BioViz *Connect*: Web application linking CyVerse cloud resources to genomic visualization in the Integrated Genome Browser

Karthik Raveendran[†], Chaitanya Kintali[†], Srishti Tiwari, Pawan Bole, Nowlan H Freese* and Ann E Loraine*

Department of Bioinformatics and Genomics, University of North Carolina at Charlotte, Kannapolis, NC 28081, USA

* Correspondence: nfreese@uncc.edu, aloraine@uncc.edu
[†] Karthik Raveendran and Chaitanya Kintali contributed equally to this work.

Abstract

Background:

To make use of large-scale data sets from genomics, biologists need computational systems to store, process, analyze, annotate, and visualize their data. Cloud-based science gateways such as CyVerse provide storage and analysis tools but offer limited visualization capability. In parallel, desktop programs such as the Integrated Genome Browser (IGB) support interactive, dynamic data visualization using local computing resources. However, CyVerse and IGB exist separate from each other, with no easy way for users to transfer data between the two.

Results:

We present BioViz *Connect*, a new web application that *connects* CyVerse and IGB using the CyVerse Terrain API. Using BioViz *Connect*, researchers can stream their data to IGB for visualization and run analyses to create new visualizations. BioViz Connect also functions as a dashboard-style application that lets users specify genome version and visual appearance for data files, thus controlling how data will look once loaded into IGB. To demonstrate BioViz *Connect*, we present an example RNA-Seq data set from *Arabidopsis thaliana* that compares gene expression between heat-treated plants and non-heat-treated controls. Using CyVerse and BioViz *Connect*, we create scaled coverage graphs and visualize them in Integrated Genome Browser, showing how the blend of cloud and desktop resources give researchers greater power to explore and understand their data. Visit https://bioviz.org/connect.html to use BioViz *Connect*.

Conclusions:

BioViz *Connect* shows how applications that integrate remote cloud resources with interactive, desktop applications boost researchers' ability to explore and understand their data.

Keywords:

Integrated Genome Browser, CyVerse, Visualization, Cloud Computing

Background

Despite decades of progress, biology researchers still face major challenges storing, processing, analyzing, and visualizing genomic data. There are many tools and systems that can perform one or more of these tasks, but rarely all four. As an example, consider data storage versus data processing and analysis systems. Biologists regularly use consumer-focused, low-cost commercial cloud storage platforms like Google Drive and Dropbox to share and store data. These and similar platforms are well-engineered and convenient, but they lack integration with data analysis tools.

To analyze their data, researchers must transfer their files to a system that offers analysis capability. For many researchers, this means moving the data onto a research-focused cluster computing environment maintained at their home institution. These systems use mature, well-documented queuing software for managing and parallelizing jobs for many users at once, and they offer tremendous computational power. Universities typically employ research computing teams whose sole focus is maintaining the cluster environment and supporting scientific users. However, using these systems requires technical skill, e.g., shell scripting, that few biologists have time to acquire.

Even more of a problem, however, is that these systems rarely make interactive visualization possible. Design decisions motivated by the need to keep these systems secure have undermined users' ability to flow data into interactive, graphical, visualization tools. By definition, such tools need to display interactive, graphical content on the user's local desktop, and the only way to do this currently is to use desktop visualization software, either running in a Web browser or in specialized desktop applications like Integrated Genome Browser [1, 2], Cytoscape [3], and others. The limitation this imposes on science cannot be underestimated. In the life sciences, where data set sizes are enormous and data are heterogeneous and complex, visualizing data is practically synonymous with understanding data. Even worse, the inability to visually interact with data leads to errors. For example, one commonly performed step in transcriptome analysis involves aligning RNA-Seq sequence reads onto a reference genome. However, the widely used tophat2 and hisat2 RNA-Seq aligners assume a default, maximum intron size (500,000 bases) that is too large for non-mammalian genomes [4, 5]. Running tophat2 with default settings and a plant reference genome generates many incorrect alignments in which sequence reads align across neighboring genes. This problem is immediately apparent upon viewing the alignments in a genome browser and almost impossible to notice otherwise. Because researchers have lacked convenient ways to visualize their data, many studies have overlooked this not-well-known aspect of spliced alignment tools [6-12].

A large community of technologists and computer scientists have been working for many years to address these problems. As often happens in science, many groups tackling the same problems have converged on similar solutions. One such common solution involves building Web sites called "science gateways" that provide a single point of access to disparate or hard-to-use HPC resources [13]. In bioinformatics, the Galaxy workflow system was one of the first to explore this idea [14]. Galaxy is a Web application implemented in python that provides a user-friendly interface to command-line tools used to process large-scale 'omics data sets [15]. Researchers upload data files to their Galaxy accounts, where they process their files using tools from a shared Galaxy Toolshed [16] in sequential pipelines assembled using a graphical workflow editor. Dozens of developers and scientists from many countries contribute to Galaxy, and many more contribute tutorials and training materials to the Galaxy Training Network [17]. These tutorials teach researchers how to properly run tools featured in the Galaxy Toolshed, and, perhaps more importantly, how to evaluate and use the results. This latter aspect of Galaxy illustrates how science gateways can combine user and developer communities that work together to improve scientific practice.

However, science gateways by design aspire to become a single location for accessing tools, and so whatever visualization tools they provide are typically built-in to the system. This makes it difficult for users to pull data into specialized visualization desktop tools that are often more flexible and user-friendly. To address this problem, Galaxy developers created a now widely used "External Viewers API" that allows developers to configure Galaxy to serve public links to data files that can then be consumed by external tools [18]. Using this API, a developer can introduce a hyperlink within the Galaxy interface that forwards a data file's URL to external viewers which in turn use the URL to retrieve and display data files hosted on Galaxy. As described elsewhere, we used this API to enable users to view their Galaxy data files in Integrated Genome Browser [1], and many other groups have done the same. However, the communication between Galaxy and external viewers is currently limited to providing only the data file URL to external applications.

Building on insights gained from the Galaxy project, the iPlant Collaborative developed the Discovery Environment (DE) web portal, a science gateway focusing on supporting plant science [19, 20]. Similar to Galaxy, the DE lets users build pipelines from command-line programs using a Web-based interface. To build the DE, iPlant engineers first built the iPlant Foundational APIs (Application Programmer Interfaces), which provided a programmatic gateway to resources for which Discovery Environment became the "front end" user interface. Thus, the project of building Discovery Environment provided immediate and realistic use cases for the API itself.

