

## Journal Information

Journal ID (publisher-id): vb  
Title: Virtual Biology  
ISSN (electronic): 2306-8140

## Article Information

Publication date (electronic): 25 March 2013  
Electronic Location Identifier: e8  
DOI: 10.12704/vb/e8

---

# BioUML Genome Browser

Tagir Valeev<sup>1, 2</sup>, Ivan Yevshin<sup>2</sup>, Fedor Kolpakov<sup>2, 3</sup>

[1] Institute of Informatics Systems SB RAS

[2] Institute of Systems Biology, Ltd

[3] Design Technological Institute of Digital Techniques, SB RAS

---

## Abstract

*Motivation:* The BioUML platform contains a number of analysis tools to process genomic data. Visualization of such data can be performed using genome browsers. Using external genome browsers is inconvenient as it requires exporting data from BioUML and importing it into a genome browser. Thus creation of an internal BioUML genome browser was considered.

*Results:* The BioUML genome browser was created. It allows easy visualization of user uploaded data, analyses results and predefined tracks. Interactive navigation, semantic zoom, feature annotation display and other capabilities were implemented. Large data volumes are also supported, which is necessary for next generation sequencing analyses.

*Availability:* The web version is available at <http://ie.biouml.org/bioumlweb/> (use “Demo” button for anonymous access).

## Keywords

Genome browser, Web-application, BioUML

---

## Introduction

A Genome browser is a software product which allows users to display and interactively navigate through genomic data obtained from a biological database. Currently there are a few dozens of free and commercial genome browsers available, both as standalone (desktop) and web applications. Standalone genome browsers include Argo Genome Browser <http://www.broad.mit.edu/annotation/argo/> [Engels et al, 2006], Artemis Genome Browser <http://www.sanger.ac.uk/resources/software/artemis/> [Carver et al, 2011], Gaggle Genome Browser <http://gaggle.systemsbiology.net/docs/geese/genomebrowser/> [Bare et al, 2010], Integrated Genome Browser <http://igb.bioviz.org/> [Nicol et al, 2009], Integrative Genomics Viewer [Thorvaldsdóttir et al, 2012] and so on. Many standalone genome browsers still need Internet access to download genomic annotation from online databases.

Notable web-based genome browsers are Ensembl genome browser <http://www.ensembl.org/> [Hubbard et al, 2002], UCSC Genome Browser <http://genome.ucsc.edu/> [Kent et al, 2002], JBrowse <http://jbrowse.org/> [Skinner et al, 2009]. Modern web technologies allow the web-applications to be as interactive as standalone ones, though Ensembl genome browser still lacks interactive features like “drag-and-scroll”. Still the only notable drawback of web-based genome browsers today is the necessity to upload user’s own data to display them in the browser.

Most of the genome browsers (both standalone and web) come as independent products or as add-ons to an online database (E.g. Ensembl genome browser). While they can display both predefined and user-uploaded tracks, they lack integration with analysis tools. Usually users have to launch separate tools to process their data, then import the results into the genome browser to visualize them. On the other hand, many notable genomic data analysis platforms like Galaxy (<http://galaxyproject.org/>) [Goecks et al, 2010] lack their own genome browsers. Galaxy can import genomic features from UCSC Genome Browser, but cannot send them back, so the user still has to download data files and send them to the genome browser in some way. These steps slow down the research. This becomes even more problematic in case of next generation sequencing (NGS) data processing due to huge data volumes.

As the BioUML platform [Kolpakov et al, 2013] contains genomic data analyses, it was rather a straightforward decision to include a genome browser there as well, so that the user can visualize the analyses results directly in the system. The integrated genome browser should be able to visualize genomic sequences, pre-defined tracks (e.g. Ensembl genes), user uploaded files and analysis results, combining tracks from various sources on a single display. All meta-information associated with track sites should be displayed on user request. The genome browser should be available both in the BioUML workbench (desktop edition) and BioUML web edition.

## Methods and technologies

BioUML web edition is client-server software. The Server code is implemented in Java 1.6 using Apache Tomcat

web-server to handle HTTP requests.

Any modern web-browser can be used as the client application. The Client code is implemented using JavaScript. AJAX technology is used to transfer data between the client and the server. The data is encoded into JSON-format. To render the graphics on the client, HTML5 Canvas technology is used. For browsers not supporting Canvas (like Internet Explorer 8) a fallback implementation is available based on excanvas project (<http://excanvas.sourceforge.net/>).

