File Formats

IGB supports multiple file formats in both compressed and uncompressed formats. See the table below for details and links (when available) to resources describing each format. IGB uses file extensions to recognize file formats, listed below.

> Third-party IGB Apps may support additional formats.

- Supported file formats
- About GFF and its variants
- About bedGraph
- Partial data loading using tabix indexed files
- Sequence File Formats
- .egr and .sin Formats

Supported file formats

Type	Extension	Description
Affymetrix XML	.axml	A mostly-obsolete XML format used internally at Affymetrix.
BAM	.bam	A binary indexed version of the SAM format used for displaying alignment data. See SAMtools for more details. **Note: Be sure you've indexed your BAM file** (you should have a .bai file as well). The index file must reside in the same folder as the BAM file unless the location is indicated using the "index" attribute in an IGB Quickload annots.xml configuration file. See About annots.xml.
SAM	.sam	Plain text version of BAM format. Index not required. Recommended for smaller files only.
BAR	.bar	Binary graph format developed by Affymetrix. Generated from tiling arrays by TAS (Tiling Analysis Software) from Affymerix, cis Genome from Hongkai Ji's research group, and others.
BED	.bed	A tabular format developed for the UCSC genome browser. IGB supports four, twelve, and fourteen column BED format. In IGB, the thirteenth and fourteenth columns of fourteen-column BED format (also called BED detail format) are interpreted as title and description, respectively.
BEDGRAPH	.bedgraph	Same as the wiggle format. See below for details.
BigBED	.bigbed	The bigBed format stores annotation items that can either be simple, or a linked collection of exons, much as bed files do. BigBed files are created initially from bed type files, using the program `bedToBigBed`. The resulting bigBed files are in an indexed binary format. The main advantage of the bigBed files is that only the portions of the files needed to display a particular region are loaded into IGB, so for large data sets bigBed is considerably faster than regular bed files. See http://genome.ucsc.edu/goldenPath/help/bigBed.html.
BigWIG	.bigwig	Like the bigBED format, this is an indexed form of a WIG file that facilitates incremental data loading and faster loading than the non-indexed, plain text version of the format. See http://genome.ucsc.edu/goldenPath/help/bigWig.html.
BGR	.bgr	Binary graph format developed by Affymetrix.
BNIB	.bnib	Binary format for sequence data originally developed for IGB by Affymetrix to speed up loading sequence data over the network. Replaced by 2bit as of IGB 7.
CRAM	.cram	A more highly compressed version of the SAM and BAM file formats. **Note: Be sure you've indexed your CRAM file** (you should have a .crai file as well). The index file must reside in the same folder as the CRAM file unless the location is indicated using the "index" attribute in an IGB Quickload annots.xml configuration file. See About annots.xml. **Note: Visualizing CRAM data in IGB requires that you use the EXACT same genome used to align your data.** If that genome is not currently available in IGB, see Custom Genomes (Genomes not in IGB).
Cytoband	.cyt	Text format for representing chromosome band (ideogram) data. Examples are available from the IGBQuickLoad.org site under human genome directories.
DAS XML files	.das, .dasxml, .das2xml	XML formats returned from DAS servers. See http://www.biodas.org. See DAS/1 specification and DAS/2 specification
Expression Graphs	.egr, .egr.txt, .sin	EGR is a tabular format representing scored genomic intervals. Files generated from Affymetrix GeneChip Operating Software (GCOS) or ExACT (Exon Array Computational Tool) software. See below for details.