Extending the concept of the Galaxy Toolshed, the Discovery Environment enables users to create, deploy, and publish data processing utilities called "Apps" that operate on user-specified input files. Apps are most often single command-line programs (like tophat2) with some or all options pre-configured by the App contributor. Once an App is published, anyone with an account on the system can use it. Another important feature of the DE and its APIs is that users can annotate their files and Apps with user-defined metadata. This enables developers to tag files with application-specific metadata, which makes supporting application-specific functionality much easier. A third key feature is that DE contains a shared "Community" file store where users can permanently publish files, often in conjunction with a peer-reviewed scientific publication. Fourth, the DE offers an open authentication system that enables users to authorize third-party access to their files. These four features make DE into a virtual community of data and compute resources for researchers and developers to use, build and populate.

In 2015, the iPlant Collaborative renamed itself CyVerse and expanded its intended audience to include all life science researchers, not just plant scientists. Throughout its history, iPlant and CyVerse continued to build and expand the iPlant Foundational APIs. In addition, the experience of building these Science Gateway APIs spawned new API-focused projects, such as the AGAVE APIs [21]. The original and always improving iPlant Foundation APIs are now available under the name "Terrain APIs." Any action a user can perform interactively using the Discovery Environment Web interface can also be done computationally using the underlying

Terrain APIs. Like Galaxy, the Discovery Environment has many thousands of users and has become an essential part of many biology projects.

Science gateway implementations have great potential to solve problems of access and usability, but like all Web-based user interfaces, they require constant maintenance and revision to keep up with rapid changes in the Web programming sphere. Web site interfaces that were innovative only one or two years ago may seem dated and clunky to today's users. Or worse, they may lack features that today's users now regard as essential, such as single sign-on using Facebook or Google. The iPlant/CyVerse infrastructure offers a partial solution to this problem of rapid obsolescence via their focus on building APIs in advance of or in tandem with application development. CyVerse infrastructure separates presentation logic from data access and data processing logic, and therefore has the potential for greater longevity as the most labile part of the infrastructure - the user interface - can be developed independently from the underlying APIs.

One important limitation of the CyVerse infrastructure, however, is that there is currently no easy way for researchers to visualize genomic data sets stored in CyVerse and thus sanity-check, explore, or visually analyze their results. To develop the concept of connecting desktop software to science gateways via APIs, we built BioViz *Connect*, which integrates CyVerse data storage and HPC resources and Integrated Genome Browser, a full-featured visualization system that runs on the user's desktop.

Implementation

BioViz *Connect* is a single-page Web application that accesses CyVerse storage and compute resources by calling Terrain API endpoints. BioViz *Connect* forwards those resources to Integrated Genome Browser by hitting local REST API endpoints within Integrated Genome Browser itself. Thus, BioViz *Connect* forms a bridge between distant systems: CyVerse-managed computing resources fronted by the Terrain APIs and Integrated Genome Browser, a rich client Java application running locally on a personal computer.

BioViz *Connect* consists of two parts: a JavaScript-based user interface that runs in the user's Web browser and a back-end server-side application that manages authentication and communication with Terrain API endpoints. The user interface code on the front end is implemented using HTML5, CSS, Bootstrap 4.3.1, JavaScript, and jQuery 1.10.2. The server-side code is implemented in python3 using the Django web application framework. BioViz *Connect* is deployed on Amazon Web Services infrastructure using the Apache Web server software, but its design is platform agnostic and could run on other cloud or local systems as needed. BioViz *Connect* code is open source and available from https://bitbucket.org/lorainelab/bioviz-connect.

BioViz *Connect* application flow, described in detail in the following sections, emphasizes responsiveness and user-friendliness while also protecting users' credentials, e.g., CyVerse account passwords. As users navigate through BioViz *Connect* screens, the front end JavaScript code running in the user's Web browser communicates securely with the server-side parts of the application, which in term communicate securely with CyVerse API endpoints via encrypted channels. The Django server-side code uses a REDIS database to store user authentication tokens during a session, which is identified by a session id stored in the database and in the user's Web browser, but apart from this, no other user data is tracked.

Integrated Genome Browser is developed using the Java programming language version 1.8. The IGB source code resides in a git repository hosted free of charge on Atlassian's bitbucket.org site, and a continuous integration/continuous deployment mechanism uses bitbucket pipelines and ansible software to build the IGB application and package it for release. A formally released, fully-tested version is clearly marked on the Web site https://bioviz.org as the recommended version for most users, and people wanting the latest features can download the most recently developed code (called the "master" branch) under the name "Early Release IGB."

IGB is a desktop software program which users download and install on their local computer systems. To make this process as easy as possible, the IGB Web site https://bioviz.org provides installers for Linux, MacOS, and Windows platforms. Installers include both IGB compiled code as well as a Java language run-time to ensure trouble-free installation and operation. IGB is open source software that is freely available for anyone to modify to suit their needs, but a core team of developers (led by Ann Loraine) is responsible for managing and executing the formal IGB release cycle via the BioViz Web domain. In addition, the core group has received complimentary licenses to use continuous integration/continuing deployment tools and resources from the Atlassian company, based in Australia. We also enjoy a complimentary license to use the Install4J installer software from EJ Technologies, based in the EU.

IGB architecture is described in detail in other publications, and so here we highlight just those aspects of IGB implementation needed to understand how IGB and BioViz *Connect* interact. IGB contains a simple Web server configured to respond to REST-style queries on an IGB-specific port on the user's local computer. As users interact with BioViz *Connect* Web pages, JavaScript functions running in the Web browser make REST-style calls to this IGB localhost endpoint, directing IGB to load and display data pulled from CyVerse. This is an enhancement of the REST-based mechanism IGB uses to interact with and consume data from other sites, including the BioAnalytics Resource (BAR) eFP-Seq browser [22] and the Galaxy workflow server [15]. The chief difference or innovation of BioViz *Connect* compared to these older schemes is that BioViz *Connect* transmits meta-data about a data set to IGB, thus enabling users to configure the appearance of their data using the BioViz *Connect* user interface as a dashboard-style application. This ensures that when colleagues or collaborators view the same data in IGB, it will appear as the owner of the data intended.

Results and Discussion

To demonstrate BioViz *Connect*, we describe the following use case scenario showing how to annotate, analyze, and visualize an example RNA-Seq data set hosted in the CyVerse Discovery Environment. For the convenience of readers, we published the example data to the CyVerse Discovery Environment "Community" folder, where all CyVerse users can view the files but not modify them.

