PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

Apollo: Democratizing genome annotation

Nathan A. Dunn[1]*, Deepak R. Unni[1], Colin Diesh[2], Monica Munoz-Torres[3], Nomi L. Harris[1], Eric Yao[2], Helena Rasche[4], Ian H. Holmes[2], Christine G. Elsik[5], Suzanna E. Lewis[1]

1 Environmental Genomics and Systems Biology, Lawrence Berkeley National Laboratory, Berkeley, California, United States of America, **2** Deptartment of Bioengineering, University of California, Berkeley, California, United States of America, **3** Translational and Integrative Sciences Lab, Oregon State University, Corvallis, Oregon, United States of America, **4** Bioinformatics Group, University of Freiburg, Freiburg, Germany, **5** Animal Sciences Division, University of Missouri, Columbia, Missouri, United States of America

* nathandunn@lbl.gov

OPEN ACCESS

Citation: Dunn NA, Unni DR, Diesh C, Munoz-Torres M, Harris NL, Yao E, et al. (2019) Apollo: Democratizing genome annotation. PLoS Comput Biol 15(2): e1006790. https://doi.org/10.1371/journal.pcbi.1006790

Editor: Aaron E. Darling, University of Technology Sydney, AUSTRALIA

Received: November 30, 2018

Accepted: January 10, 2019

Published: February 6, 2019

Data Availability Statement: Apollo (https://github.com/GMOD/Apollo) is licensed under BSD-3. It can be built from source, and is also available via docker (https://github.com/GMOD/docker-apollo). It requires a Java Development Kit (JDK) 1.8 and Node.js 6 or better. We also provide a User Guide, a public demo (http://genomearchitect.org) and information about joining our mailing list (apollo@lists.lbl.gov).

Funding: This work was funded by R01-GM080203 from the National Institute of General Medicine Sciences (https://www.nigms.nih.gov/)

Abstract

Genome annotation is the process of identifying the location and function of a genome's encoded features. Improving the biological accuracy of annotation is a complex and iterative process requiring researchers to review and incorporate multiple sources of information such as transcriptome alignments, predictive models based on sequence profiles, and comparisons to features found in related organisms. Because rapidly decreasing costs are enabling an ever-growing number of scientists to incorporate sequencing as a routine laboratory technique, there is widespread demand for tools that can assist in the deliberative analytical review of genomic information. To this end, we present Apollo, an open source software package that enables researchers to efficiently inspect and refine the precise structure and role of genomic features in a graphical browser-based platform. Some of Apollo's newer user interface features include support for real-time collaboration, allowing distributed users to simultaneously edit the same encoded features while also instantly seeing the updates made by other researchers on the same region in a manner similar to Google Docs. Its technical architecture enables Apollo to be integrated into multiple existing genomic analysis pipelines and heterogeneous laboratory workflow platforms. Finally, we consider the implications that Apollo and related applications may have on how the results of genome research are published and made accessible.

This is a *PLOS Computational Biology* Software paper.

Introduction

Apollo's design is based on the premise that the best genomic descriptions ('annotations') can be produced by starting with automatically-generated sequence features and then providing expert researchers with interactive editing tools to examine these multiple sources of evidence and collaboratively refine the genomic annotations. The first version of Apollo was a stand-alone desktop application [1]. As software technologies advanced, each new generation of

http://grantome.com/grant/NIH/R01-GM080203-01 to the PI Suzanna Lewis used to fund the majority of this work (https://taggs.hhs.gov/Detail/AwardDetail?arg_AwardNum=R01GM080203&arg_ProgOfficeCode=127). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing interests: The authors have declared that no competing interests exist.

Apollo took advantage of these to improve the user experience. The most fundamental change occurred circa 2010 when Apollo transitioned to running inside a web browser [2]. Once Apollo became a web-based application that permits real-time collaboration, the user base grew to include research and teaching environments studying a wide variety of species. Our most recent version of Apollo [3] offers a broad range of researchers an accessible way to improve the biological precision of their genomic feature descriptions.

Organizations that use Apollo include Echinobase [4], Hymenoptera Genome Database [5], i5k Workspace [6], PhytoPath [7], TreeGenes [8], Vectorbase [9] and XenBase [10]. To date, the i5K Workspace has supported publication of seven insect genomes that were manually curated with Apollo [11–17]. Other projects that have used Apollo include genomes of additional insects [18–20], human parasites [21–23], birds [24,25], the sea lamprey [26], plants [27–29], fungi [30–35] and a plant pathogenic nematode [36]. Projects such as the re-annotation of the whipworm genome by hundreds of high school students in the UK, supported by the Institute for Research in Schools (IRIS) [37], and the curation of 33,044 gene loci in the kiwifruit genome by 93 annotators, are evidence of Apollo's robust support for large dispersed projects.

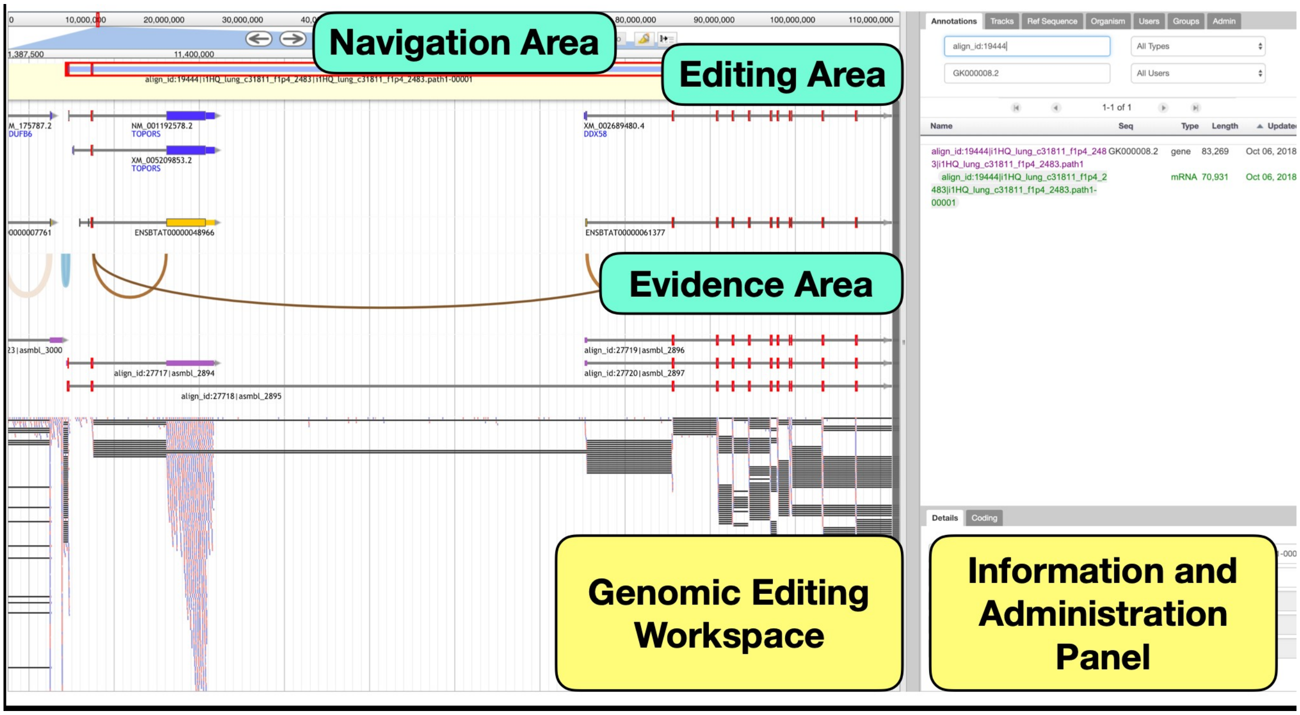
The ease of setting up Apollo makes it appealing to small projects as well as large. For example, one small group used Apollo to annotate 14 genes of a fungal mitochondrial genome [32]. Other reported Apollo use cases include annotating gene loci that pose challenges in automated gene prediction, such as the MHC-B region in the genome of the Mikado pheasant [25] and the effector complement of the flax rust pathogen Melampsora lini [33]. Through the process of gene model curation, the use of Apollo can reveal species-specific genome characteristics that can be used to improve automated gene prediction. For example, curation of some gene models of the yellow potato cyst nematode, *Globodera rostochiensis*, using RNAseq alignments as evidence, revealed a high frequency of non-canonical splice sites. Subsequent use of these manually curated genes as a training set markedly improved the automated gene predictions [36].

