperature or helicity. More details can be found in the articles published by Tøstesen.

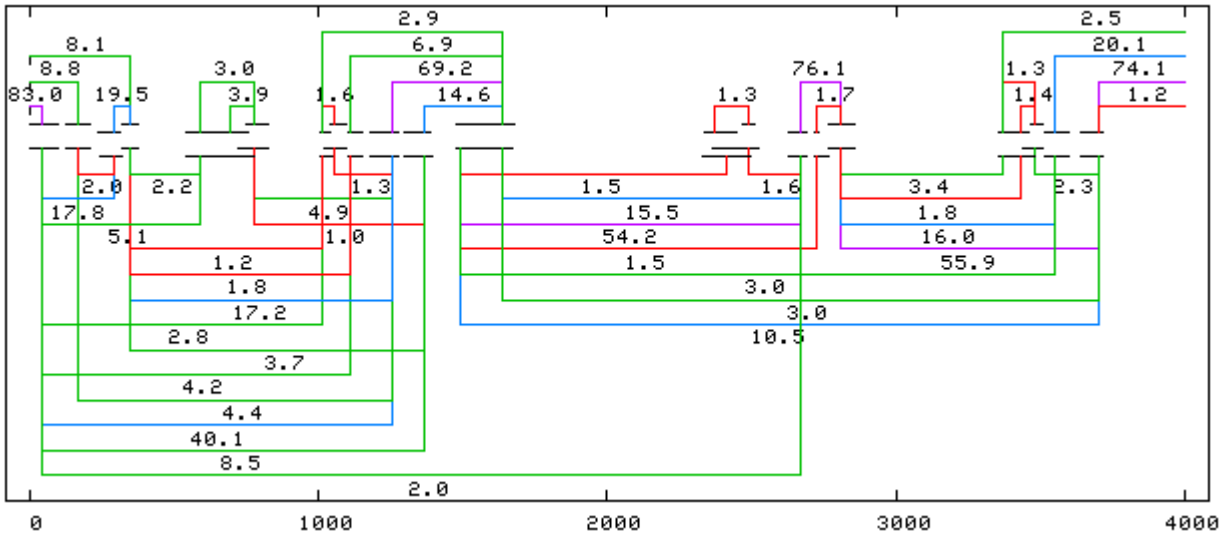


Figure 6- Example of a stitch profile plot of a DNA sequence.

Figure 6 shows the probabilities of opened (melted) and closed (non-melted) areas of a sequence. The profile is horizontally divided in two parts where there are drawn stitches above and below. The open stitch is drawn above, and the closed stitch is drawn below. The description 'stitch profile' comes from the idea that the profile resembles sewn stitches.

5.1 Predicting coding and non-coding regions in a sequence

This hierarchic pattern shows the probability of the different conformations that a melting DNA can have and they are shown as 'stitches' labeled with a probability. The purpose of the algorithm is to calculate a profile that shows the stable/unstable regions in a sequence of interest supplied by a biologist. These stable regions may be coding-regions and thus lead to the discovery of new genes or become supplementary evidence to existing knowledge.

A melting profile shows the probability of melting across the sequence for a given temperature. But this profile only show the average melting for the whole sequence and not the several conformations a DNA can have for a given temperature. A stitch profile, on the other hand, is a hierarchical model that shows the different conformations possible, labeled with their probability.

The coding regions in a sequence seem to be more stable when we test and analyze the physical properties of DNA. Yeramian [6] used the Plasmodium falciparum genome to analyze connections between (non-)coding regions and (un-)stable regions and found close connections between the predicted (un-)stable boundaries and the annotated (non-)coding boundaries.

5.2 The algorithm

The stitch profile uses 3 types of stitches, which are the leading, trailing and complete stitches where the complete stitch is most interesting one. The algorithm is looking at the physical properties between neighbor nucleotides and creates a hierarchical organization of the energy landscape which is the stitch profile. In order to make a good stitch it is then important to have a lot of sequence information both before and after the stitch since the algorithm calculates the long-range effect of the melting properties.

The algorithm calculates in polynomial time $O(N^2)$ and this makes whole genome calculations not possible within reasonable time. A sequence with 50Kbp takes about 15 minutes, while 100Kbp takes 1 hour on a regular desktop computer. This will lead to restrictions when on-the-fly calculations are being performed such as we did with the online service. It is possible to calculate a sequence and store the results in a file, and then to use this file to draw the profiles. To plot the profile from a premade stitch file is fast enough because it takes perhaps 5-20seconds depending on sequence length.

A newer version of stitch profile has a running time of O(N log N) compared with the one currently implemented that is $O(N^2)$, and the new one is therefore generally faster for long sequences. The new version was available too late to be fully incorporated into the current implementation of the system. This new version makes it possible to create a pre-rendered chromosomal stitch profile much faster. Both versions have linear memory usage.

5.3 Stitchprofiles.uio.no

At *stitchprofiles.uio.no* the web user is presented with an interface to the Stitch profile algorithm. It is here possible to analyze a DNA sequence and get the stitch profile of that sequence. It is this service that we wish to incorporate into Ensembl so that this algorithm can be used more efficiently. We also wish to test the algorithm's validity against other annotations. An article has been published in NAR about this online service [4] that describes how it works and how to use it.

The website is running on perl and CGI and writes Strict xHTML code. The service is simple in function and presentation because the more complex issues are implemented under the interactive API. The implementation is divided into separate modules that acts like classes, but it is not object oriented. It is therefore a functional implementation with encapsulation to avoid namespace pollution. Integrating this system into Ensembl should be seamless because there is no shared namespace between the stitch profile and Ensembl.

The web service can now make several profiles for given a DNA sequence. These are the melting curve and the temperature profile, probability profile and stitch profile. The other functions are present in order to compare the stitch profile to the other plots if this should be of interest. See Figure 5 for a comparison between the two of the plots created by the website.

To analyze a sequence, the user needs to specify a DNA sequence that might be inserted directly, or specify a sequence identifier. The next steps are to choose the type of plot to make and set the parameter values before executing the production of the plot.

The plots available at stitchprofile.uio.no are made using Gnuplot and drawing routines implemented by E. Tøstesen. The stitch profile algorithm makes a flat file with the results from the calculation. The stitch profile plot function then reads this file and makes the plot. An example of a plot is shown in Figure 6

On the website there is a 50 000 bp length limitation on sequences that can be analyzed on the fly. Even a calculation on a sequence of this length will take about 15-20 minutes. In Ensembl this calculation time is not very practical because users would then need to wait 15-20 minutes each time they try to navigate when Ensembl is showing 50000 bp. This means that the stitch profile calculation file must be precomputed and then the user can navigate in Ensembl without having to wait for the calculation of the stitch profile. The website was made to present this new algorithm to the public and to give scientists something concrete to work with. It is easier to see how the algorithm works when it is possible to test it out.

6 Distributed annotation system (DAS)

DAS [13] is an XML standard that can provide simple annotations for a specific gene. It was created and designed to decentralize the annotations while presenting the annotations in one location. The DAS system consists of 3 separate systems; Annotation server, genome server and an annotation viewer.

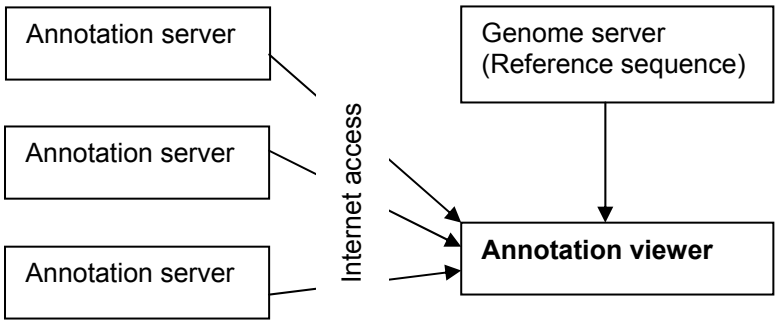


Figure 7 - DAS server setup

Figure 8 shows how the servers are connected and how they work together.

6.1 Annotation server

The idea behind the annotation server is that it can be located anywhere in the world through the internet, and that the annotator is in control of the annotations which is stored on this server. The annotator will be following an XML protocol on how to create DAS compatible annotations. These annotations will then be available to any DAS Viewer that has implemented the DAS specification.

An annotation in DAS is mapped to a genome through position and length. In DAS this is called ‘entry points’. Each chromosome consists of several contigs which is called ‘superlinks’ and inside the ‘superlinks’ we have smaller contigs that is referred to as ‘links’.

6.2 Genome server – Reference sequence

This is the server that contains the genome sequence and will be serving sequences to the annotation viewer. The genomes in the server are mapped up with the ‘superlinks’ which is previously mentioned and is accessible through the use of these links. Any annotation must have one ‘entry_point’ and length to be able to link the feature data to the genome.

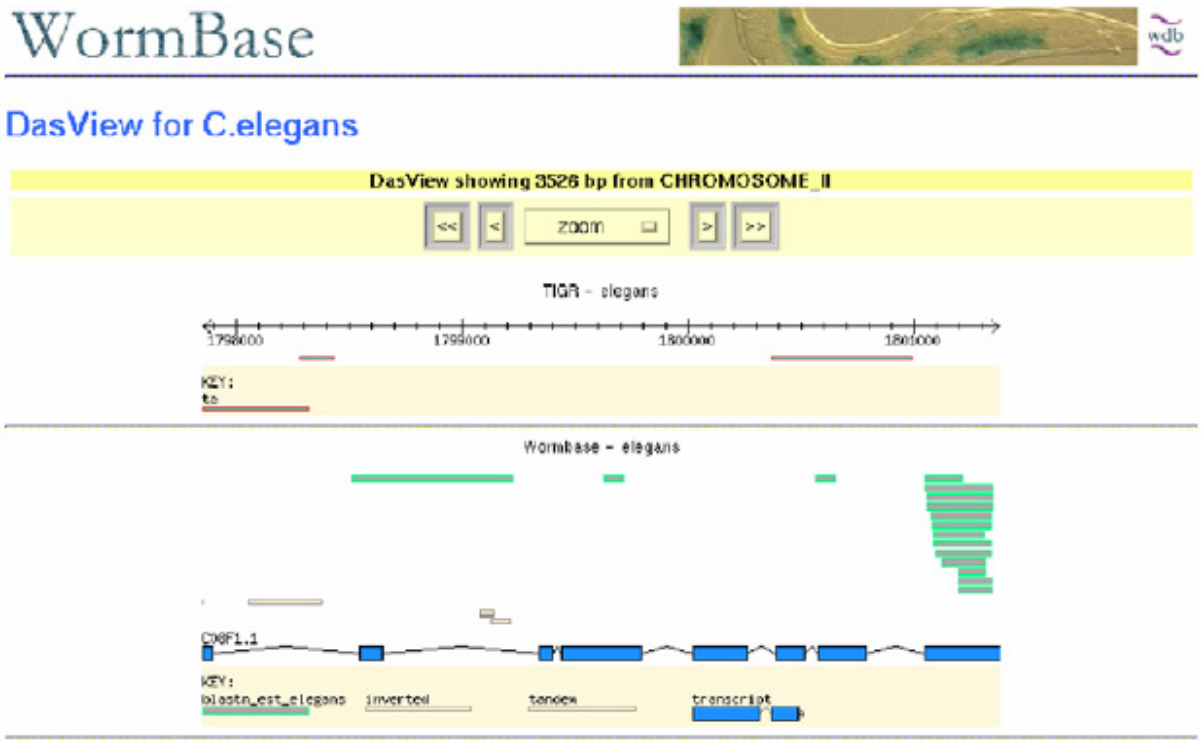


Figure 9 - Example of WormBase DasView

6.3 Annotation viewer

This server is the centralized unit that unites and presents the annotations from the annotation servers to the scientist in a graphical form. There is many ways to represent the annotations, and the Ensembl browser is capable to show annotations from an annotation server. It will show up in Ensembl like a normal annotation, but it will be a custom made annotation which only the annotator can see unless the annotation server is a shared (public) server.

DasView, as seen in Figure 7, is an implementation of an 'Annotation viewer'. The boxes on the bottom of the figure are some annotations to this genome, and represent a feature which belongs to the selected sequence. A box gets created based on primary 3 parameters; entry_point, length and genome ID.

6.4 Extensible Markup Language (XML) standard

The annotations are to be created according to the DAS XML specification. This makes the annotations more human readable and easier for any application to parse. The content of the specification is closely connected to the General Feature Format (GFF)[3] standard, so this is a XML version of the GFF standard.

6.5 Ensembl

It is possible to make your own public annotation server and upload the server info to Ensembl in order to add an annotation track and then compare your annotations against Ensembl's build-in annotations. However the capabilities of DAS are too rudimentary to be used to show something like the stitch profile figure.

In Ensembl's Vega version, which is a version that is specially made to store manually curated annotations, the DAS system was suggested as a way of uploading external annotation information. This suggestion was then rejected because DAS was too simple and could not support all the functions as Ensembl needed.

[3] http://www.sanger.ac.uk/Software/formats/GFF/GFF_Spec.shtml

7 Ensembl

7.1 Expanding the ways of annotation

The official release of Ensembl (v33) does not have the ability to display 2D plot information in the stitch profile figure, but there have been created packages/extensions to the Ensembl framework like the Statistical Viewer [1]. This viewer can display 2D plots in a window that is integrated with Ensembl, but requires a local installation of Ensembl. The glyphs (drawing constructs) used to create this viewer were already implemented in Ensembl since they are internal implementations, but they can not be used with the regular gene annotations.

It is possible to use Ensembl's drawing routines with the glyph system to make custom made plots, but the implementation of these is difficult.

7.2 The Ensembl Web Site: Mechanics of a Genome Browser

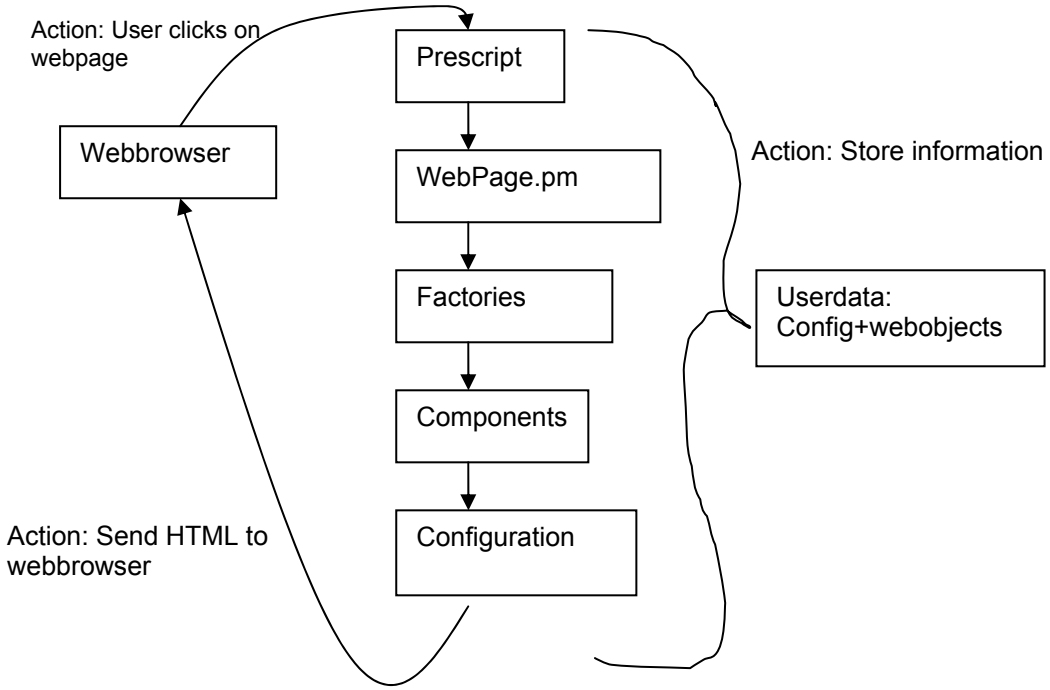


Figure 10 – Ensembl's production pathway [3]

Ensembl is an open-source program released under an Apache-style license [15]. Ensembl is based on Perl, MySQL and Apache together with JavaScripts. The core of Ensembl is based on the BioPerl library, but Ensembl have grown much bigger than this one. They have also rewritten much of the original material so there is no longer any connection to the BioPerl libraries.

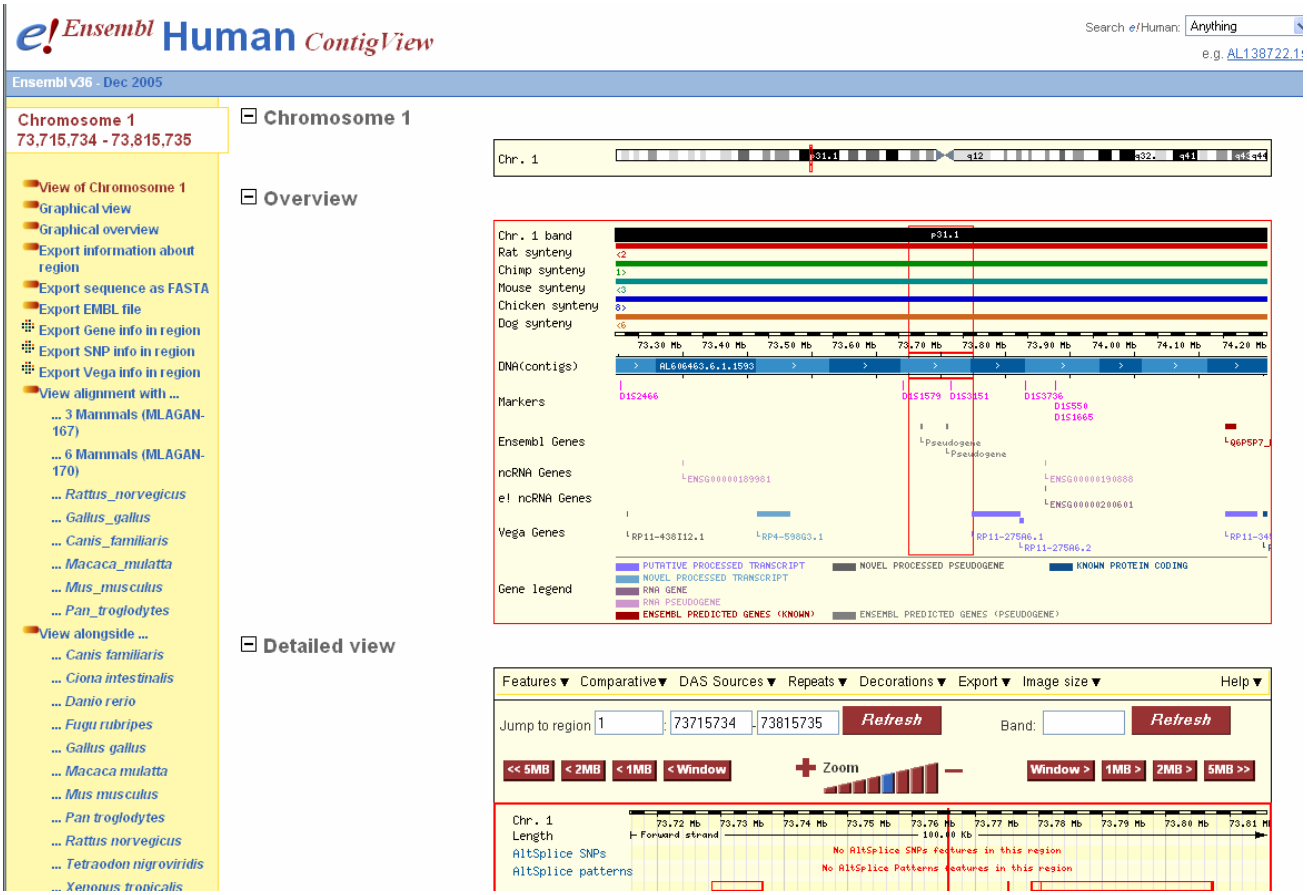


Figure 11 - Ensembl's ContigView

7.2.1 Architecture

As seen in Figure 10, a user clicks on interactive material on an Ensembl webpage and then Ensembl computes the desired information and sends that back to the users web browser. To create a webpage such as the ContigView in Figure 11, Ensembl's production line goes through many steps before emerging on the other side with a webpage. When the web user begins to browse, the first script to activate is an Apache preflight configuration script, then control passes to another script that sends web user request to a centralized web manager called in a module called Webpage.pm. This module is context sensitive and is designed to be flexible. Based on request, the Webpage.pm module performs the necessary actions to create the desired webpage.