FASTA	.fa, .fasta, .fna, .fsa, .mpfa, .fas	Sequence data in a simple ASCII format. For larger sequence files (e.g., the human genome) use 2Bit. See Sharing data for a custom genome not already part of IGB QuickLoad.
GenBank	.gb, .gen	NCBI's file format. IGB has limited support for GenBank files.
GFF (General Feature Format)	.gff, gtf, .gff3	General Feature Format. There are several types of GFF file that use incompatible syntax. The original GFF format is GFF1. A variant called GTF is also used. GFF3 has been proposed to extend on GFF and to constrain the specification more tightly to avoid mutually-incompatible versions of GFF. If IGB has difficulty reading your GFF file, make sure the header includes the GFF version, as indicated in the GFF specification documents.
GR	.gr	Tab-delimited graph format. A simple text format containing two columns of numbers separated by a single space or tab. The first number is the base position; the second number is the score. Because this format does not include chromosome names, we recommend you use .sgr or .wig formats instead.
PSL	.psl, .psl3,	PSL is a tabular format used for representing alignments in UCSC's BLAT tool.
Link.psl	.link.psl	Link.psl represents alignments of Affymetrix target sequences and the location of probe set probes within those sequences. Used to display genomic alignments of Affymetrix probe sets. Ann Loraine wrote some python code for creating link.psl files. See https://bitbucket.org/lorainelab/affyprobesetsforigb and Visualizing probe sets
PSLX	.pslx	PSLX is an extension to the PSL format that includes the aligned sequence. Aligned sequences are displayed similar to BAM files.
Scored Intervals	.sin, .egr, .egr.txt	See below for details,
Scored Map	.map	An outdated format, replaced now by EGR files.
SGR	.sgr	Tab-delimited graph format. Sequence graph files that show base coordinate scores. These files are generated by CNAT (the Affymetrix Chromosome Copy Number Analysis Tool software). The format of .sgr text files is: chromosome identification, then two columns of numbers separated by a single space or a tab. The first number is the base position; the second number is the score
TALLY	.tally	Tally files are created by the bam_tally program (using options -P -B 0), and contain mismatch pileup information. The display is identical to the MisMatch Pileup view mode. The Tally files contain the sequence reference. The plugin will use a tabix index if available.
USeq	.useq	USeq is a binary indexed format used to display graph and annotation data. Supported in IGB 9.1.8 and earlier. For more information about it, see: http://useq.sourceforge.net.
VCF	.vcf	Variant Call Format (VCF) is a flexible and extendable format for variation data such as single nucleotide variants, insertions/deletions, copy number variants and structural variants. More information on the VCF file format can be found here: https://github.com/samtools/hts-specs
Wiggle	.wig	This is a text format for graphical data designed for the UCSC genome browser. IGB supports all 3 subtypes: BED, variableStep, fixedStep. For more information, see the UCSC Web page describing the format: http://genome.ucsc.edu/goldenPath/help/wiggle.html. Files in wiggle format can use UCSC track lines to specify colors and other properties.
2Bit	.2bit	2Bit is a compact format for DNA sequences developed by UCSC. See http://genome.ucsc.edu/FAQ/FAQformat.html#format7 for more information about it.

About GFF and its variants

GFF stands for 'general feature format' or 'gene finding format'; it is a tab-delimited file with 9 columns. There are several types of GFF files that use incompatible syntax. The original GFF format is GFF1. A variant called GTF is also used. GFF3 has been proposed to extend on GFF and to constrain the specification more tightly to avoid mutually-incompatible versions of GFF. Some GFF files created by Affymetrix make use of extensions to GFF that are specific to IGB. These are indicated in the file headers by lines beginning with "##IGB-".

IGB can handle most versions of GFF/GTF, but may have difficulty with some rarely-used advanced features. IGB does not read any FASTA data that is included in some GFF3 files. If IGB has difficulty reading your GFF file, make sure that there is a line in the header similar to **`##gff-version 2`** that identifies the correct format number 1, 2 or 3.

The GFF format is described at http://www.sanger.ac.uk/Software/formats/GFF/GFF_Spec.shtml

The GTF format is described here http://genes.cs.wustl.edu/GTF2.html

The GFF3 format is described here http://song.sourceforge.net/gff3-jan04.shtml

About bedGraph

A bedGraph file associates numerical values (e.g., read coverage) with regions of a genome assembly. Note that the data for an entire genome can reside in a single file. If using this format, you should sort it and index it using tabix for faster and more memory-efficient data loading.

```
track type=bedGraph name="Pollen RNA-Seq coverage"
chr1        0        3722        0
chr1        3722        3797        1
chr1        3797        5890        0
chr1        5890        5893        4
chr1        5893        5897        5
chr1        5897        5902        6
chr1        5902        5910        7
chr1        5910        5939        8
chr1        5939        5944        9
chr1        5944        5957        10
```

Note that the top line of the file contains information (meta-data) about the data set, including its name. When you open the file in IGB, the name will appear as the track label.