Thanks to its ability to simplify and accelerate annotation efforts for both large and small projects, Apollo's user base continues to grow. Since 2015, Apollo has had an annual growth rate of roughly 70% for returning users, peaking at over 2,700 unique users one day in late 2017, with a current average of around 1,000 unique users per month.

Apollo's integrated graphical environment allows users to browse and modify the location (s) and other information for a variety of feature types and streamlines common editing tasks by providing built-in calculations for features such as predicted proteins, splice sites, and gene set membership. An overview of the interface is shown in Fig 1.

To briefly describe the basic capabilities, Apollo's Genomic Editing Workspace (bottom left of Fig 1) displays tracks of information gathered from upstream pipelines and individual users' analyses. These provide the evidence (predictions and alignment) for refining genomic annotations. Any combination of features can be dragged from the evidence area into the editable area, where researchers carry out their edits without affecting the features from the evidence area. When evidence features are dropped into the editable track, they are assigned a default feature type of "protein coding transcript" and the longest open reading frame is automatically calculated, as well as its gene membership based on overlap with the CDS in the same reading frame as existing transcripts. Exon boundaries can be set either by dragging them upstream or downstream, or by using a menu option to set them to the nearest upstream or downstream splice junction (these are automatically calculated based on the configured donor and acceptor dinucleotides).

Apollo provides several ways to customize the display. From the track tab, in the information and administration panel on the right, users can select the specific evidence tracks they



align_id:19444

All Types

GK000008.2

All Users

1-1 of 1

Name	Seq	Type	Length	Updated
align_id:19444|i1HQ_lung_c31811_f1p4_2483|i1HQ_lung_c31811_f1p4_2483.path1	GK000008.2	gene	83,269	Oct 06, 2018
align_id:19444|i1HQ_lung_c31811_f1p4_2483|i1HQ_lung_c31811_f1p4_2483.path1-00001		mRNA	70,931	Oct 06, 2018

Fig 1. The Apollo Genome Editor has two main panels: A Genomic Editing Workspace and a closeable Information and Administration Panel which contains a range of configurable tabs. Within the **Genomic Editing Workspace**, the **Navigation Area** offers several ways to move to a region of interest. A user may: move upstream or downstream in fixed units; zoom in or out; or enter the coordinates or feature identifier to center on an exact genomic location. The **Evidence Area** contains data imported from local or remote files. In the **Editing Area** users can create annotations by dragging up evidence to create editable features of various types: coding and non-coding transcripts, pseudogenes, repeat regions, transposons, variant calls, transcription and translation start sites, and others. In the above example, the evidence area shows that there are reads spanning across exons that belong to two separate, previously known transcripts—NM_001192578.2 and XM_002689480.4. In the editing area, we add these known transcripts as annotations and then merge the two transcripts to create a single transcript. This newly created annotation then goes through additional refinements, to ensure that the transcript is a faithful representation of the evidence observed.

https://doi.org/10.1371/journal.pcbi.1006790.g001

want to view, categorize and filter tracks, and change the track order. The annotation tab lists every annotation across the genome, and can be searched by scaffold, identifier, researcher, or biological type. Information such as the gene symbol, description, cross-references, Gene Ontology functional class, links to publications, or general comments on each annotation may also be added from this tab. The reference sequence tab provides a sortable and searchable list of every scaffold, including the length, name, and number of annotations on each, for navigation across the genome.

Design and implementation

Apollo's design has always been driven by its users; their engagement in the development process has been a critical factor in Apollo's success. Over time the demographic of Apollo users has changed, with concomitant changes to Apollo's requirements. Notably, as sequencing costs have fallen, there are now a burgeoning number of projects targeting specific organisms, clades, or populations that frequently lack the funds or expertise to create their own software tools from scratch and are therefore reliant on available open source applications. Because members of these projects may be geographically distributed, they need tools that enable **real-time collaborative editing**. Additionally, annotating the effects different **variants** have on known genes has become a high priority research focus. And finally, particularly for collaborative projects, tracking the complete **annotation history** is crucial, not only for undo/redo operations but also to review the changes that have been made over time by different individuals.

Real-time collaborative editing

Apollo was designed with a standard client-server architecture (Fig 2) that can be run within a servlet container (e.g., Tomcat) and works with most relational database engines (e.g., PostgreSQL). The architecture provides a uniform authorization layer for external applications using its web services. For example, the i5K's project management software leverages Apollo web services to register new users and set appropriate user and group membership. The newly added users then have the necessary credentials to perform manual edits or utilize the same web services, allowing them to perform operations such as uploading bulk annotations.

Apollo's client interface is built as a JBrowse [38] plugin, a popular genome browser written in JavaScript. It provides the ability to import and export standard genomic data formats, flexible display of multiple types of genomic features, and fast scrolling and zooming. The primary editing client is a single-page application that embeds JBrowse. The server is built using Grails [39], an open source framework for developing web applications using Groovy [40] and other JVM languages. The Grails framework enables us to leverage well established technologies such as Spring (https://spring.io) for event control, the Grails Object Relational Mapper (GORM), Hibernate (http://hibernate.org) for efficiently mapping data objects to a backend persistent store, Ivy and Maven for build and plugin dependencies, and Grails plugins for security and navigation. Communication between the client and the server is provided through a REST API secured by the Apache Shiro library (https://shiro.apache.org/). To support integration into larger workflows, the web services that support user-interface functionality are fully exposed.