Component:
- is a hash structure
- with default values
- have component functions

+

Configuration:
- initializes and updates component based on request
- is context sensitive

Figure 12 – Webpage content

A webpage consists of Components which gets configured by Configuration scripts as shown in Figure 12. Webpage.pm creates the correct Components based on the web user's action, and calls for Configuration scripts to make the Components work. Every piece of information in Ensembl is stored in hashes, and namespaces are protected by the object oriented design in Perl. But as long as everything is stored in hashes, then the program still has access to everything. With these hashes, the programmer only need to access the hash and extract whatever is needed regardless of namespace.

7.2.2 WebPage.pm

This package/class takes care of creating most of the webpages on Ensembl. Each webpage object contains links and hashes that relate to that object. The implementation minimizes the amount of code by letting the same code do several different things, but in a context sensitive manner. This leads to flexible code, but it may be hard to maintain and newcomers may face a steep learning curve.

7.2.3 Factories

Ensembl implement methods to allow add-ons to be added to Ensembl without changing any of the original code, and Factories are made to be generic production houses for any type of data.

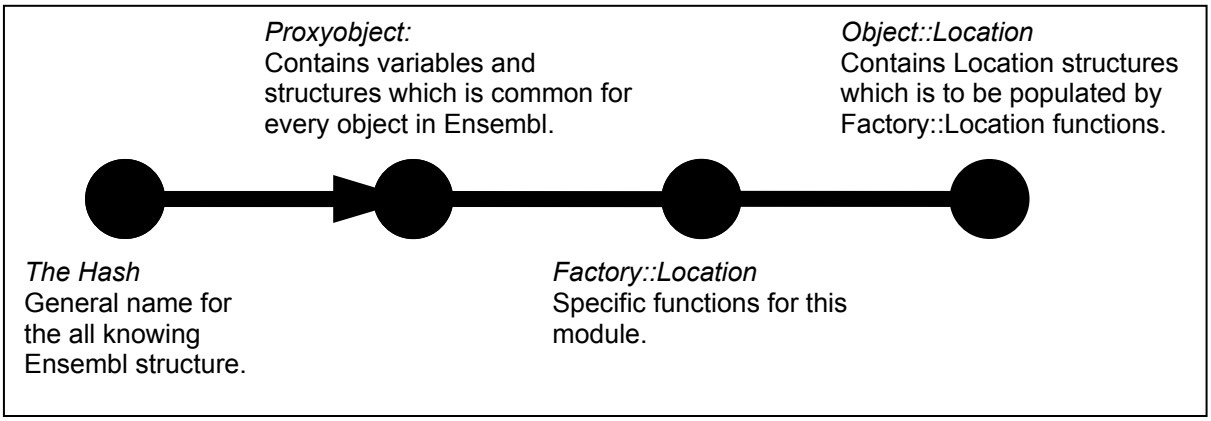


Figure 13 – Example of the Location module information structure

The Proxy, Factory and Object implementation technique makes it easier to add functionality to Ensembl because the programmer only has to take care of his/her own code. This is because Ensembl's own necessary structure and functions are either added with this technique or inherited from parent classes/modules.

Ensembl's web code is formed into a hierarchic inheritance structure, where every module inherits from EnsEMBL::Web::Root. There are also several hashes which are common for many of the trees. This can be described as the Static variables and functions in a Java application.

7.2.4 Components

Components consist of functions which can be used when creating views such as Contigview. These functions print out certain features, like Overview, Detail view and Basepair view. Components are instantiated and stored in Configuration objects(hashes). Together with the global information hash these components contains the necessary information depending on the type of component. An example of a component is the "Component::Location" (hereby called CP:Location) component, and this module type will be used throughout this chapter as a reference.

CP::Location contains functions such as `contigviewbottom`() which prints out the detailed view:

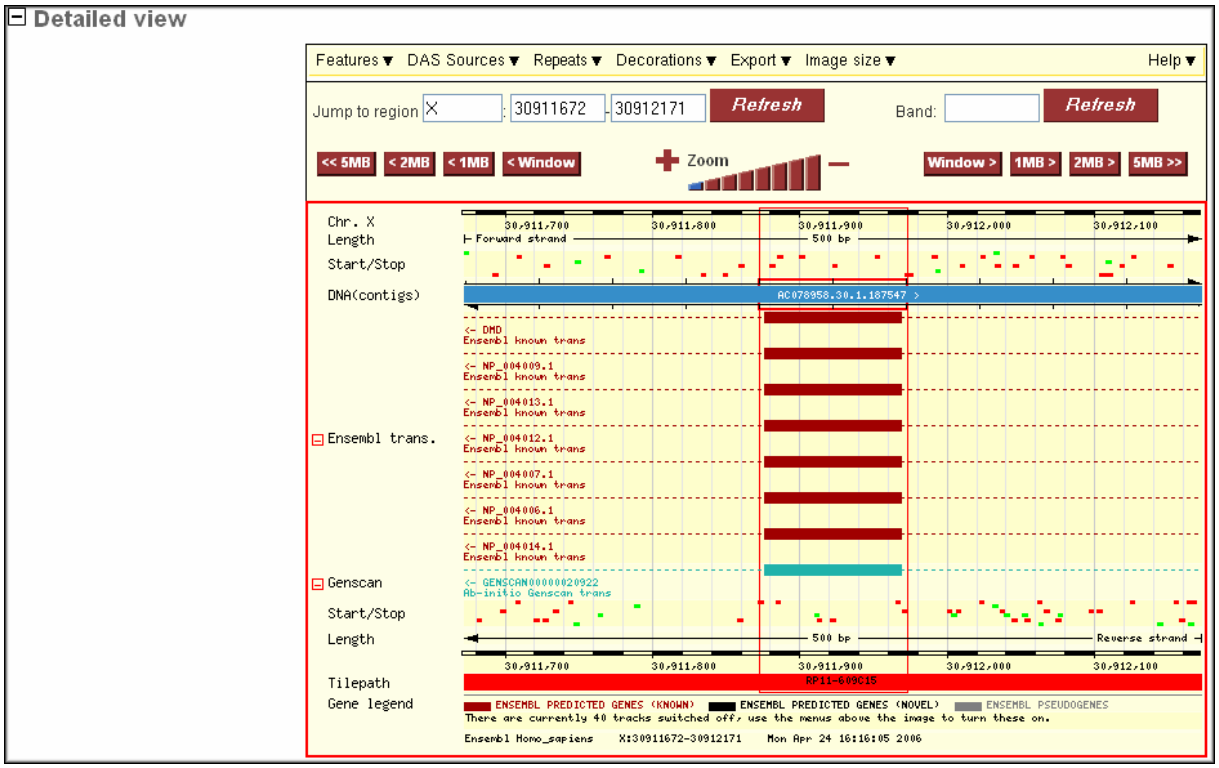


Figure 14 - Example of Detailed View

The function has connections to a panel which is a compilation of different components, and this panel is connected to Apache's print function which inserts the information compiled from the panel.

`contigviewbottom`() also configures the view with the correct Slice (genomeselection), height, width and other parameters.

7.2.5 Configuration

Each component has a corresponding configuration module. The Location module also has a configuration module called Configuration::Location (CF:Location) which compiles functions from CP::Location into a panel where they will be executed.

```
$bottom->add_components(qw(
  menu   EnsEMBL::Web::Component::Location::cytoview_menu
  nav    EnsEMBL::Web::Component::Location::cytoview_nav
  image  EnsEMBL::Web::Component::Location::cytoview
));
```

Figure 15 - Code snippet from CF::Location

In Figure 15, `$bottom` is a panel into which some of the functions from the CP::Location module are added. When Ensembl is printing out the results to the browser, these functions will be executed and the results will be printed into the correct panel.

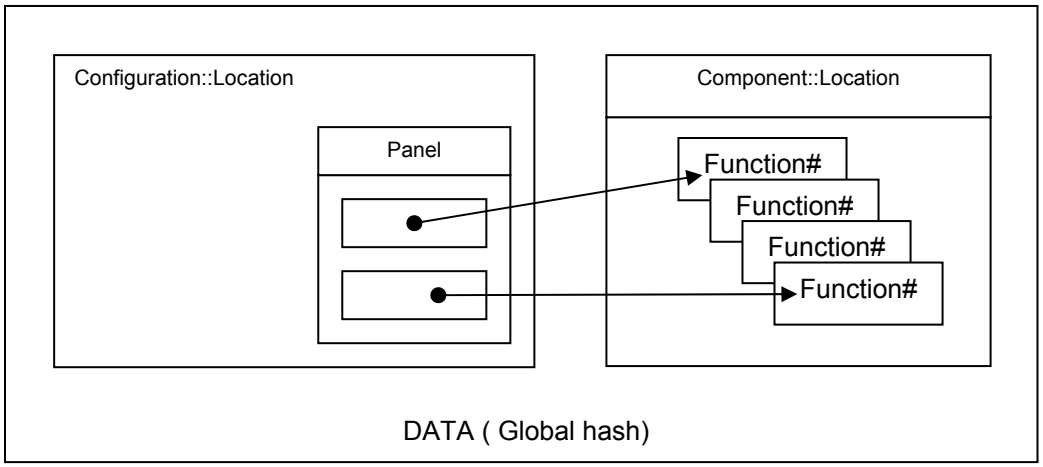


Figure 16 – Adding components to a panel

7.2.6 Shaping information in iterations

Ensembl adds information to the objects in several steps in such a way that it can be described as shaping information into what is requested. The process of making a webpage is like adding information in iterations.

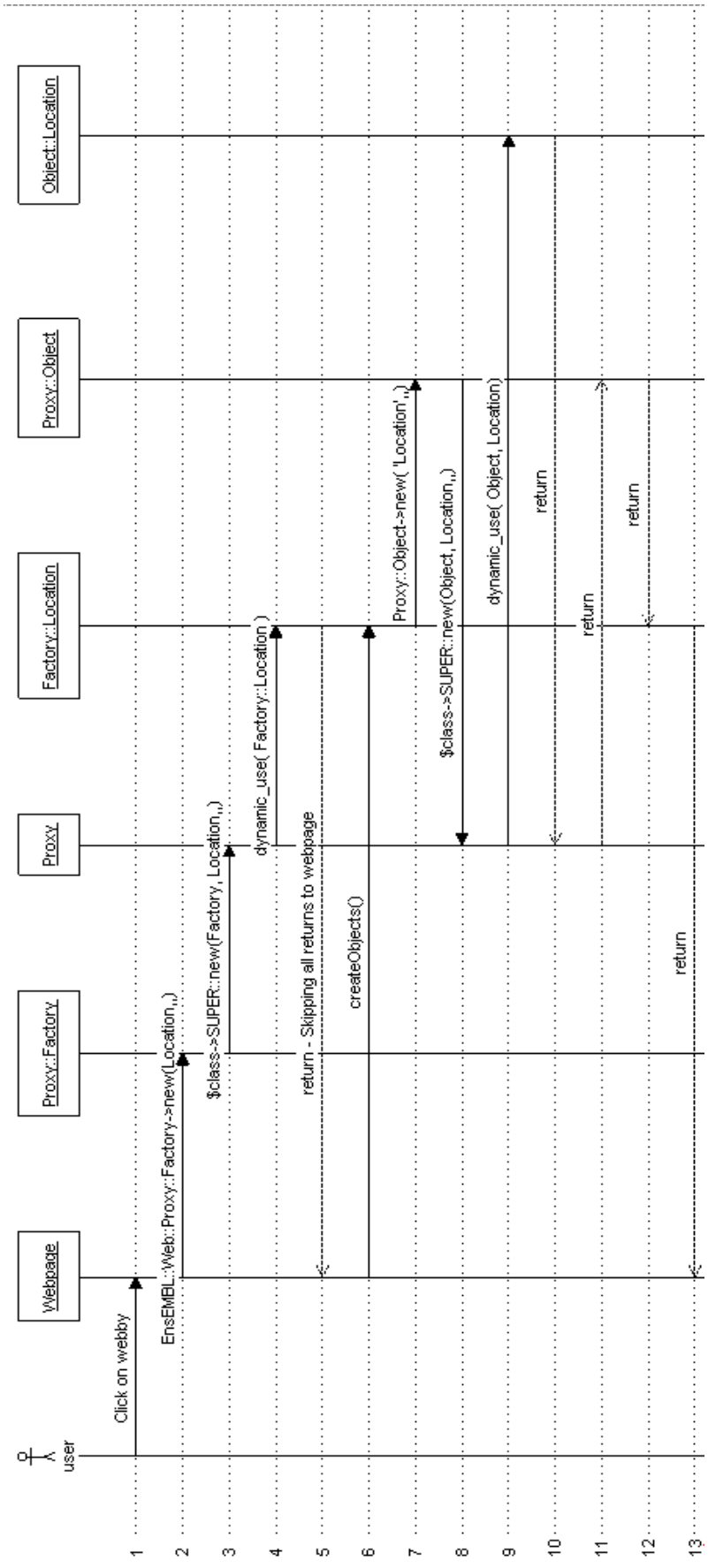


Figure 17 - Create webobjects

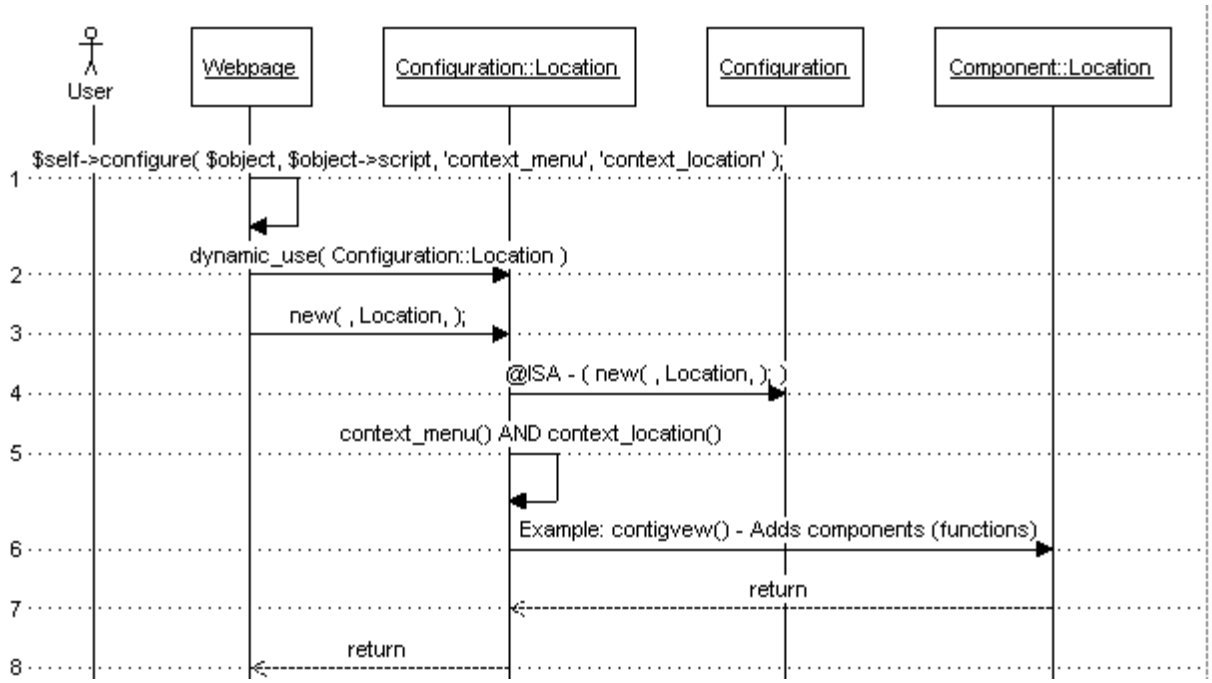
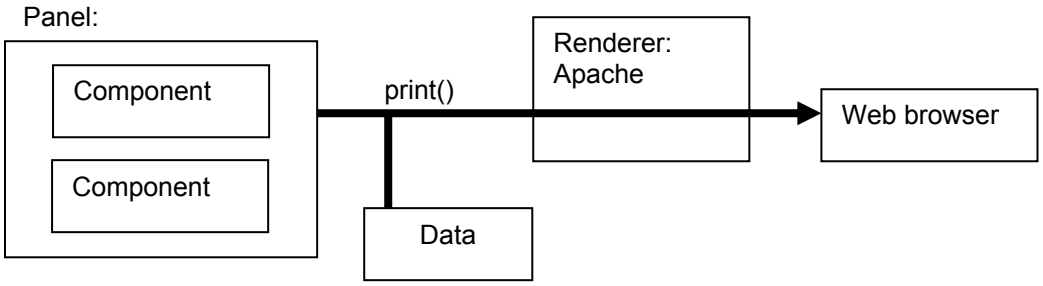


Figure 18 - Configure components

7.2.7 Adding panels

A panel consists of a pointer to a renderer that will print the results of running Component functions. The renderer may be Apache, the web server.



7.3 The Ensembl Core Software Libraries

Ensembl stores its sequences with stable identifiers in order to be handle new releases of the different genomes. These IDs are based on the contig coordinate system since chromosomal coordinate system changes with each new assembly of a genome.

7.3.1 Ensembl graphical library

Ensembl uses a graphical library called “GD Graphics Library”[4] which is an open source graphics library that is able to do most forms of drawing. Stitch profiles uses Gnuplot[5] which is also very capable of making plots, but is more of a step-by-step drawing machine, while GD supplies API wrappers for Perl, PHP and other languages.

GD is generally used in PHP when creating dynamic images on a webpage.

Ensembl also have its own library of drawing routines, but these are not very well documented. This library implements “glyphs” which are predefined figures/structures which can be added to an image container. These glyphs will then be configured according to their use in the image with for example position, height, width, color and more.

Ensembl stores the information needed to draw images in hashes. When the image has been set up, the system calls the drawing routines to create the image and send it to the web browser that requested it.

[4] http://www.boutell.com/gd/
[5] http://www.gnuplot.info/

7.3.2 Ensembl – Code, implementation principals and structure

Ensembl's main parts are the web code, BioPerl and Apache. The code is only partially documented. Most of the code is object oriented to the degree this is possible with Perl. Ensembl's code is also designed to be flexible and non-redundant. This leads to complex code that contain functions that call a list of other functions, where those functions are added while running. This allows the creation of function calls such as this:

```
while(
@modulearray = dynamic load modules in a directory;
for all $modules in @ modules array do{
            $modules.functioncall();
      }
}
```

This small pseudo code example shows how Ensembl handles add-ons.

