Using GBrowse 2.0 to visualize and share next-generation sequence data

Lincoln D. Stein

Submitted: 27th September 2012; Received (in revised form): 18th December 2012

Abstract
GBrowse is a mature web-based genome browser that is suitable for deployment on both public and private web sites. It supports most of genome browser features, including qualitative and quantitative (wiggle) tracks, track uploading, track sharing, interactive track configuration, semantic zooming and limited smooth track panning. As of version 2.0, GBrowse supports next-generation sequencing (NGS) data by providing for the direct display of SAM and BAM sequence alignment files. SAM/BAM tracks provide semantic zooming and support both local and remote data sources. This article provides step-by-step instructions for configuring GBrowse to display NGS data.

Keywords: bioinformatics; genomics; DNA sequencing; genome browser; data visualization; data sharing

INTRODUCTION
GBrowse was among the first web-based genome browsers [1] and was the first to be widely used outside its site of origin. Originally developed for use with WormBase (www.wormbase.org), it was released as a standalone project in January 2002 and has continued to develop on a steady basis for the past decade. Support for next-generation sequencing (NGS) data was introduced in version 2.0, released in January 2010. GBrowse supports both DNA-seq and RNA-seq NGS alignments and can display the data at multiple resolutions from a whole-chromosome coverage histogram to individual base pairs (Figure 1). NGS data can be uploaded directly to the browser, linked to via a URL or manually added to the server. Uploaded and linked sequencing data can be made public or shared selectively with collaborators. Version 2.0 also added a new client-side architecture that enhanced the browser’s interactivity and performance.

GBrowse is intended for environments in which groups wish to display and share genome annotations in a format that can be accessed casually without preinstallation of desktop software. Hence, it is suitable for installation on public web sites, as well as the web sites of small-to-medium collaborations among several geographically separate groups. It is particularly well suited to collaborative environments in which some annotation tracks are public while others are restricted to individuals or groups, as GBrowse provides a highly configurable track-level security model that is able to integrate with a variety of popular enterprise authentication systems (gmod.org/wiki/GBrowse_Configuration/Authentication).

Although it can be used as a single user’s desktop genome browser, GBrowse is not as convenient for this purpose as IGV or other desktop genome browsers. Public sites that use GBrowse include WormBase, COSMIC (www.sanger.ac.uk/perl/genetics/CGP/cosmic), modENCODE (www.modencode.org), the human HapMap project (www.hapmap.org), BeeBase (www.beebase.org), FlyBase (flybase.org), the Database of Genetic Variants (projects.tcag.ca/variation) and many others.

Although it can be used on its own, GBrowse integrates well with the other bioinformatics tools in the Generic Model Organism Database.
These include Chado [2], a database schema for genomic data, several genome synteny browsers [3–5], the Galaxy workflow engine [6], the Apollo genome editor [7], the MAKER genome annotation pipeline [8] and the BioMart federated data mining engine [9].

GBrowse is well supported by a mailing list, a WIKI, a help desk and both physical and online tutorials. As of 2012, major new features were not being added to GBrowse and development prioritized bug, performance and stability fixes. Instead, new development efforts are going to JBrowse [10], GBrowse’s designated replacement in the GMOD suite. JBrowse, which uses a pure client-side architecture, provides a much improved user experience over GBrowse, but does not yet support all of GBrowse’s features.

GBROWSE TECHNOLOGIES

GBrowse is a web application that is divided between code that runs on the web server and on the web browser client. The server side of GBrowse is written in Perl with a little C code thrown in to accelerate critical functions. The server manages a series of databases containing genome annotation information, receives requests from the web browser to view regions of interest and renders these regions as PNG, SVG or PDF images. On the web browser side, a series of Javascript functions handle the user interface, allowing you to pan and zoom across the genome, select a region via click-and-drag, configure tracks via popup menus and upload track data.

Support for a wide range of genome databases and views is one of GBrowse’s most flexible features.
A series of data adapter plugins allow GBrowse to run on top of flat files loaded into memory, large SQL and NoSQL databases, remote data sources and specialized file formats such as BAM alignment files. Genomic features can be represented by a large number of reusable ‘glyphs’ (roughly 75 in all), which range from generic colored boxes, to highly specific representations of linkage disequilibrium among SNP haplotype blocks.