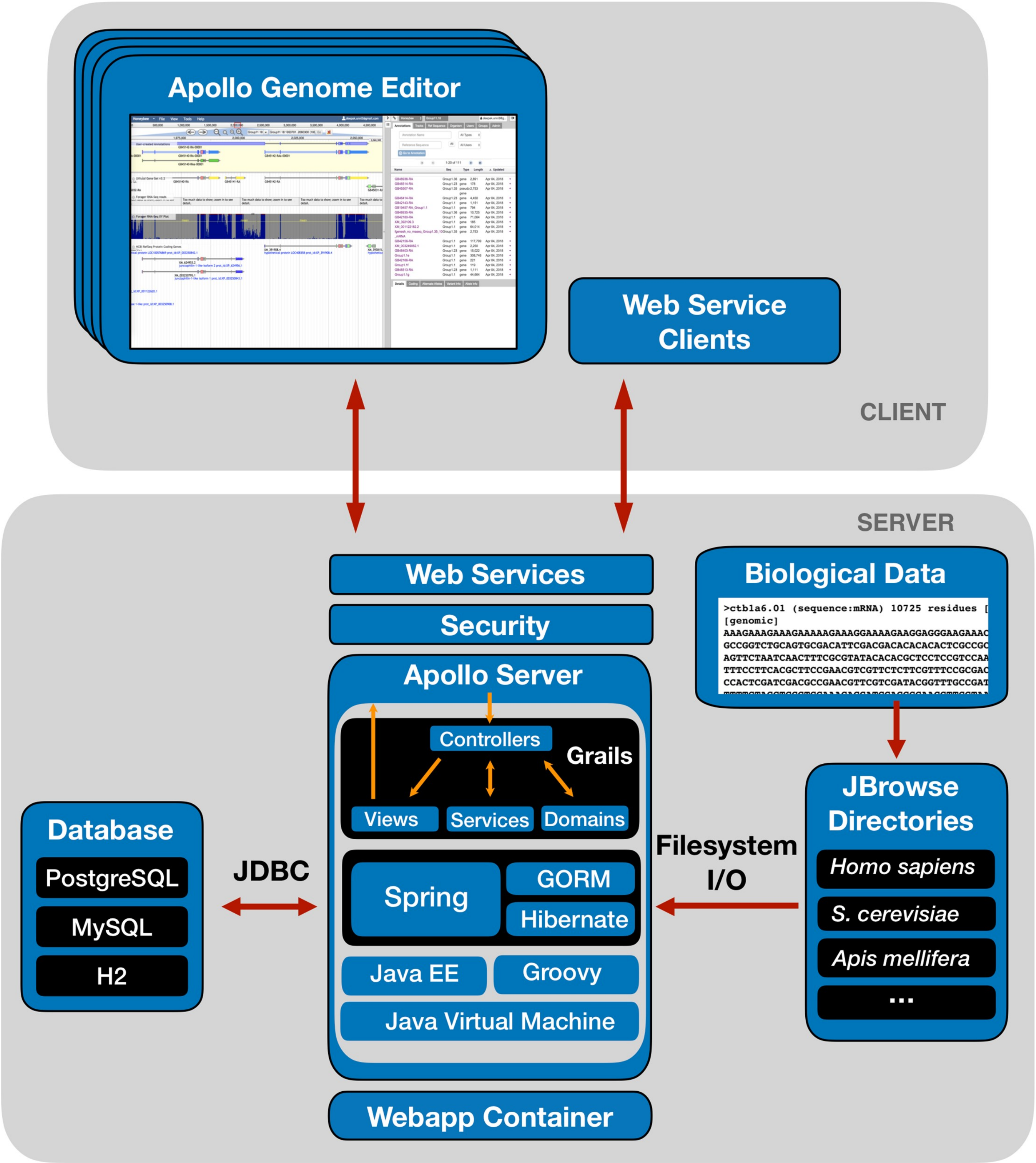
Concurrent editing by multiple users is implemented via WebSockets. WebSockets are well-supported in most recent web browsers and are an ideal technology to support push operations to all connected clients efficiently in real-time. Once a user's client connects to the server, WebSockets keep the line open for subsequent communication, including any structural and functional editing operations. This makes every annotation update in one client instantaneously visible in every other client. Apollo uses the STOMP (Streaming Text Oriented Messaging Protocol) protocol which uses a publish and subscribe communication style, minimizing communication overhead. WebSockets provide a robust and performant solution for pushing updates to multiple web clients that can fall back to a more traditional long-polling approach when client support is lacking as in older browsers.

Variants

In addition to allowing genomic features to be viewed and edited, Apollo provides the ability to annotate alterations in the underlying genomic sequence and visualize their impact (Fig 3). These may be assembly error *corrections*, to correct errors introduced in the sequencing and/or assembly process (a common issue when dealing with low-coverage genome sequences). Or these may be naturally occurring *variants*, genomic differences found among different members of a population. The effect of the annotated variants are reflected within the annotated genomic features they intersect with.

Annotation history

As researchers progressively refine the sequence features on a genomic region, information is automatically recorded for every change they make: what change was made; the time and date



```
>ctbla6.01 (sequence:mRNA) 10725 residues [
[genomic]
AAAGAAAGAAAGAAAAAGAAAGGAAAAGAAGGAGGGAAGAAAC
GCCGGTCTGCAGTGCGACATTCGACGACACACACACTCGCCGC
AGTTCTAATCAACTTTCGCGTATACACACGCTCCTCCGTCCAA
TTTCCTTCACGCTTCCGAACGTCGTTCTCTTCGTTTCCGCGAC
CCACTCGATCGACGCCGAACGTTCGTCGATACGGTTTGCCGAT
```

Fig 2. Apollo's client-server architecture. The server is built over the Grails framework using a relational database backend (RDBMS), for example PostgreSQL, MySQL or H2. Genomic evidence is provided by pointing to existing JBrowse directories that contain processed biological data. The Apollo Genome Editor and Web Services clients communicate with the server via REST and WebSockets.