8 Statistical viewer

The Statistical viewer [1] is software that is able to insert an extra drawing window into the Ensembl Genome Browser in which custom made plots can be added. The drawing routines uses Ensembl's own routines with the glyph library. Annotations in Ensembl are drawn with 1D (x1,x2) lines where each line represent an annotation. That is why it is not possible to add 2D (x,y) plots to Ensembl in its native form. In order to make 2D plots, we need to add the functionality to Ensembl in a modified local installation. This statistical viewer is an example of such a modification, however this software is designed towards use of linkage data and not the types of graphs used in the stictch profiles..

In an early stage of this thesis work, an approach similar to that of the Statistical viewer was pursued for the integration of stitch profiles into Ensembl. However, it was realised that this approach would require a rewrite of the stitch profile drawing routines and a customization of the entire code. This approach was therefore abandoned.

9 Ensembl - Integrating an annotation window

In this chapter different ways of integrating a 2D plot from the stitch profile algorithm into the Ensembl framework will be discussed.

9.1 By using DAS

Limitations of DAS

Inititially, DAS was considered as a possible approach to make 2D plots in Ensembl. However, after realising its limitations, both in the creation of annotations and to Ensembl, DAS was dropped. Since Ensembl does not support 2D plots, DAS could not be used to upload the stitch profile annotations.

Even if we make a 2D plot drawing routine inside Ensembl, DAS would be lacking syntax to make the necessary annotations because they are two different types of annotations. A typical 2D plot needs point coordinates and DAS can not serve this in a good manner.

We considered using DAS by making lots of 1D lines through the DAS protocol representing the stitch profile, but this would not work because the layout on the Ensembl webpage would make the profile incomprehensible.

9.2 HTML frames

Using HTML frames it could be possible to have Ensembl in one frame and the stitch profile in the another frame.

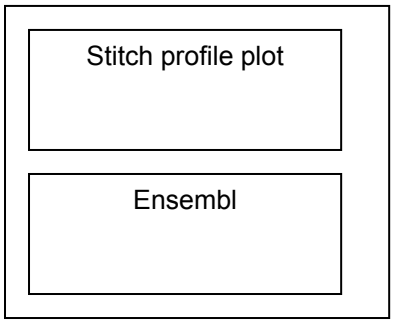


Figure 19 – HTML Frames

Figure 19 illustrates the concept of this approach where we include the functional website of the stitch profiles[6] in the top frame and the official Ensembl in the bottom frame. When navigating in the stitch profile frame, a script could be called to get the Ensembl's webpage for the selected region. This will allow for vertical visual comparison.

The problem is that this solution does not look good, and might not be as functional as we would hope. Navigating through frames and then via some script seems also a little hard way of doing it, and it would not appeal to any user. It will also be difficult to compare annotations from Ensembl's frame against the stitch profile plot because of the frames and layout.

9.3 A separate window

Another approach would be to implement one more annotation window inside Ensembl using Ensembl implementation principles and use this window for the 2D stitch profile plot. This approach has been pursued in such a way that a piece of the Ensembl code has been copied, altered and added to the ContigView. The new window will work as a container with a configuration to fit into the rest of Ensembl. This approach requires a locally installed Ensembl. This solution is described in more detail in chapter 10.

[6] Stitchprofiles.uio.no

9.4 Using Ensembl's drawing routines to create the annotation

This approach uses a window that can be added to Ensembl and draws the 2D plots with the build-in graphical library that comes with Ensembl.

A problem with this approach is that this form of drawing is rather difficult to understand and use for those unfamiliar with glyphs. Therefore, converting the drawing routines already made for Gnuplot in the stitch profile system into Ensembl's drawing routines will probably take too long time.

9.5 By manipulation of the Ensembl HTML code

This approach does not require the installation of a local Ensembl version or alterations to its source code. The idea is to set up a proxy http server that manipulates the Ensembl serverside made HTML code by a script that inserts the relevant stitch profile plot into the HTML code. This way we can use Ensembl's online server together with the navigation provided in the system. A problem with this approach is that installing the proxy and the script seems difficult and it also requires special software such as Squid[7].

[7] http://www.squid-cache.org/

10 Implementing

10.1 Design

After evaluating several solutions as described in the previous chapter, the one integrating the stitch profile through a container in Ensembl was chosen. With this solution, there is no need to implement new drawing routines based on the Ensembl's glyph library in order to create a stitch profile because the profile image is made by the original stitch profile code.

This solution also allows the stitch profile to be navigated through Ensembl's navigation functions enabling the user of the system to use the normal ways of browsing the genome. When adding a container to Ensembl, the same calling functions used on other containers are also sent to the stitch profile container. This way, the stitch profile code gets access to Ensembl's code seamlessly.

10.1.1 The print function

Ensembl uses Apaches native print function to stream data to the web user's browser, and this is used directly in the stitch profile container to print out the html code. So Ensembl creates all the usual frames, containers and annotations, and also the stitch profile container which contains its own custom print function.

10.1.2 Object oriented

Ensembl is more or less object oriented at the high levels. The stitch profile is also an object that is created through the container. Even though the stitch profile is more module oriented, the object layer is operating like an intermediate between Ensembl and the stitch profile. The only thing that Ensembl gets from the stitch profile is the web code in order to view the profile, and stitch profile only gets the information necessary to create the profile. This way the intermediate layer has access to everything in both ends.

With this design, the implementation is very flexible and it is easy to incorporate other windows/containers in a similar way, such as the melting map container which is described later in this chapter.

10.1.3 Why not use the add-on architecture in Ensembl?

Ensembl has its own way of adding customized functionality, but this is rather cumbersome when integrating a container into Ensembl's own hard coded views. On the other hand, this architecture made it possible to integrate stitch profile like it is described here because of the way Ensembl creates the containers.

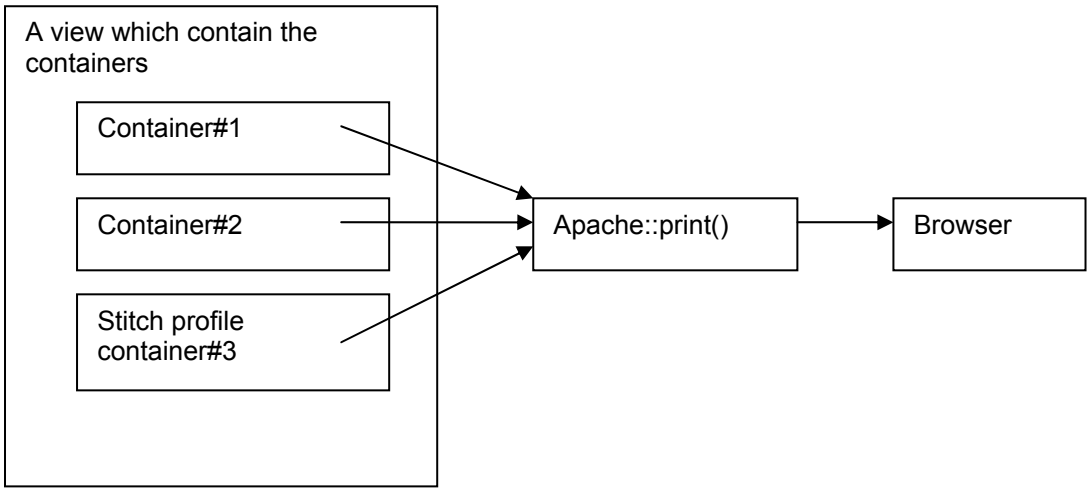


Figure 20 - Ensembl container architecture

Figure 20 describes how Ensembl adds containers to a view and executes them. Containers are functions and not objects. The containers are added to a list which allows an arbitrary number of containers to be executed. The containers are then executed from a function that goes through this list and calls them with the same parameters.

This way of adding functions to an indirect execution is a normal implementation technique used in all parts of Ensembl. With this architecture, Ensembl makes it possible to add any functionality without the need of cascading implementation when adding extra functionality to a program, because only the function to be executed is written, and then this function is added to the list to be executed. Ensembl then takes care of the rest.

10.2 Processing the stitch profile

10.2.1 Ensembl side integration

Without considering the massive amount of function calls Ensembl performs to configure and add views, the stitch profile integration is implemented into 2 packages.

File#1 - EnsEMBL::Web::Configuration::Location
This package contains the list of functions to be executed in the ContigView. Configuration packages describe the content which belongs to a certain view. This content consists of Components. Stitch profile is added as a container component to this particular view.

```
sub contigview {
...
    $bottom->add_components(qw(
      menu  EnsEMBL::Web::Component::Location::contigviewbottom_menu
      nav   EnsEMBL::Web::Component::Location::contigviewbottom_nav
      image EnsEMBL::Web::Component::Location::contigviewbottom
      comp  EnsEMBL::Web::Component::Location::contigview_comparison
    ));
    $self->add_panel( $bottom );
  }
...
}
```

Figure 21 - Adding stitch profile container (comp) to the list of executables

The line "`comp EnsEMBL::Web::Component::Location::contigveiw_comparison`" in Figure 21 adds the stitch profile container to the list of containers to be executed. When a container gets executed, it is given the Apace handler which can be used to print out text to the browser. When Ensembl is executed, this handler streams html code to the browser as it gets produced.

File#2 - EnsEMBL::Web::Component::Location
This package contains the components that can be used in the relative configuration package, and in this package the stitch profile container is implemented. The container consists of the stitch profile object which creates the profile depending on what the web user is currently browsing on. The object returns the profile image and produces the html code in order to present it to the web user.

```
sub contigview_comparison {
  my($panel, $object) = @_;
  my $slice = $object->database('core')->get_SliceAdaptor()->fetch_by_region(
                          $object->seq_region_type, $object->seq_region_name,
                          $object->seq_region_start, $object->seq_region_end, 1);
  ...
  # Adding annotation object:
  my $ens_img_obj = EnsemblAnnotationImage::new($object->real_species);

  # Make Stitch profile
  # USE: make_stitchprofile_annotation( start, end, chromosome, ensembl_hash);
  my $stitchprofile_png = $ens_img_obj->make_stitchprofile_annotation(
                                        $object->seq_region_start,
                                        $object->seq_region_end,
                                        $object->seq_region_name,
                                        $slice);

  $panel->print( " <div style=\"text-align:center;padding-left:10em\"><img src=\"$stitch
profile_png \" alt=\"Stitch profile\" title=\"\" style=\"width: 608px; height: 304\" /><
/div> </br>" );

 # Make meltingmap annotation:
  if ($object->real_species eq "Homo_sapiens"){
    my $meltingmap_plot = $ens_img_obj->make_meltingmap_annotation($object->seq_region_n
ame, $object->seq_region_start,$object->seq_region_end);
    $panel->print( " <div style=\"text-align:center;padding-left:7em\"><img src=\"$melti
ngmap_plot \" alt=\"Melting map\" title=\"\" style=\"width: 640px; height: 350\" /></div
> </br>" );
  }
  return 0;
}
```

Figure 22 - The creation of the stitch profile within the container

The Figure 22 describes the stitch profile container that gets executed when Ensembl produces ContigView. Here we use Ensembl internals to make the stitch profile directly by sending the '`$slice`' hash into the intermediate code. This contains the information of the sequence the web user is currently browsing on.

10.2.2 Stitch profile side integration

The intermediate package which handles the communication between stitch profile and Ensembl is called: "EnsemblAnnotationImage.pm", and activates the stitch profile program which produces the profile image.

10.2.3 On the fly computations and precalculated stitch profiles

Currently the web user can use Ensembl to compare annotations with the stitch profile, but the profile is calculated on the fly. This means that for sequences of length 20Kbp and more, the web user will have to wait a very long time. But for the *Saccharomyces cerevisiae* genome, w we haveprecalculated the entire chromosomes VII and VIII. This means that the user does not have to wait until the calculation of the profile, but merely for the image to be made. The stitch profile calculation details are stored in a stitch profile file containing drawing routines that draw the profile image which again is presented to the user.

Precomputation have also been carried out for the dystrophin gene on human chromosome X between positions 30891993 and 33122215.

10.3 Adding the melting map annotation

An manuscript in preparation [16] describes a melting map for the whole human genome. This map has been integrated into this Ensembl installation and is browsable together with all other annotations. After having discovering the method for integrating a new container/window, it is almost trivial to add features. The screenshot below shows an example of the integration of both the stitch profile and the melting map.

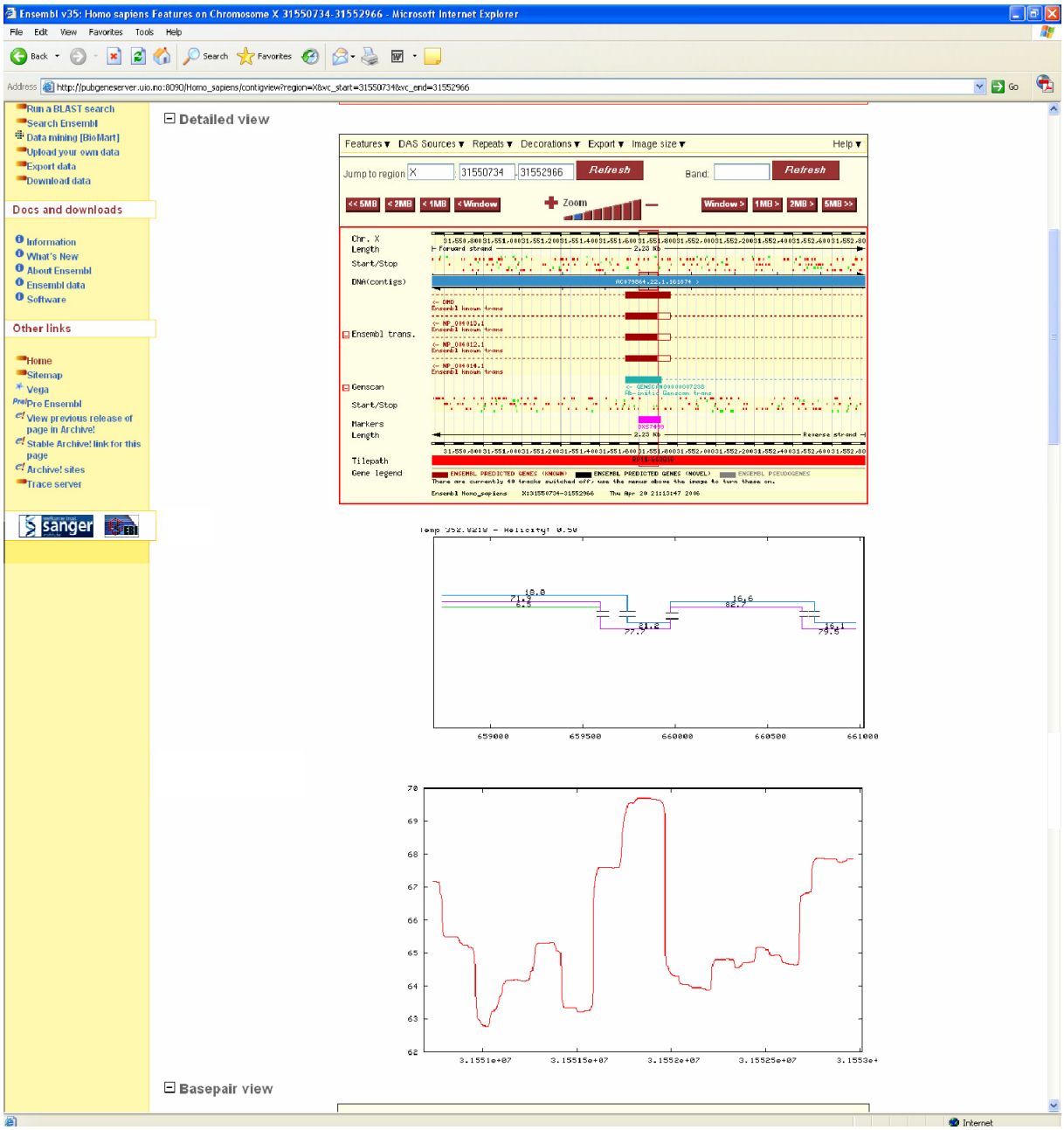


Figure 23 - Screenshot of the stitch profile and the melting map integrated with Ensembl

In this screenshot we can also see a connection between the stitch profile and the melting map around the position 31551850 that is located in the middle. The high peak in the melting map correspond to the closed stitches, which again corresponds to the annotated coding region in Ensembl.

10.4 Limitations

10.4.1 Server stability

After a few uses of the locally installed Ensembl, the server will eventually halt when it comes to creating the stitch profile. This is caused by the implementation of the stitch profile algorithm and an unknown factor from the server. Apparently, the Perl process which created the stitch profile is not terminated when Ensembl has finished processing the web page.

Occasionally the process gets terminated, and if the web user continues to browse, the user will get another correctly produced web page such as seen in Figure 23. When it halts, meaning when Ensembl waits for the stitch profile to be produced, the previous stitch profile process is still stored within that last browser session and will use that object when creating the next stitch profile.

Since Ensembl/Apache does not terminate the process after producing the web page, this will trigger a bug in the stitch profile program. With the current implementation of the stitch profile, it is not possible create a new plot using the same object because there are some architectural flaws in the stitch profile program relating to how it uses libraries and namespace. The bug was discovered a little late, and that is why the bug has not been corrected. It does not seem to be a trivial bug, because it might also relate to the way Ensembl uses registers to save web user information. A simple refresh on the browser will describe this feature where the browser may have cached up some of the webpage to allow faster browsing. The browser saves some of the information, while Ensembl also seems do so. To perform a proper refresh the user must click on the ‘Refresh’ button on the webpage.

There is a workaround to both of the problems, which is to make the stitch profile be made in a separate process with no ties to the Ensembl with the use of a bash script which makes a proper process each time it is activated.

10.4.2 Sequence length

There are some sequence length restrictions when browsing the stitch profile through Ensembl. Usually 15Kbp is the max length of the profile because the browser will eventually time out while waiting for the process to finish, and also because the profile gets cluttered with the amount of stitches. Using pre-calculated stitch profiles as it has been done with the dystrophin gene, makes it faster to browse and possible to make long stitch profiles. But again, the profile gets crowded with stitches over 15Kbp. This can be solved by adding some extra features to the integration implementation such as the ability to specify parameters to the stitch profile.

11 Comparing stitch profiles with Ensembl's annotations to find biological correlations

Stitch profiles shows the melting profile of a DNA sequence in such a way that it is easier to visually compare it directly to other annotations such as manually curated genome annotations. Ensembl is a visual tool where we can compare annotations at a high level of detail, leaving the analysis mainly to the eye of the beholder. With these results, the scientist can go on to the next step and use more low level details and advanced statistics to get more accurate data.

We wish to find biological relevance to the stitch profile by comparing it to Ensembl's annotations, and while doing this we learn more about accuracy of the stitch profile. The service at stitchprofile.uio.no is also a visual tool, but it does not have an easy way of comparing the results against other annotations. If a scientist wants to compare it to other annotations such as the Ensembl's or other annotation service, then it would be a cumbersome process of image refitting and scaling. Integrating stitch profiles into Ensembl's framework, enables the scientist to make a visual comparison on the fly between the different annotations.

There are two hypotheses which we will investigate:
1) There exists a correlation between a *closed stitch* in a stitch profile and a *coding region* annotated in Ensembl

2) There exists a correlation between an *open stitch* in a stitch profile and a non-*coding region* annotated in Ensembl

As mentioned earlier in this thesis when describing the stitch profile, there have been some earlier publications about these hypotheses where the melting profile has been compared to genes. This chapter will try to establish how accurate the stitch profile predicts these (non-)coding regions of a genome, and also document how and why the stitch profiles fails to do so in certain regions. Mapping melting profiles to gene regions have been done before, but not like this, since the stitch profile presents the melting profile in a different way. It is easier to do a visual comparison with the stitch profile because of the hierarchical ordering of the stitches which have been labeled with probabilities.

