BIOINFORMATICS APPLICATIONS NOTE
Vol. 28 no. 7 2012, pages 1054–1056
doi:10.1093/bioinformatics/bts046

Databases and ontologies

Advance Access publication January 28, 2012

Chado Controller: advanced annotation management with a community annotation system

Valentin Guignon[1,2,*], Gaëtan Droc[1], Michael Alaux[3], Franc-Christophe Baurens[1], Olivier Garsmeur[1], Claire Poiron[1†], Tim Carver[4], Mathieu Rouard[2] and Stéphanie Bocs[1,*]

[1]CIRAD, UMR AGAP, F-34398 Montpellier, [2]Bioversity International, CfL programme, F-34397 Montpellier, [3]Unité de Recherche en Génomique-Info, UR 1164, INRA Centre de Versailles-Grignon, Versailles, France and [4]Wellcome Trust Sanger Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SA, UK

Associate Editor: Janet Kelso

ABSTRACT

Summary: We developed a controller that is compliant with the Chado database schema, GBrowse and genome annotation-editing tools such as Artemis and Apollo. It enables the management of public and private data, monitors manual annotation (with controlled vocabularies, structural and functional annotation controls) and stores versions of annotation for all modified features. The Chado controller uses PostgreSQL and Perl.

Availability: The Chado Controller package is available for download at http://www.gnpannot.org/content/chado-controller and runs on any Unix-like operating system, and documentation is available at http://www.gnpannot.org/content/chado-controller-doc

The system can be tested using the GNPAnnot Sandbox at http://www.gnpannot.org/content/gnpannot-sandbox-form

Contact: valentin.guignon@cirad.fr; stephanie.sidibe-bocs@cirad.fr

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on September 19, 2011; revised on January 13, 2012; accepted on January 23, 2012

1 INTRODUCTION

With the growth of large community annotation projects due to the rapid progress of the next-generation sequencing technologies, efficient and optimized genomic data management is critical. Community Annotation Systems (CASs) are suitable for curators to annotate from all over the world via the Web. Some genomic information systems (Flicek, *et al.*, 2011; Fujita, *et al.*, 2011; St Pierre and McQuilton, 2009) have been provided to the community but only a few of them allow manual interactive, dynamic and user-friendly curation (Legeai, *et al.*, 2010; Vallenet, *et al.*, 2006). Generic Model Organism Database (GMOD, http://gmod.org) is a collaborative project to develop a set of interoperable open-source software for visualizing, annotating and managing biological data. In this project are developed popular software such as Chado (Mungall and Emmert, 2007) a modular database schema that underlies many GMOD tools such as the Generic Genome Browser, GBrowse, (Stein, *et al.*, 2002). In conjunction with genome annotation-editing tools such as Apollo (Lewis, *et al.*, 2002) and Artemis (Carver, *et al.*, 2008), these interfaces are foundations for a generic and robust CAS. However, additional components are required to consolidate these components. For instance, software to deal with confidential and unpublished data which is an important requirement for a system to be more widely adopted by biologists that prefer to deal with flat files on a local file system. Also required is a monitoring system to keep track of the annotation process and highlight annotation inconsistencies. Currently, when a modification is made, pre-existing manual annotations are overwritten and only basic data quality controls are made. To address these issues, we decided to extend the Chado schema. Our approach was driven by the GMOD philosophy and we propose a generic, modular, seamless, easy-to-install and highly configurable controller called the Chado Controller (CC).

2 IMPLEMENTATION

Chado is a modular schema driven by ontologies and controlled vocabularies and partitioned into modules for different biological domains linked to appropriate ontologies. The modules targeted by the CC are those related to genomic sequences: core modules (general usage, ontologies and controlled vocabularies) and the sequence feature module. The CC is based on a Model–View–Controller (MVC) architecture. In this publication, the 'model' is Chado, 'views' are GBrowse and Artemis and the 'controller' is the CC. The CC is embedded in the Chado 1.1 database as PostgreSQL views, procedures and triggers in order to intercept and process any query made on the genomic features. Chado 1.2 has been tested and no schema conflicts have been detected. Several Perl scripts and modules were developed to install and administer the CC. Some patches have been written for GBrowse 1.70 and 2.40 and Artemis 13.2.8 to apply the new features brought by the CC. For Apollo 1.11.4, a new data adapter has been added. The architecture of the CC modules is described in Supplementary Figures S1–S4 and in technical documentation.

2.1 User access restriction module

The access restriction module required the following: (i) new tables in the generic database schema; (ii) a new PostgreSQL view

*To whom correspondence should be addressed.
†Present address: UM2, LIGM, CNRS, UPR IGH 1142, IMGT®, F-34396 Montpellier, France.