The BioViz *Connect* launch page is https://bioviz.org/connect.html. Clicking the "Log in to CyVerse" link at BioViz *Connect* takes us to a page hosted on CyVerse where we enter our CyVerse username and password to log in. (Anyone can obtain an account free of charge from https://de.cyverse.org/de/.) This page is the entry point for the CyVerse Central Authentication Service (CAS). When we log in using this page, the CyVerse site and BioViz *Connect* communicate behind the scenes, exchanging data that allows BioViz *Connect* to access and interact with CyVerse resources on our behalf. This service ensures we never expose our CyVerse password outside the CyVerse domain itself. Another benefit is that CyVerse infrastructure manages user accounts; no BioViz-specific accounts or passwords are required.

Upon successful login, BioViz *Connect* receives an authorization code directly from CyVerse. BioViz *Connect's* server-side code uses the authorization code to retrieve an access token from CyVerse, which it then uses to access Terrain API endpoints. At no point is the access token exposed to the Web browser running on our local computer. Protecting the access token in this way is essential as it provides unfettered access to a user's account.

The token expires after a short period. Following expiration, we must log in again to continue using BioViz Connect. We can also sign out whenever we like. When we log out, BioViz *Connect* deletes our access token from its server-side database, deletes a BioViz *Connect* Web browser cookie with session identifier from our Web browser, and then redirects us to a CyVerse CAS logout page to notify CyVerse that the access token is now no longer valid. Many other sites use the same process to manage logging in and out, and so this flow feels familiar and comfortable.

As soon as we log in, our Web browser displays the BioViz *Connect* user interface, which shows a browsable, sortable, paginated view of our account's CyVerse home directory and its contents (Fig. 1A). This view of files and data resembles the interface for Google Drive, a deliberate design choice that leverages researchers' increasing familiarity with this commonly used cloud storage system. Similar to Google Drive, clicking an item selects it, and double-clicking a folder opens it and displays the contents. A bread crumb display at the top of the panel shows the path from the root folder to the currently opened folder, a view that helps us understand and remember how we have organized our data. One improvement on Google Drive is that users can copy the file path for any file or folder. Links shown in the browser's URL bar reflect the organization of one's files, and we can use the browser's forward, back, and bookmarking functions to navigate the virtual file system.

A search entry form at the top of the page lets us to search for files or folders by name. Searches use implicit wild cards. For example, we can search for ".bam" to retrieve all binary alignment (bam format) files in the home folder. Only files for which we have read access are returned.

Navigation buttons linking home, shared and community data folders appear in a side panel left of the central file and folder view. As noted above, the demonstration RNA-Seq data set resides in a subfolder of the "Community" folder, which holds data files shared across all user accounts. Our example data reside in file path "BioViz / rnaseq / A_thaliana_Jun_2009 / SRP220157 / reads." The "reads" folder contains bam format files with alignments between RNA-Seq sequence reads and the most recent (June 2009) release of the Arabidopsis thaliana reference genome, also called "TAIR10" and "TAIR9" [23]. The RNA-Seq data are from a study of how heat and desiccation stresses affect gene expression and RNA splicing in 21-day old *Arabidopsis* plants. The original sequence data are publicly available from the Sequence Read Archive (SRA) [24] as Bioproject PRJNA509437.

To illustrate using BioViz *Connect*, we first discuss RNA-Seq alignment files SRR10060893.bam and SRR10060894.bam, named for their corresponding "run" accessions in the Sequence Read Archive. They both represent control samples and are biological replicates of each other. The BioViz *Connect* middle panel view shows that SRR10060893.bam has size 1.61 GB, about twice the size of SRR10060894.bam, which is 0.669 GB. Using their "run" ids, we can look them up in the Sequence Read Archive, where we find that sample SRR10060893 produced around 37.6 sequences (called "spots") and sample SR1006094 produced 36.7 sequences. This shows that the two samples were sequenced to about the same depth, and so their resulting alignment files ought to have similar sizes. The fact that they do not suggests there could be a problem with one or both samples. Visualizing the data in Integrated Genome Browser will likely expose any possible problems with the data.

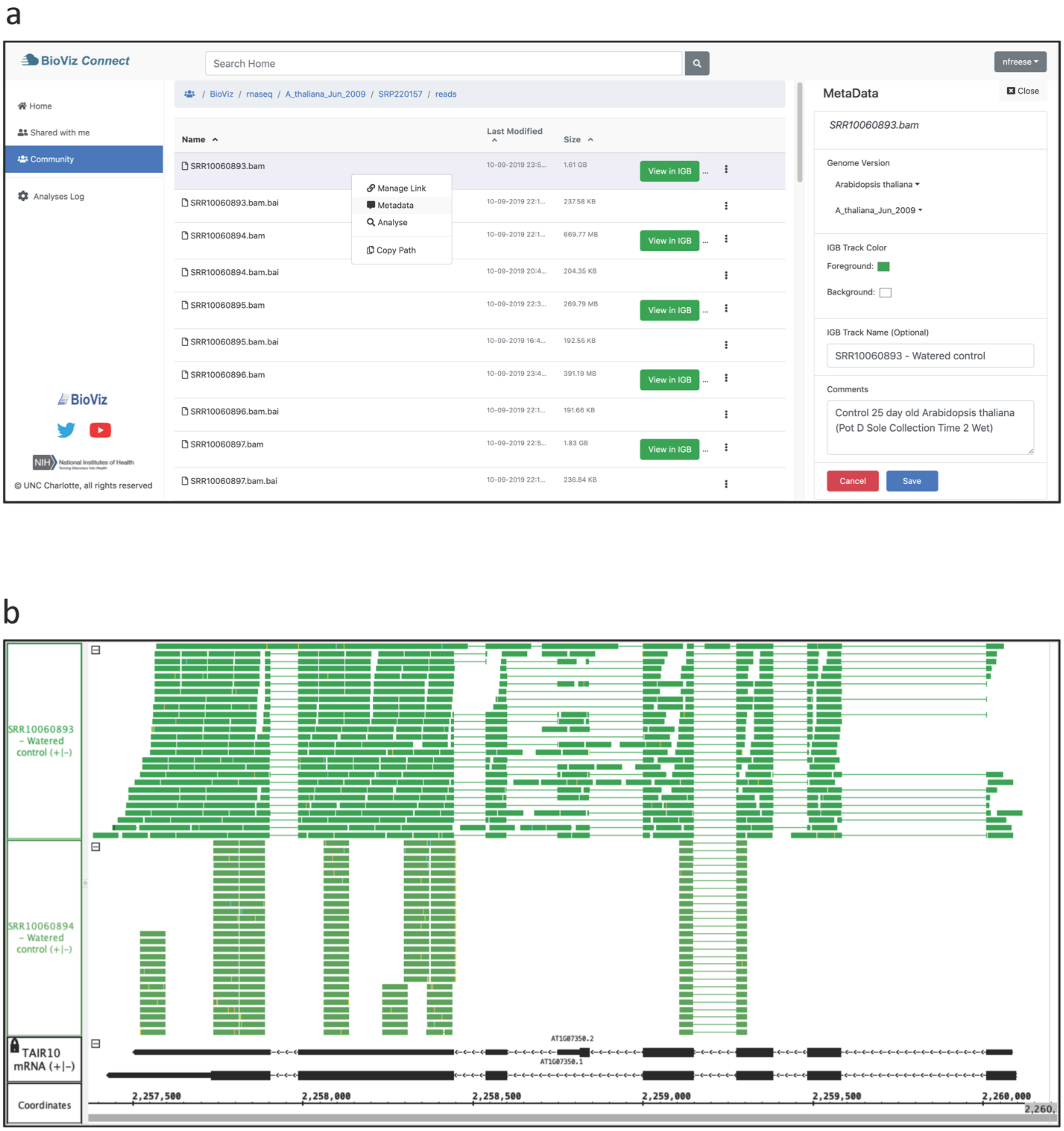
To view the data, we start Integrated Genome Browser. Next, we return to BioViz *Connect* in our Web browser and click the green button labeled "View in IGB". IGB then opens the correct genome version and adds the file as a new track to its display. Clicking the "View in IGB" button causes BioViz *Connect* JavaScript code, running in the Web browser, to send a command to Integrated Genome Browser running on our desktop. The command includes parameters that tell IGB where to find the data on the CyVerse site along with meta-data parameters associated with the track, including genome version and style information. If IGB is not currently running when we click the "View in IGB" button, then the JavaScript code instead displays a notice letting us know that in order to view the data in IGB, we first must start IGB. Using BioViz *Connect* requires IGB version 9.1.2 or later.