Describing the notion of the (non-)coding region
A melting map is based on several parameters such as the empirical thermal models and salt concentration. Then there is the sequence which is to be analyzed, and a melting algorithm uses the molecular energy bounds between base pairs to calculate the temperature which corresponds to the set probability. The algorithm favors DNA sequences which have low GC% content if the goal is to find genes. Generally, genes contain more GC% content than intergenic regions and thus the bonds between the DNA strands are stronger in the genes. A higher general GC-content will make it more difficult to distinguish the two regions because the energy landscape does not contain the same recognizable peaks.

GC-rich regions contains strong molecular bindings, whereas AT-rich regions have less strong molecular bindings. Strong bindings lead to higher melting temperature because it demands more energy to break it.

Stitch profiles maps the (non)-coding to open and closed stitches for a set helicity (or set temperature). The open stitches then represent the weak spots of the sequence, while the closed stitches represents the strong spots of the sequence. Therefore the stitch profile is a map over the non-/coding regions in the analyzed sequence where the coding regions might be a structural strong point in a gene or other coding region outside the gene such as noncoding DNA, provided the hypothesis holds.

11.1 Quality and testing

11.1.1 Investigation of stitch profile patterns by a step-wise raising of the temperature

As an approach to investigate patterns in the stitch profile, we made a set of tests on both *S. cerevisiae* and the *Homo sapiens* genomes. The test was to study how the stitch profile changed as the temperature increased, and this was carried out by making a set of stitch profiles with helicity decreasing from 100% to 20%. The stepsize was ~1% for each stitch profile. The range from 100% to 20% produced 80 stitch profiles for each of the sequences that were analyzed, and from this test an overview figure was made showing the main differences between the profiles. Each sequence has its own overview figure, and with this we try to show how the stitch profile behaves while increasing the temperature. The temperatures in the figures are shown in Kelvin degrees.

11.1.2 Sequence length

We know that the stitch profile algorithm need a lot of padding sequence around the region of interest, but the precise amount has not been tested systematically. Visual inspection indicates that a few thousand basepairs padding is good enough if the region of interest is shorter than 1000 bp. If the coding region is large then we will need to include more sequence on both ends for the analysis in order to get an accurate profile. Currently, we have examined the *S. cerevisiae* chromosomes VII and VIII as well as the dystrophin gene on human chromosome X.

The sequences in the following figures in this chapter have been calculated with 1000-3000bp padded on each end of the region of interest. The surrounding areas to the region of interest also have some structural information which is useful when considering the other genome mechanics which applies to the coding region such as the control region of a gene. Therefore additional padding is added in order to discover these mechanics.

The stitch profile algorithm also creates some artifacts at the edges of the sequence. Since the algorithm uses basepair binding energy to predict the stitches, the calculations seem more accurate when a certain amount of sequence is “in-line-of-sight”. Meaning, the accuracy increases as the sequence is getting processed, and decreasing in accuracy when reaching the end of the sequence. Again, no systematic testing has validated this statement. For now, it is an observation.

All in all, the more sequence the better for the accuracy of the calculation. The sequence melting abilities gets more stable with increased sequence length. With short sequences, padding the sequence changes a lot of the calculation. The point is that the sequence of interest gains a more stable stitch profile when padding enough sequence.

11.2 Biological aspects with stitch profiles

It might be interesting comparing already existing annotations to the stitch profile to find additional genomic attributes. Stitch profiles might be used to uncover introns in genes that were previously not noticed, and this might lead to a more comprehensive understanding of the gene structure. It might be useful when aligning cDNA to a genome since it is possible to score the introns and then only compare those regions of the genome that are transcribed and present in the mature (spliced) RNA.

Since the stitch profile calculates the melting structure based on physical properties, the other genomic regulating proteins inside the nucleus is somewhat disregarded such as surrounding proteins that also helps to regulate transcribing. This makes it even more important to compare the profile to other gene finding algorithms which is basing their calculation on other biological data.

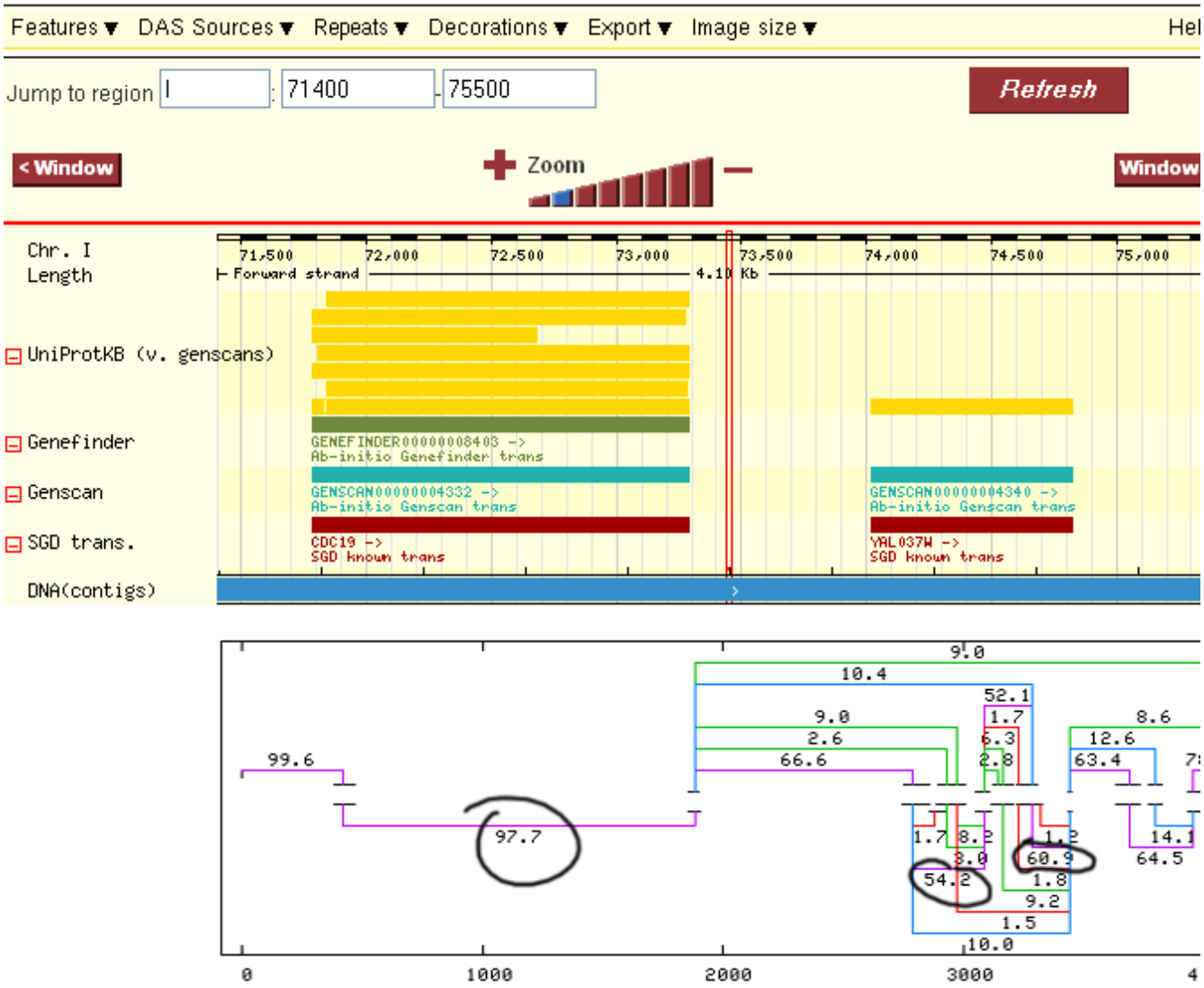


Figure 24 - Comparing Ensembl's annotations with the stitch profile (*S. cerevisiae*)

Figure 24 shows a comparison between several different types of genome annotations against the stitch profile for this sequence produced with 50% helicity. The 'SGD transcript' annotation is the source and builder of the SC genome. When comparing the stitch profile to these annotations, the profile indicate good similarity between the known CDC19 gene and the 97, 7% closed stitch. In the YAL037W gene, the stitch profile shows 2 closed stitches with high probability 54, 2% and 60, 9%.

11.3 A visual comparison between the annotation and the stitch profile

On the following pages, a comparison is made between a stitch profile and Ensembl's annotations for the same sequence. The genome is *S. cerevisiae*, where the region of interest lies between position 204300 and 222489 on chromosome VIII. This analysis will investigate how accurate the stitch profile is against 'known' genes.

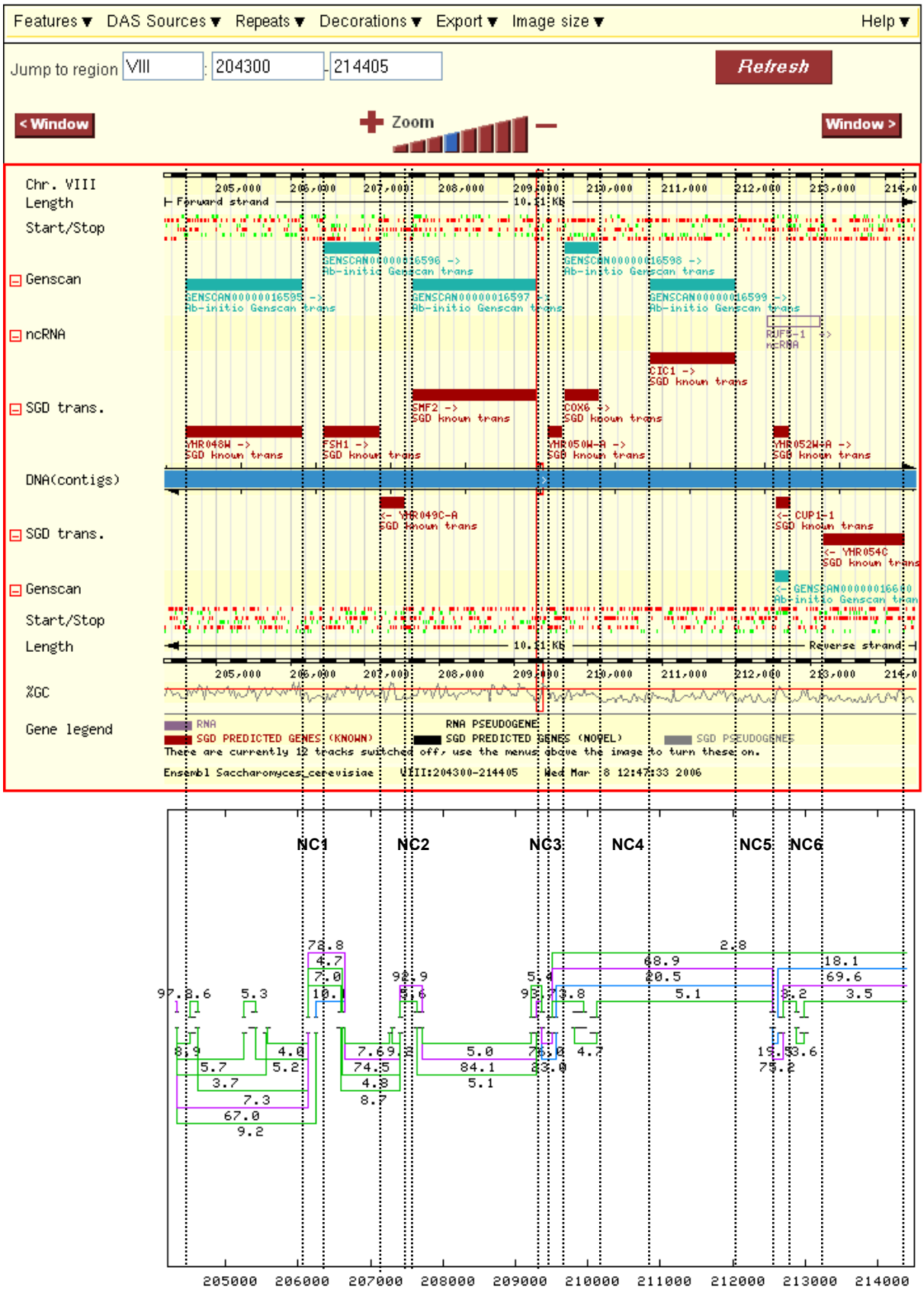


Figure 25 - SC - Chr.8 [204300, 214405] with vertical visual supporting lines

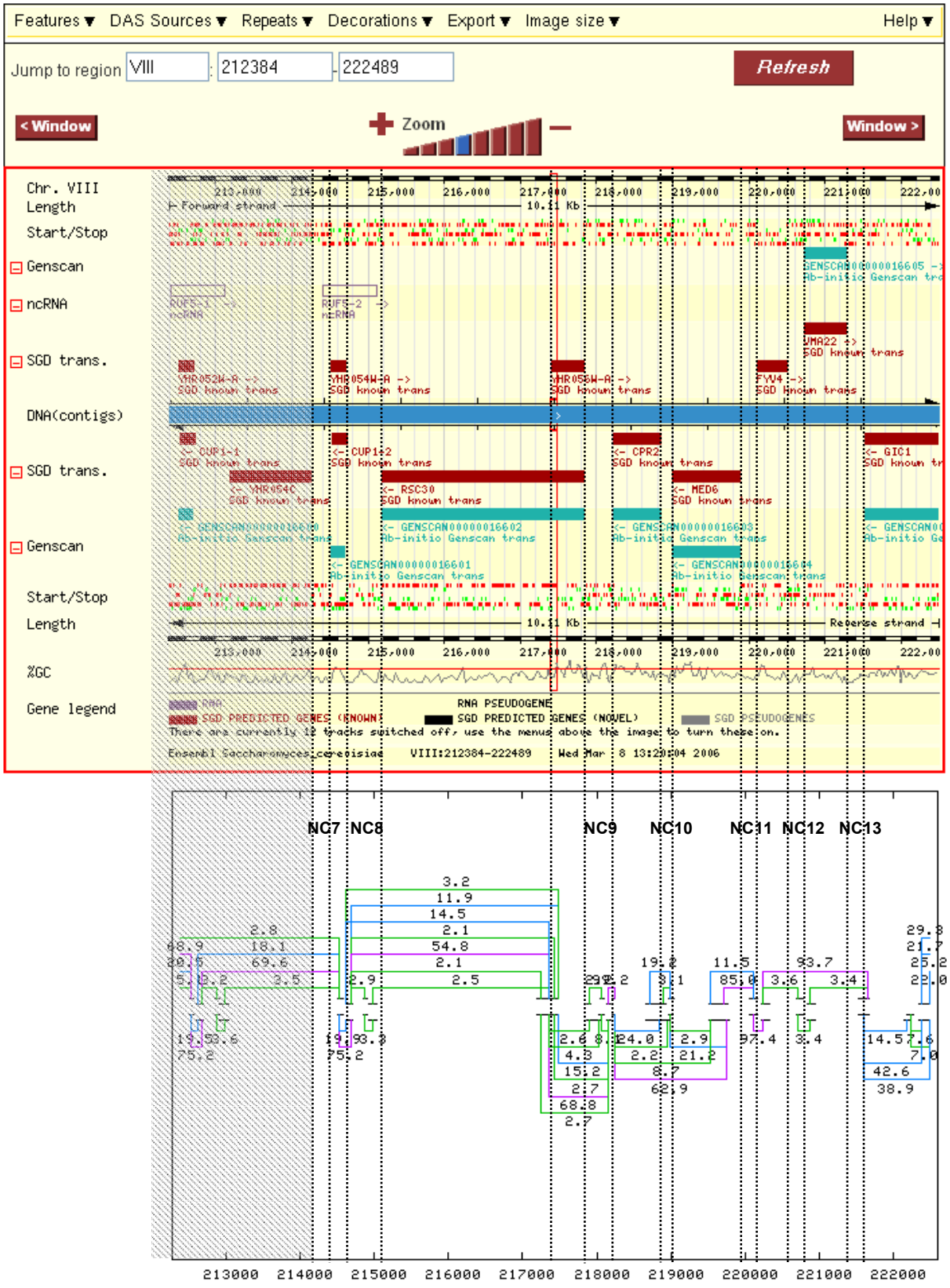


Figure 26 - SC - Chr.8 [212384, 222489] with vertical visual supporting lines

Gene to stitch profile similarity

			Hit closed %			*Hit open %*			
	Gene#	*Strand*	*start*	*middle*	*end*	*start*	*middle*	*end*	*Visual hit*
#1	YHR048W	Forward	67	67	67				Y
#2	*NC1*		67				72,8	72,8	Y
#3	FSH1	Forward		74,5	74,5	72,8			Y
#4	YHR049C-A	Reverse	74,5	74,5				92,9	Y, but it might be because of hit with FSH1
#5	*NC2*					92,9	92,9	92,9	Y
#6	SMF2	Forward		84,1	84,1	92,9			Y
#7	*NC3*				76	93,7	93,7		Y
#8	YHR050W-A	Forward	76				73,8	73,8	N
#9	COX6	Forward				68,9	68,9	68,9	N
#10	*NC4*					68,9	68,9	68,9	Y
#11	CIC1	Forward				68,9	68,9	68,9	N
#12	*NC5*					68,9	68,9	68,9	Y
#13	YHR052W-A	Forward	75,2	75,2				68,9	Y
#14	CUP1-1	Reverse	75,2	75,2				68,9	Y, but it might be because of hit with YHR052W-A
#15	*NC6*			53,6		69,6		69,6	Y
#16	YHR054C	Reverse				69,6	69,6	69,6	N
#17	*NC7*					69,6	69,6	69,6	Y
#18	YHR054W-A	Forward		75,2	75,2	69,6			Y
#19	*CUP1-2*	Reverse		75,2	75,2	69,6			Y, but it might be because of hit with YHR054W-A
#20	*NC8*					54,8	54,8	54,8	Y
#21	RSC30	Reverse				54,8	54,8	54,8	N
#22	YHR056W-A	Forward	68,8	68,8	68,8				Y
#23	*NC9*		68,8	68,8		30	30		Y
#24	CPR2	Reverse	62,9	62,9	62,9			19,2	Y
#25	*NC10*		62,9	62,9	62,9	19,2	19,2	19,2	Y
#26	MED6	Reverse	62,9	62,9				85	Y
#27	*NC11*					85	85	85	Y
#28	FYV4	Forward	97,4				93,7	93,7	Y
#29	*NC12*					93,7	93,7	93,7	Y
#30	VMA22	Forward				93,7	93,7	93,7	N
#31	*NC13*					93,7	93,7	93,7	Y
#32	GIC1	Reverse	42,6	42,6	42,6				Y

Table 1 - Gene to stitch profile similarity

11.3.1 Explanation of Figure 25 and Figure 26

Table 1 analyses the 'SGD known transcripts' annotations to the stitch profile for that same region.The region which is compared was chosen almost randomly. The only requirement of the region was that it needed to contain a moderate density of coding areas.

The stitch profile is calculated with the parameters:

- θ = 50% helicity
- Maximum depth: $D_{max=}3$
- Probability cutoff: $p_c=0.01$
- Empiric thermodynamic parameter set: "Blossey and Carlon 2003"
- Salt concentration $[Na^+] = 0.075$

Chromosome 8 has been calculated as a whole using the stitch profile algorithm, and the profile in the figure is a plot from this calculation. Since the algorithm considers long-range effects, the more sequence data the better the profile. The figures are edited screenshots from the locally installed Ensembl.