Partial data loading using tabix indexed files

IGB supports partial data loading of several file types using tabix indexing. Supported file types include SAM, BED, BEDGRAPH, PSL, and PSLX.

Indexed files allow for faster searching and loading. The indexed file and its index (.tbi file) must reside in the same folder either on your local computer or on a server. More about tabix can be found here.

Sequence File Formats

IGB supports fasta and 2bit formats. Older versions of IGB also support an IGB-specific format called bnib. Newer versions of IGB will probably still open bnib files, but as of IGB 7.0, we are no longer including the bnib format in our testing process.

FASTA files contain sequence data in a simple ASCII format. For details, Google search fasta.

> ⚠ We recommend using FASTA for short sequences only. For loading data into IGB and setting up a QuickLoad site, we use the 2bit sequence format.

BNIB is an older format developed for IGB that makes it possible to represent sequence data in a very compact format. 2bit, developed at UCSC is also a compact, binary format for representing sequence data, but a number of open source tools are available for working with this format and so for this reason, IGB now uses 2bit instead of bnib.

.egr and .sin Formats

EGR (also known as Scored Interval, .sin, format.files) are TAB-delimited files with a header. They can contain one or more scores associated with named annotations or with named or unnamed genomic regions. They have an optional header section which is a list of tag-value pairs, one per line, in the form: **# tag = value** Currently the only tags used by the parser are of the form **score$i** (score name tags are optional). If score name tags are present, then score number **$i** will be named according to the value of the **score$i** tag. If any score name tags are missing, default names will be created.

It is recommended that a tag value pair with the genome version, such as **#genome_version = H_sapiens_May_2004** , be included to indicate which genome assembly the sequence coordinates are based on. This will ensure that the file is being compared to other annotations from the same assembly.

There are three versions of this format. They can all be described this way, where the parentheses indicate optional elements:(annot_id) ((seqid) min_coord max_coord strand) [score]*

1. `seqid` is word string [a-zA-Z_0-9]+
2. `min_coord` is int
3. `max_coord` is int
4. `strand` can be '+', '-', or '.' for "unknown"
5. `score` is float
6. `annot_id` is word string [a-zA-Z_0-9]+

All lines must have same number of columns. Format 1 has tab-delimited lines with 4 required columns, any additional columns are scores:seqid min_coord max_coord strand [score]* Format 2 has tab-delimited lines with 5 required columns, any additional columns are scores:annot_id seqid min_coord max_coord strand [score]* Format 3 has tab-delimited lines with 1 required column, any additional columns are scores:annot_id [score]* The IGB parser should be able to distinguish between these, based on combination of number of fields, and presence and position of the strand field. For use in IGB, EGR version 3 is dependent on prior loading of annotations with matching ids.

Examples

- Format 1:# genome_version = H_sapiens_Apr_2003

1. score0 = A375
2. score1 = FHS
   gene1 chr22 14433291 14433388 + 140.642 175.816
   gene2 chr22 14433586 14433682 + 52.3838 58.1253
   gene3 chr22 14434054 14434140 + 36.2883 40.7145

- Format 2:# genome_version = H_sapiens_Apr_2003

1. score0 = A375
2. score1 = FHS
   chr22 14433291 14433388 + 140.642 175.816
   chr22 14433586 14433682 + 52.3838 58.1253
   chr22 14434054 14434140 + 36.2883 40.7145

- Format 3:(assumes annotations with the names gene1, gene2, and gene3 are already loaded.)# genome_version = H_sapiens_Apr_2003

1. score0 = A375
2. score1 = FHS
   gene1 140.642 175.816
   gene2 52.3838 58.1253
   gene3 36.2883 40.7145