Many third-party libraries are required for GBrowse to work. In particular, to display NGS sequencing data, GBrowse requires the Samtools and BigWig libraries. Because installation of these dependencies can be tedious and confusing for the newcomer, GBrowse has recently been packaged in preconfigured virtual machines that can be run on the desktop or in the Amazon Cloud. These VMs allow the user to bring up a starter genome in minutes and to start building on top of it immediately.

WORKING WITH NGS DATA IN GBROWSE
This section describes the process of installing GBrowse, configuring a data source and loading NGS tracks.

Initial installation
GBrowse will run on any recent Linux distribution and hardware. For viewing large BAM files and gene annotation databases, a minimum of 4 GB RAM and 200 GB of free disk space is recommended. One can install GBrowse from source code or install it from binaries using the ‘apt’ and ‘rpm’ package managers. There are also prebuilt virtual machine images for GBrowse in Amazon and VirtualBox formats. These provide you with basic setups for the human, worm, fly and yeast genomes which you can then build on.

Installation of the GBrowse package is described in detail at http://gmod.org/wiki/GBrowse_2.0_Install_HOWTO. The most hassle-free installation method is to run GBrowse in one of the prebuilt virtual machines. This will provide you with full functionality and performance without making any modifications to your own system. You have the option of downloading the virtual machine to your local laptop/desktop or running it on Amazon’s EC2 cloud.

Installing locally
Local installation requires you to have the VirtualBox machine virtualization software installed. VirtualBox is free software that runs on Windows, Linux and Macintosh OS computers. To obtain it, go to www.virtualbox.org and download the version appropriate for your operating system. You may also install it using a package manager such as ‘apt’.

Users of the commercial VMWare Workstation or VMWare Player applications (www.vmware.com) can also run the GBrowse2 VM. (During import you may receive a warning message about the VM not meeting compliance checks, but this may be safely ignored).

Once VirtualBox is installed, you may download and install the GBrowse2 VM. Go to gmod.org/wiki/GBrowse2_VMs and find the link for the latest version of the GBrowse2 VirtualBox ‘appliance’. Download the file to your local disk. From the VirtualBox File menu, select ‘Import Appliance’ and choose the downloaded file. This will give you a new virtual machine named ‘GBrowse2 (version number)’. To launch the machine, select it and click ‘Start’ in the VirtualBox main screen.

After booting, the GBrowse2 VM will boot automatically into a restricted ‘gbrowse’ account that provides access to the genome browser and documentation running in a browser window. You may test out GBrowse from within the virtual machine or connect to it from a web browser running on the host (real) machine by opening URL: http://localhost:8081/fgb2/gbrowse.

To administer the browser, you must log out of the restricted account by selecting ‘Logout’ from the Menu at the top left of the screen. This will take you to a login window. Select the ‘Administrator’ account and provide the password ‘gbrowse’. This will take you to a desktop in which all administrative functions are enabled. To gain access to the command line, which you will need to configure GBrowse, go to Menu and select ‘Accessories->LXTerminal’. You will find GBrowse’s configuration files in ‘/opt/gbrowse/etc’ and its databases in ‘/opt/gbrowse/databases’. Less-frequently accessed directories, such as those used to store uploaded tracks, are located in ‘/opt/gbrowse/lib/gbrowse2’.

You may use secure shell (ssh) to log into the VM from the host machine using the IP address 192.168.56.10:

ssh admin@192.168.56.10
Remote ssh access from non-host machines is disabled by default, but you can enable it by configuring an ethernet bridge adaptor as described in Chapter 6 of the Virtual Box manual (www.virtualbox.org/manual/ch06.html). Note that if you enable remote access to the VM, it is a very good idea to change the Administrator password, which you can do by issuing the ‘passwd’ command from the command line or by selecting ‘System->Users and Groups’ and using the graphical user interface to change Administrator’s password.

**Using the Amazon virtual machine image**

The Amazon GBrowse2 virtual machine allows you to run GBrowse2 on top of Amazon’s Elastic Compute Cloud (EC2). This gives you an Internet-connected server with essentially no set up required and considerable flexibility. The downside is that you pay a fixed charge for every hour the server is running. However, the cost is not very much (8–12 cents per hour), and so this method is a great way to try the system out with little investment of time or effort, particularly if you are already an EC2 user.