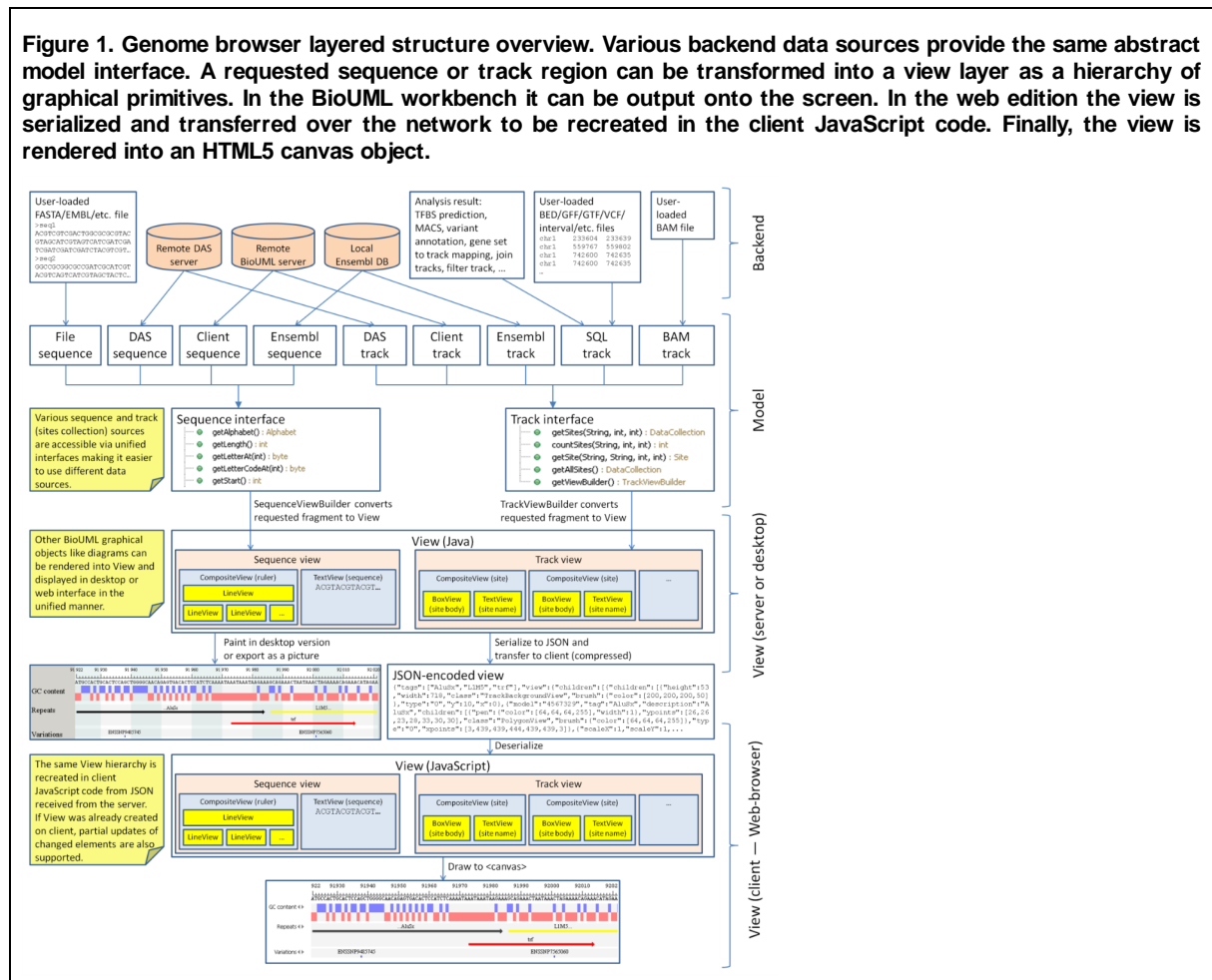
The BioUML workbench is a standalone Java 1.6 application which is based on the Swing user interface library.

## Implementation

In general most genome browser implementations contain genomic sequence and a set of individual tracks. Each track contains a number of “features” or “sites”. Every site has genomic coordinates (chromosome, start and end positions) and other information such as name, strand, intensity, and so on, which can be used to display it correctly.

The BioUML genome browser implementation schema is displayed in Fig. 1. It consists of several software layers:

1. Backend: data sources and storage;
2. Model: abstract model-level interfaces to the data sources;
3. View: the hierarchy of graphical primitives describing visual representation of the selected part.



Let’s consider these layers in more detail.

### Backend

Tracks and sequences displayed in the genome browser are gathered from various sources. The currently supported sequence sources are:

- Locally installed Ensembl database [Hubbard et al, 2002].
- User-uploaded sequence file. Several sequence formats are supported including FASTA, FASTQ [Cock et al, 2009], EMBL, GenBank. Indexes are created for the file to seek faster.
- Sequences from remote web-services accessible via DAS protocol [Jenkinson et al, 2008].
- Sequences from a remote BioUML server.

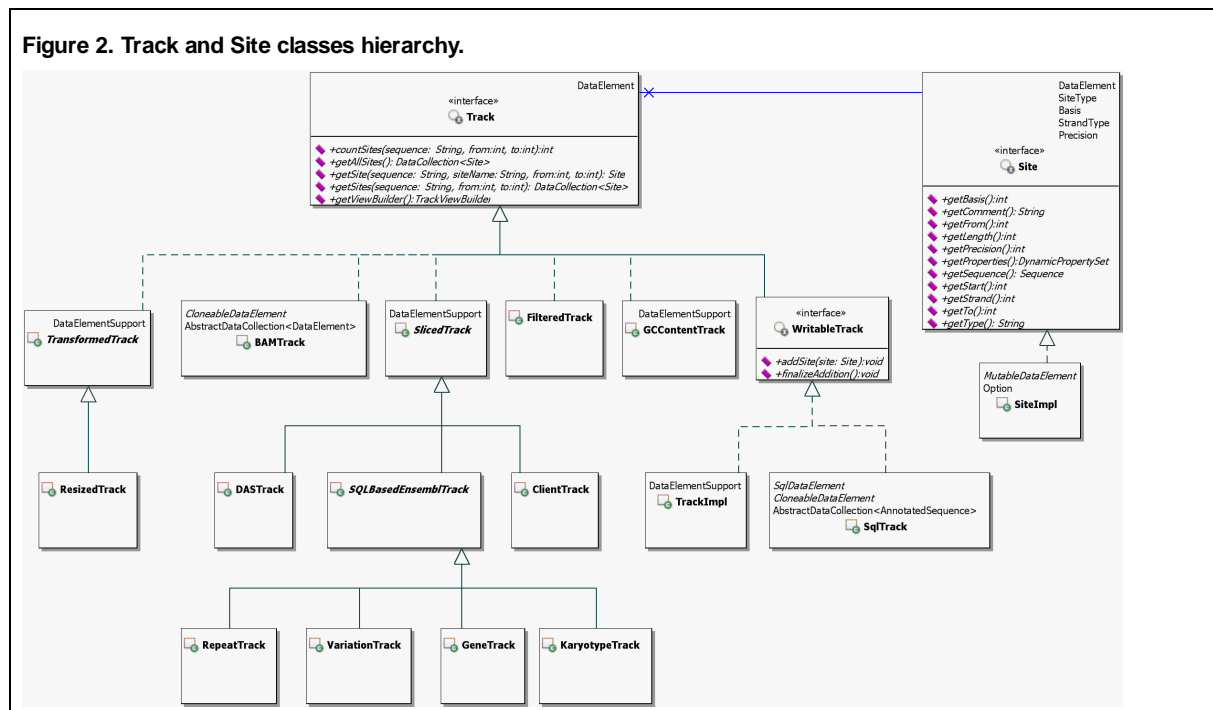
The currently supported track sources are:

- Locally installed Ensembl database provides 5 tracks: karyotype, genes, repeats, variations and GC content.