When we first open a genome, IGB shows the first listed chromosome in a pre-sorted list, typically whichever chromosome is named "1" or "Chr1." In this case, Chr1 encompasses more than 30 million bases. Because the RNA-Seq experiment contains millions of alignments, we need to zoom to a smaller region before attempting to load the data. In this example, our goal is to sanity-check the two data files by comparing them, and for this purpose, the best approach is to view the data for a familiar gene whose behavior is already well-understood. We know from prior work that the gene SR45a, encoding an RNA-binding protein, exhibits interesting splicing and expression patterns under stresses [25, 26], and so we use IGB's search function to zoom and pan to SR45a. (SR45a is also called AT1G07350, which is the gene's Arabidopsis Genome Initiative or AGI code.)

To load the alignments into IGB, we click the "Load Data" button at the top right of the IGB window. Once the data load, we almost immediately notice a problem with SRR10060894 (Fig. 1B). The alignments for this sample appear in tidy, vertical stacks covering approximately 30% of the gene's exons. By contrast, the alignments for the other sample fully cover the gene body and include many spliced reads split across introns. The sparser pattern observed in SRR10060894 typically arises when the library synthesis process included too many PCR amplification cycles, which reduces the diversity of resulting sequence data. Unfortunately, this means we must not use this file for further analysis. This example illustrates the importance of visualizing data at multiple steps in the data analysis process, not just at the end.

BioViz *Connect* links CyVerse and IGB by using the Terrain API to attach meta-data to files. The Terrain API represents metadata items as three values - Attribute, Value, and Unit (AVU) - and is built on top of the iRods data management system, the underlying data store for CyVerse. Within a metadata item, "Attribute" refers to its label, "Value" refers to its specific value, and "Unit" indicates metadata type. Within BioViz *Connect*, we use a custom, IGB-specific Unit to retrieve and display IGB-relevant metadata. BioViz *Connect* passes these metadata values to IGB to when we click "View in IGB" to specify how the file's data will look once loaded and also to make sure that IGB overlays the data onto the correct reference assembly.

To view and modify BioViz *Connect*-related meta-data for a file, we right-click the file name in the middle panel and select "View meta-data". This opens a right-side panel with IGB-specific meta-data (Fig. 1A). The most important meta-data for genomic data files is shown at the top of the right panel display: species and reference genome assembly names. If these are not correctly set, IGB will fail to switch to the correct genome assembly version when we click the "View in IGB" option. Because the goal of BioViz *Connect* is to facilitate visualization in IGB, it uses an IGB-specific genome version nomenclature which includes species name (*thaliana*), the first letter of the genus (A for *Arabidopsis*), and the month and year that the genome assembly version was released. Style meta data control how the track will look in IGB, such as track colors and track name. A free-text comment field lets the file's owner describe in human-readable terms what the file represents. If our account has write-permission to the file, we can change the meta-data values, thus using the BioViz *Connect* user interface to control how IGB will represent the data once loaded. This meta-data display thus acts as a dashboard application that lets us annotate data and configure its appearance.



/ BioViz / rnaseq / A_thaliana_Jun_2009 / SRP220157 / reads

Name	Last Modified	Size		
SRR10060893.bam	10-09-2019 23:5...	1.61 GB	View in IGB ...	⋮
SRR10060893.bam.bai	10-09-2019 22:1...	237.58 KB		⋮
SRR10060894.bam	10-09-2019 22:1...	669.77 MB	View in IGB ...	⋮
SRR10060894.bam.bai	10-09-2019 20:4...	204.35 KB		⋮
SRR10060895.bam	10-09-2019 22:3...	269.79 MB	View in IGB ...	⋮
SRR10060895.bam.bai	10-09-2019 16:4...	192.55 KB		⋮
SRR10060896.bam	10-09-2019 23:4...	391.19 MB	View in IGB ...	⋮
SRR10060896.bam.bai	10-09-2019 22:1...	191.66 KB		⋮
SRR10060897.bam	10-09-2019 22:5...	1.83 GB	View in IGB ...	⋮
SRR10060897.bam.bai	10-09-2019 22:1...	236.84 KB		⋮

MetaData

SRR10060893.bam

Genome Version

Arabidopsis thaliana ▾

A_thaliana_Jun_2009 ▾

IGB Track Color

Foreground: 🟩

Background: □

IGB Track Name (Optional)

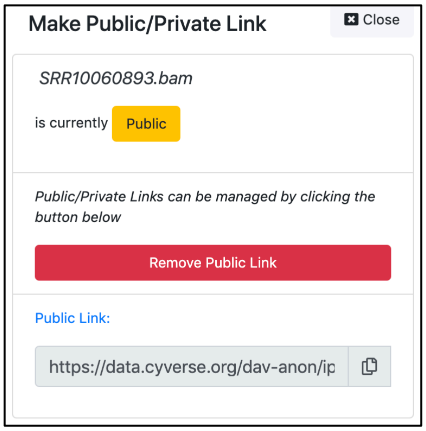
SRR10060893 - Watered control

Comments

Control 25 day old Arabidopsis thaliana
(Pot D Sole Collection Time 2 Wet)

Fig. 1 Example BioViz *Connect* main page and data visualization. **a** BioViz *Connect* main page. The left panel shows shortcuts to home, shared, community folders. The middle panel lists files and folders. The right panel shows the selected file's metadata. **b** SRR10060893.bam and SRR10060894.bam files viewed in IGB overlapping the SR45a gene of *Arabidopsis thaliana*.