You will need to have an EC2 account, which you can set up in a few minutes by visiting aws.amazon.com (have a credit card ready). During the signup process, you will get several types of credentials: (1) a login username and password for the Amazon console; (2) an access key and secret access keypair for use with EC2’s command-line tools and (3) a ssh public/private keypair for use in logging into the GBrowse server.

You will need a ssh client to log into the GBrowse server. If your desktop or laptop is a Macintosh or Linux machine, then the command-line program ‘ssh’ will already be installed. If your desktop runs Windows, then you will need to install a suitable ssh client. I recommend PuTTY (www.chiark.greenend.org.uk/~sgtatham/putty/).

Go to the GBrowse VMs page at gmod.org/wiki/GBrowse2_VMs and find the link to the latest Amazon Machine Image (AMI). Clicking on this link will take you directly to the ‘Request Instances Wizard’ which leads you through the process of launching a virtual machine. Alternatively, you may search Amazon for the most recent GBrowse AMI. To do this, log into the Amazon Web Services (AWS) Console, select the EC2 service, navigate to ‘AMIs’ and use the search box to filter for public images named ‘GBrowse’. Right click on the AMI with the latest version number and select ‘Launch instance’ to bring up the request instances wizard.

The wizard will prompt you for a number of properties of the virtual machine to launch. The most important of these is the Instance Type, which controls the number and speed of CPUs and the amount of memory that the VM will have. For GBrowse, you should choose at least the ‘Small’ instance. For better performance, choose the ‘Medium’ or ‘Large’ instances. Faster instances cost more per hour.

Later during the instance creation process, you will be asked to select the ssh keypair to use for logins; choose the one you created during registration. Toward the end, you will also be asked to configure a ‘security group’, which is Amazon’s term for a firewall. I recommend that you select ‘Create a new Security Group’ and use the wizard to create a security group named ‘web+ssh’ that allows SSH and HTTP access from all Internet addresses (indicated by the default ‘0.0.0.0/0’).

After you complete the wizard, you can watch the instance start from the AWS Console’s ‘Instance’ page. When its status has changed from ‘pending’ to ‘running’, determine its DNS name from the ‘Public DNS’ column. This will be the hostname you use for web access to GBrowse2 and ssh access to the server.

To browse the starter genomes that are installed on the cloud image, go to http://public-dns-name/. This will bring you to a page that lists the starter genomes as well as pointers to the GBrowse tutorial and documentation.

To log into the machine in order to administer GBrowse, you will use ssh. Find the location of your public ssh keypair and log in like this:

```
ssh -i ~/path/to/keyfile admin@public-dns-name
```

Where ~/path/to/keyfile is the path to the ssh keypair file created during AWS registration, and public-dns-name is the DNS name of the running instance. This will take you to a command line prompt.

**Adding additional genomes and chromosomes**

The VirtualBox edition of GBrowse2 comes with preinstalled ‘starter databases’ for yeast (Saccharomyces
cerevisiae, 11 April 2012, SacCer_Apr2011/sacCer3) and the nematode (Caenorhabditis elegans, October 2010, WS220/ce10). The Amazon Virtual Machine edition includes the yeast and nematode genomes, as well as human (Homo sapiens, February 2009, GRCh37/hg19). These databases contain the chromosome sizes, genomic DNA and a set of reference gene models and noncoding RNAs.

To add additional databases, both virtual machines come with the ‘import_ucsc_db.pl’ script, which creates starter databases from information in the UCSC genome browser (genome.ucsc.edu). This can be used to add the human hg19 genome build to the VirtualBox edition, which because of the size of the data, does not include a preinstalled version. The command to use is

```
import_ucsc_db.pl hg19 'H. sapiens genome (hg19)'
```

Where the first argument is the UCSC build name, and the second optional argument is a description to use for the database. This command will fetch the FASTA files for each chromosome, initialize the database of chromosome sizes and fetch reference genes and noncoding RNAs. An optional –remove-chr argument will remove the ‘chr’ prefix that UCSC places in front of each chromosome name. This is recommended if you work frequently with non-UCSC data sources, such as the model organism databases or Ensembl, and is how the default databases on the Amazon and VirtualBox VMs were created.