https://doi.org/10.1371/journal.pcbi.1006790.g002

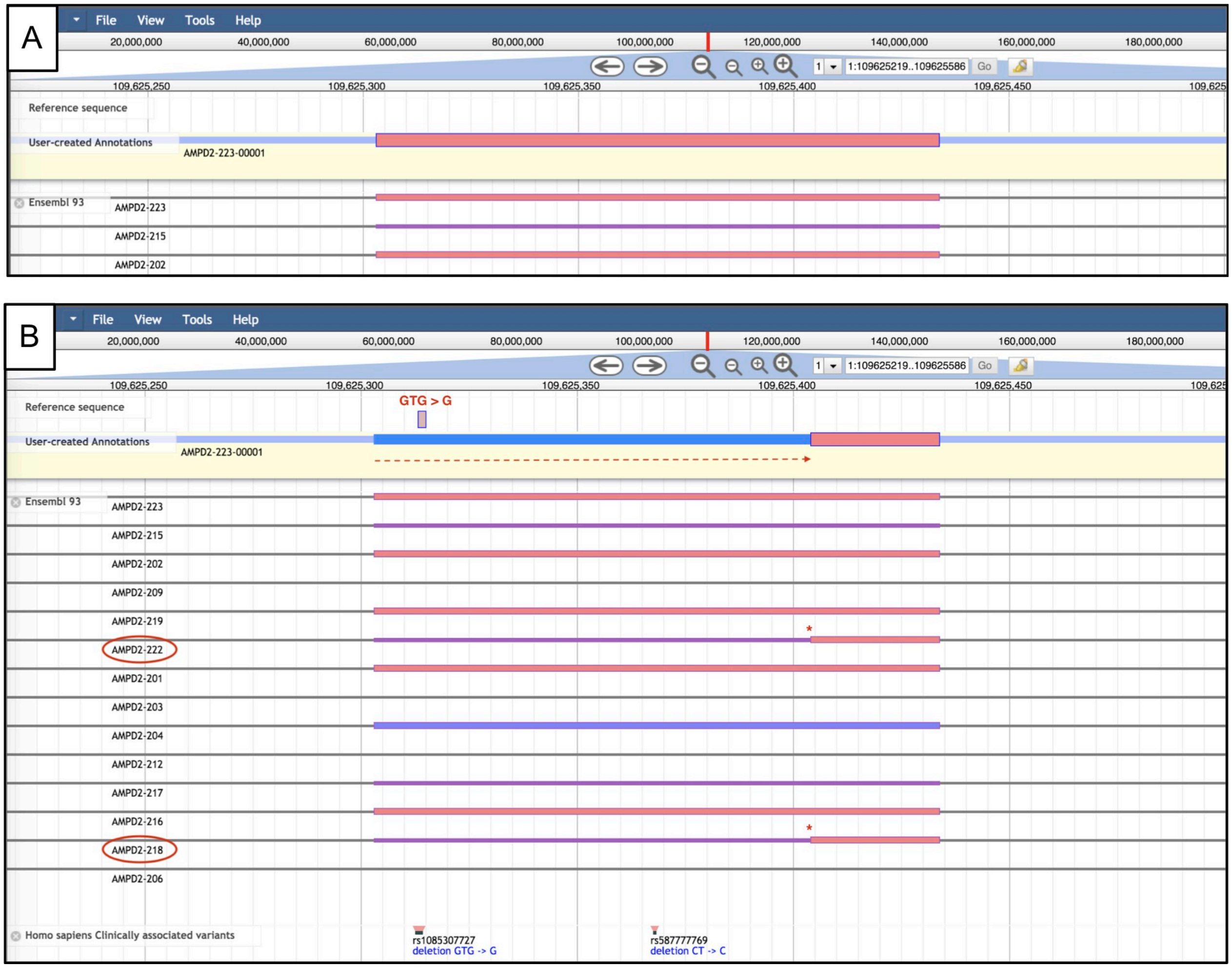


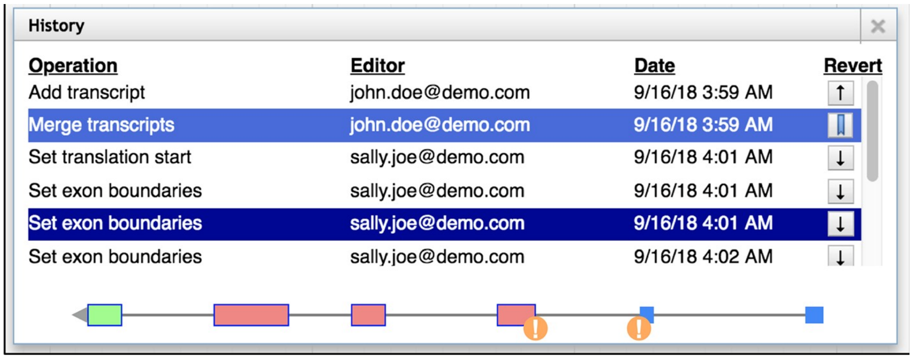
Fig 3. Example of variant annotation in Apollo. A) AMPD2-223, an isoform of gene AMPD2 as seen from the Evidence Area (truncated for space). From AMPD2-223, an isoform is dragged into the Editing Area. B) The deletion variant rs1085307727, from the 'Homo sapiens Clinically associated variants' track, overlaps with AMPD2-223-00001. Creating a corresponding deletion in the Editing Area of the Sequence Track allows visualization of the effect of the variant on transcript AMPD2-223-00001. Here, the transcription start site has moved further downstream, as indicated by the red dashed line. In this case, the altered form of the transcript recapitulates other alternate isoforms for this gene (AMDP2-218 and AMPD2-222), which are circled and starred for clarity.

https://doi.org/10.1371/journal.pcbi.1006790.g003

of the change; and the username (or email) of the editor. This edit history was a key design requirement, ensuring that all changes made are captured in a revertible, visual history of structural edits (Fig 4), which lets users graphically navigate through the different versions and roll back if necessary.

Results

Apollo's wide appeal across research projects of various sizes that focus on various organisms owes much to the many years of engagement between Apollo developers and its user community. In working with its users to maximize Apollo's utility for their breadth of organisms and purposes, it became clear to the development team that successful widespread uptake of Apollo depends on ensuring 1) reliable **scalability** so it can transparently handle very large genomes, a large number of genomes, and multiple users; 2) smooth **integration** into each group's technical environment; 3) a range of **customization** to accommodate different biological situations



History

Operation	Editor	Date	Revert
Add transcript	john.doe@demo.com	9/16/18 3:59 AM	↑
Merge transcripts	john.doe@demo.com	9/16/18 3:59 AM	▮
Set translation start	sally.joe@demo.com	9/16/18 4:01 AM	↓
Set exon boundaries	sally.joe@demo.com	9/16/18 4:01 AM	↓
Set exon boundaries	sally.joe@demo.com	9/16/18 4:01 AM	↓
Set exon boundaries	sally.joe@demo.com	9/16/18 4:02 AM	↓

Fig 4. The history navigator allows visual navigation of genomic edits as well as the ability to return to previous versions. The current version is indicated with a bookmark icon (in the Revert column). Users can select any version from the history, and make edits starting from that version if desired. The orange circles with an exclamation point indicate non-canonical splice sites.

https://doi.org/10.1371/journal.pcbi.1006790.g004

and project arrangements; and 4) direct engagement with users to encourage feedback and support **community contributions.**

Scalability

One of the major objectives when designing the architecture of the current version of Apollo was the ability for a single server to handle different dimensions of scale, whether it is thousands of genomes or large numbers of simultaneous users. We have encountered situations where a research group is studying many species in a particular clade; large, geographically distributed teams focused on a particular genomic region; and many students in a class working on team projects. Minimal requirements for Apollo are at least 500 MB of memory, or as much as several GB for optimal performance. However, with that allocation, we have optimized Apollo such that a single server can be successfully scaled to support several hundred genome projects and researchers. We tested and improved Apollo's performance and reliability via a combination of improved algorithms, optimized I/O requests, and more efficient database queries. As part of the testing process, we used a test suite that utilized the Apache JMeter load test tool, allowing the tool to simulate extraordinarily heavy read and write load over a sustained period. Additionally, we were able to scale up by modeling all organisms and users in a single database instance.

Ease of integration

Biological data and tools do not exist in a vacuum. To enjoy wide use, bioinformatics environments such as Apollo need to be able to smoothly integrate with multiple analysis tools and user interfaces.

Web services

Documented and secure web services are key to integrating any software into different bioinformatics ecosystems. Apollo exposes the methods used to drive the user interface as a web service, as well as providing services that support integration into different laboratories' existing environments. All methods are secured and require the same user permissions they would from the interactive browser application. Web services documentation is automatically generated from annotations within the software. There are many workflow environments that Apollo has been integrated into, typically after multiple alignment, filtering, and automated genome annotation steps. These environments include Galaxy [41] via the G-OnRamp project [42], GenSAS [43,44], DNA Subway [45], and the i5K workspace [6]. The i5K project leverages the user registration services, and the Galaxy Genome Annotation (GGA) project [46] automatically generates new projects in Apollo from data created via its biological workflow. The GGA project also provides a Python library for interacting with the Apollo API [47] and is used by projects such as BioInformatics Platform for Agroecosystem Arthropods (BIPAA) [48] and Texas A&M University Center for Phage Technology (TAMU-CPT) [49].

Import and export

Importing new information as it becomes available is essential for revealing additional genomic insights. Likewise, exporting the curated annotations provides corrected information for downstream analysis, such as protein motif profiling. In either direction, a variety of standard genomic data formats, such as GFF3, BAM, GTF, GVF, GenBank, VCF, BED, BigWig, or the Chado database [50] are supported. These import/export capabilities are also available via a REST endpoint for direct programmatic use in other applications. Additionally, JBrowse has a large number of other input/output plugins, and associated visualization widgets, (https://gmod.github.io/jbrowse-registry/), which can be made available within Apollo.