- SQL-track: table in MySQL database representing genomic intervals with arbitrary annotation. It can be created in a number of ways:
  - User-uploaded track file. A variety of file formats are supported including the following:
    - BED format;
    - GFF (General Feature Format);
    - GTF (Gene Transfer Format);
    - WIG (Wiggle Track Format);
    - VCF (Variant Call Format).
  - Output of many programs including the following:
    - Bowtie [Langmead et al, 2009];
    - BreakDancer [Chen et al, 2009];
    - Pindel [Ye et al, 2009];
    - VarScan [Koboldt et al, 2009].
  - Output of many built-in BioUML analyses.
- BAM-track: BAM (Binary sequence Alignment/Map [Li et al, 2009]) file, either uploaded by the user or created by an analysis tool. An index is created for the file to retrieve information faster.
- DAS-track: sites (features) are retrieved from the web-service via DAS protocol.
- Client track: track data is retrieved from a remote BioUML server.
- Virtual track: track based on existing track(s) with some transformation applied:
  - Filtered track: track where some of sites are removed according to condition.
  - Resized track: track where sites are shrunk or enlarged to some extent.

## Model

A set of interfaces is used to encapsulate tracks and sequences implementation. The main interfaces here are Sequence and Track, which can be used to obtain sequence data (nucleotide letters) and track data (sites information) in a convenient way. The Site interface represents a single feature on the track. The Site and Track interfaces with some of the implementing classes are shown in Fig. 2.



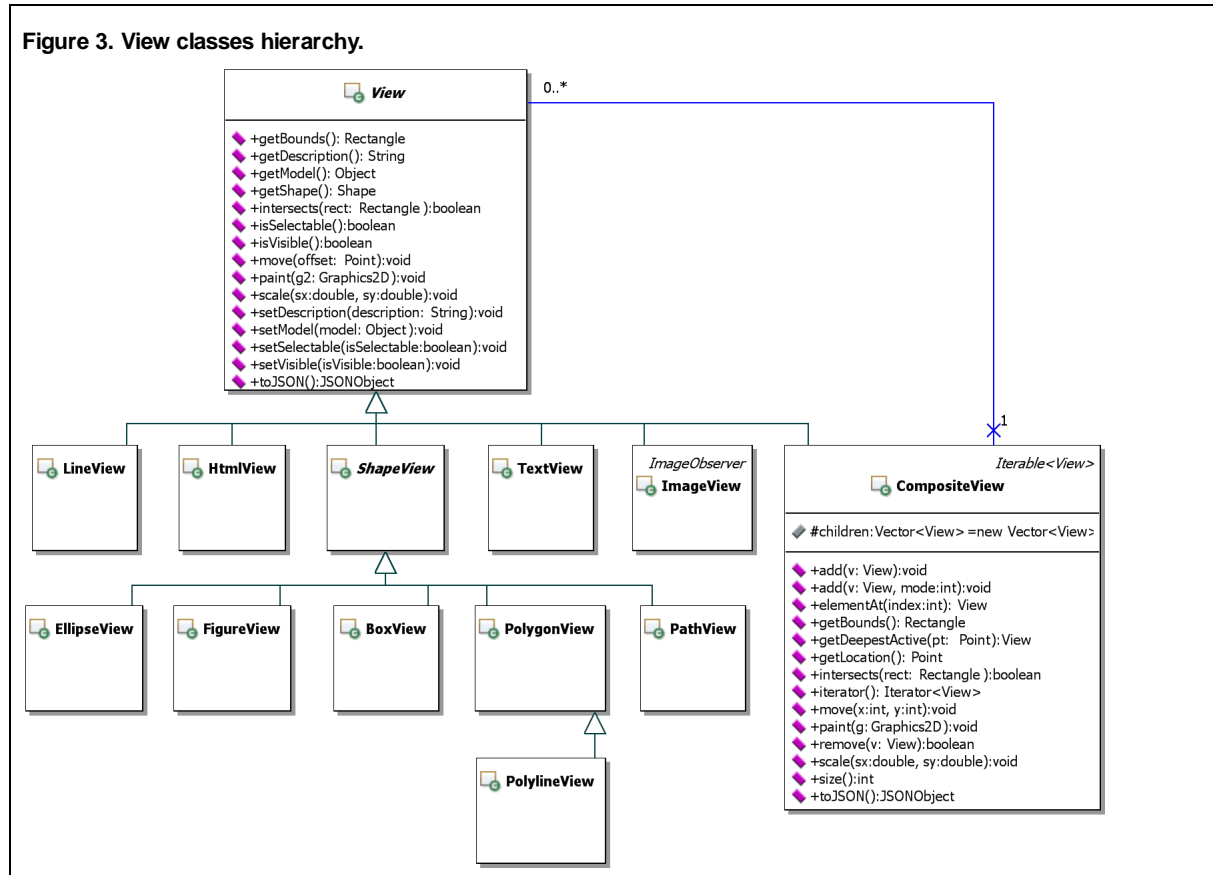
The Track interface provides a number of methods to retrieve either collection of all sites or sites located in a specific region of a specific sequence (chromosome). The SlicedTrack abstract class supports partial loading of track data from the back-end. The WritableTrack interface allows adding sites to the track. Two WritableTrack implementations are available: SqlTrack, which serializes the track data into MySQL database, and TrackImpl, which stores all the sites in the memory (this can be useful inside some analyses). ClientTrack is used as a proxy for Track located on a remote BioUML server.

The Site interface allows retrieving different site information such as sequence and coordinates where the site is located, strand, site type and so on. Any custom information associated via site can be retrieved using the getProperties() method.

There is also TrackViewBuilder class which can build a view of a specific region of a track given specific visualization options. Most tracks use the default builder, but the custom builder is necessary sometimes to display special tracks like karyotype.

## View

To store and display graphical representation of the selected genomic region, the View class hierarchy is used. This hierarchy represents a number of graphical primitives as well as the CompositeView class, which can combine several views together. The class diagram is shown in Fig. 3.



The View abstract class defines basic methods for all graphical elements including manipulation methods (move, scale, setModel, setSelectable), informational methods (getBounds, isSelectable, getModel), rendering (paint) and the serialization method (toJSON). The CompositeView class is a type of View which may contain other views.

Created View can be either painted (in the BioUML workbench) or serialized into JSON and transferred to the client (web-browser). On the client side View is deserialized into hierarchy of JavaScript objects and painted into HTML5 canvas.