Each coding and non-coding region are compared to the stitch profile to see whether they correspond to 'closed' or 'open' regions of the genome. 'Closed' is the so-called stable regions and 'open' is the unstable regions. The vertical dotted lines in the comparison image represent the start and end of the coding and non-coding regions. They are included to make it a little easier to compare positions. The regions called NC<number> means **N**on-**C**oding regions and is only used in the figures to make a unique name for each non-coding region so that it can be compared to the 'open' regions in the stitch profile.

The 'Hit Closed %' table with (start, middle, end) describes the hit percentage on different positions on Gene#. As for the gene 'YHR048W', it hits a stitch with 67% in the start, middle and end position of the gene and therefore it is a good hit for the gene. 'Hit Open %' works in the same way, but only for non-coding areas.

The column 'Visual hit' describes the manual visual annotation comparison where the values are [Yes, No] and Yes-values are given when annotations matches with the stitches. One example is the gene 'YHR048W' which have the value 'Yes' because the gene corresponds to a closed stitch in this region. Such a visual comparison is the key to compare the annotations since it gives a better overview of structure and recognizable patterns than might be discovered by a simpler inspection. Later it might be better to perform a one-to-one base comparison of the annotations and perform statistics on the results. Then it would be possible to find an average hit value for the gene annotation against the corresponding stitch profile for that region.

When doing visual comparisons, it is easy to see which regions hits perfectly and which regions misses totally. With this in mind, the next step will be to try find out why the algorithm misses on coding regions.

11.3.2 Finding (non)-coding regions

By looking at Figure 25 and Figure 26, it is apparent that the coding region does not always correspond correctly to the closed regions of the stitch profile. By doing visual comparison it is possible to distinguish the almost perfect matches versus the not-so-perfect matches. For instance, the gene 'COX6' does not have a corresponding closed region at all. Further investigation of 'COX6' might reveal the reasons why the stitch profile algorithm does not recognize this particular coding region.

The stitch profile seems a little fuzzy with its predictions, and this must be taken into account in the comparisons. For instance, in "Table 1 - Gene to stitch profile ", the coding regions usually have two hits in a closed region, while those regions that have none or only one hit in a closed region are usually not true coding regions.

11.3.3 Stitch profile to gene annotations comparison

Only stitches with more than 50% are considered.

	Stitch value:		
Stitch#	*Non-coding*	*Coding*	*Annotation confirmed (>50%)*
1		67	X
2	72,8		X
3		74,8	X
4	92,9		X
5		84,1	X
6	93,7		X
7		76	X
8	68,9		
9		75,2	X
10	69,6		
11		75,2	X
12	54,8		
13		68,8	X
14	50		X
15		62,9	X
16	85		X
17		92,4	X
18	93,7		

	NonCoding	Coding
TRUE	5	9
FALSE	4	0

True total:	14
False total:	4

Table 2 - Stitch profile to gene annotations comparison

The non-coding regions in Table 1 are completely correct. Every non-coding regions in the Ensembl gene annotation figure matches open regions in the corresponding stitch profile. But in “Table 2 - Stitch profile to gene annotations comparison” a reverse comparison is being done, and there it is found 4 false positive hits from open regions versus Ensembl’s gene annotations. This means that there exists larger amount of open regions in the stitch profile than non-coding regions in the Ensembl, making it easier for the Ensembl to hit open regions in the stitch profiles.

11.4 Analyzing stitch profile with decreasing temperature

In this analysis, a stitch profile was created for each temperature between 20% and 80% helicity with a 1% increase creating a total of 61 profiles. These profiles were then sorted on increasing helicity and put together in an animation[8].In this animation, we observed that the position of each stitch basically remained the same, but the predicted percentage of each stitch changed as temperature decreased and several open and closed stitches were added or dissolved.

With analyzing the behavior in the stitch profile in this one-frame-pr-profile, it becomes easier to identify the stable regions because some of the high value prediction stitches are more resilient to change than other stitches. It is also possible to see that the coding region of the sequence has more conformations than the non-coding region, and those regions that are false positives according to the Ensembl annotation have a behavior which is different compared to the true positives regions.

All these observations are based on the visual presentation, and do not have the accuracy that is possible with running statistics on the comparison, but it represents a visual overview that might produce ideas which can lead to more understanding about the genomic structure.

What is interesting about this animation is that it shows that the stitches do not move around in the profile as temperature decreases. The stitches do however dissolve into bigger stitches or lesser ones during the decrease as expected.

[8] http://pubgeneserver.uio.no:8090/SP_decreasing_temperature.gif

Comparing a set of stitch profiles with increased temperature with Ensembl's annotations:

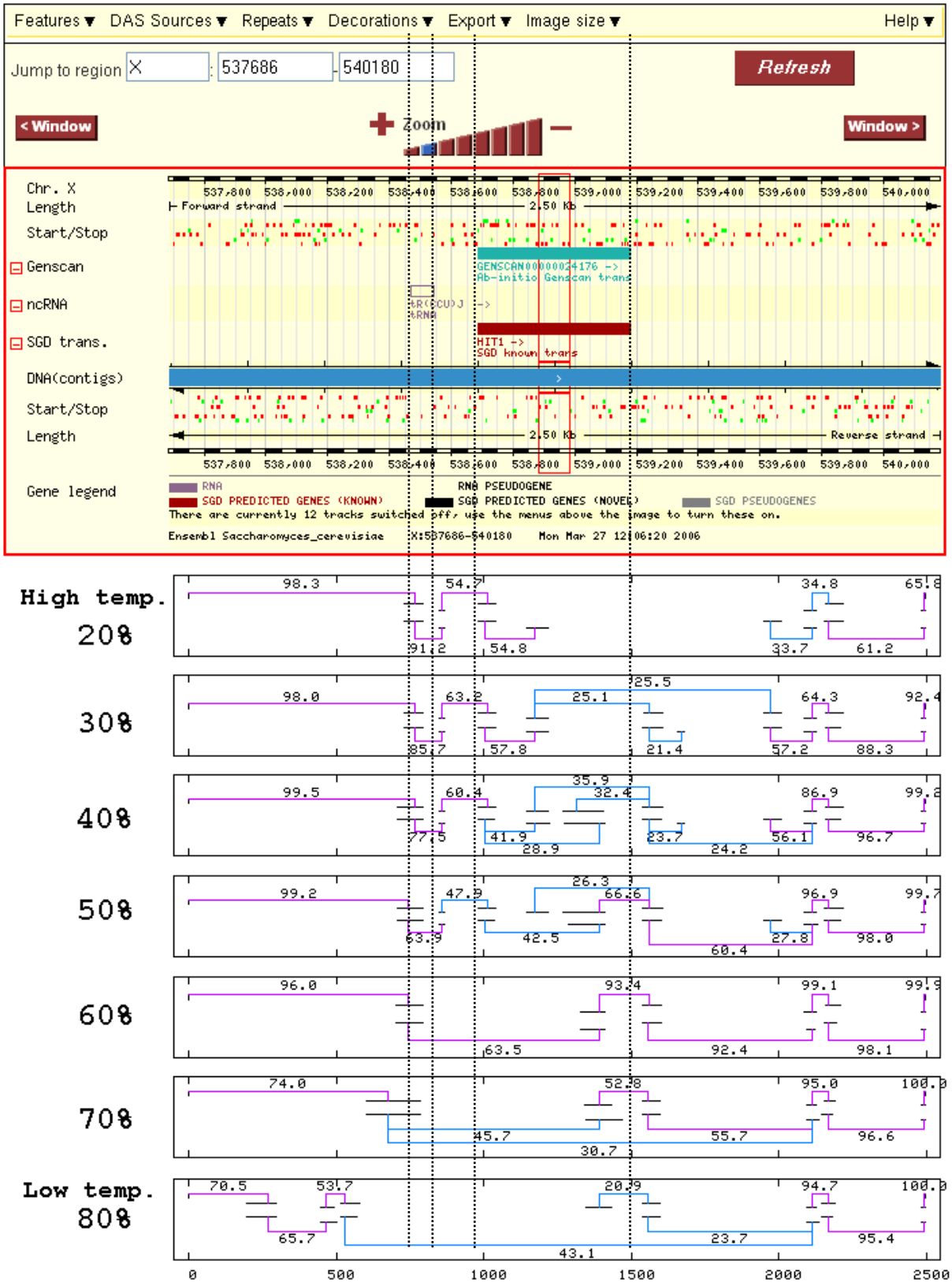


Figure 27 - Decreasing temperature (20% high temperature, 80% low temperature)

The Figure 27 is a reduced version of the animation with a probability cutoff: p_c=0.3 in order to get a better overview of the major stitches. Here it is easy to see that the prediction of each stitch varies when the temperature decreases, and how some stitches remain almost through all the tested temperatures. The coding region with the gene ‘HIT1’ in Ensembl also seems to gain more conformations as temperature decreases compared to the non-coding regions.

The tRNA (CCU)J also has a noticable corresponding stable stitch. As the temperature increases both the tRNA and HIT1 gets one common closed stitch. The false positive stitches at the left and right of the HIT1 gene seem to have less activity and conformations compared to the coding region.

11.5 Stitch profiles compared to Ensembl's Saccharomyces cerevisiae annotations

These genes from the genome *S. cerevisiae* were analyzed:

Figure	*Gene*	*Chr*	*Start*	*Stop*	*Length (bp)*
Figure 28	CDC8	X	542478	545972	3495
Figure 29	PKT2	X	544274	548969	4696
Figure 30	CBF1-YJR061W	X	548180	553500	5321
Figure 31	NTA1-RPA12-CCT5	X	552700	557756	5057
Figure 32	MOG1-HOC1	X	572580	575600	3021
Figure 33	CDC11-MIR1	X	574996	580000	5005

Table 3 - Overview of the genes analyzed

The genes were more or less randomly chosen starting off with the CDC8 gene and then moving the analysis from that point to the CDC11-MIR1 genes.

In this analysis, each stitch profile has been created by the sequence described in Table 3.

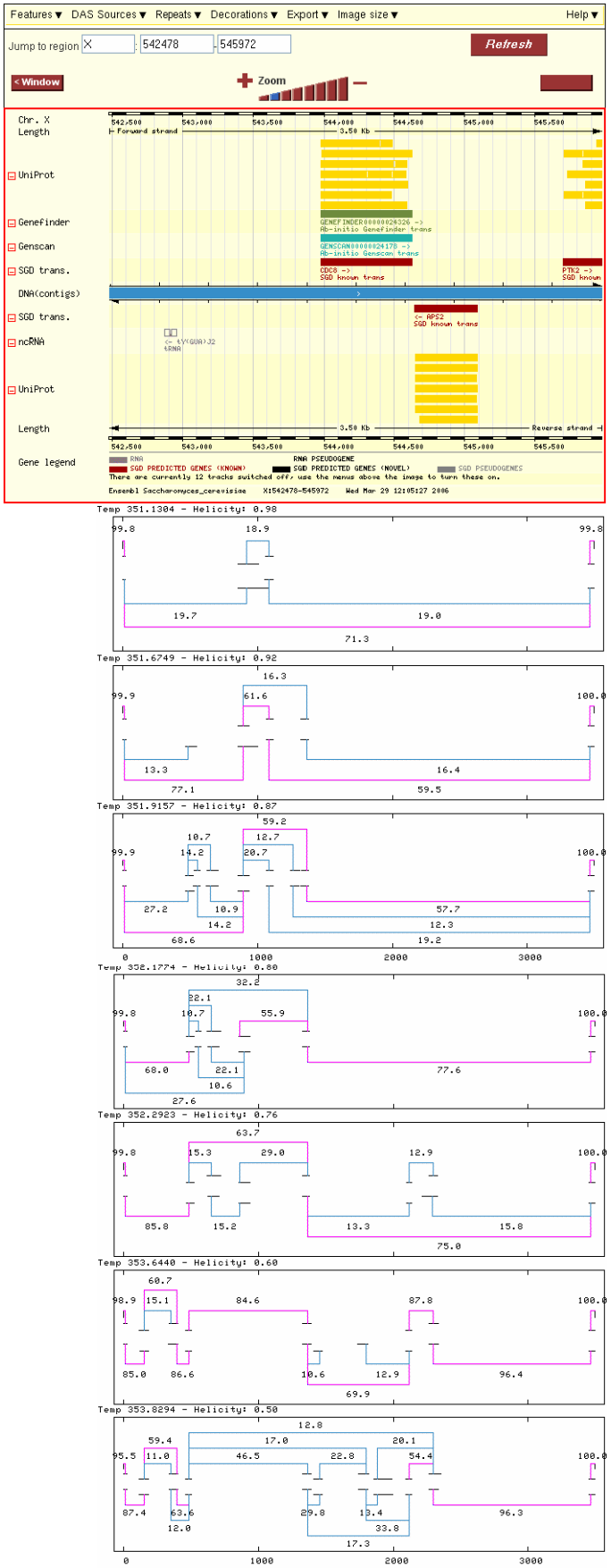


Figure 28 – Stitch profiles (Helicity:98%->50%) compared to the gene: CDC8

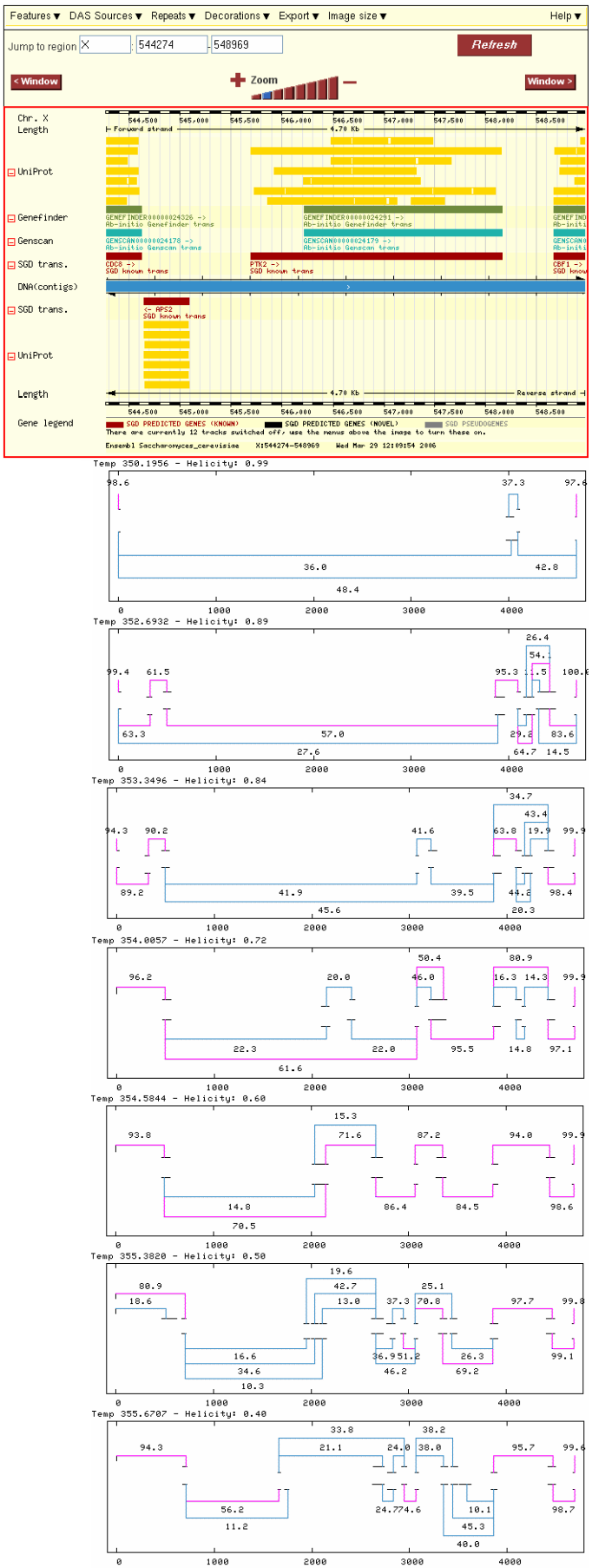


Figure 29 - Stitch profiles (Helicity:99%->40%) compared the gene: PTK2

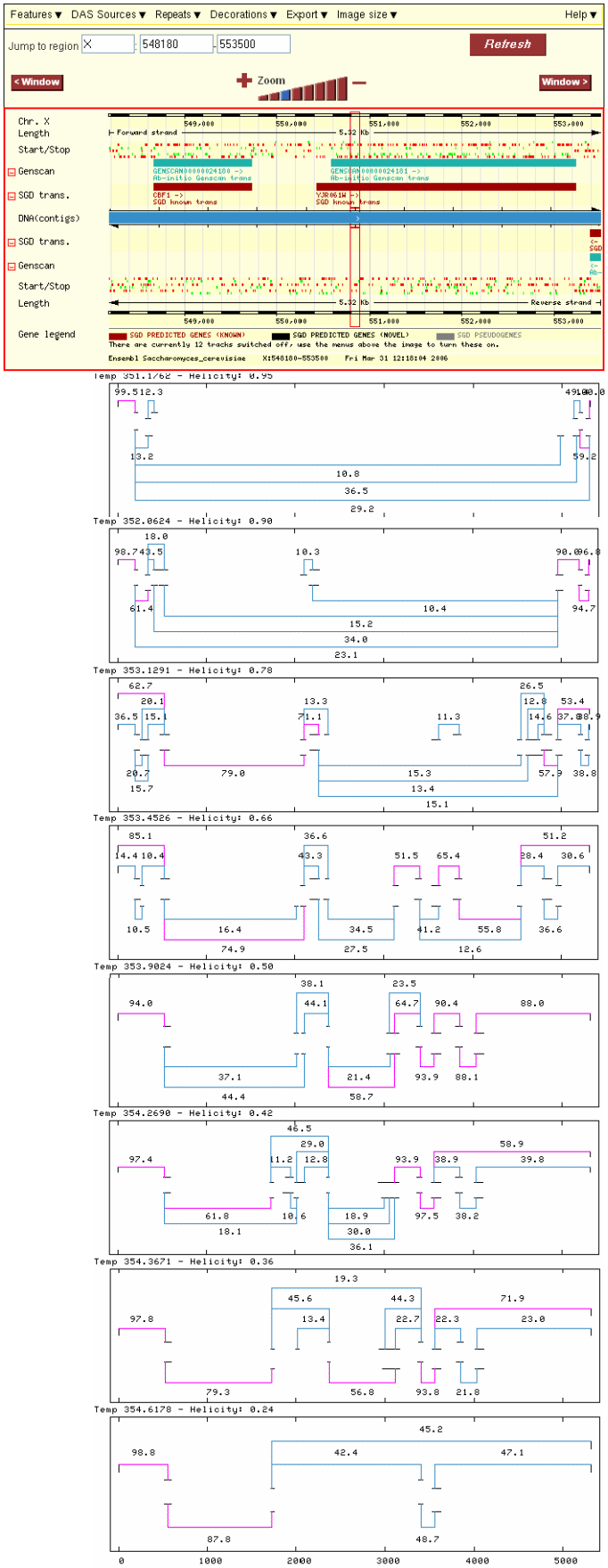


Figure 30 - Stitch profiles (Helicity:98%->24%) compared to the genes: CBF1-YJR061W