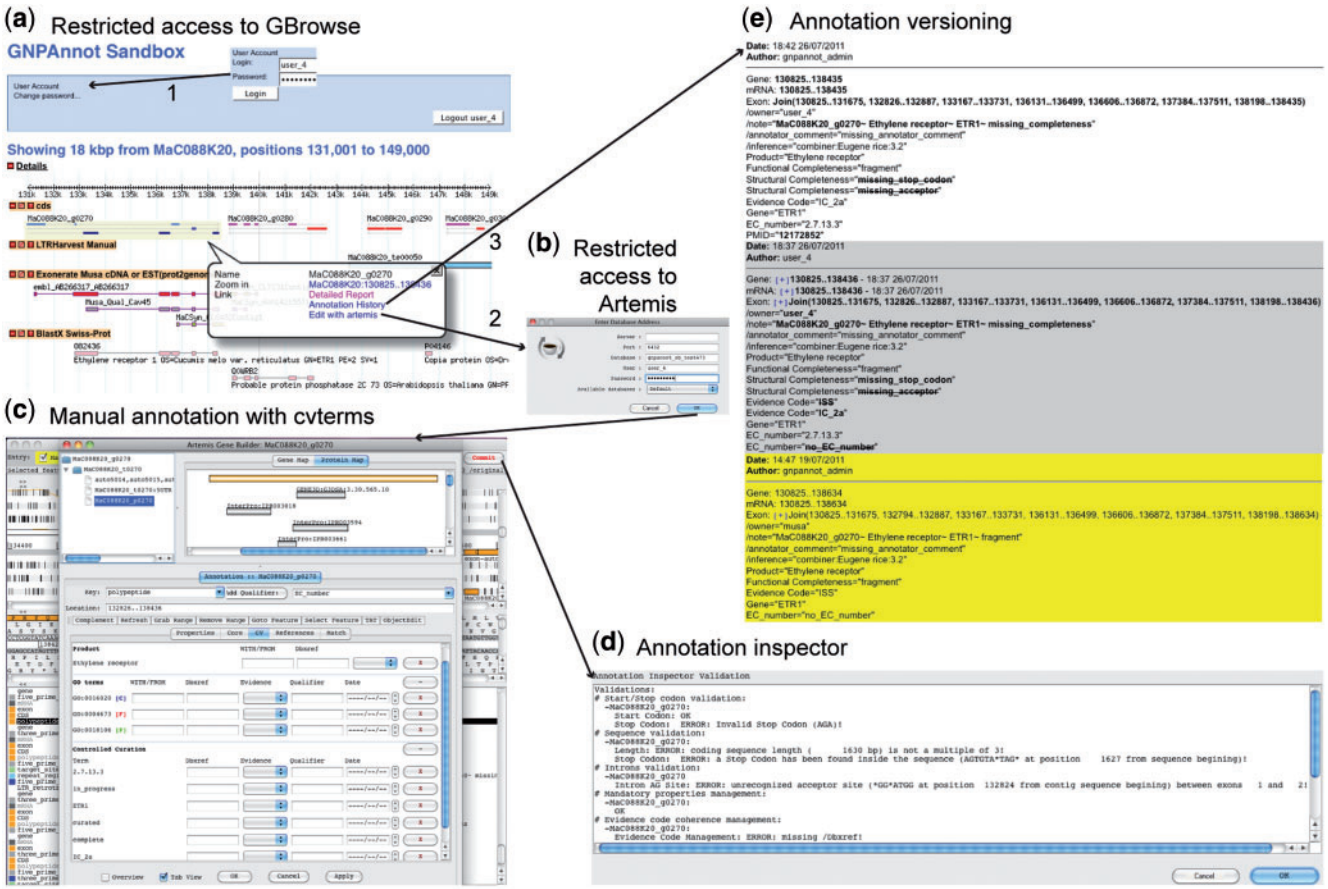

Fig. 1. GNPAnnot Community Annotation System round trip at ATGC South Green Bioinformatics Platform. (**a**) Illustration of GBrowse that uses the access restriction module of the CC from the GNPAnnot Sandbox. (**b**) Artemis connection using the CC access restriction module. (**c**) Feature editing, both the structure and the function using cv terms. (**d**) Clicking on the commit button calls the CC annotation inspector module. (**e**) Feature annotation history using the CC annotation versioning module.

that restricts access; (iii) a login and password area for GBrowse (Supplementary Fig. S1 and Fig. 1a); and (iv) an administrator web interface developed to manage users or group permissions. A view of the feature table with rules (for insert/update/delete) ensures access-restrictions. For user-management, a table for users/groups and a table to associate the access-level of features with users and groups were created.