The flow of data from CyVerse into IGB depends on data files having publicly accessible URLs hosted on servers that IGB can use to retrieve data. It also depends on the server software supporting HTTP byte range requests, which allows client software to request arbitrarily small or large sections using byte addresses and thus avoid downloading entire files during the visualization session. To ensure users have total control of file visibility, BioViz *Connect* provides an easy way to activate or de-activate data file URLs when needed. The context menu shown in Figure 1A includes an option to "Manage Link," which opens the right-panel display (Fig. 2). Similar to sharing in commercial cloud storage products, users can choose to make individual files public or private. Public links are viewable to anyone with the link, including non-CyVerse users. This lets researchers easily share data with colleagues without their needing to log into CyVerse first. They can open the file from inside IGB by using its "Open URL" feature. As a convenience for users, clicking the "View in IGB" button on private files triggers a dialog for creating the public URL for the data set.



SRR10060893.bam

is currently Public

Public/Private Links can be managed by clicking the button below

Public Link:

https://data.cyverse.org/dav-anon/ip

Fig. 2 Right-panel showing Public/Private options for a file. The file is currently set as public, with a public link displayed at the bottom.

In addition to the above example, the experiment includes another two files from biological replicates with different sizes. The file SRR10060911.bam is 1.83 Gb, but SRR10060912.bam is only 0.454 Gb. Viewing the "Notes" section in the meta-data for each file shows that both samples are from plants that underwent a 3-hour, heat stress treatment. When we open and view these samples in IGB, we see that the pattern of alignments within them is similar, but there are fewer alignments from the smaller file (Fig. 3).

To summarize the amount of sequencing that was done, we use a visual analytics feature within IGB that lets us create coverage graphs using data from the read alignment tracks. To make a coverage graph, we right-click a track label for a read alignment track and choose option "Track Operations > Depth Graph (All)". This generates a new derived track showing a graph in which the y-axis indicates the number of sequences aligned per position indicated on the x-axis, corresponding to base pair positions in the genome. IGB displays a vertical, yellow highlight bar to signal when a track derives from another track and does not come from a separate data file. After modifying the y-axis lower and upper boundary values (using controls in IGB's Graph tab), we see that although the pattern of alignments was similar between the two samples, the overall level of sequencing was different. Sample SRR10060912 has less read coverage across SR45a than SRR10060911 (Fig. 3). Thus, the file size difference reflects different amounts of sequencing rather than a problem with the library synthesis, as was the case in the previous example.

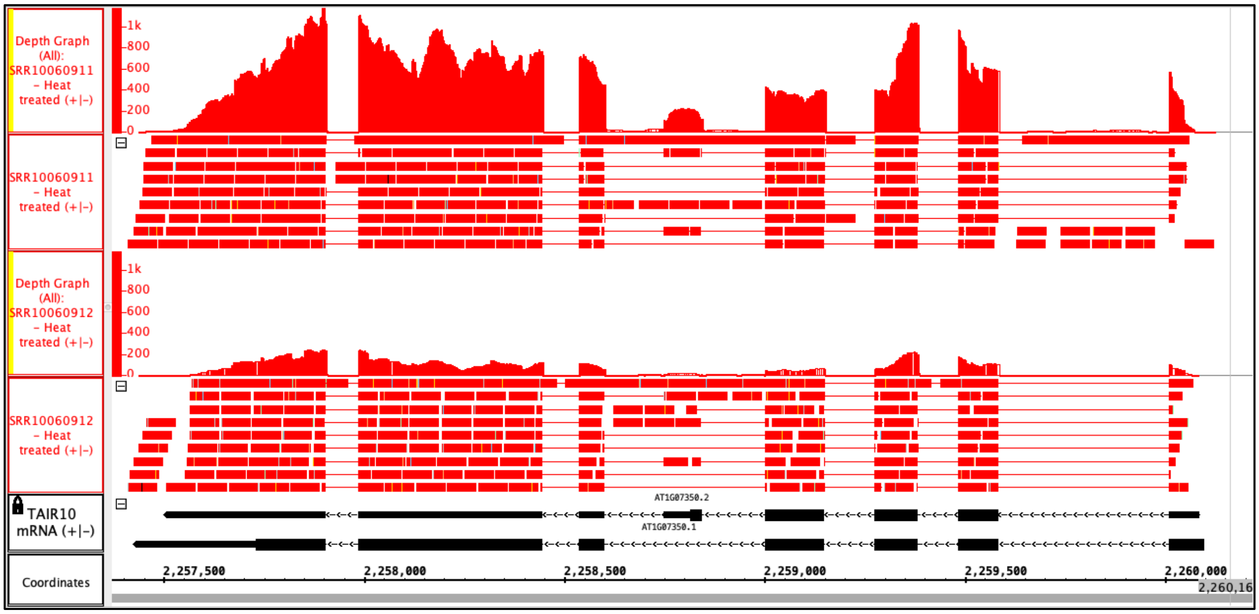

Depth Graph (All): SRR10060911 – Heat treated (+|-)
1k
800
600
400
200
0
SRR10060911 – Heat treated (+|-)
Depth Graph (All): SRR10060912 – Heat treated (+|-)
1k
800
600
400
200
0
SRR10060912 – Heat treated (+|-)
TAIR10 mRNA (+|-)
AT1G07350.2
AT1G07350.1
Coordinates
2,257,500
2,258,000
2,258,500
2,259,000
2,259,500
2,260,000
2,260,16

Fig. 3 Heat treated samples viewed in IGB. Alignment tracks and their derived depth graph tracks are shown. The y-axis values represent the number of aligned sequences per base pair position indicated on the coordinates track. Gene models are from the SR45a gene.