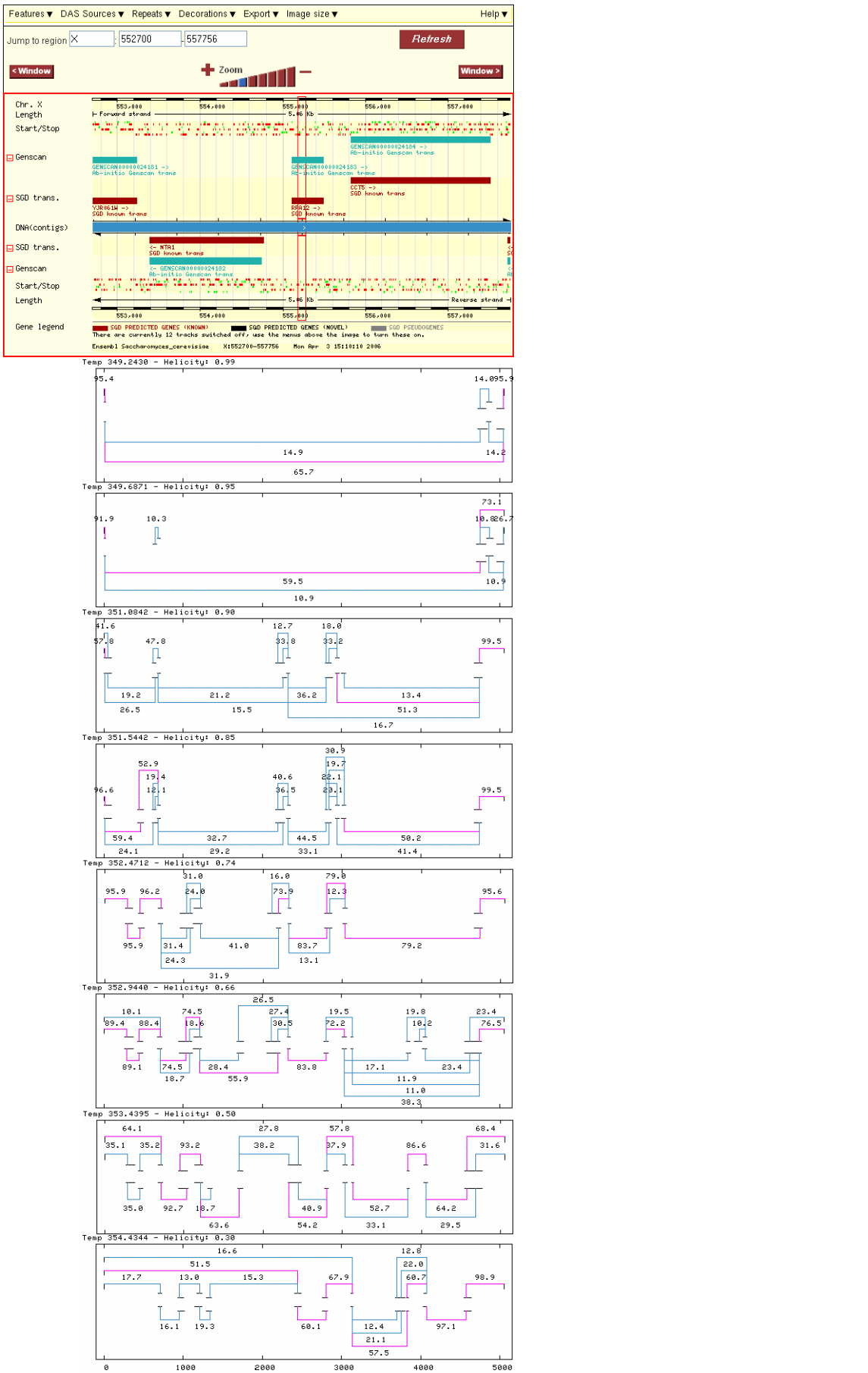


Figure 31 - Stitch profiles (Helicity:98%->38%) compared to the genes: NTA1-RPA12-CCT5

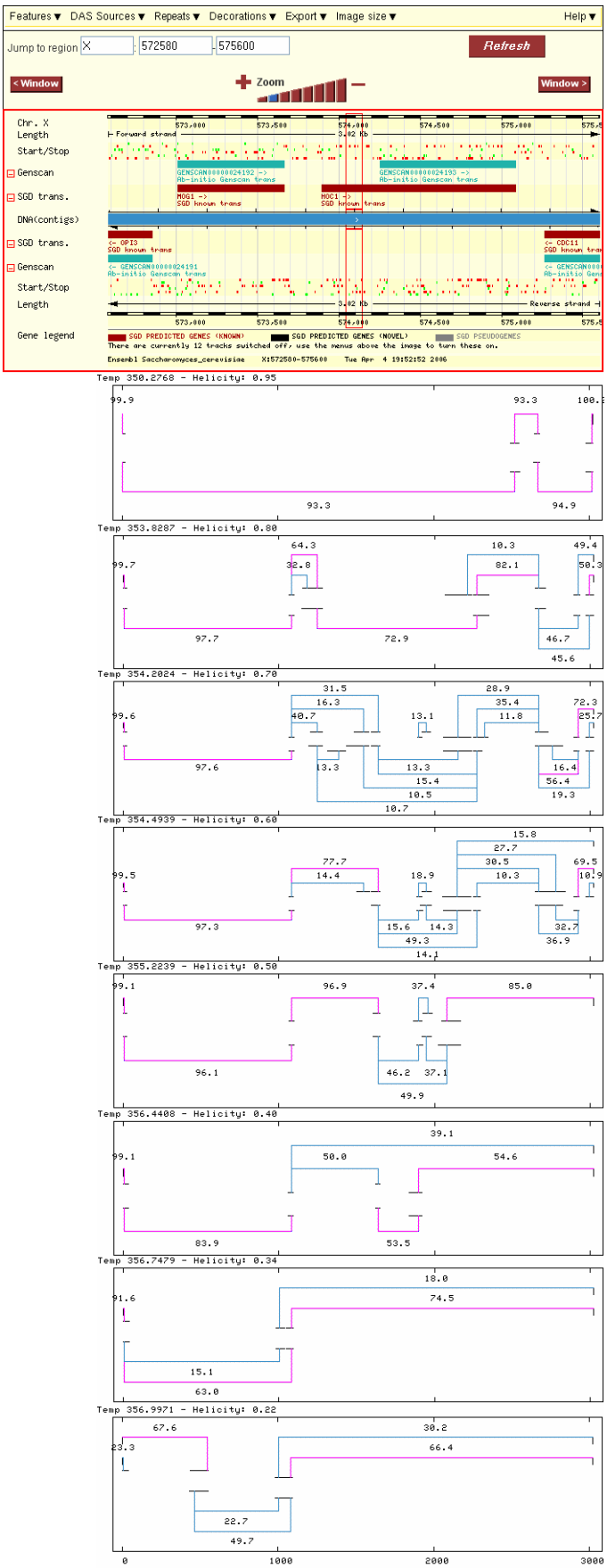


Figure 32 - Stitch profiles (Helicity:95%->22%) compared to the genes: MOG1-HOC1

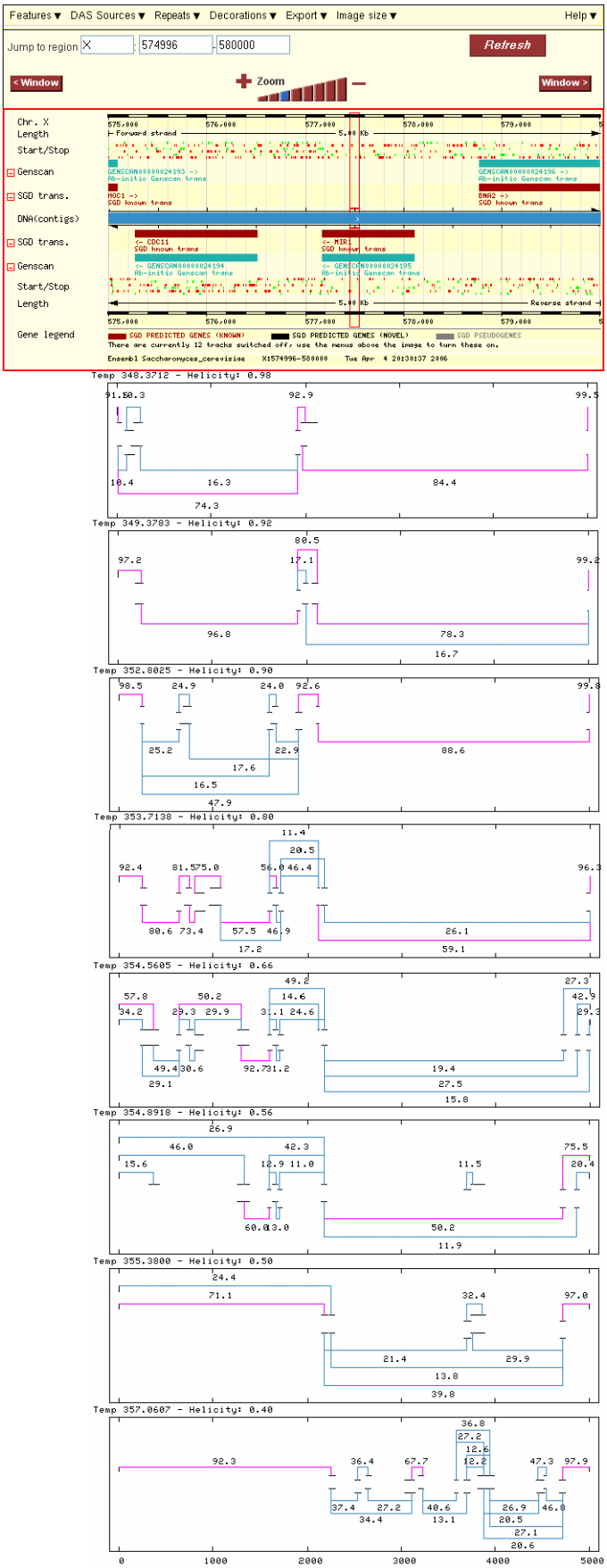


Figure 33 - Stitch profiles (Helicity:95%->40%) compared to the genes: CDC11-MIR1

11.5.1 Analyzing the figures:

Instead of analyzing these figures with great depth such as done with Figure 25 and Figure 26, they will be visually compared with the annotations in Ensembl without vertical lines and named non-coding regions.

In this analysis, it would be interesting to find some answer to these questions to find more biological relevance to the stitch profile:

- Does the stitches correspond to the annotation?
- Is there any consistency to the false positives in the stitch profile?
- Is there any structural similarity to the gene melting which is consistent with all the profiles

Do the stitches correspond to the annotation?
Looking at Figure 28, we observe that the region of the CDC8 gene is stable as the temperature increases, and also that the surrounding sequence of the CDC8 have a higher probability to melt early. At 50% helicity it is also possible to see that the profile have about three stable regions which correspond to the annotation. The images in Figure 28 shows which helix is most likely to dissociate first as the temperature increases, and this behavior is similar for all the figures in this this test and was the reason why this test with increasing temperature was created. As shown with the analysis of Figure 25 and Figure 26, stitch profiles is particularly good in finding non-coding regions, but the number of false positives is also higher (meaning more open stitches than annotated).

Looking at all the figures, the first 98-90% helicities open stitches are almost without false positives, but the accuracy decreases as the temperature increases.

When considering closed stitches when looking at all the figures, the stitch profile is much better at predicting coding regions in the forward strand of the sequence. Coding regions in the reverse strand shows a tendency of being not discovered. The stitch profile does not make a difference if the coding region is either forward or reverse, so it is a bit strange that reverse strand genes have a tendency to be less predicted. The stitch profile calculates both strands and puts the results together, and since these two calculations should be symmetric, these results should not be occurring.

Is there any consistency to the false positives in the stitch profile?
The false positives usually occur more in the open stitches when comparing the stitches to the annotation, while it is the coding regions that have the false positives when comparing the annotations to the stitch profile.

Is there any structural similarity to the gene melting which is consistent with all the profiles
Particularly the open stitches at high helicities (100%-90%) are almost always true positives which is a good way to find non-coding regions within a genome. This behavior is true to both strands of the sequence.

The figures show that if there exist a coding region within a sequence, an open stitch will be opening in the control region to the coding region. This behavior is also true to both strands, but not always.

If a coding or a non-coding region is particularly long, it will break up into a smaller more complex stitch structure. This will lead to confusion if we are looking for coding regions, and it also shows that the stitch profile is better when it is compared with annotations.

It is also noticeable that this genome is without introns and other more complex genome structure which the stitch profile has problems predicting correctly. Also we have used ‘windows’ of sequence which enhances the performance of the stitch profile algorithm. Such as padding sequence and placing the region of interest in the middle. The genomic structure of this genome is less complex than the humane genome, and that might make it easier for the algorithm to find coding regions..

11.6 Stitch profiles compared to Ensembl's Homo sapiens annotations

These genes from the genome *Homo sapiens* were analyzed:

Figure	*Gene*	*Chr*	*Start*	*Stop*	*Length (bp)*
Figure 34	POU3F4	X	82567430	82572920	5491
Figure 35	CYLC1	X	82930000	82937001	7002
Figure 36	Q8IW7_HUMAN	X	83994000	83998000	4001
Figure 37	LAMR1P15	X	86762000	86767833	5834
Figure 38	CPXCR1	X	87808000	87817000	9001

Table 4 - Overview of the genes analyzed

We made a stitch profile for the dystrophin gene which is located in the chromosome X and has a length of 2,4 Mbp. This is a very big stitch profile, and therefore the profile is more accurate than the shorter ones. The profile has been calculated with 50% helicity and the relevant region of this profile is shown at the bottom in the next overview figures.

The transcript from the dystrophin gene is randomly chosen and is called 'NP_004012.1'. From this transcript, some of the first exons have been analyzed.

Figure	*Exons*	*Chr*	*Start*	*Stop*	*Length (bp)*
Figure 39	ENSE00001400386	X	31931145	31935243	4099
Figure 40	ENSE00001258651	X	31744113	31748288	4176
Figure 41	ENSE00001213257	X	31707854	31712001	4148
Figure 42	ENSE00001258622	X	31650962	31655147	4186
Figure 43	ENSE00001258614	X	31612992	31616193	3202

Table 5 - Overview of which exons in a dystrophin transcript that have been analyzed

Also in this analysis, each stitch profile has been created by the sequence as described in Table 4 and Table 5. There has also been added an extra stitch profile into the overview figures of the dystrophin transcript.