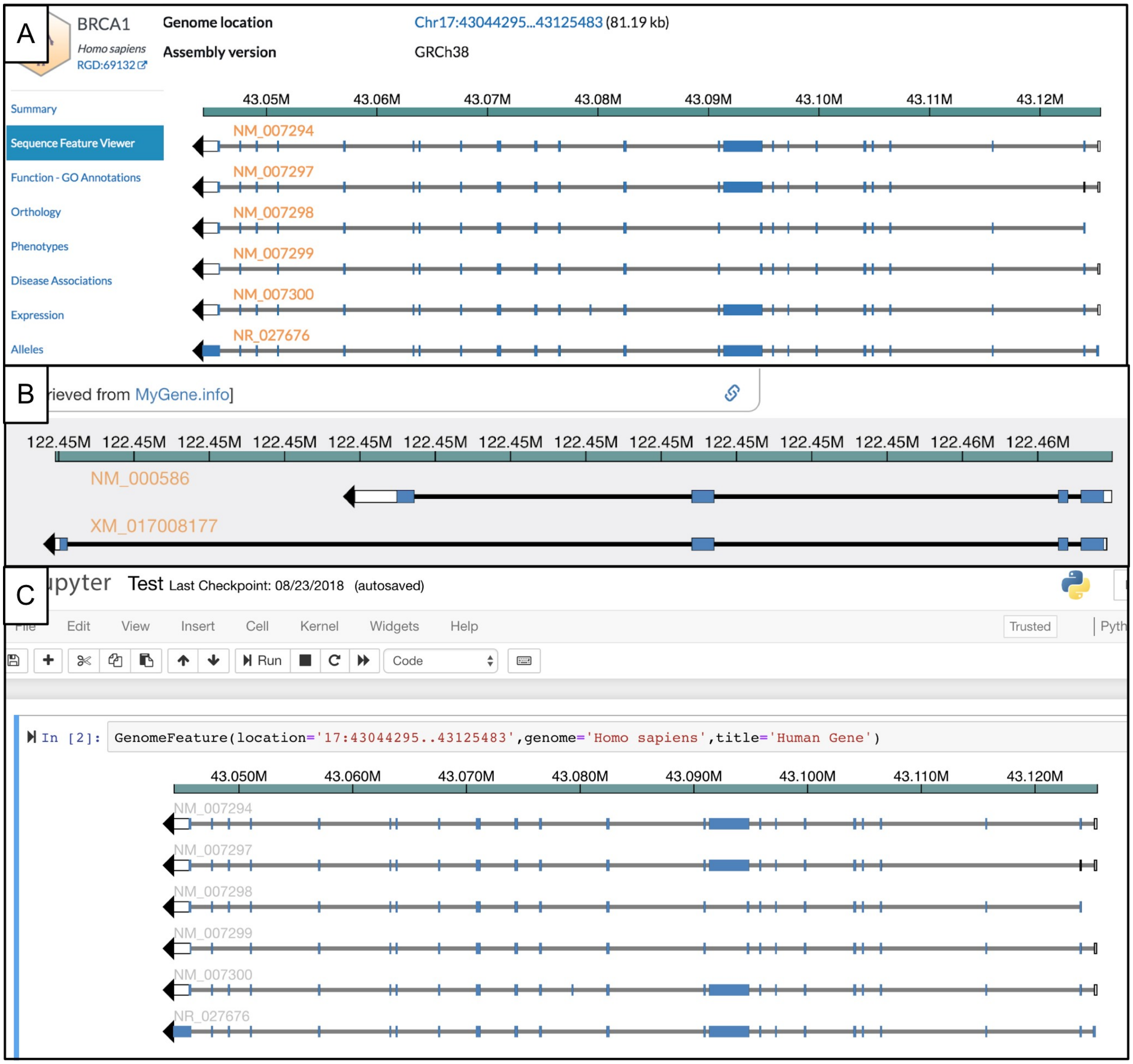
Customization

Apollo's collection of configuration options enable it to meet the unique biological and organizational needs of individual projects. Options include: which organism genomes the server will host; the appropriate codon translation table to use for each genome; organism-specific acceptor and donor sites; how deep the 'undo' stack should be; which algorithm to use when determining if transcripts are isoforms of the same gene; and many others.

In addition to the particular biological configuration, each project can specify the permissions granted to specific users or user-groups that may correspond, for example, to a laboratory or organism within a larger project. For more information about configuring Apollo, see http://genomearchitect.org/users-guide/.

Community contributions

As it has evolved, Apollo has greatly benefited from community contributions via bug reports, comments, feature suggestions, as well as directly from code changes submitted by external developers via pull requests. Many of Apollo's newer features are based on contributions from or joint development projects with members of the bioinformatics community. One recent example was the creation of the Genome Feature Widget (https://www.npmjs.com/package/genomefeaturecomponent) to provide a lightweight overview of genomic features in order to embed them within a web page. Working with external developers at the Human Phenotype Ontology [51] the Mouse Genome Database [52] and Wormbase [53], we expanded the Apollo web services to serve pieces of genomic evidence as JSON snippets that can be digested by the widget. The Genome Feature Widget is now being used by the Monarch Initiative [54] and the Alliance of Genome Resources (AGR) [55] in some of their web pages (Fig 5A, Fig 5B), as well as to embed Apollo visualizations in other platforms such as Jupyter Notebooks (Fig 5C). Other examples of community contributions include addition of an "Instructor" administrator role to allow a teacher who does not handle the administration of the the project to more easily use Apollo in classes. Additionally, users have added web services, the ability to select tracks, and numerous build improvements.



```
GenomeFeature(location='17:43044295..43125483',genome='Homo sapiens',title='Human Gene')
```

Fig 5. Three demonstrations of the Genome Feature Component npm widget (https://www.npmjs.com/package/genomefeaturecomponent) show examples that leverage Apollo's web services by consuming snippets of data for particular regions. A) The Alliance for Genome Resources web page (https://www.alliancegenome.org/gene/HGNC:1100) visualizes the Human BRCA1 gene. B) the Monarch Initiative (https://monarchinitiative.org/) web page visualizes the human IL2 gene. C) We embed the npm widget within a Jupyter Notebook widget to be called directly from a Python command-line script.

https://doi.org/10.1371/journal.pcbi.1006790.g005

Availability and future directions

Availability

Apollo is freely available (https://github.com/GMOD/Apollo) under the BSD-3 license. A User Guide and demo are provided at http://genomearchitect.org/, while numerous configuration directions are documented (https://genomearchitect.readthedocs.io/en/latest/). We welcome improvements submitted as GitHub pull requests by the community.

- A local **installation** of Apollo requires Java JDK 1.8+ and Node.js 6+. Installing, running, and testing are all accomplished using a provided bash script. We also provide a complete Docker implementation [56]. Additionally, after every Apollo release, an Amazon Web Services EC2 public image is provided.

Future directions

As we work to increase Apollo's repertoire of visual exploration and visual analytics tools, several major enhancements are currently under development. First is improving the visualization of variants and their predicted effects to help in identifying disease-causing variants across diverse groups. Second is sequence coordinate transforms, which will combine different sequence regions into a single, synthetic region. This will allow the visualization of two or more genomic regions, from the length of entire chromosomes to just a few exons, within a single artificially constructed genomic region. Artificially joining scaffolds facilitates annotation of genomic features that were split in a fragmented assembly, or it can hide intra- and intergenic regions to provide a more densely information-rich visualization of the genome. Additionally, we plan to simplify the annotation workflow by eliminating the need for manual server-side preprocessing of genomes and genomic evidence during initial installation and allowing all configuration to be done via the web interface. Finally, we are hoping to further improve Apollo's performance by using graph databases.