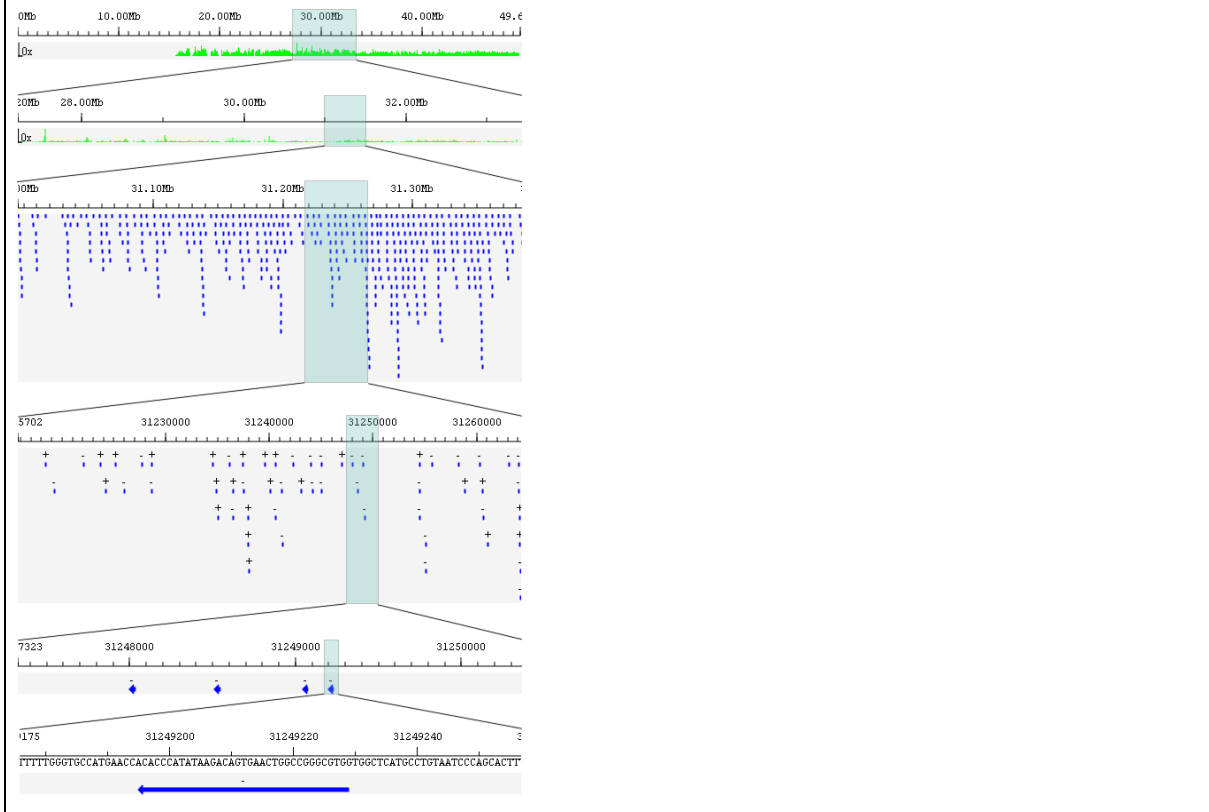
A model can be associated with an individual view. In our case it allows to bind the view representing an individual site with the site itself. Thanks to this the user can just select a site to get all the annotation associated with it.

## Features

### Semantic zoom

To make the genome browser more user-friendly the semantic zoom feature is implemented. Depending on the current zoom level and number of visible sites a different view is displayed for the user as shown in Fig. 4.

**Figure 4. Semantic zoom capabilities of BioUML genome browser (web edition).** The coverage profile can be seen when a whole chromosome is displayed. On whole chromosome view coverage profile is displayed. Upon zooming (in) individual reads become visible. Further zooming adds feature titles (strand direction in this case), then arrow-like view of features. Finally the nucleotide sequence appears.



Here a BED file uploaded by the user is used as an example track. The BED file contains aligned reads of a ChIP-Seq experiment performed on FoxA1 protein. The genome browser in Fig. 4. displays Homo sapiens chromosome 22 (NCBI36 genome build). As you can see, the genome browser is capable of displaying this track in any scale from per-nucleotide to chromosome-wide. Upon zooming in the coverage profile view is changed to the individual reads view. Further zooming adds reads' titles (strand direction is used as the read title in the example) then reads become displayed as arrows. On the most detailed scale the nucleotide sequence becomes visible.

## Asynchronous track loading

In BioUML web edition the genome browser supports asynchronous track loading. Some tracks may be loaded and displayed slowly depending on their source, requested region and settings applied. For example, a DAS track can work slowly due to network latency or delays on the remote DAS server. Thanks to asynchronous track loading genome browser doesn't stall because of slow tracks: while they are being loaded, the BioUML genome browser displays other tracks with all the functionality working.

## Results

### BioUML web edition genome browser

The genome browser is available as a part of the BioUML platform. The web-version is freely available on <http://ie.biouml.org/bioumlweb/>. Users may either press "Demo" button for anonymous access or register to get a personal project folder.

Example genome browser views are available in the "Data" tree tab, under "data/Examples/GenomeBrowser/Data/Views". You may use a shortcut URL to get right there: <http://ie.biouml.org/samples/gb>. Fig. 5. shows a screenshot of this demonstration.

**Figure 5. BioUML genome browser in action. Homo sapiens chromosome 1 (NCBI36) is displayed. Three Ensembl tracks are added (Karyotype, GC content, Genes) along with one DAS-track (CytoChip3).**

The screenshot shows the BioUML genome browser interface. The main display area shows a genomic track for chromosome 1 (NCBI36) with the following tracks: Karyotype, GC content, Genes, and CytoChip3. The Karyotype track shows bands for p36.33, p36.32, and p36.31. The GC content track shows a zoomed-in view of the GC content with a text box indicating 'Zoom in to see GC content' and a position of 4 710 000. The Genes track shows various gene models, including PRDM16, RP4-785P20, RP11-168B8, RP4-703E10, RP1-120G22, CAMTA1, RP1-283E3, RP3-395M20, RP11-46F15, RP1-37J18, RP11-49J3, RP11-242F24, RP1-140A9, RP11-333E3, RP1-286D6, and RP11-58A11. The CytoChip3 track shows various microarray sites, including 5-902P8, RP11-181G12, RP4-785P20, RP11-168B8, RP4-703E10, RP1-120G22, RP1-283E3, RP3-395M20, RP11-46F15, RP1-37J18, RP11-49J3, RP11-242F24, RP1-140A9, RP11-333E3, RP1-286D6, and RP11-547D24.