You may also add new tracks or whole species by loading BED, GFF3, SAM or BAM files downloaded from a suitable source. The process for doing this is described in detail in the GBrowse online documentation at gmod.org/wiki/GBrowse_2.0_HOWTO. The rest of this article focuses on the process of installing NGS files.

### Uploading a SAM/BAM file

You can view aligned NGS data contained in either BAM or SAM formats (samtools.sourceforge.net/). Alignment files can be uploaded via GBrowse’s web interface, linked to from a remote FTP site or web server or installed on the server using the command line. Because alignment files can be quite large, direct uploading is only recommended for smaller BAM/SAM files (less than a couple hundred megabytes).

We will first discuss uploading. For this example, we use a small (5.4 M) modENCODE (www.modencode.org) SAM file obtained by performing RACE sequencing of the 3’-UTRs of *C. elegans* L1 larval RNA. Download the file using either the full URL ftp://data.modencode.org/all_files/cele-signal-1/2327_L1.ws220.sam.gz or its equivalent ‘tiny URL’ is http://tinyurl.com/9ns9fjz. If you try this with your own SAM file, be careful to match the genome build (WS220/ce10) and the naming convention for the chromosomes. The starter GBrowse databases all use unadorned chromosome names, such as ‘1’ and ‘III’. This is consistent with the NCBI GenBank and Ensembl convention, but conflicts with the UCSC Genome Browser convention of ‘chr1’ and ‘chrIII’.

Start the GBrowse server by launching either the VirtualBox or Amazon editions and navigate your browser to the *C. elegans* database by choosing *C. elegans* (WS220/ce10) from the welcome page or Data Source popup menu in the browser itself.

Click on ‘Custom Tracks’ in the menu bar at the top of the genome browser, and select ‘Add custom tracks: [From a file]’ at the bottom of the custom tracks panel. Choose the SAM file that you downloaded previously, and click the ‘Upload’ button. Depending on your network speed, it will take about 20s to upload and fully process this file. When the processing is finished, summary information about the upload will appear (Figure 2).

You may now click on the Browser menu item to return to the main genome browser view. This is a low coverage RNA sequencing experiment, and so you may have to zoom out a bit in order to see the data. To see an example of how the data are represented, search for ‘icl-1’ in the ‘Landmark or Region’ search box. This will display the gene icl-1 as well as a histogram of coverage of the uploaded SAM file (Figure 3). This alignment suggests that the real 3’-end of the icl-1 gene lies about 50 bp downstream of the annotated end.

To view this region in more detail, zoom in on it by clicking on the ruler at the top of the panel and dragging across the coverage region. Do this repeatedly until the histogram is replaced by the reads themselves. When you increase the detail to a region of ~100, the bases themselves come into view (Figure 1, bottom). Mismatches and deletions relative to the reference genome are shown in red, while insertions are shown in green. Clicking on one of the reads brings up an information page which
Figure 2: Summary information about an uploaded SAM file. The summary information includes the name and description of the uploaded data, which can be edited by clicking on the respective fields, and information about the date and size of the upload. The ‘Sharing’ area allows the user to enable sharing of the track with select collaborators or with the public as a whole.

Figure 3: Uploaded SAM file in histogram mode. Since this is 3’-RACE data, the reads are concentrated at the 3’-end of the gene and show that the gene’s 3’-UTR should be extended.

shows details about the read and a text representation of the alignment. To change the appearance of the sequence alignment track, click on the toolbox icon that appears in the track’s titlebar. This will bring up a dialog that allows you to change colors, size, the presence or absence of read names and various other features.

You may upload a BAM file in exactly the same manner as you uploaded a SAM file. The advantage of this over SAM format is that processing will be quicker because the server does not have to convert it into BAM internally. You may upload as many BAM/SAM tracks as you like and select which ones are displaying using the ‘Select Tracks’ panel.

Uploaded files are inaccessible to other GBrowse users unless they are explicitly shared. However, the files are readable by anyone who can log into the virtual machine.