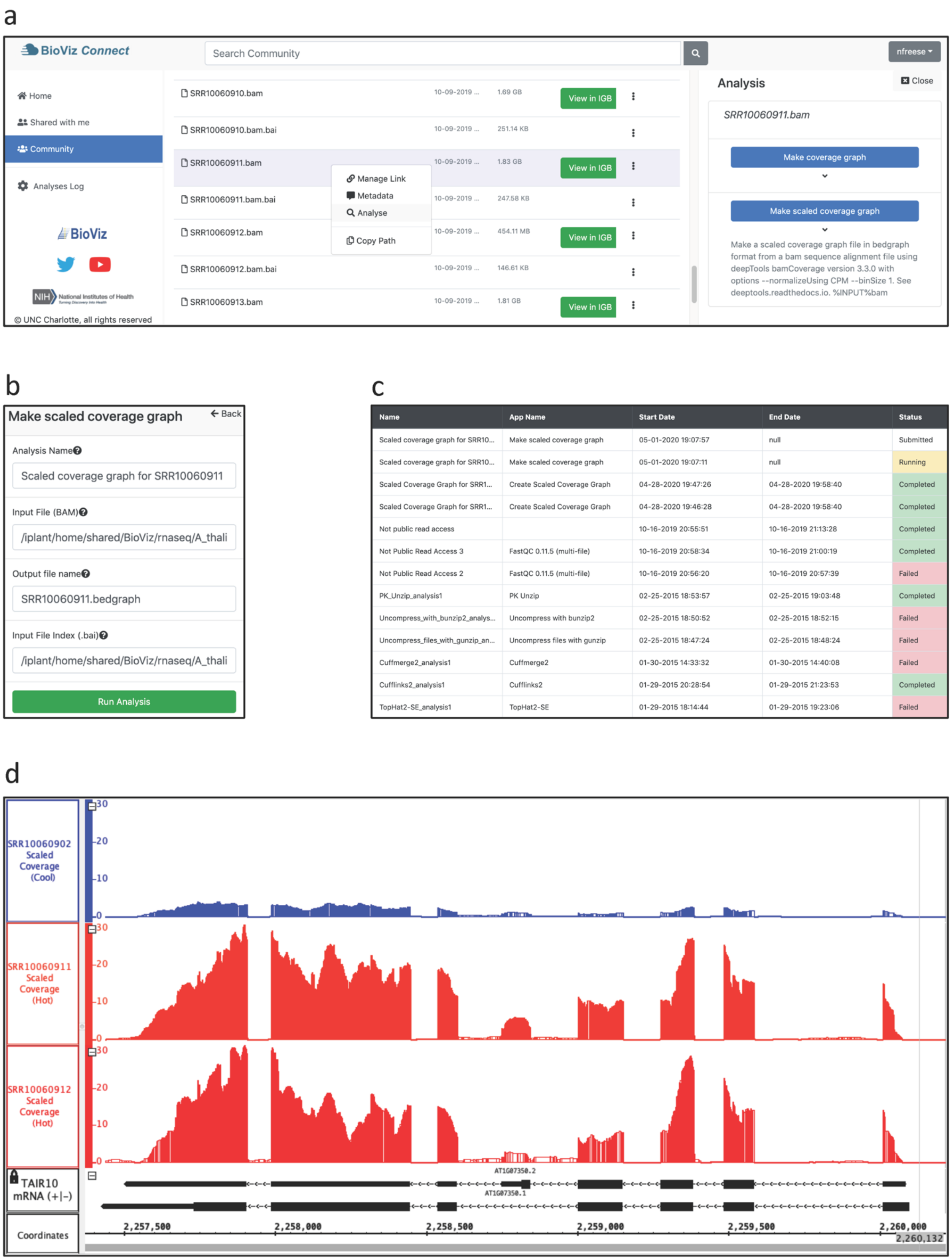
Summarizing read alignments with coverage graphs can provide a fast way to compare expression across sample types, but only if the libraries were sequenced to approximately the same depth. To properly use coverage graphs to assess expression, we need to scale or normalize the graphs to account for differences in sequencing depth. Performing this transformation within IGB itself is impractical, however, as it would require downloading, reading, and processing the entire bam-format alignments file. A much better approach is to off-load computationally intensive visual analytics tasks to CyVerse HPC resources. To do this, we again take advantage of Terrain APIs, this time using APIs that provide access to pre-built data analysis functions called "CyVerse Apps." These Apps provide high-performance-computing

(HPC) via APIs, and BioViz *Connect* uses these to generate scaled genome-wide coverage graphs for researchers without taxing their local resources.

To create a scaled coverage graph, we return to BioViz *Connect*, right-click a bam format file, and choose "Analyse." This opens the Analysis right-panel display, which lists CyVerse Apps that can accept the selected file type as input (Fig. 4A). The CyVerse ecosystem has a large number of Apps with similar names that are maintained by different groups, and so to avoid confusing users, BioViz *Connect* displays a curated subset of published Apps that produce output relevant to data visualization in IGB. When a user selects a file and clicks "Analyse", BioViz *Connect* checks its database of curated Apps to determine which can accept the file as input and then displays them in the right panel. When we select an App, BioViz *Connect* retrieves App details from the Terrain API and dynamically creates an entry form (using HTML5) we use to enter options for how we want the App to run.

Selecting "Make scaled coverage graph" opens a form with options for creating the graph using the genomeCoverage function from the DeepTools suite of python-based command line genome analysis tools (Fig. 4B) [27]. To run the tool, we enter a name for the file that will be created from the bam file and also select the BAM file's companion index (bai) file, a required input. Next, we click "Run Analysis," which submits a request to run the App using the CyVerse HPC resources. Once we submit our request, also called a "job," we can check its status by looking it up in our Analyses Log (Fig. 4C). Jobs are listed as Queued, Running, Failed, or Completed. The CyVerse infrastructure sends us an email when the job finishes, and we can find the output, a bedgraph format file, in the same folder as the input bam file. If we do not have permission to write to this folder, as is the case with Community folder, then the output file is saved to our home directory. Clicking the job name in the Analyses Log opens the folder where the output data are stored.