Graph databases for performance improvement. Apollo relies on a traditional relational database, a well-established and performant technology that provides schema enforcement and transaction support, which are both requirements for a reliable curation tool. Genomic features are represented using a nested data model similar to Chado [50] and thus require multiple joins in order to retrieve them from the database, which is inherently inefficient, especially over larger sections of the genome. This is problematic if a user wants to promote an entire evidence track to the editing window, an operation which vastly simplifies downstream merging of evidence. While denormalization is possible, the data is constantly changing due to edits, requiring a cascade of changes to ensure consistency. A coming solution, and one which improves the modeling of the data, will be to replace the relational database with a graph database. Experiments have suggested that they offer an order of magnitude speedup while still providing schema enforcement, transaction support, and a more adaptive schema.

Genome publication. The plummeting price of sequencing is leading to an explosion of genomic sequencing. This in turn is producing a growing trove of information from which to gain insights into each new genome's encoded features. Projects such as the joint Wellcome Trust Sanger Centre and Beijing Genome Institute project to sequence every vertebrate genome [57] are the tip of the iceberg. While large genomic resource centers may have funding for staff members to maintain genome curation efforts for a handful of organisms, this will not scale to the annotation effort needed to cover the rapidly accumulating genomes of other organisms or strains. Annotation on this larger scale requires contributions from a much wider community of researchers, who have the biological expertise to improve annotations, but require an efficient user interface that is collaborative and accessible through a web browser. Apollo provides a free, open source annotation platform that these researchers can integrate into their workflow, thereby helping to democratize the process of genome annotation.

Frequently, when a genome analysis project is completed, gene annotations and metadata generated during the life of the project become inaccessible to other researchers unless they are integrated into a stably supported central resource [58]. To overcome this, annotations could be saved to a central track hub registry (such as Ensembl or UCSC), as a read-only

JBrowse snapshot of the annotations. This would not only preserve the data in a GFF3 file, but would also offer a means of viewing it. A JBrowse registry hub, where indexed snapshots are listed, would ensure the long-term preservation of the evidence trail that supports each annotation and its micro-attribution. This archive methodology has been shown to be successful within the G-OnRamp group's Galaxy workflow (https://github.com/goeckslab/jbrowse-archive-creator).

Expanding on the idea of the track hub 'publication' of a genome, Apollo establishes a new data capture and dissemination paradigm that can benefit the individual researcher as well as the wider community. By recording their genome annotations precisely, Apollo makes it possible for researchers to claim professional credit for their work when it is utilized in subsequent research. Citable contributions could derive from creation, structural changes, and for enriching an annotation with additional information such as the biological function associated with a gene. The annotations produced by a particular author, identified in Apollo by their Open Researcher and Contributor ID (ORCID, https://orcid.org/), would become citable micro-publications, and could be included in data exports to show the provenance of the annotations. A 'genome press release' in which the contributors release a summary of their genome annotation set findings would bring the annotations of new organisms and clades to the attention of the wider community and provide appropriate credit to the authors.

Acknowledgments

Thanks to the Apollo and JBrowse communities for bringing issues to our attention, requesting new features, contributing code, integrating and using our product. Some notable contributors, in addition to those in the author list: Yating Liu, Luke Sargent, and Antony Bretaudeau.

We also thank Chris Childers and Monica Poelchau at the National Agricultural Library for use cases, bug reports, feedback and stress-testing.

Author Contributions

Conceptualization: Ian H. Holmes, Suzanna E. Lewis.

Data curation: Monica Munoz-Torres, Christine G. Elsik.

Funding acquisition: Ian H. Holmes, Christine G. Elsik, Suzanna E. Lewis.

Investigation: Christine G. Elsik, Suzanna E. Lewis.

Methodology: Nathan A. Dunn.

Project administration: Ian H. Holmes, Christine G. Elsik, Suzanna E. Lewis.

Resources: Monica Munoz-Torres.

Software: Nathan A. Dunn, Deepak R. Unni, Colin Diesh, Eric Yao, Helena Rasche, Suzanna E. Lewis.

Supervision: Nathan A. Dunn, Ian H. Holmes, Suzanna E. Lewis.

Validation: Monica Munoz-Torres.

Visualization: Eric Yao, Helena Rasche, Ian H. Holmes.

Writing – original draft: Nathan A. Dunn, Deepak R. Unni, Nomi L. Harris, Suzanna E. Lewis.

Writing – review & editing: Nathan A. Dunn, Deepak R. Unni, Colin Diesh, Nomi L. Harris, Ian H. Holmes, Christine G. Elsik, Suzanna E. Lewis.

