Connecting Integrated Genome Browser to a huge genome database using its open API solves one problem and creates another – and an opportunity.

Ann Loraine^{1,2}, Jaya Sravani Sirigineedi², Nowlan Freese²

¹Contact Email: <u>aloraine@charlotte.edu</u>

² University of North Carolina at Charlotte, 9201 University City Blvd, Charlotte, NC 28223

Project Website: https://bioviz.org, http://igbquickload-main.bioviz.org/quickload Source Code: https://bitbucket.org/lorainelab/integrated-genome-browser License: Common Public License Version 1.0

Integrated Genome Browser (IGB, pronounced "ig-bee") is a fast, feature-rich, open-source desktop genome browser thousands of researchers have used to explore and analyze genomic data. To support our user audience, we maintain data delivery Web sites called "IGB Quickloads" that supply IGB with 97 reference genome assemblies as of this writing. Of course, IGB can open user-provided genome assembly files, but if a researcher's desired assembly already exists in an IGB Quickload, they can avoid this inconvenient work. However, we are finding it increasingly difficult to update these Quickload sites as new assemblies are published. Fortunately, many genome database systems now offer robust computational access to their data. By accessing these computational resources, IGB could show new assemblies without our first re-formatting and copying them to a Quickload. To test this idea, we developed a new IGB version that consumes and displays data from one such resource, a JSON-emitting API (application programming interface) from the UCSC Genome Browser system. Now available as an "early access" version at the BioViz.org Web site, this new IGB version can display more than 200 assemblies visible in the UCSC Browser, along with hundreds of experimental data and annotation tracks the API also provides.

This wealth of annotations and experimental data tracks gives us a new problem to solve, but also an opportunity. The API provides information like track names and data formats, but little about what the data represent. IGB can offer users the option to load these tracks, but without meta-data documenting what they show, how can users understand and interpret what they see? We urge API authors to augment their APIs with richer meta-data about the data sets, such as references to publications, documentation on the data providers' own web sites, or even ontology terms that formally classify the data sets. With such meta-data, we could create search interfaces for users to find all available data relevant to their scientific interests, reducing their risk of overlooking relevant, previously published work. By providing more complete meta-data, the UCSC team and others like them could make their amazing resources accessible to even more audiences – developers like us and the worldwide communities we serve.

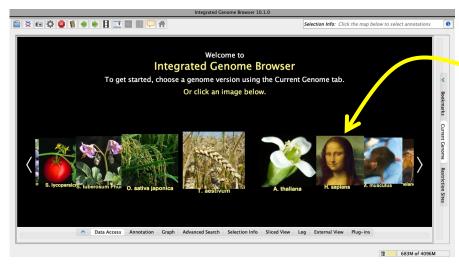
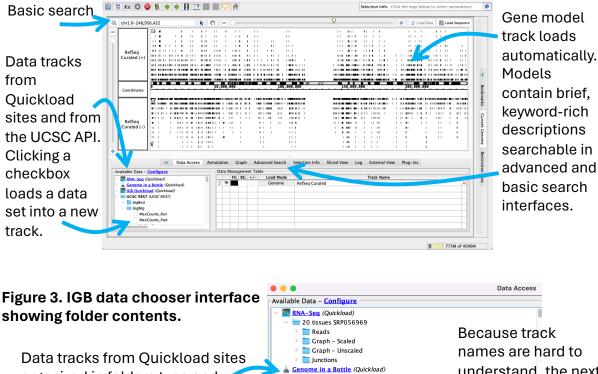


Figure 1. IGB start screen showing species with available genomes

Clicking an image opens the newest assembly available via from IGB Quickload for the selected species. Reference gene models annotated with brief functional descriptions also load. Keeping these data up-to-date is hard.



AshkenazimTrio ChineseTrio

> MaxCounts_Fwd ReMapDensity

VeinAdult_CNhs12844_tpm_rev VeinAdult_CNhs12844_tpm_fwd

VaginaAdult_CNhs12854_tpm_rev VaginaAdult_CNhs12854_tpm_fwd

UterusFetalDonor1_CNhs11763_tpm_rev UterusFetalDonor1_CNhs11763_tpm_fwd

NA12878
IGB Quickload (Quickload)
UCSC REST (UCSC REST)

🚞 bigBed

bigWig MaxCounts Rev

Figure 2. IGB showing the newest human genome assembly available via Quickload.

Data tracks from UCSC API sites organized in folders by data format, labeled using UCSC track names.

organized in folders type and

source.

We hope new API versions will provide more information about these experiment, including links to publications and other documentation. This would help users decide which tracks to load and interpret what they see once the data appear in IGB. Because track names are hard to understand, the next IGB iteration will also show "short labels" and "long labels" provided by the API.

We also plan to add Web search capability.

UterusAdultPool1 CNhs11676 tpm rev UterusAdultPool1_CNhs11676_tpm_fwd UrethraDonor2_CNhs13464_tpm_rev UrethraDonor2 CNhs13464 tpm fwd UniversalRNAHumanNormalTissuesBiochainPool1_CNhs10612_tpm_rev UniversalRNAHumanNormalTissuesBiochainPool1_CNhs10612_tpm_fwd UmbilicalCordFetalDonor1 CNhs11765 tpm rev UmbilicalCordFetalDonor1_CNhs11765_tpm_fwd TracheaFetalDonor1_CNhs11766_tpm_rev TracheaFetalDonor1 CNhs11766 tpm fwd TracheaAdultPool1_CNhs10635_tpm_rev TracheaAdultPool1_CNhs10635_tpm_fwd TonsilAdultPool1 CNhs10654 tpm rev TonsilAdultPool1_CNhs10654_tpm_fwd TongueFetalDonor1_CNhs11768_tpm_rev TongueFetalDonor1 CNhs11768 tpm fwd TongueEpidermisFungiformPapillaeDonor1_CNhs13460_tpm_rev TongueEpidermisFungiformPapillaeDonor1_CNhs13460_tpm_fwd TongueAdult CNhs12853 tpm rev TongueAdult_CNhs12853_tpm_fwd ThyroidFetalDonor1_CNhs11769_tpm_rev ThyroidFetalDonor1 CNhs11769 tom fwd ThyroidAdultPool1_CNhs10634_tpm_rev ThyroidAdultPool1_CNhs10634_tpm_fwd ThymusFetalPool1 CNhs10650 tpm rev ThymusFetalPool1_CNhs10650_tpm_fwd ThymusAdultPool1_CNhs10633_tpm_rev ThymusAdultPool1 CNhs10633 tpm fwd ThroatFetalDonor1 CNhs11770 tpm rev