Figure 4D shows scaled coverage graphs created for the two RNA-Seq samples described above. Examining the scaled coverage graphs for SR45a shows that both heat-treated samples had similar scaled expression (Fig. 4D). This indicates that the difference observed in the unscaled coverage graph was likely an artifact of lower sequencing depth rather than depressed expression of the gene in one replicate versus the other. Figure 4D also shows a scaled coverage graph (labeled "cool") from a control sample that received no heat stress treatment. The control sample's scaled coverage graph exhibits approximately 10-fold lower expression (measured as y-axis values) across the spliced regions of the gene, visually demonstrating that the heat stress treatment increased SR45a transcript abundance during stress [25].



a

BioViz Connect

Search Community

nfreese

Home
Shared with me
Community
Analyses Log

SRR10060910.bam	10-09-2019 ...	1.69 GB	View in IGB
SRR10060910.bam.bai	10-09-2019 ...	251.14 KB	
SRR10060911.bam	10-09-2019 ...	1.83 GB	View in IGB
SRR10060911.bam.bai	10-09-2019 ...	247.58 KB	
SRR10060912.bam	10-09-2019 ...	454.11 MB	View in IGB
SRR10060912.bam.bai	10-09-2019 ...	146.61 KB	
SRR10060913.bam	10-09-2019 ...	1.81 GB	View in IGB

Manage Link
Metadata
Analyse
Copy Path

BioViz

NIH National Institutes of Health

© UNC Charlotte, all rights reserved

Analysis

Close

SRR10060911.bam

Make coverage graph

Make scaled coverage graph

Make a scaled coverage graph file in bedgraph format from a bam sequence alignment file using deepTools bamCoverage version 3.3.0 with options --normalizeUsing CPM --binSize 1. See deeptools.readthedocs.io. %INPUT%bam

b

Make scaled coverage graph

Back

Analysis Name: Scaled coverage graph for SRR10060911

Input File (BAM): /iplant/home/shared/BioViz/rnaseq/A_thali

Output file name: SRR10060911.bedgraph

Input File Index (.bai): /iplant/home/shared/BioViz/rnaseq/A_thali

Run Analysis

c

Name	App Name	Start Date	End Date	Status
Scaled coverage graph for SRR10...	Make scaled coverage graph	05-01-2020 19:07:57	null	Submitted
Scaled coverage graph for SRR10...	Make scaled coverage graph	05-01-2020 19:07:11	null	Running
Scaled Coverage Graph for SRR1...	Create Scaled Coverage Graph	04-28-2020 19:47:26	04-28-2020 19:58:40	Completed
Scaled Coverage Graph for SRR1...	Create Scaled Coverage Graph	04-28-2020 19:46:28	04-28-2020 19:58:40	Completed
Not public read access		10-16-2019 20:55:51	10-16-2019 21:13:28	Completed
Not Public Read Access 3	FastQC 0.11.5 (multi-file)	10-16-2019 20:58:34	10-16-2019 21:00:19	Completed
Not Public Read Access 2	FastQC 0.11.5 (multi-file)	10-16-2019 20:56:20	10-16-2019 20:57:39	Failed
PK_Unzip_analysis1	PK Unzip	02-25-2015 18:53:57	02-25-2015 19:03:48	Completed
Uncompress_with_bunzip2_analys...	Uncompress with bunzip2	02-25-2015 18:50:52	02-25-2015 18:52:15	Failed
Uncompress_files_with_gunzip_an...	Uncompress files with gunzip	02-25-2015 18:47:24	02-25-2015 18:48:24	Failed
Cuffmerge2_analysis1	Cuffmerge2	01-30-2015 14:33:32	01-30-2015 14:40:08	Failed
Cufflinks2_analysis1	Cufflinks2	01-29-2015 20:28:54	01-29-2015 21:23:53	Completed
TopHat2-SE_analysis1	TopHat2-SE	01-29-2015 18:14:44	01-29-2015 19:23:06	Failed

Fig. 4. Example analysis in BioViz *Connect* with output visualized in IGB. **a** BioViz *Connect* main page with analysis right panel open. **b** Scaled coverage graph analysis options for naming the analysis, selecting input file, output file name, and index file selection. **c** Analyses log showing the status of current and previous jobs. **d** SRR10060902 (control), SRR10060911 (heat treated), and SRR10060912 (heat treated) scaled bedgraph files viewed in IGB overlapping the SR45a gene of *Arabidopsis thaliana*.

Conclusions

We built BioViz *Connect*, a Web application uniting the extensive data storage and HPC resources provided by CyVerse with the powerful desktop interactivity provided by Integrated Genome Browser. BioViz *Connect* offers a familiar user interface for viewing files and folders, attaching metadata to files, running analyses, and sending data to IGB for visualization. As one of the first web applications to use the Terrain API outside the CyVerse ecosystem, BioViz *Connect* demonstrates the power and flexibility of combining cloud resources with local desktop performance to optimize whole genome analyses and interactive visualization.

Acknowledgments

We thank Paul Sarando, Sarah Roberts, Ian McEwen, Ramona Walls, and Reetu Tuteja for their assistance with the Terrain API and publishing CyVerse apps, which was made possible through CyVerse's External Collaborative Partnership program.

Abbreviations

IGB: Integrated Genome Browser; HPC: High Performance Computing; iRods: Integrated Rule-Oriented Data System; REST: REpresentational State Transfer; API: Application Programming interface; AVU: Attribute Value Unit; HTML: HyperText Markup Language; CSS: Cascading Style Sheets

Funding

Research reported in this publication was supported by the National Institute of General Medical Sciences of the National Institutes of Health under award number 5R01GM121927-02.

Availability of data and materials

BioViz *Connect* is available from https://bioviz.org/connect.html. RNA-Seq data files described here are available from https://data.cyverse.org/dav-anon/iplant/home/shared/BioViz/rnaseq.

Authors' contributions

NHF and AEL conceived of and supervised the project. KR, CK, ST, and PB planned and developed BioViz *Connect*. NHF, AEL, KR, CK, ST, PB tested and debugged BioViz *Connect*. NHF, KR, CK, and AEL wrote the draft manuscript. All authors read and approved the final manuscript.

Competing interests

The authors declare that they have no competing interests.

REFERENCES CITED

1. Freese NH, Norris DC, Loraine AE: **Integrated genome browser: visual analytics platform for genomics**. *Bioinformatics* 2016, **32**(14):2089-2095.
2. Nicol JW, Helt GA, Blanchard SG, Jr., Raja A, Loraine AE: **The Integrated Genome Browser: free software for distribution and exploration of genome-scale datasets**. *Bioinformatics* 2009, **25**(20):2730-2731.
3. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T: **Cytoscape: a software environment for integrated models of biomolecular interaction networks**. *Genome research* 2003, **13**(11):2498-2504.
4. Kim D, Langmead B, Salzberg SL: **HISAT: a fast spliced aligner with low memory requirements**. *Nature methods* 2015, **12**(4):357-360.
5. Kim D, Pertea G, Trapnell C, Pimentel H, Kelley R, Salzberg SL: **TopHat2: accurate alignment of transcriptomes in the presence of insertions, deletions and gene fusions**. *Genome biology* 2013, **14**(4):R36.