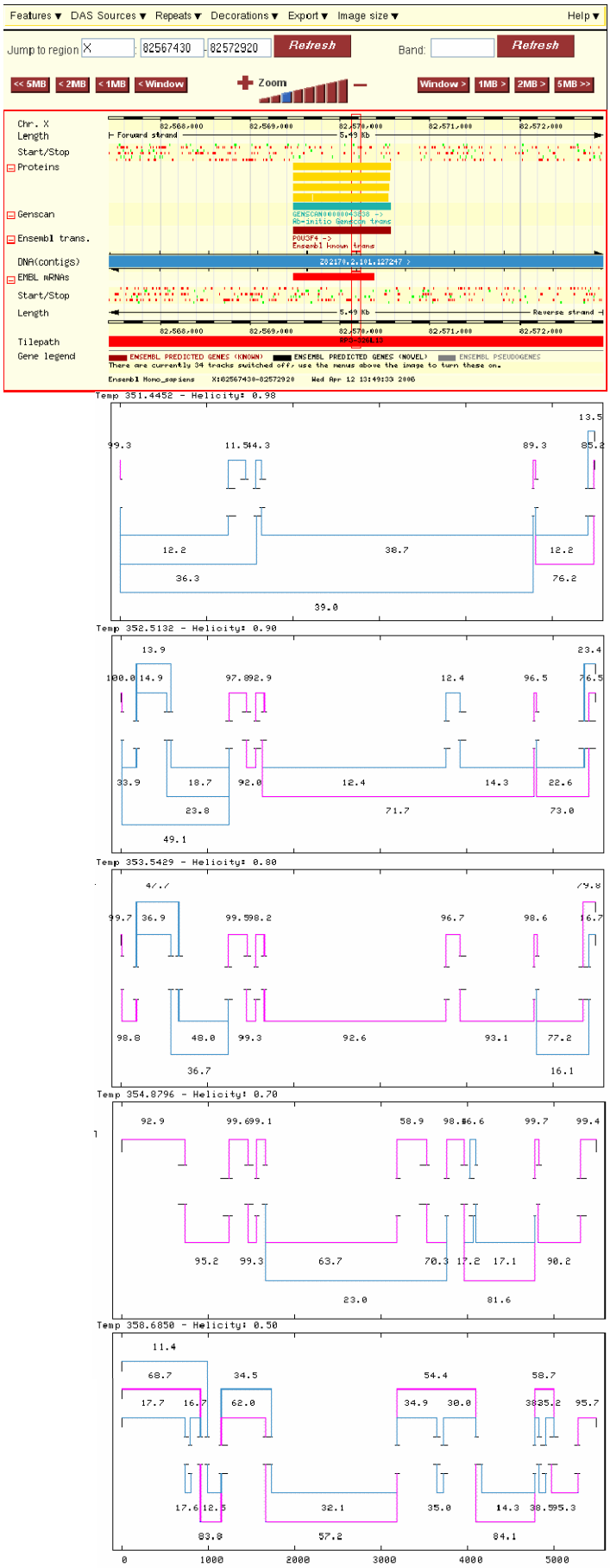


Figure 34 - Stitch profiles (Helicity:98%->50%) compared to the gene: POU3F4

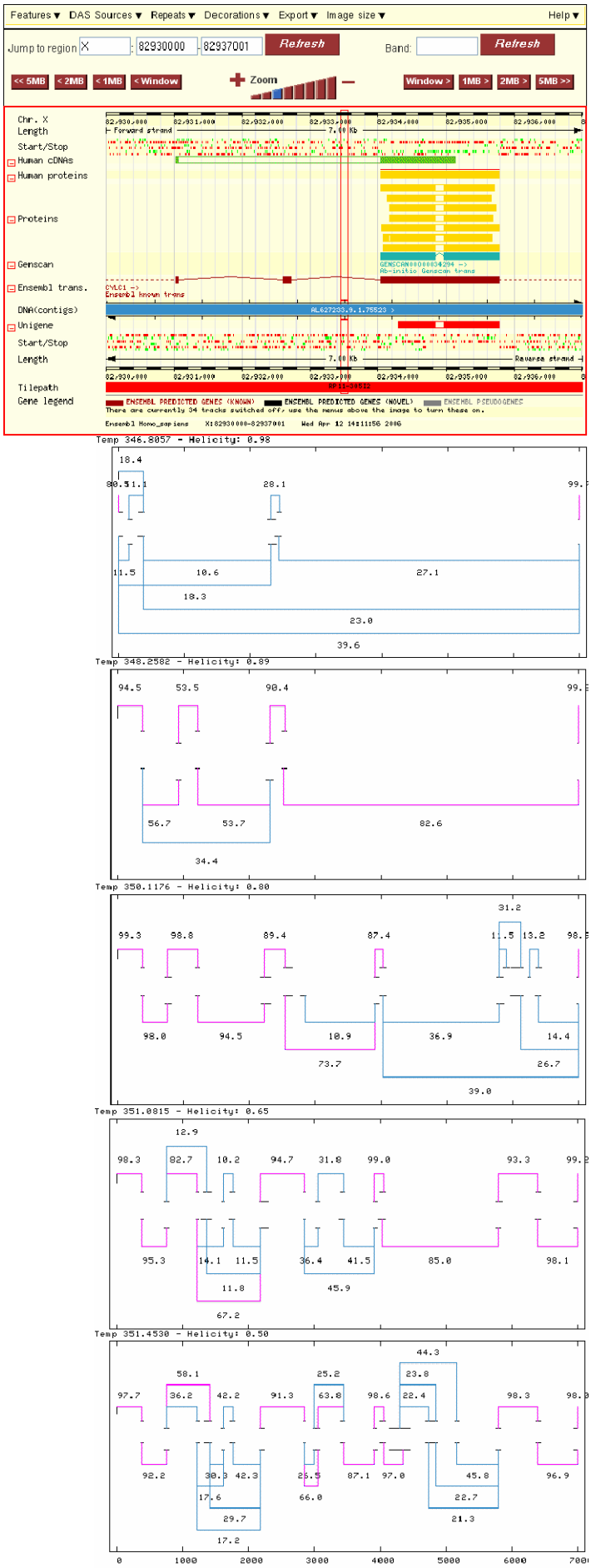


Figure 35 - Stitch profiles (Helicity:98%->50%) compared to the gene: CYLC1

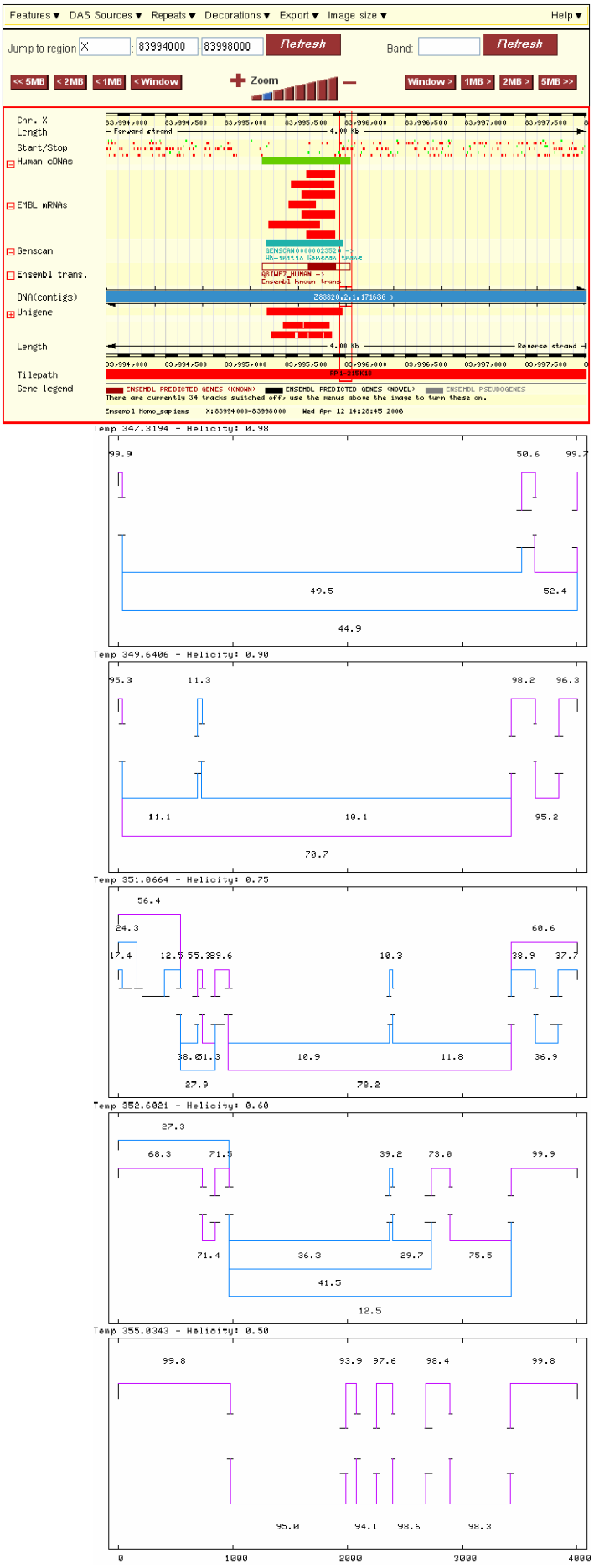


Figure 36 - Stitch profiles (Helicity:98%->50%) compared to the gene: CYLC1

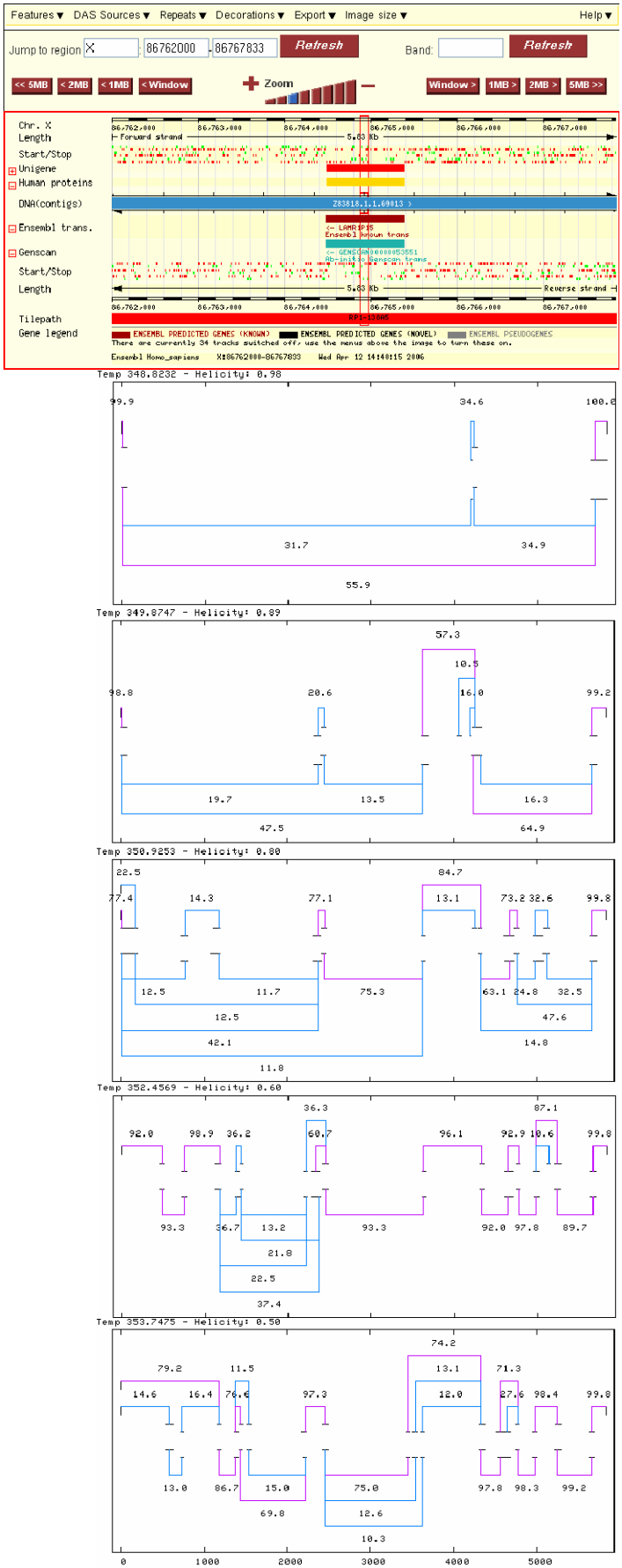


Figure 37 - Stitch profiles (Helicity:98%->50%) compared to the gene: LAMR1P15

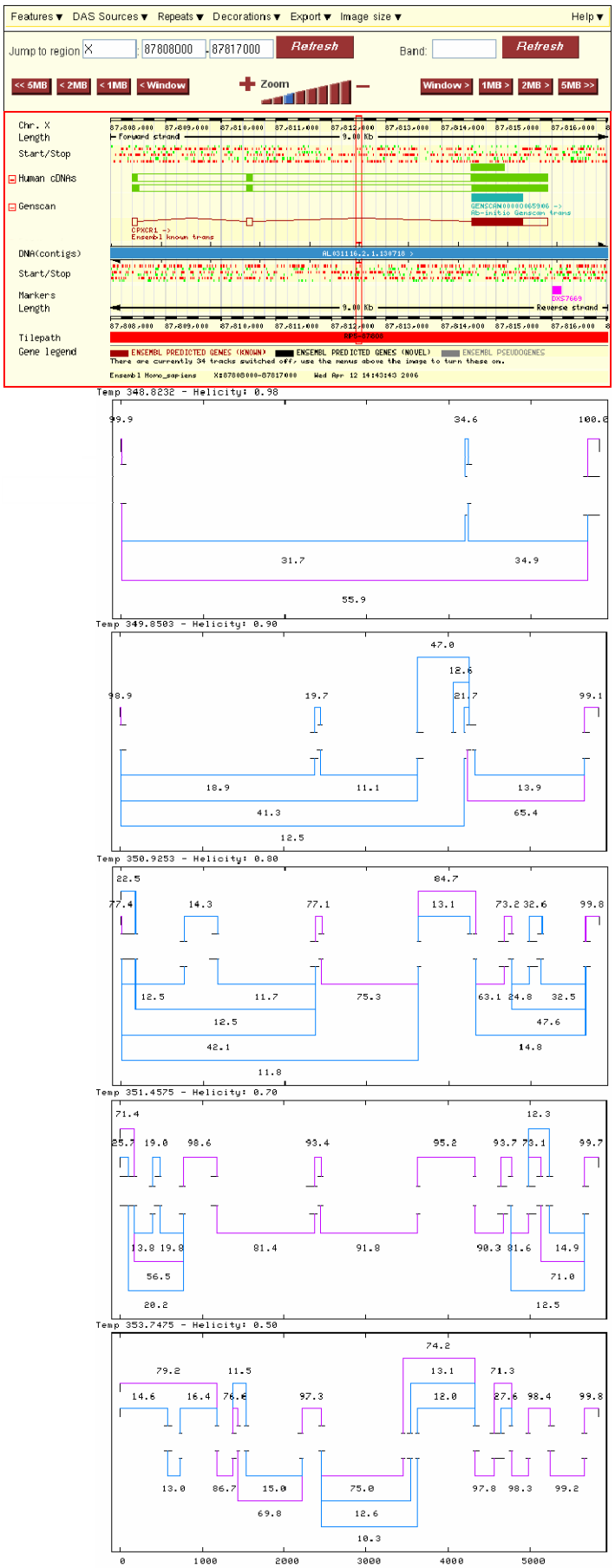
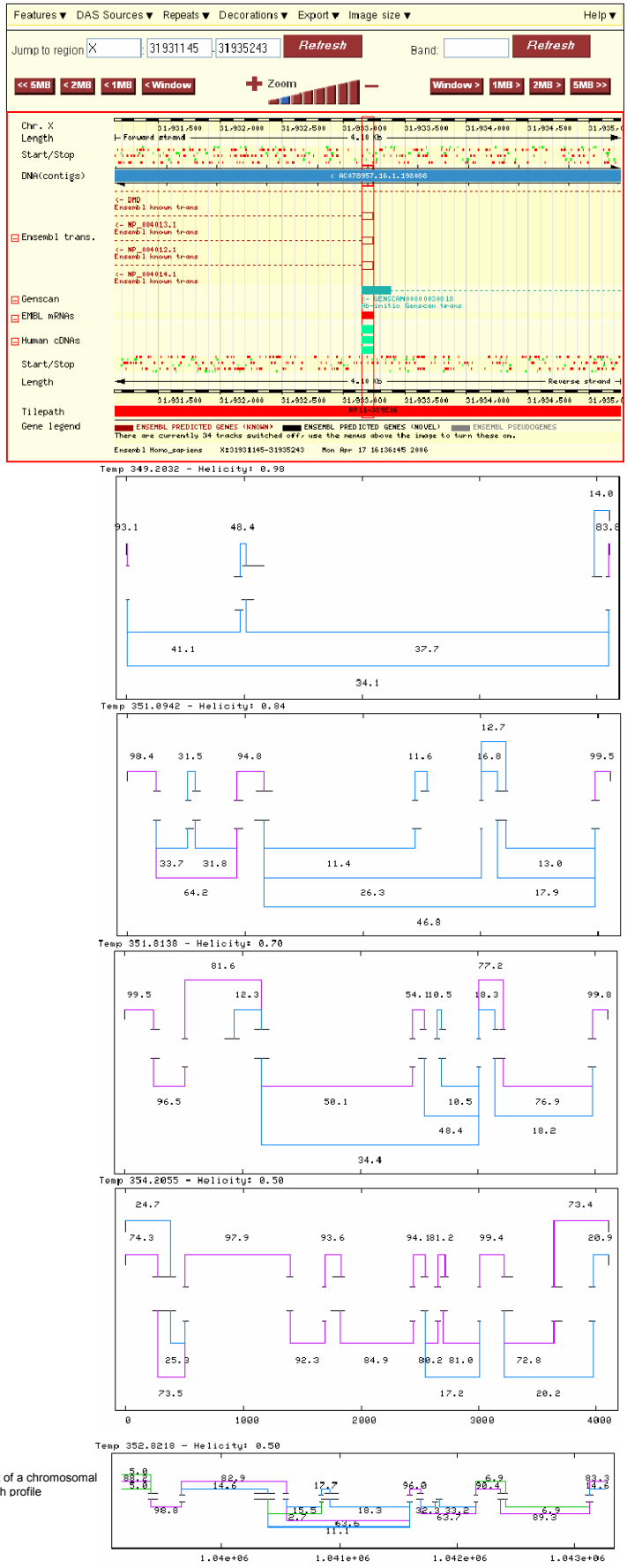
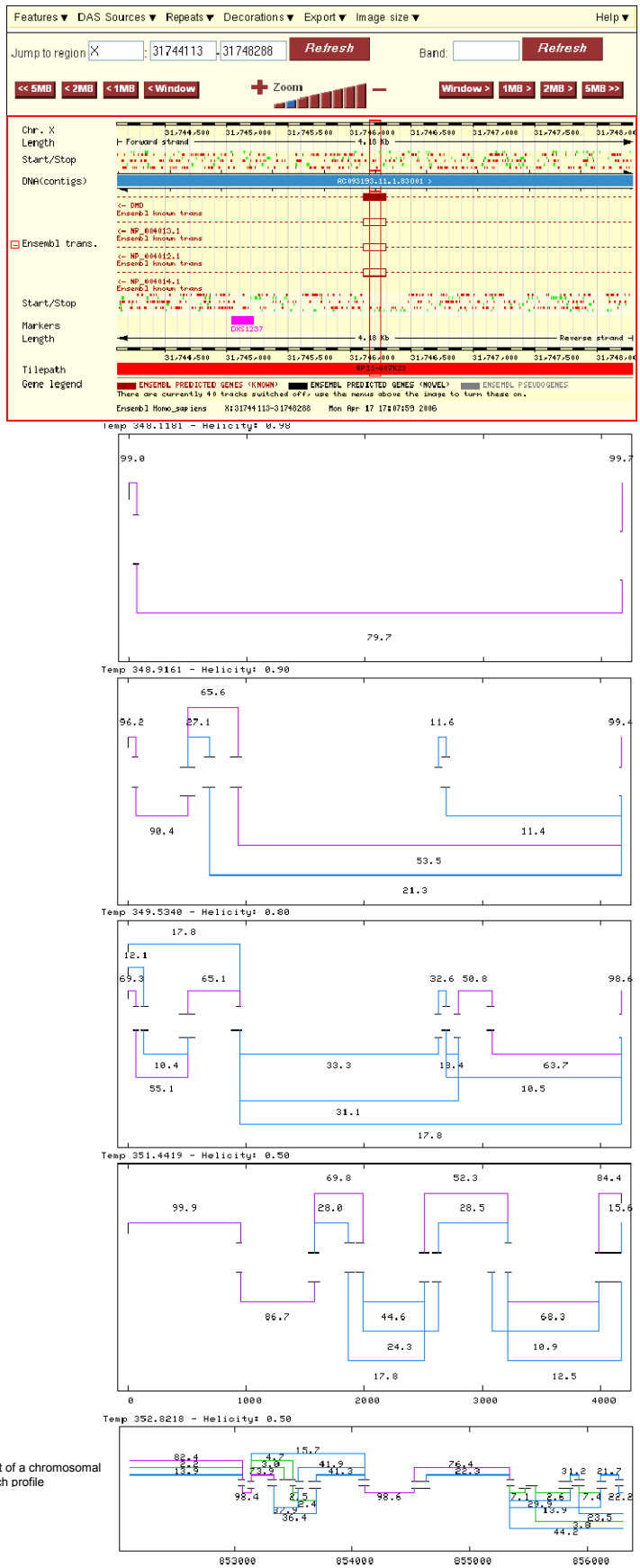