Track sharing
Uploaded BAM/SAM tracks can be shared with collaborators. To do this, go back to ‘Custom Tracks’ and click on the ‘Sharing’ popup menu. Select ‘Casual’ sharing to get a sharing link, and email this link to whoever you wish. They can get access to the track by clicking on this link in the received email.

To make a track public, select ‘Custom Tracks->Sharing->Public’. This will enable anyone to find and view your track using the search features of the ‘Community Tracks’ panel. To aid in sharing, you should give your public track a good descriptive name and description, which you can do by clicking on the upload’s name and description fields.

The last sharing option is called ‘Group’. In this mode, you can share the track with a specific set of named collaborators. For this to work, you will need to know the collaborators’ email addresses or GBrowse login names. Select ‘Custom Tracks->Sharing->Group’, and then type in a portion of the first collaborator’s email address or login name in the ‘Enter a username’ text field (autocomplete will help you select the correct user). Click ‘Add’ to
authorize this user. You may repeat this process multiple times to add additional collaborators.

When you are finished with your upload(s), go to ‘Custom Tracks’ and click on the trash can icon to delete the ones you wish.

**Uploading a BAM/SAM file as the administrator**

With a slight modification of the above recipe, you can upload a NGS file in a way that allows it to become listed as a public track. The only difference is that you must log into GBrowse as the administrator before uploading the file(s). From the genome browser’s main page, click on ‘Log in’ in the upper right hand corner. When prompted for a username and password type username ‘admin’ and password ‘gbrowse’. As long as you are logged in as the administrator, any track that you create via the ‘Custom Tracks’ panel becomes visible to the world and can be found in the standard ‘Select Tracks’ panel.

Note that it is recommended you change the admin password before making the server public. The GBrowse VM page tells you how to do this. Also be aware that the admin password used for logging into GBrowse’s web interface is not shared with the Unix account of the same name: if you are using the Amazon VM, the ‘admin’ login has no password, but can only be accessed using a ssh key.

**Linking to a BAM file**

It takes a long time to upload a large SAM or BAM file. In cases when the file is more than 100 MB in size, GBrowse users are encouraged to use the software’s remote BAM feature. This feature allows the browser to fetch alignment data on as needed basis, allowing you to view the data right away.

For this to work, the alignment data must be in sorted BAM format, must have been indexed against the correct genome build and must be placed on a Web or FTP server at a location where the GBrowse server can reach it via the network. If you are using the VirtualBox VM, this means that the Web or FTP server may either be a public internet site, or may be located on your LAN (including on the host machine that runs the virtual machine). If you are using the Amazon VM, then the Web/FTP server must be internet accessible.

For this example, we are going to use an indexed BAM file from the 1000 genomes project, a high-coverage Illumina sequence from an anonymous individual, mapped onto chromosome 1 of the GRCh37 build of the reference genome.

If you are using the VirtualBox VM, you will first need to install a starter human hg19/GRCh37 database. Log into the VM as the ‘admin’ user (password ‘gbrowse’), open a terminal window and type the following command:

```
import_ucsc_db.pl –remove-chr hg19 'H. sapiens (hg19/GRCh37)'
```

This will contact the UCSC genome browser to fetch the DNA for the build and reference gene models and noncoding RNAs, consuming roughly 3.5 GB of additional disk space. The –remove-chr argument is required because UCSC appends ‘chr’ to the beginnings of each of its chromosome names, while the 1000 genome project does not. After the data are installed, the script will restart the web server. If you refresh your browser, you will find the database installed.

If you are using the Amazon VM, then the human reference data have already been installed for you, and the proceeding step is not required.

With your web browser, navigate to GBrowse and select **H. sapiens** from the ‘Data Source’ menu. Click on ‘Custom Tracks’ in the menubar at the top of the page and then click ‘Add custom tracks: [From a URL]’. This will pull down a text box. Cut and paste the following URL into the box:

```
ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/pilot2_high_cov_GRCh37_bams/data/NA12878/alignment/NA12878.chrom1.ILLUMINA.bwa.CEU.high_coverage.20100311.bam
```

You may now view any region of the chromosome 1, although I suggest that you limit the region to less than 100 kb to avoid network timeouts. For example, search for gene **PLEKHN1**. This will show a coverage histogram across the gene. Then zoom down to 1 kb using the Scroll/Zoom menu. This will show the paired-end read alignment details (Figure 4). As before, once you zoom down to ~100 bp, the base pairs and mismatches will be displayed.