6. Clark S, Yu F, Gu L, Min XJ: **Expanding Alternative Splicing Identification by Integrating Multiple Sources of Transcription Data in Tomato**. *Front Plant Sci* 2019, **10**:689.

7. Liu Y, Wei H, Ma M, Li Q, Kong D, Sun J, Ma X, Wang B, Chen C, Xie Y *et al*: **Arabidopsis FHY3 and FAR1 Regulate the Balance between Growth and Defense Responses under Shade Conditions**. *The Plant cell* 2019, **31**(9):2089-2106.

8. Peng Y, Xiong D, Zhao L, Ouyang W, Wang S, Sun J, Zhang Q, Guan P, Xie L, Li W *et al*: **Chromatin interaction maps reveal genetic regulation for quantitative traits in maize**. *Nature communications* 2019, **10**(1):2632.

9. Sang Q, Pajoro A, Sun H, Song B, Yang X, Stolze SC, Andres F, Schneeberger K, Nakagami H, Coupland G: **Mutagenesis of a Quintuple Mutant Impaired in Environmental Responses Reveals Roles for CHROMATIN REMODELING4 in the Arabidopsis Floral Transition**. *The Plant cell* 2020, **32**(5):1479-1500.

10. Smith NMA, Yagound B, Remnant EJ, Foster CSP, Buchmann G, Allsopp MH, Kent CF, Zayed A, Rose SA, Lo K *et al*: **Paternally-biased gene expression follows kin-selected predictions in female honey bee embryos**. *Molecular ecology* 2020, **29**(8):1523-1533.

11. Wang C, Yu H, Luo L, Duan L, Cai L, He X, Wen J, Mysore KS, Li G, Xiao A *et al*: **NODULES WITH ACTIVATED DEFENSE 1 is required for maintenance of rhizobial endosymbiosis in Medicago truncatula**. *The New phytologist* 2016, **212**(1):176-191.

12. Zhang K, Yu L, Pang X, Cao H, Si H, Zang J, Xing J, Dong J: **In silico analysis of maize HDACs with an emphasis on their response to biotic and abiotic stresses**. *PeerJ* 2020, **8**:e8539.

13. Wilkins-Diehr N, Gannon D, Klimeck G, Oster S, Pamidighantam S: **TeraGrid Science Gateways and Their Impact on Science**. *Computer* 2008, **41**(11):32-41.

14. Giardine B, Riemer C, Hardison RC, Burhans R, Elnitski L, Shah P, Zhang Y, Blankenberg D, Albert I, Taylor J *et al*: **Galaxy: a platform for interactive large-scale genome analysis**. *Genome research* 2005, **15**(10):1451-1455.

15. Afgan E, Baker D, Batut B, van den Beek M, Bouvier D, Cech M, Chilton J, Clements D, Coraor N, Grüning BA *et al*: **The Galaxy platform for accessible, reproducible and**

collaborative biomedical analyses: 2018 update. *Nucleic Acids Res* 2018, **46**(W1):W537-W544.

16. Blankenberg D, Von Kuster G, Bouvier E, Baker D, Afgan E, Stoler N, Taylor J, Nekrutenko A: **Dissemination of scientific software with Galaxy ToolShed**. *Genome biology* 2014, **15**(2):403.

17. Batut B, Hiltemann S, Bagnacani A, Baker D, Bhardwaj V, Blank C, Bretaudeau A, Brillet-Guéguen L, Čech M, Chilton J *et al*: **Community-Driven Data Analysis Training for Biology**. *Cell systems* 2018, **6**(6):752-758.e751.

18. Blankenberg D, Chilton J, Coraor N: **Galaxy External Display Applications: closing a dataflow interoperability loop**. *Nature methods* 2020, **17**(2):123-124.

19. Goff SA, Vaughn M, McKay S, Lyons E, Stapleton AE, Gessler D, Matasci N, Wang L, Hanlon M, Lenards A *et al*: **The iPlant Collaborative: Cyberinfrastructure for Plant Biology**. *Front Plant Sci* 2011, **2**:34-34.

20. Merchant N, Lyons E, Goff S, Vaughn M, Ware D, Micklos D, Antin P: **The iPlant Collaborative: Cyberinfrastructure for Enabling Data to Discovery for the Life Sciences**. *PLoS Biol* 2016, **14**(1):e1002342-e1002342.

21. Allen WJ, Gabr RE, Tefera GB, Pednekar AS, Vaughn MW, Narayana PA: **Platform for Automated Real-Time High Performance Analytics on Medical Image Data**. *IEEE journal of biomedical and health informatics* 2018, **22**(2):318-324.

22. Sullivan A, Purohit PK, Freese NH, Pasha A, Esteban E, Waese J, Wu A, Chen M, Chin CY, Song R *et al*: **An 'eFP-Seq Browser' for visualizing and exploring RNA sequencing data**. *Plant J* 2019, **100**(3):641-654.

23. Berardini TZ, Reiser L, Li D, Mezheritsky Y, Muller R, Strait E, Huala E: **The Arabidopsis information resource: Making and mining the "gold standard" annotated reference plant genome**. *Genesis (New York, NY : 2000)* 2015, **53**(8):474-485.

24. Leinonen R, Sugawara H, Shumway M: **The sequence read archive**. *Nucleic Acids Res* 2011, **39**(Database issue):D19-21.

25. Gulledge AA, Roberts AD, Vora H, Patel K, Loraine AE: **Mining Arabidopsis thaliana RNA-seq data with Integrated Genome Browser reveals stress-induced alternative**

splicing of the putative splicing regulator SR45a. *American journal of botany* 2012, **99**(2):219-231.

26. Yoshimura K, Mori T, Yokoyama K, Koike Y, Tanabe N, Sato N, Takahashi H, Maruta T, Shigeoka S: **Identification of alternative splicing events regulated by an Arabidopsis serine/arginine-like protein, atSR45a, in response to high-light stress using a tiling array**. *Plant & cell physiology* 2011, **52**(10):1786-1805.

27. Ramirez F, Ryan DP, Gruning B, Bhardwaj V, Kilpert F, Richter AS, Heyne S, Dundar F, Manke T: **deepTools2: a next generation web server for deep-sequencing data analysis**. *Nucleic Acids Res* 2016, **44**(W1):W160-165.