The bottom-left panel shows the information for the selected site (RP4-785P20:3214521:3355092). The information includes the Site ID, Type (CytoChip3), Sequence name (1), Position (3214521 - 3355092 (140572)), and Properties (Category, Label, Orientation, Type, Type name).

The bottom-right panel shows the list of visible sites on the selected track. The list is as follows:

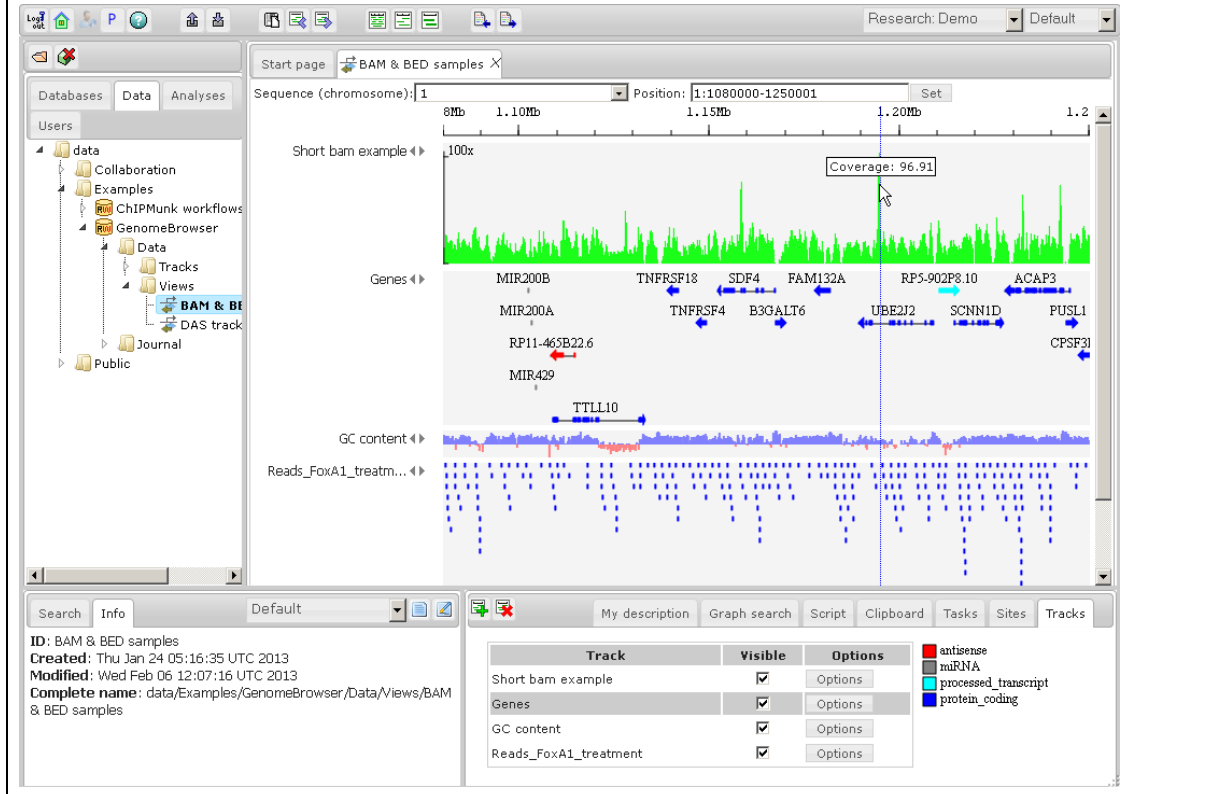
ID	Sequence (chromosome) name	From	To	Length	Strand	Type	Property: name	Property: stain
1	1	1	2300000	2300000	x	p36.33	p36.33	gneg
2	1	2300001	5300000	3000000	x	p36.32	p36.32	gpos25
3	1	5300001	7100000	1800000	x	p36.31	p36.31	gneg

This stored view contains the chromosome 1 sequence from Ensembl database which is located at “databases/EnsemblHuman52”. Three tracks are added from the same Ensembl database. A Fourth track is the DAS-track, which queries data from EBI genomic DAS-server (<http://www.ebi.ac.uk/das-srv/genomicdas>) using DAS source “CytoChip v3”. Some of the available interactive features are listed below:

- Click and drag with your mouse to navigate along the track.
- Select a range with the mouse while holding Ctrl or Shift key to zoom into this range.
- Select a sequence in “Sequence (chromosome)” drop-down list or type a new position in the “Position” box to navigate into the new range.
- Use the mouse wheel while holding Ctrl key to zoom in or out.
- Use “Semantic zoom out” and “Semantic zoom in” toolbar buttons to zoom in or out.
- Move the mouse pointer over the genome browser display to get context-sensitive information like current position or site name.
- Click on an individual site to get the site information in the Info box in the bottom-left corner.
- Drag track names up and down to rearrange them.
- Drag tracks from the tree to the genome browser to add them into the view.
- Use “Sites” tab to see the list of all visible sites on the selected track along with the annotation.
- Use “Tracks” tab to rename or remove tracks, setup their options and see the legend.

Another sample displaying a BAM-track and an SQL-track (user loaded BED file) is located in “data/Examples /GenomeBrowser/Data/Views/BAM & BED samples”. Navigate to this location in the tree and then double-click to open it. This view is shown in Fig. 6.

**Figure 6. BAM file (Short bam example) and BED file (Reads\_FoxA1\_treatment) visualization in the genome browser. Genes and GC content are Ensembl tracks.**



Here Homo sapiens (GRCh37) chromosome 1 is displayed from Ensembl database. The first track called “Short bam example” is a user-uploaded BAM-file. At the current zoom-level it’s displayed in the profile mode. Next two tracks are loaded from Ensembl and the last track called “Reads\_FoxA1\_treatment” is a user-loaded BED file which was also displayed in Fig. 4.

To create your own genome browser view you should first select the genome. You may navigate, for example, into “databases/Ensembl/Sequences/chromosomes GRCh37” and double-click on this folder or on the chromosome name. Alternatively you may upload your own sequences (for example, in FASTA format) and open them. Anonymous users have a shared project which is located under “data/Collaboration/Demo/Data”. You may import your data into this location.