Note that the paired-end read relationships are shown by default. If you prefer a more compact display that does not keep the paired ends aligned, you may change it by going to the ‘Custom Tracks’
section, finding the link to the track ‘Configuration’ file and clicking `[edit]`. This will display an editable box containing the following information:

```plaintext
[ftp_ftp.1000genomes.ebi.ac.uk_voll_ftp_technical_pilot2_high_cov_GRCh37_bams_data_NA12878_alignment_NA12878.chrom1.ILLUMINA.bwa.CEU.high_coverage.20100311.bam
database = database_0  # do not change this!
feature = read_pair
glyph = segments
draw_target = 1
show_mismatch = 1
mismatch_color = red
bgcolor = blue
fgcolor = blue
height = 3
label = 1
label_density = 50
bump = fast
key = Shared track from ftp://ftp.1000genomes.ebi.ac.uk/voll/ftp/
```

Find the line that reads ‘feature = read_pair’ and change ‘read_pair’ to ‘match’. Other customizations that you can perform at this level are described in gmod.org/wiki/GBrowse_NGS_Tutorial and gmod.org/wiki/GBrowse_2.0_HOWTO.

**Configuring a BAM track on the server**

The last way to add an NGS alignment track to GBrowse is via installing it directly on the server. This gives you the greatest ability to customize the appearance and behavior of the track.

To do this, log into the server as the ‘admin’ user and create a directory in which the BAM or SAM files will be installed. By convention, the GBrowse server’s databases are stored in `/opt/gbrowse/databases/<source>`, where ‘source’ is the genome build name (such as ‘hg19’). You are encouraged to follow this convention. For the purpose of example, we create a directory named ‘NGS_alignments’, and
then make it owned by the admin user. We use the ‘sudo’ command to gain root privileges to allow this:

```bash
sudo mkdir /opt/gbrowse/databases/hg19/NGS_alignments
sudo chown admin /opt/gbrowse/databases/hg19/NGS_alignments
```

Next, copy one or more alignment files into this directory. You may use BAM or SAM files, and the SAM files may be compressed with gzip if you wish. To get the files, you may use the ‘wget’ command to copy files from internet sites, or ‘scp’ to use the ssh to copy files from your home directory or other private sites. The page at gmod.org/wiki/GBrowse2_VMs provides a few tips on how to do this.

We will again use the human 1000 genomes data as an example, but use a smallish (50 MB) exon-targeted file from chromosome 1. For conciseness, we will use a Tiny URL to fetch the file.

```bash
cd /opt/gbrowse/databases/hg19/NGS_alignments
wget http://tinyurl.com/97kw5zu
```

You may do this for the BAM files from additional chromosomes if you wish. Use ‘samtools merge’ to merge all chromosomes into a single BAM file before you proceed to the next step.

Next, run the ‘bamToGBrowse.pl’ tool, providing it with the path to the NGS_alignments directory and the FASTA file containing the chromosomal DNA. In this case, we wish to work with the current directory (‘.’). The chromosomal DNA can be found a level above in `/opt/gbrowse/databases/hg19/chromosomes/:

```bash
cd /opt/gbrowse/databases/hg19
bamToGBrowse.pl NGS_alignments chromosomes/chromosomes.fa
```

In a short time (~10 s for the example), the script will create various indexes and then write out a track configuration file in the same directory named ‘gbrowse.conf’. This file needs to be appended to the hg19 configuration file, located at ‘/opt/gbrowse/etc/hg19.conf’, which can be done from the command line using:

```bash
sudo sh -c ‘cat NGS_alignments/gbrowse.conf’>/opt/gbrowse/etc/hg19.conf
```

The ‘sudo’ is needed because hg19.conf is normally owned by the ‘root’ user, although you are free to change this.

```
Restart the web server with:

```bash
sudo service apache2 restart
```

You will now be able to see the alignment track in the ‘Select Tracks’ section of the genome browser page (remember that only chromosome 1 is represented in the downloaded file!). If you wish to customize any aspect of the track, such as its name, you simply edit the appropriate track configuration section of ‘/opt/gbrowse/etc/hg19.conf’, as described in gmod.org/wiki/GBrowse_NGS_Tutorial and gmod.org/wiki/GBrowse_2.0_HOWTO.

### FUTURE DIRECTIONS

As noted in the ‘Introduction’ section, GBrowse has reached a state of maturity and is no longer adding major new features. Future releases will focus on performance and stability. In particular, as genome annotation databases grow, the strain on the underlying GBrowse databases increases and performance suffers. GBrowse already provides a master/slave architecture in which the task of querying databases and rendering tracks is handed off to a farm of network-connected servers, so that the main server does not bear the full load. However, in practice, this architecture is seldom used due to the complexity of deploying and maintaining the slave servers.

The Amazon cloud version of GBrowse provides a solution for this. The development team plans to enhance the Amazon VM with the option of automatically launching slaves into the Amazon cloud automatically when the load hits predefined limits. Another advantage of running on the cloud is that it enables the use of distributed ‘Big Data’ databases such as HBase and MongoDB. Under this scenario, genomic data can be uploaded into a flexible pool of relatively low-end database servers. GBrowse will be able to search for annotations across this pool, avoiding a bottleneck on a single database server or filesystem and hopefully seeing significant performance improvements.

### Key Points

- GBrowse 2.0 fully supports next-generation sequencing data from both DNA and RNA sequencing experiments.
GBrowse runs in a web server and is accessed via any modern web browser. Next-generation sequencing data tracks can be installed permanently as public tracks, uploaded on an as-needed basis, imported via URLs and selectively shared with other users. The software is most suitable in a collaborative environment where visualization of sequencing data is shared among multiple local and remote collaborators. GBrowse is available as preconfigured virtual machines running on the desktop or the Amazon Elastic Compute Cloud, as well as in source code and binary form.

Acknowledgements
The author wishes to thank Dr Scott Cain for assistance with configuring and testing the VMs and four anonymous reviewers who contributed many helpful suggestions during manuscript preparation.

FUNDING
This work was funded by grant #P41 G02223 from the National Human Genome Research Institute at the US National Institutes of Health, and by the Ministry of Economic Development and Innovation, Ontario (in part).

References
1. Stein LD, Mungall C, Shu S, et al. The generic genome browser: a building block for a model organism system database. Genome Res 2002;12(10):1599–610.
2. Mungall CJ, Emmert DB; FlyBase Consortium; A Chado case study: an ontology-based modular schema for representing genome-associated biological information. Bioinformatics 2007;23(13):i337–46.
3. McKay SJ, Vergara IA, Stajich JE. Using the Generic Synteny Browser (GBrowse_syn). Curr Protoc Bioinformatics 2010;Chapter 9:Unit 9.12.
4. Pan X, Stein L, Brendel V. SynBrowse: a synteny browser for comparative sequence analysis. Bioinformatics 2005;21(17):3461–8.
5. Youens-Clark K, Faga B, Yap IV, et al. CMap 1.01: a comparative mapping application for the Internet. Bioinformatics 2009;25(22):3040–2.
6. Blankenberg D, Coraor N, Von Kuster G, et al. Integrating diverse databases into an unified analysis framework: a Galaxy approach. Database 2011.
7. Lewis SE, Searle SM, Harris N, et al. Apollo: a sequence annotation editor. Genome Biol 2002;3(12):1–14.
8. Cantarel BL, Korf I, Robb SM, et al. MAKER: an easy-to-use annotation pipeline designed for emerging model organism genomes. Genome Res 2008;18(1):188–96.
9. Zhang J, Haider S, Baran J, et al. BioMart: a data federation framework for large collaborative projects. Database 2011.
10. Skinner ME, Uzilov AV, Stein LD, et al. JBrowse: a next-generation genome browser. Genome Res 2009;19(9):1630–8.
11. Li H, Handsaker B, Wysoker A, et al.; 1000 Genome Project Data Processing Subgroup. The Sequence Alignment/Map format and SAMtools. Bioinformatics 2009;25(16):2078–9.