Figure 38 - Stitch profiles (Helicity:98%->50%) compared to the gene: CPXCR1



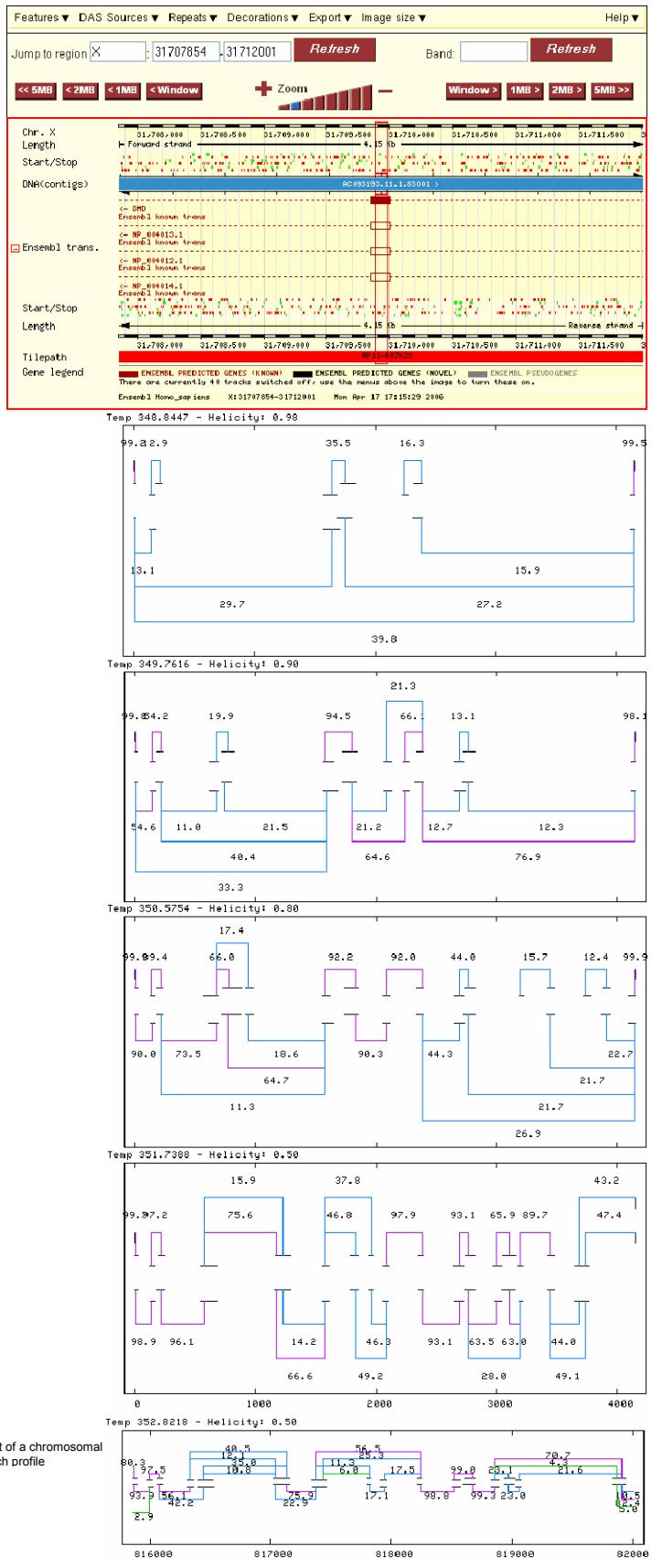
Part of a chromosomal stitch profile

Figure 39 – Stitch profiles compared to ENSE00001400386



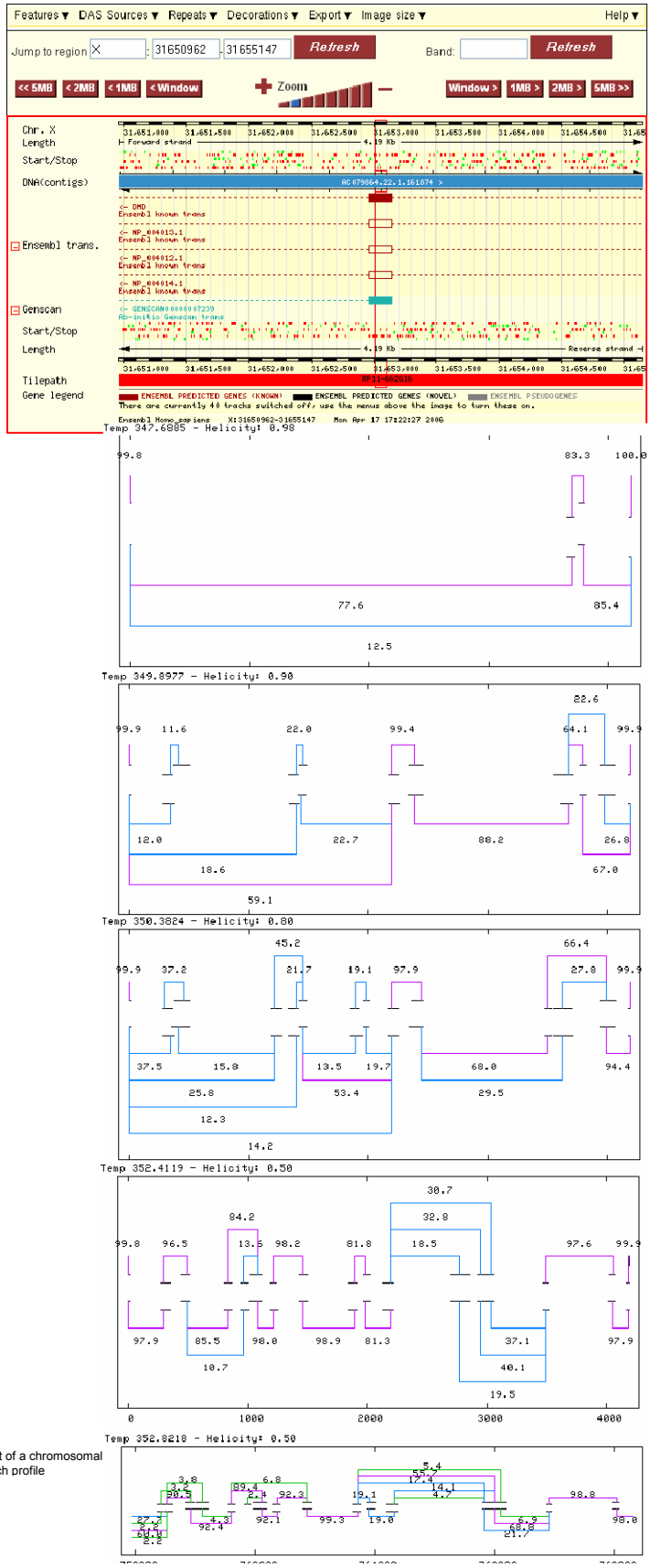
Part of a chromosomal stitch profile

Figure 40 - Stitch profiles compared to ENSE00001258651



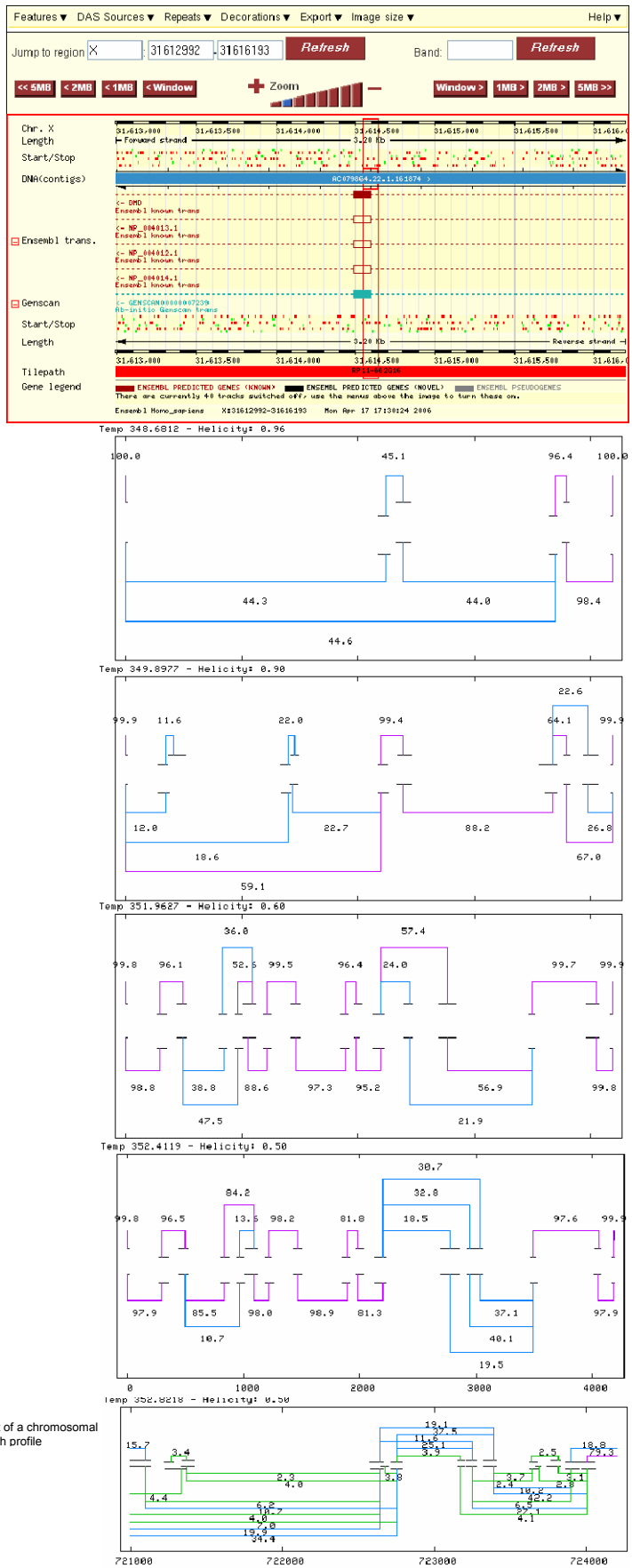
Part of a chromosomal stitch profile

Figure 41 - Stitch profiles compared to ENSE00001213257



Part of a chromosomal stitch profile

Figure 42 - Stitch profiles compared to ENSE00001258622



Part of a chromosomal stitch profile

Figure 43 - Stitch profiles compared to ENSE00001258614

11.6.1 Analyzing the figures:

The humane genome is more complex in structure than the previous analyzed genome, and possibly because of this the stitch profile is less accurate with its predictions.

Compared with the stitch profiles of the SC genome, the profiles in these sequences have a lot more false positives in both closed and open stitches. Again we see that the weakest points of the sequence gets good prediction according to the annotations, but there is also a lot more weak points because of the length of the non-coding region. The higher density of coding regions in the SC genome makes it easier for the stitch profile to predict the open and closed stitches because the fluctuations in the melting map are changing closer to each other. This is not the case with the human genome, and makes the prediction more difficult.

Genes Figure 34 to Figure 38
The stitch profile have some difficulties predicting the control regions and also the coding region, but there was almost no false positives comparing the coding region to the stitch profile down to 50%. All sequences melt when enough temperature is applied, so that some of the figures show that the coding regions are stable until the temperature gets too high. It is possible to assume that a stable stitch through temperature increase, is a good indication of a strong sequence region.

Dystrophin Figure 39 to Figure 43
Even though there is a lot of false positives in both annotation compared to profile and vice versa, there is no false positives comparing the coding region of the annotation to the corresponding closed stitch. It is also noticeable that the open stitches in the range 98-90% helicity correspond to the control regions of the annotated coding regions. So when considering these two features combined it is a good prediction, but there are a lot of false positives around them.

By looking at the figures, the stitch profile algorithm provides again a good prediction of the control regions to the coding regions. The stitch profile shown at the bottom of the overview figure is the complete dystrophin calculation which is more accurate than the shorter sequence calculation. This profile is a calculation of the whole dystrophin gene and because of this, it has been possible to detect open and closed regions which are connected from greater distance.

11.7 A summary of the comparison test

The stitch profile shows that it has a more accurate prediction on genomes with simpler sequence structure after having analyzed the temperature behavior in the *Saccharomyces cerevisiae* and *Homo sapiens* genome. We also found that the coding region in a sequence have a structure where the control regions are more likely to melt first when we melt the sequence from 100% helicity and down.

We have also seen that the stitch profile is better at predicting smaller coding regions than long ones. This can be noticed in Figure 29, Figure 30 and Figure 31.

In the yeast genome, the stitch profile is better at predicting whole coding regions than in the human genome, but in both genomes it is still better at predicting non-coding regions. This is true because the genes in the yeast genome are shorter in length, while in the humane genome the genes are sometimes too large to be calculated.

In these analyses, the annotations of Ensembl has been the reference for determining the accuracy to the stitch profile. So that the false positives when comparing the annotations are related to this reference. What we did not investigate was the possibility that the profile was showing some other genomic structure that was not annotated. It would be the natural next step to research other possible structures that might apply, where this may explain more about the correlation between the stitch profile and the genomic structure.

12 Conclusion and summary

12.1 Implementation

The integration of stitch profile into Ensembl made it possible to create stitch profiles while browsing through the genomes which are available and able to compare these profiles with the other annotations. At the moment there is a mix of creating stitch profiles on the fly and creating them out of pre-calculated stitch files such as was done with dystrophin gene. With the new version of stitch profiles (Tøstesen, to be published) it is possible to calculate stitch profiles for very long sequences within reasonable time. We have decided not to calculate the whole genome for either human and yeast, but to focus on doing the temperature analyses. This was because the time spent on implementing and calculating would be too time consuming when there were other analyses that were more important to perform.

We also considered another version of the stitch profile which could perform chromosomal calculation with moving calculation windows based on max and minima points on the melting profile, but got instead another version that was faster without loosing precision. As mentioned before, this algorithm was available too late to be substituted with the version which currently is implemented.

The implementation design also made it possible to add other annotations such as the melting map into Ensembl in the same way as stitch profiles was added. The melting map is available for the whole human genome, and can be compared with the other annotations in Ensembl in addition to the stitch profile.

Ensembl was a good software package to work with where its high modularity and implementation principals make it possible to add own code to the system without the need to change too many lines of code.

Hopefully, there will be a stable version of the implementation at some point in time so that scientists can browse the human genome with the stitch profile annotation.

12.2 Annotation comparison

We analyzed 6 yeast sequences and 10 human sequences, and found the algorithm to visually predict a coding region with high probability, but it is hard to predict coding regions without the support of other annotations because of the high amount of false positives. We saw that the stitch profile depends on sequence and genome, and that the stitch profile algorithm works better on less complex genomic structures. The comparisons are done visually and manually, and therefore we loose some precision. These predictions would have been more correct and reliable if we had used statistics instead of visual comparisons.

12.3 Future:

12.3.1 Implementation

Clickable profiles
The ability to click on each stitch to get the stitch details would be nice feature to have on the stitch profile, but that is not possible at the moment. To add this functionality, there are several solutions such as mapping the pixels on the image to a hyperlink, or to add vector based graphics with clickable units.
This would allow the user to get more detailed information out of the stitch profile.

Chromosomal calculations
Since the current implementation does not have the entire genome available, this would be next necessary step for full coverage of the genomic calculation of the human genome.

Stitch profiles – new version
The new version of stitch profile is faster than the one implemented, and this should be corrected.

12.3.2 Statistics

This would have been the biggest and most useful addition to this system, but it would also take a long time to design and implement because of the amount of unknown factors. With statistics we would be able to analyze base-for-base correlations between the stitch profile and the gene annotation. This would require finding out how to extract stitch details for a given position, and also finding out how to extract Ensembl annotation position information. After having solved that, we could have performed analyses such as:

- What is the probability for a selected annotation to have a closed stitch?
- How accurate does the stitch profile predict a large human gene with introns and exons?
- How accurate does the stitch profile predict a gene in the human genome compared to a randomly generated DNA sequence?
- How many false positives between the profile and the annotation does there exist for a selected gene?
- Finding new possible coding regions in the human genome using machine-learning algorithms
- Since the stitch profiles algorithm depends on GC% content, it would be have been interesting to analyze the GC% of the sequences to find any relations on how accurate the prediction was compared to level of GC% content.

These were just a few examples of how useful statistics could be when comparing the annotations, and they only demonstrate a small portion of the material that could be analyzed in the future.

12.4 Acknowledgements

I would like to thank my supervisors Eivind Tøstesen, Eivind Hovig and Torbjørn Rognes for having shown patience and for giving guidance throughout my work on this thesis, and for having introduced me to the world of melting DNA, Ensembl and annotations. Additional thanks go to the Department of Informatics (University of Oslo) and to my colleagues at the CMBN Bioinformatics Group, for their help and support.

Appendix A. Reference list:

1. Stenger JE, Xu H, Haynes C, Hauser ER, Pericak-Vance M, Goldschmidt-Clermont PJ, Vance JM.
Statistical Viewer: a tool to upload and integrate linkage and association data as plots displayed within the Ensembl genome browser.
BMC Bioinformatics. 2005 Apr 12;6(1):95.

2. Hubbard,T., Barker,D., Birney,E., Cameron,G., Chen,Y., Clark,L., Cox,T., Cuff,J., Curwen,V., Down,T. et al. (2002)
The Ensembl genome database project.
Nucleic Acids Res., 30, 38–41.

3. Stalker J, Gibbins B, Meidl P, Smith J, Spooner W, Hotz HR, Cox AV.
The Ensembl Web site: mechanics of a genome browser.
Genome Res. 2004 May;14(5):951-5.

4. E. Tøstesen, G. I. Jerstad and E. Hovig (2005):
Stitchprofiles.uio.no: Analysis of partly melted DNA conformations using stitch profiles.
Nucl. Acids Res 33, w573-w576.Doi:10.1093/nar/gki424.

5. Stabenau A, McVicker G, Melsopp C, Proctor G, Clamp M, Birney E.
The Ensembl core software libraries.
Genome Res. 2004 May;14(5):929-33. Review.

6. Yeramian,E. (2000b)
The physics of DNA and the annotation of Plasmodium falciparum.
Gene, 255, 151–168.

7. Dowell RD, Jokerst RM, Day A, Eddy SR, Stein L.
The distributed annotation system.
BMC Bioinformatics. 2001;2(1):7. Epub 2001 Oct 10.

8. J. L. Ashurst, C.-K. Chen, J. G. R. Gilbert, K. Jekosch, S. Keenan, P. Meidl, S. M. Searle, J. Stalker, R. Storey, S. Trevanion, L. Wilming and T. Hubbard
The Vertebrate Genome Annotation (Vega) database
Nucleic Acids Res. 2005 Jan 1;33 Database Issue:D459-D465.

9. Val Curwen, Eduardo Eyras, T. Daniel Andrews, Laura Clarke, Emmanuel Mongin, Steven M.J. Searle, and Michele Clamp
The Ensembl Automatic Gene Annotation System
Genome Res. 2004 May; 14(5):942-950.

10. Yeramian E, Bonnefoy S, Langsley G.
Physics-based gene identification: proof of concept for Plasmodium falciparum.
Bioinformatics. 2002 Jan;18(1):190-3.

11. Galtier, N and Lobry JR.
Relationships between genomic G+C content, RNA secondary structures, and optimal growth temperature in prokaryotes.
J Mol Evol 1997 Jun;44(6)632-6

12. Yeramian E.
Genes and the physics of the DNA double-helix.
Gene. 2000 Sep 19;255(2):139-50.

13. Robin D Dowell , Rodney M Jokerst , Allen Day , Sean R Eddy and Lincoln Stein
The Distributed Annotation System
BMC Bioinformatics 2001, 2:7 doi:10.1186/1471-2105-2-7

14. www.ensembl.org

15. http://www.ensembl.org/info/about/code_licence.html

16. Fang Liu, Eivind Tøstesen, Jostein K. Sundet, Tor-Kristian Jenssen, William G. Thilly and Eivind Hovig
The human genomic melting map
Manuscript

17. W. James Kent et al.
The Human Genome Browser at UCSC
Genome Res. 2002 May; 12(6):996-1006.

18. Wheeler et al.
Database resources of the National Center for Biotechnology
Nucleic Acids Res. 2003, 31, No. 1 28-33

19. Kim D. Pruitt, Tatiana Tatusova and Donna R. Maglott
NCBI Reference Sequence (RefSeq): a curated non-redundant sequence database of genomes, transcripts and proteins
Nucleic Acids Res. 2005, Vol33, Database Issue:D501-D504.

20. A. S. Hinrichs et al.
The UCSC Genome Browser Database: update 2006
http://www.ncbi.nlm.nih.gov/projects/CCDS/
Nucleic Acids Res. 2006, Vol 34, Database Issue:D590-D598.

21. The International SNP Map Working Group
A map of human genome sequence variation containing 1.42 million single nucleotide polymorphisms
Nature 409, 928-933 (15 February 2001)

22. The International Human Genome Mapping Consortium
A physical map of the human genome
Nature 409, 934-941 (15 February 2001)

23. Watson, James D. and Francis H.C. Crick
A structure for Deoxyribose Nucleic Acid
Nature 171, 737–738 (25 April 1953)

24. Zody MC, Garber M, Sharpe T, Young SK, Rowen L, O'Neill K, Whittaker CA, Kamal M, Chang JL, Cuomo CA, Dewar K, FitzGerald MG, Kodira CD, Madan A, Qin S, Yang X, Abbasi N, Abouelleil A, Arachchi HM, Baradarani L, Birditt B, Bloom S, Bloom T, Borowsky ML, Burke J, Butler J, Cook A, DeArellano K, DeCaprio D, Dorris L 3rd, Dors M, Eichler EE, Engels R, Fahey J, Fleetwood P, Friedman C, Gearin G, Hall JL, Hensley G, Johnson E, Jones C, Kamat A, Kaur A, Locke DP, Madan A, Munson G, Jaffe DB, Lui A, Macdonald P, Mauceli E, Naylor JW, Nesbitt R, Nicol R, O'Leary SB, Ratcliffe A, Rounsley S, She X, Sneddon KM, Stewart S, Sougnez C, Stone SM, Topham K, Vincent D, Wang S, Zimmer AR, Birren BW, Hood L, Lander ES, Nusbaum C.
Analysis of the DNA sequence and duplication history of human chromosome 15.
Nature. 2006 Mar 30;440(7084):671-5.)