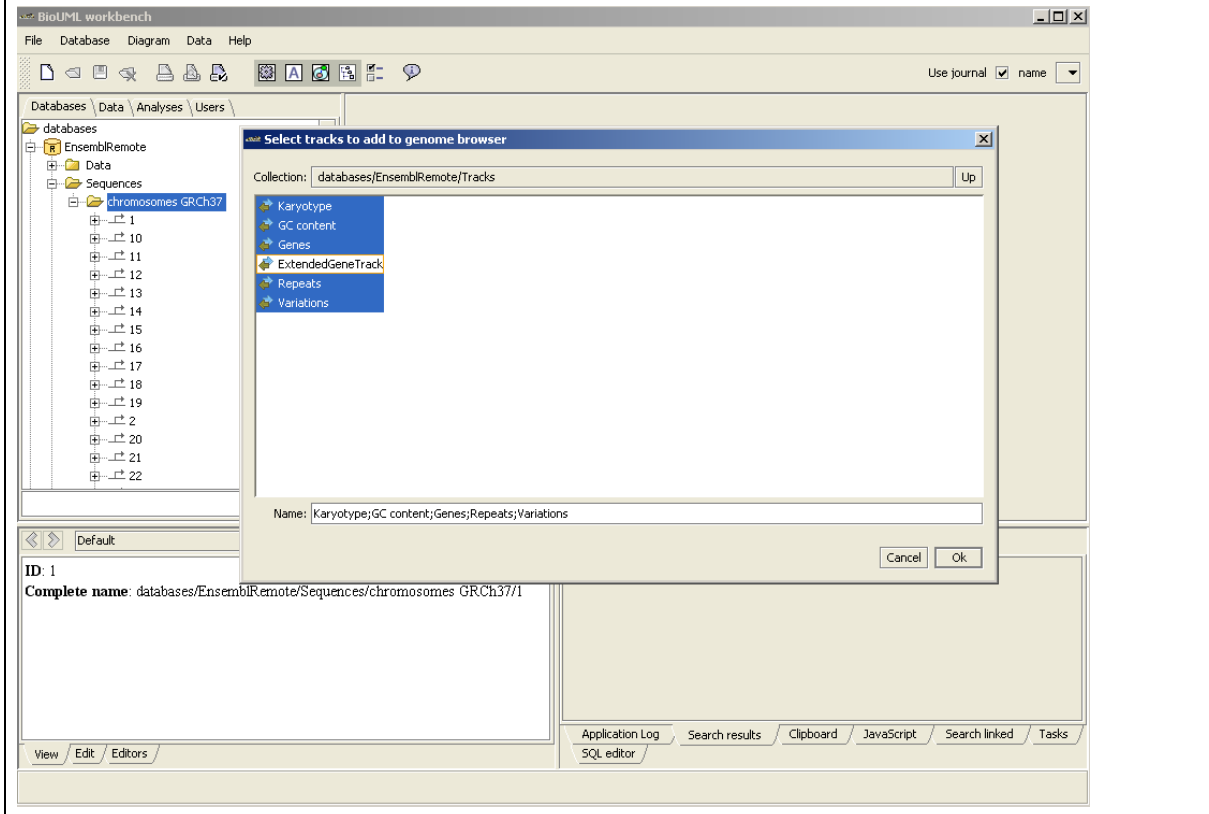
After opening a sequence you may be prompted to add some tracks associated with it. You can always add more tracks later simply dragging them from the tree. To save the resulting view you can press the “Save view” button on the toolbar.

## BioUML workbench genome browser

The BioUML workbench can be downloaded and installed from the BioStore service (<https://bio-store.org/>). After free registration and login navigate into “Downloads” area and download the BioUML workbench installer from there. Please note that Java 1.6 or later is required for the BioUML workbench to work correctly.

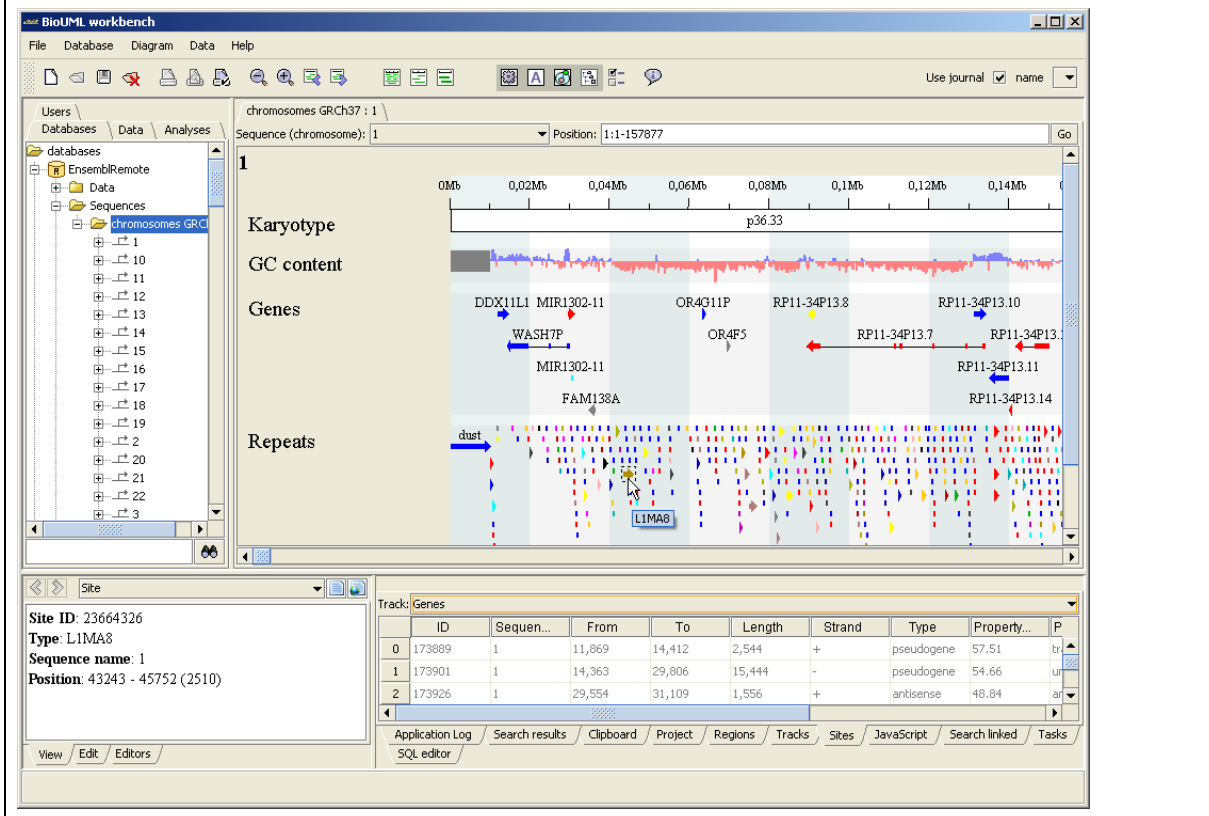
Upon successful installation you can launch the BioUML workbench. It has preinstalled “EnsemblRemote” collection, which is connected to the “databases/Ensembl” collection on ie.biouml.org BioUML server. Open it, navigate into “Sequences” folder and double-click on “chromosomes GRCh37”. The Dialog box will appear asking you which tracks you want to add to the genome browser as it’s displayed in Fig. 7. Select some of them (use Ctrl or Shift to select several) and press “Ok”.