References

1. Lewis SE, Searle SMJ, Harris N, Gibson M, Lyer V, Richter J, et al. Apollo: a sequence annotation editor [Internet]. Genome Biol. 2002. p. research0082.1. https://doi.org/10.1186/gb-2002-3-12-research0082 PMID: 12537571
2. Lee E, Helt GA, Reese JT, Munoz-Torres MC, Childers CP, Buels RM, et al. Web Apollo: a web-based genomic annotation editing platform. Genome Biol. 2013; 14: R93. https://doi.org/10.1186/gb-2013-14-8-r93 PMID: 24000942
3. Unni D, Dunn N, Yao E, Buels R, Li Y, Holmes I, et al. GMOD/Apollo: Apollo2.1.0(JB#d3827c) [Internet]. 2018. https://doi.org/10.5281/zenodo.1295754
4. Kudtarkar P, Cameron RA. Echinobase: an expanding resource for echinoderm genomic information. Database. 2017;2017. https://doi.org/10.1093/database/bax074 PMID: 29220460
5. Elsik CG, Tayal A, Diesh CM, Unni DR, Emery ML, Nguyen HN, et al. Hymenoptera Genome Database: integrating genome annotations in HymenopteraMine. Nucleic Acids Res. 2016; 44: D793–800. https://doi.org/10.1093/nar/gkv1208 PMID: 26578564
6. Poelchau M, Childers C, Moore G, Tsavatapalli V, Evans J, Lee C-Y, et al. The i5k Workspace@NAL—enabling genomic data access, visualization and curation of arthropod genomes. Nucleic Acids Res. 2015; 43: D714–9. https://doi.org/10.1093/nar/gku983 PMID: 25332403
7. Pedro H, Maheswari U, Urban M, Irvine AG, Cuzick A, McDowall MD, et al. PhytoPath: an integrative resource for plant pathogen genomics. Nucleic Acids Res. 2016; 44: D688–93. https://doi.org/10.1093/nar/gkv1052 PMID: 26476449
8. Neale DB, Wegrzyn JL, Stevens KA, Zimin AV, Puiu D, Crepeau MW, et al. Decoding the massive genome of loblolly pine using haploid DNA and novel assembly strategies. Genome Biol. 2014; 15: R59. https://doi.org/10.1186/gb-2014-15-3-r59 PMID: 24647006
9. Giraldo-Calderón GI, Emrich SJ, MacCallum RM, Maslen G, Dialynas E, Topalis P, et al. VectorBase: an updated bioinformatics resource for invertebrate vectors and other organisms related with human diseases. Nucleic Acids Res. 2015; 43: D707–13. https://doi.org/10.1093/nar/gku1117 PMID: 25510499
10. James-Zorn C, Ponferrada VG, Burns KA, Fortriede JD, Lotay VS, Liu Y, et al. Xenbase: Core features, data acquisition, and data processing. Genesis. 2015; 53: 486–497. https://doi.org/10.1002/dvg.22873 PMID: 26150211
11. Poynton HC, Hasenbein S, Benoit JB, Sepulveda MS, Poelchau MF, Hughes DST, et al. The Toxicogenome of Hyalella azteca: A Model for Sediment Ecotoxicology and Evolutionary Toxicology. Environ Sci Technol. 2018; 52: 6009–6022. https://doi.org/10.1021/acs.est.8b00837 PMID: 29634279
12. McKenna DD, Scully ED, Pauchet Y, Hoover K, Kirsch R, Geib SM, et al. Genome of the Asian longhorned beetle (Anoplophora glabripennis), a globally significant invasive species, reveals key functional and evolutionary innovations at the beetle-plant interface. Genome Biol. 2016; 17: 227. https://doi.org/10.1186/s13059-016-1088-8 PMID: 27832824
13. Linnen CR, O'Quin CT, Shackleford T, Sears CR, Lindstedt C. Genetic Basis of Body Color and Spotting Pattern in Redheaded Pine Sawfly Larvae (Neodiprion lecontei). Genetics. 2018; 209: 291–305. https://doi.org/10.1534/genetics.118.300793 PMID: 29496749
14. Schoville SD, Chen YH, Andersson MN, Benoit JB, Bhandari A, Bowsher JH, et al. A model species for agricultural pest genomics: the genome of the Colorado potato beetle, Leptinotarsa decemlineata (Coleoptera: Chrysomelidae). Sci Rep. 2018; 8: 1931. https://doi.org/10.1038/s41598-018-20154-1 PMID: 29386578
15. Papanicolaou A, Schetelig MF, Arensburger P, Atkinson PW, Benoit JB, Bourtzis K, et al. The whole genome sequence of the Mediterranean fruit fly, Ceratitis capitata (Wiedemann), reveals insights into the biology and adaptive evolution of a highly invasive pest species. Genome Biol. 2016; 17: 192. https://doi.org/10.1186/s13059-016-1049-2 PMID: 27659211
16. Kanost MR, Arrese EL, Cao X, Chen Y-R, Chellapilla S, Goldsmith MR, et al. Multifaceted biological insights from a draft genome sequence of the tobacco hornworm moth, Manduca sexta. Insect Biochem Mol Biol. 2016; 76: 118–147. https://doi.org/10.1016/j.ibmb.2016.07.005 PMID: 27522922
17. Benoit JB, Adelman ZN, Reinhardt K, Dolan A, Poelchau M, Jennings EC, et al. Unique features of a global human ectoparasite identified through sequencing of the bed bug genome. Nat Commun. 2016; 7: 10165. https://doi.org/10.1038/ncomms10165 PMID: 26836814
18. Fu Y, Yang Y, Zhang H, Farley G, Wang J, Quarles KA, et al. The genome of the Hi5 germ cell line from Trichoplusia ni, an agricultural pest and novel model for small RNA biology. Elife. 2018; 7. https://doi.org/10.7554/eLife.31628 PMID: 29376823

19. Gouin A, Bretaudeau A, Nam K, Gimenez S, Aury J-M, Duvic B, et al. Two genomes of highly polyphagous lepidopteran pests (Spodoptera frugiperda, Noctuidae) with different host-plant ranges. Sci Rep. 2017; 7: 11816. https://doi.org/10.1038/s41598-017-10461-4 PMID: 28947760

20. Chen X-G, Jiang X, Gu J, Xu M, Wu Y, Deng Y, et al. Genome sequence of the Asian Tiger mosquito, Aedes albopictus, reveals insights into its biology, genetics, and evolution. Proc Natl Acad Sci U S A. 2015; 112: E5907–15. https://doi.org/10.1073/pnas.1516410112 PMID: 26483478

21. Zhu Y, Engström PG, Tellgren-Roth C, Baudo CD, Kennell JC, Sun S, et al. Proteogenomics produces comprehensive and highly accurate protein-coding gene annotation in a complete genome assembly of Malassezia sympodialis. Nucleic Acids Res. 2017; 45: 2629–2643. https://doi.org/10.1093/nar/gkx006 PMID: 28100699

22. Ifeonu OO, Simon R, Tennant SM, Sheoran AS, Daly MC, Felix V, et al. Cryptosporidium hominis gene catalog: a resource for the selection of novel Cryptosporidium vaccine candidates. Database. 2016;2016. https://doi.org/10.1093/database/baw137 PMID: 28095366

23. Ifeonu OO, Chibucos MC, Orvis J, Su Q, Elwin K, Guo F, et al. Annotated draft genome sequences of three species of Cryptosporidium: Cryptosporidium meleagridis isolate UKMEL1, C. baileyi isolate TAMU-09Q1 and C. hominis isolates TU502_2012 and UKH1. Pathog Dis. 2016; 74. https://doi.org/10.1093/femspd/ftw080 PMID: 27519257

24. Colquitt BM, Mets DG, Brainard MS. Draft genome assembly of the Bengalese finch, Lonchura striata domestica, a model for motor skill variability and learning. Gigascience. 2018; 7: 1–6. https://doi.org/10.1093/gigascience/giy008 PMID: 29618046

25. Lee C-Y, Hsieh P-H, Chiang L-M, Chattopadhyay A, Li K-Y, Lee Y-F, et al. Whole-genome de novo sequencing reveals unique genes that contributed to the adaptive evolution of the Mikado pheasant. Gigascience. 2018; 7. https://doi.org/10.1093/gigascience/giy044 PMID: 29722814

26. Smith JJ, Timoshevskaya N, Ye C, Holt C, Keinath MC, Parker HJ, et al. The sea lamprey germline genome provides insights into programmed genome rearrangement and vertebrate evolution. Nat Genet. 2018; 50: 270–277. https://doi.org/10.1038/s41588-017-0036-1 PMID: 29358652

27. Pilkington SM, Crowhurst R, Hilario E, Nardozza S, Fraser L, Peng Y, et al. A manually annotated Actinidia chinensis var. chinensis (kiwifruit) genome highlights the challenges associated with draft genomes and gene prediction in plants. BMC Genomics. 2018; 19: 257. https://doi.org/10.1186/s12864-018-4656-3 PMID: 29661190

28. Li Y, Wei W, Feng J, Luo H, Pi M, Liu Z, et al. Genome re-annotation of the wild strawberry Fragaria vesca using extensive Illumina- and SMRT-based RNA-seq datasets. DNA Res. 2017; https://doi.org/10.1093/dnares/dsx038 PMID: 29036429

29. Xu Z, Luo H, Ji A, Zhang X, Song J, Chen S. Global Identification of the Full-Length Transcripts and Alternative Splicing Related to Phenolic Acid Biosynthetic Genes in Salvia miltiorrhiza. Front Plant Sci. 2016; 7: 100. https://doi.org/10.3389/fpls.2016.00100 PMID: 26904067

30. Chen L, Gong Y, Cai Y, Liu W, Zhou Y, Xiao Y, et al. Genome Sequence of the Edible Cultivated Mushroom Lentinula edodes (Shiitake) Reveals Insights into Lignocellulose Degradation. PLoS One. 2016; 11: e0160336. https://doi.org/10.1371/journal.pone.0160336 PMID: 27500531