2.2 Annotation inspector module

A module to check manual annotation, called the inspector, was integrated in the implementation of Chado, with database triggers that calls SQL functions for monitored events. This module automatically adds some basic properties to any given genomic feature (e.g. '/owner', '/color'). Additional procedures to check the integrity of the structure of curated genes are run by the genome editor (e.g. start/stop codon, coding sequence length, splicing sites). For instance, Artemis has been modified to call the initialization procedure once connecting to Chado. Then, when a change is validated using the commit button, a procedure returns the inspection report in a java dialog box (Fig. 1d). Depending on the user choice, Artemis can then either commit the change or rollback to the original state. The list of data quality controls is available in Supplementary Tables S1–S4.

2.3 Annotation versioning module

The annotation versioning module keeps track of any change in database content. This functionality is based on a slightly modified version of the 'audit module' provided in Chado. The name of the annotator responsible for the modification is now recorded in the database. By default, all the tables available in the database at the time of installation are audited. Any annotation changes made by Artemis or even data loaded in database are recorded. The 'GBrowse_history' web page can be used to view the history of changes on a given feature (Fig. 1e). It displays the modifications made by each annotator in chronological order.

2.4 Chado controller package

The CC comes with several utilities, including an installer, a compatibility management script (see 'readme' file and documentation) and a controlled vocabulary management script. The installer can be used to install, update or uninstall the CC. By default, the installation process will modify the database, update the

GBrowse configuration file, add new modules and scripts and patch what is needed. In addition, the CC has various options to be finely tuned and change its functioning.

3 DISCUSSION

3.1 Architecture

Due to some limitations of the association of Chado, GBrowse, Apollo and Artemis, annotation communities have been compelled to set up multiple Chado databases. The CC can greatly simplify the informatics architecture. As it manages private data and enables annotation versioning, the number of Chado databases to be maintained per project, genome or institute can be reduced. Thus, it can contribute to a more effective management of the bioinformatics architecture.

3.2 Data loading and performance

Benchmarking was carried out (using custom benchmark files available for download and summarized in Supplementary Table S5) and the CC causes only a slight delay on starting a new database connection. Read access was found to be almost as fast as it is without the controller. Write operations can be slow, although acceptable (up to three times longer). The quality control carried out by the inspector can take time and depends mainly on the number of features to be checked. In our test database, it took $\sim$1 s per gene. People who cannot afford the inspection delay can disable or uninstall the inspector. However, we found that losing a few seconds using the inspector is significantly less constraining than having an administrator's annotation validation step.

3.3 High quality annotations support

This system has been effective to manage annotation of BAC sequences of plant genomes (e.g. 51 *Musaceae* and 11 *Poaceae*). It enhanced the curation of 637 out of 1319 genes (48%; see Statistics at http://www.gnpannot.org/content/south-monocots-statistics). This high-quality annotation contributed to answering a number of biological questions (Bocs, *et al.*, 2010; Garsmeur, *et al.*, 2011).

ACKNOWLEDGEMENTS

We would like to acknowledge and thank the GMOD and GNPAnnot communities in particular to Fabrice Legeai, Joelle Amselem and Baptiste Brault.

Funding: French National Research Agency (ANR Genoplante) (grant ANR-07-GPLA-004).

Conflict of Interest: none declared.

REFERENCES

Bocs,S. *et al.* (2010) Mechanisms of haplotype divergence at the RGA08 nucleotide-binding leucine-rich repeat gene locus in wild banana (Musa balbisiana). *BMC Plant Biol.*, **10**, 149.

Carver,T. *et al.* (2008) Artemis and ACT: viewing, annotating and comparing sequences stored in a relational database. *Bioinformatics*, **24**, 2672–2676.

Flicek,P. *et al.* (2011) Ensembl 2011. *Nucleic Acids Res.*, **39**, D800–D806.

Fujita,P.A. *et al.* (2011) The UCSC Genome Browser database: update 2011. *Nucleic Acids Res.*, **39**, D876–D882.

Garsmeur,O. *et al.* (2011) High homologous gene conservation despite extreme autopolyploid redundancy in sugarcane. *New Phytol.*, **189**, 629–642.

Legeai,F. *et al.* (2010) AphidBase: a centralized bioinformatic resource for annotation of the pea aphid genome. *Insect Mol. Biol.*, **19** (Suppl. 2), 5–12.

Lewis,S.E. *et al.* (2002) Apollo: a sequence annotation editor. *Genome Biol.*, **3**, RESEARCH0082.

Mungall,C.J. and Emmert,D.B. (2007) A Chado case study: an ontology-based modular schema for representing genome-associated biological information. *Bioinformatics*, **23**, i337–i346.

St Pierre,S. and McQuilton,P. (2009) Inside FlyBase: biocuration as a career. *Fly (Austin)*, **3**, 112–114.

Stein,L.D. *et al.* (2002) The generic genome browser: a building block for a model organism system database. *Genome Res.*, **12**, 1599–1610.

Vallenet,D. *et al.* (2006) MaGe: a microbial genome annotation system supported by synteny results. *Nucleic Acids Res.*, **34**, 53–65.