**Figure 7. BioUML workbench.** To open genome browser navigate into “EnsemblRemote/Sequences” in the “Databases” tab, then double-click on “chromosomes GRCh37”. You will be asked to select tracks.



The Genome browser will be opened on the first chromosome as it's displayed in Fig. 8. Most of the functions work in the same manner as for BioUML web edition: you can navigate through the sequence, switch to another chromosome, obtain sites information, and so on. The “Sites” and “Tracks” tabs below the genome browser are similar to those on the web.

**Figure 8. BioUML workbench genome browser.**



## Comparison

Genome browsers can be classified in a number of ways. One of the main characteristics is the application type: either



standalone or web. Web-based genome browsers differ in how they render the graphics: images can be generated on server or client-side.

Another interesting characteristic is the list of data formats supported. Currently there are many track data formats including BED, GFF, GTF, VCF, Wiggle, BigWig, SAM, BAM and so on. To make a genome browser convenient most of these formats should be supported. Other data sources are also interesting: DAS-tracks, locally installed NCBI or Ensembl database and so on.

Some genome browsers allow the user to display custom sequences. This may be crucial for users working with custom genome builds or with genomes not available in public databases.

For BAM files it's important which visualization features are supported: coverage (profile) visualization, reads visualization, phred quality profiles and variations (SNP) visualization. One more important BAM-related feature is automatic indexing. Although there is a separate tool to index BAM-files in samtools package [Li et al, 2009], automatic indexing makes it easier to upload BAM-files.

Sometimes it's necessary to get the tabular representation of the features displayed along with their coordinates and annotation. Some genome browsers support this feature.

Table 1 shows a feature comparison between the BioUML genome browser and some other well-known genome browsers including web-based and desktop ones.

**Table 1. Genome browsers comparison.**

<sup>1</sup> cannot be uploaded, must be prepared manually on the server side via special Perl script.

<sup>2</sup> cannot be uploaded, only accessed by URL.

<sup>3</sup> soft-clipped reads are displayed incorrectly.

<sup>4</sup> via separate application called "Table Browser".

<sup>5</sup> user uploaded tracks cannot be shared.

<sup>6</sup> view (session) can be saved into file and passed along with user tracks to a colleague.

Software title		BioUML genome browser	Ensembl genome browser	UCSC genome browser	JBrowse	Integrative Genomics Viewer	Integrated Genome Browser
Type		Web-client and desktop	Web-server	Web-server	Web-client	Desktop	Desktop
Implementation language		Java, JavaScript	Perl	Perl, Java, C	JavaScript, Perl	Java	Java
Data sources	Main backend database	Ensembl	Ensembl	UCSC	Perl Bio::DB packages	Broad Institute hosted genomes (UCSC, NCBI, Ensembl, etc.)	IGB Quickload sites
	User sequences support (FASTA, etc.)	+	-	-	+ <sup>1</sup>	+	+
	User tracks upload/import from URL	+	+	+	+	+	+
	DAS tracks support	+	+	-	-	+	+
Track formats support	BED	+	+	+	+ <sup>1</sup>	+	+
	VCF	+	+	+	-	+	+
	GFF	+	+	+	+	+	+
	WIG	+	+	+	+ <sup>1</sup>	+	+
	BigBed	-	+	+	-	+	+
	BigWig	-	+	+	+	+	+
Working with BAM files	BAM support	+	+ <sup>2</sup>	+ <sup>2</sup>	+	+	+
	Index BAM files automatically	+	-	-	-	-	-
	Coverage visualization	+	+	-	+	+	+
	Reads visualization	+	+	+	+ <sup>3</sup>	+	+
	Phred quality visualization	+	-	-	-	-	-
	Variations visualization	+	+	+	+	+	+
User interface features	Drag and scroll	+	-	+	+	+	+
	Zoom selection	+	+	+	+	+	+
	Tabular view	+	-	+ <sup>4</sup>	-	-	-
	Share view	+	+	+	+ <sup>5</sup>	+ <sup>6</sup>	+ <sup>6</sup>

## Conclusion

Fully functional BioUML genome browser was created and proved to be useful in genomic research. For analysis results obtained in BioUML it's much more convenient to visualize them in the BioUML genome browser than in any external genome browser. Most of the features available in other modern genome browsers are supported by the BioUML genome browser as well.

The BioUML genome browser is capable of working with very large tracks. For example, Ensembl Human version 65 Variation track contains more than 40,000,000 individual variations. User uploaded BED and VCF files containing more than 50,000,000 were tested successfully. BAM files containing more than 500,000,000 aligned reads were also visualized without any problems. Thus the BioUML genome browser is suitable for NGS data as well.

## Application to virtual biology

Genome browser is an essential tool for NGS data analysis, including whole genome analysis (personal genome interpretation and personalized medicine) and ChIP-seq, transcriptome, ribo-seq data analysis (virtual cell construction).