31. Frantzeskakis L, Kracher B, Kusch S, Yoshikawa-Maekawa M, Bauer S, Pedersen C, et al. Signatures of host specialization and a recent transposable element burst in the dynamic one-speed genome of the fungal barley powdery mildew pathogen. BMC Genomics. 2018; 19: 381. https://doi.org/10.1186/s12864-018-4750-6 PMID: 29788921

32. Jelen V, de Jonge R, Van de Peer Y, Javornik B, Jakše J. Complete mitochondrial genome of the Verticillium-wilt causing plant pathogen Verticillium nonalfalfae. PLoS One. 2016; 11: e0148525. https://doi.org/10.1371/journal.pone.0148525 PMID: 26839950

33. Nemri A, Saunders DGO, Anderson C, Upadhyaya NM, Win J, Lawrence GJ, et al. The genome sequence and effector complement of the flax rust pathogen Melampsora lini. Front Plant Sci. 2014; 5: 98. https://doi.org/10.3389/fpls.2014.00098 PMID: 24715894

34. Schuelke TA, Westbrook A, Broders K, Woeste K, MacManes MD. De novo genome assembly of Geosmithia morbida, the causal agent of thousand cankers disease. PeerJ. 2016; 4: e1952. https://doi.org/10.7717/peerj.1952 PMID: 27168971

35. Syme RA, Tan K-C, Hane JK, Dodhia K, Stoll T, Hastie M, et al. Comprehensive Annotation of the Parastagonospora nodorum Reference Genome Using Next-Generation Genomics, Transcriptomics and Proteogenomics. PLoS One. 2016; 11: e0147221. https://doi.org/10.1371/journal.pone.0147221 PMID: 26840125

36. Eves-van den Akker S, Laetsch DR, Thorpe P, Lilley CJ, Danchin EGJ, Da Rocha M, et al. The genome of the yellow potato cyst nematode, Globodera rostochiensis, reveals insights into the basis of parasitism and virulence. Genome Biol. 2016; 17: 124. https://doi.org/10.1186/s13059-016-0985-1 PMID: 27286965

37. Genome Decoders: The Human Whipworm [Internet]. 28 Sep 2017 [cited 25 Sep 2018]. Available: https://www.sanger.ac.uk/news/view/uk-students-working-scientists-help-prevent-childhood-parasite-infection

38. Buels R, Yao E, Diesh CM, Hayes RD, Munoz-Torres M, Helt G, et al. JBrowse: a dynamic web platform for genome visualization and analysis. Genome Biol. 2016; 17: 66. https://doi.org/10.1186/s13059-016-0924-1 PMID: 27072794

39. Smith G, Ledbrook P. Grails in Action [Internet]. Manning; 2014. Available: https://market.android.com/details?id=book-ZyCdmwEACAAJ

40. The Apache Groovy programming language [Internet]. 2018. Available: http://groovy-lang.org/

41. Afgan E, Baker D, Batut B, van den Beek M, Bouvier D, Cech M, et al. The Galaxy platform for accessible, reproducible and collaborative biomedical analyses: 2018 update. Nucleic Acids Res. 2018; 46: W537–W544. https://doi.org/10.1093/nar/gky379 PMID: 29790989

42. G-OnRamp–Create Genome Browsers for Genome Annotation [Internet]. 25 Sep 2018 [cited 25 Sep 2018]. Available: http://gonramp.wustl.edu/

43. Lee T, Peace C, Jung S, Zheng P, Main D, Cho I. GenSAS—An online integrated genome sequence annotation pipeline. 2011 4th International Conference on Biomedical Engineering and Informatics (BMEI). 2011. pp. 1967–1973. https://doi.org/10.1109/BMEI.2011.6098712

44. Humann JL. GenSAS v5.1: A Web-Based Platform for Structural and Functional Annotation and Curation of Genomes. PAG—Plant and Animal Genome XXVI Conference (January 13–17, 2018). Washington State University; 2018. Available: https://pag.confex.com/pag/xxvi/meetingapp.cgi/Paper/28336

45. Hilgert U, McKay S, Khalfan M, Williams J, Ghiban C, Micklos D. DNA Subway: Making Genome Analysis Egalitarian. Proceedings of the 2014 Annual Conference on Extreme Science and Engineering Discovery Environment. ACM; 2014. p. 70. https://doi.org/10.1145/2616498.2616575

46. Bretaudeau A, Dunn N, Gladman S, Grüning B, Rasche H, Seemann T. Galaxy Genome Annotation project: Integrating Galaxy and GMOD for genome annotation. F1000Res. 2018; 7. https://doi.org/10.7490/f1000research.1116180.1

47. Rasche H. Apollo Python Integration [Internet]. 2017. Available: https://pypi.org/project/apollo/

48. Bretaudeau A. Deployment of genome databases for insects using Galaxy Genome Annotation [Internet]. F1000Research; 2017 Jul 11. https://doi.org/10.7490/f1000research.1114390.1

49. Rasche H, Grüning B, Dunn N, Bretaudeau A. GGA: Galaxy for genome annotation, teaching, and genomic databases. F1000Res. 2018; 7. https://doi.org/10.7490/f1000research.1116181.1

50. Mungall CJ, Emmert DB, FlyBase Consortium. A Chado case study: an ontology-based modular schema for representing genome-associated biological information. Bioinformatics. 2007; 23: i337–46. https://doi.org/10.1093/bioinformatics/btm189 PMID: 17646315

51. Köhler S, Vasilevsky NA, Engelstad M, Foster E, McMurry J, Aymé S, et al. The Human Phenotype Ontology in 2017. Nucleic Acids Res. 2017; 45: D865–D876. https://doi.org/10.1093/nar/gkw1039 PMID: 27899602

52. Smith CL, Blake JA, Kadin JA, Richardson JE, Bult CJ, Mouse Genome Database Group. Mouse Genome Database (MGD)-2018: knowledgebase for the laboratory mouse. Nucleic Acids Res. 2018; 46: D836–D842. https://doi.org/10.1093/nar/gkx1006 PMID: 29092072

53. Lee RYN, Howe KL, Harris TW, Arnaboldi V, Cain S, Chan J, et al. WormBase 2017: molting into a new stage. Nucleic Acids Res. 2018; 46: D869–D874. https://doi.org/10.1093/nar/gkx998 PMID: 29069413

54. McMurry JA, Köhler S, Washington NL, Balhoff JP, Borromeo C, Brush M, et al. Navigating the Phenotype Frontier: The Monarch Initiative. Genetics. 2016; 203: 1491–1495. https://doi.org/10.1534/genetics.116.188870 PMID: 27516611

55. Alliance of Genome Resources [Internet]. [cited 22 Nov 2018]. Available: https://www.alliancegenome.org/

56. Dunn N, Rasche H, Paulini M. GMOD/docker-apollo: Apollo 2.1.0 Docker+PostgreSQL [Internet]. 2018. https://doi.org/10.5281/zenodo.1296537

57. Researchers reboot ambitious effort to sequence all vertebrate genomes, but challenges loom. In: Science | AAAS [Internet]. 13 Sep 2018 [cited 19 Nov 2018]. https://doi.org/10.1126/science.aav4025

58. Gibney E, Van Noorden R. Scientists losing data at a rapid rate. Nature News. 2013; https://doi.org/10.1038/nature.2013.14416