## References

- Bare J C, Koide Tie, Reiss David J, Tenenbaum Dan, Baliga Nitin S, authors. **Integration and visualization of systems biology data in context of the genome.** BMC Bioinformatics. 19–7;2010;11:382 DOI:10.1186/1471-2105-11-382 [PMID:20642854]
- Carver Tim, Harris Simon R, Berriman Matthew, Parkhill Julian, McQuillan Jacqueline A, authors. **Artemis: an integrated platform for visualization and analysis of high-throughput sequence-based experimental data.** Bioinformatics. 22–12;2011;(4)28:464–469. DOI:10.1093/bioinformatics/btr703 [PMID:22199388]
- Chen Ken, Wallis John W, McLellan Michael D, Larson David E, Kalicki Joelle M, Pohl Craig S, McGrath Sean D, Wendl Michael C, Zhang Qunyan, Locke Devin P, Shi Xiaoqi, Fulton Robert S, Ley Timothy J, Wilson Richard K, Ding Li, Mardis Elaine R, authors. **BreakDancer: an algorithm for high-resolution mapping of genomic structural variation.** Nat Methods. 9–8;2009;(9)6:677–681. DOI:10.1038/nmeth.1363 [PMID:19668202]
- Cock Peter J A, Fields Christopher J, Goto Naohisa, Heuer Michael L, Rice Peter M, authors. **The Sanger FASTQ file format for sequences with quality scores, and the Solexa/Illumina FASTQ variants.** Nucleic Acids Res. 16–12;2009;(6)38:1767–1771. DOI:10.1093/nar/gkp1137 [PMID:20015970]
- Engels Reinhard, Yu Tamara, Burge Chris, Mesirov Jill P, DeCaprio David, Galagan James E, authors. **Combo: a whole genome comparative browser.** Bioinformatics. 18–5;2006;(14)22:1782–1783. DOI:10.1093/bioinformatics/btl193 [PMID:16709588]
- Goecks Jeremy, Nekrutenko Anton, Taylor James, authors. **Galaxy: a comprehensive approach for supporting accessible, reproducible, and transparent computational research in the life sciences.** Genome Biol. 25–8;2010;(8)11: DOI:10.1186/gb-2010-11-8-r86 [PMID:20738864]
- Hubbard T, Barker D, Birney E, Cameron G, Chen Y, Clark L, Cox T, Cuff J, Curwen V, Down T, Durbin R, Eyras E, Gilbert J, Hammond M, Huminciecki L, Kasprzyk A, Lehvaslaiho H, Lijnzaad P, Melsopp C, Mongin E, Pettett R, Pockock M, Potter S, Rust A, Schmidt E, Searle S, Slater G, Smith J, Spooner W, Stabenau A, Stalker J, Stupka E, Ureta-Vidal A, Vastrik I, Clamp M, authors. **The Ensembl genome database project.** Nucleic Acids Res. 1–1;2002;(1)30:38–41. [PMID:11752248]
- Jenkinson Andrew M, Albrecht Mario, Birney Ewan, Blankenburg Hagen, Down Thomas, Finn Robert D, Hermjakob Henning, Hubbard Tim J P, Jimenez Rafael C, Jones Philip, Kähäri Andreas, Kulesha Eugene, Macías José R, Reeves Gabrielle A, Prlić Andreas, authors. **Integrating biological data--the Distributed Annotation System.** BMC Bioinformatics. 22–7;2008;9 Suppl 8: DOI:10.1186/1471-2105-9-S8-S3 [PMID:18673527]
- Kent W J, Sugnet Charles W, Furey Terrence S, Roskin Krishna M, Pringle Tom H, Zahler Alan M, Haussler David, authors. **The human genome browser at UCSC.** Genome Res. 2002;(6)12:996–1006. DOI:10.1101/gr.229102. Article published online before print in May 2002 [PMID:12045153]
- Koboldt Daniel C, Chen Ken, Wylie Todd, Larson David E, McLellan Michael D, Mardis Elaine R, Weinstock George M, Wilson Richard K, Ding Li, authors. **VarScan: variant detection in massively parallel sequencing of individual and pooled samples.** Bioinformatics. 19–6;2009;(17)25:2283–2285. DOI:10.1093/bioinformatics/btp373 [PMID:19542151]
- Kolpakov F A, Valeev T F, Tolstyh N I, Kiselev I N, YuV Kondrakhin, Kutumova E O, Yevshin I S, authors. **BioUML - open source integrated platform for building virtual cell and virtual physiological human.** Virtual Biology. 2013;(1) DOI:10.12704/vb/e12
- Langmead Ben, Trapnell Cole, Pop Mihai, Salzberg Steven L, authors. **Ultrafast and memory-efficient alignment of short DNA sequences to the human genome.** Genome Biol. 4–3;2009;(3)10: DOI:10.1186/gb-2009-10-3-r25 [PMID:19261174]
- Li Heng, Handsaker Bob, Wysoker Alec, Fennell Tim, Ruan Jue, Homer Nils, Marth Gabor, Abecasis Goncalo, Durbin Richard, authors. **The Sequence Alignment/Map format and SAMtools.** Bioinformatics. 8–6;2009;(16)25:2078–2079. DOI:10.1093/bioinformatics/btp352 [PMID:19505943]
- Nicol John W, Helt Gregg A, Blanchard Steven G Jr, Raja Archana, Loraine Ann E, authors. **The Integrated Genome Browser: free software for distribution and exploration of genome-scale datasets.** Bioinformatics. 4–8;2009;(20)25:2730–2731. DOI:10.1093/bioinformatics/btp472 [PMID:19654113]
- Skinner Mitchell E, Uzilov Andrew V, Stein Lincoln D, Mungall Christopher J, Holmes Ian H, authors. **JBrowse: a next-generation genome browser.** Genome Res. 1–7;2009;(9)19:1630–1638. DOI:10.1101/gr.094607.109 [PMID:19570905]
- Thorvaldsdóttir Helga, Robinson James T, Mesirov Jill P, authors. **Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration.** Brief Bioinform. 19–4;2012;(2)14:178–192. DOI:10.1093/bib/bbs017 [PMID:22517427]
- Ye Kai, Schulz Marcel H, Long Quan, Apweiler Rolf, Ning Zemin, authors. **Pindel: a pattern growth approach to detect break points of large deletions and medium sized insertions from paired-end short reads.** Bioinformatics. 26–6;2009;(21)25:2865–2871. DOI:10.1093/bioinformatics/btp394 [PMID:19561018]