ballaxy: web services for structural bioinformatics

Anna Katharina Hildebrandt1,*, Daniel Stöckel1, Nina M. Fischer2, Luis de la Garza2, Jens Krüger2, Stefan Nickels1, Marc Röttig2, Charlotta Schärfe2, Marcel Schumann2, Philipp Thiel2, Hans-Peter Lenhof1, Oliver Kohlbacher2 and Andreas Hildebrandt3,*

1Center for Bioinformatics, Saarland University, 66041 Saarbrücken, 2Applied Bioinformatics, Center for Bioinformatics, Quantitative Biology Center, University of Tübingen, 72076 Tübingen and 3Chair for Software-Engineering and Bioinformatics, Institute for Informatics, Johannes-Gutenberg-University Mainz, 55128 Mainz, Germany

ABSTRACT

Motivation: Web-based workflow systems have gained considerable momentum in sequence-oriented bioinformatics. In structural bioinformatics, however, such systems are still relatively rare; while commercial stand-alone workflow applications are common in the pharmaceutical industry, academic researchers often still rely on command-line scripting to glue individual tools together.

Results: In this work, we address the problem of building a web-based system for workflows in structural bioinformatics. For the underlying molecular modelling engine, we opted for the BALL framework because of its extensive and well-tested functionality in the field of structural bioinformatics. The large number of molecular data structures and algorithms implemented in BALL allows for elegant and sophisticated development of new approaches in the field. We hence connected the versatile BALL library and its visualization and editing front end BALLView with the Galaxy workflow framework. The result, which we call ballaxy, enables the user to simply and intuitively create sophisticated pipelines for applications in structure-based computational biology, integrated into a standard tool for molecular modelling.

Availability and implementation: ballaxy consists of three parts: some minor modifications to the Galaxy system, a collection of tools and an integration into the BALL framework and the BALLView application for molecular modelling. Modifications to Galaxy will be submitted to the Galaxy project, and the BALL and BALLView integrations will be integrated in the next major BALL release. After acceptance of the modifications into the Galaxy project, we will publish all ballaxy tools via the Galaxy toolshed. In the meantime, all three components are available from http://www.ball-project.org/ballaxy. Also, docker images for ballaxy are available at https://registry.hub.docker.com/u/anih/ballaxy/dockerfile/. ballaxy is licensed under the terms of the GPL.

Supplementary information: Supplementary data are available at Bioinformatics online.

Contact: anna.hildebrandt@bioinf.uni-sb.de or andreas.hildebrandt@uni-mainz.de

Received on January 29, 2014; revised on August 7, 2014; accepted on August 20, 2014

*To whom correspondence should be addressed.
2 IMPLEMENTATION

The first step of our approach consists in an extension of the BALL project (Hildebrandt et al., 2010), where common functionality has been encapsulated in command-line tools. These have then been supplemented with a mechanism for the generation of description files for workflow systems—Galaxy in particular—which then allows for the integration of the tools into a workflow. In the process, we also extended the Galaxy system to support molecular file formats and automatically load the BALL tools.

To solve the problem of providing workflow functionality in combination with a versatile modelling environment, we then extended BALLView (Moll et al., 2005, 2006) by a plugin that tightly integrates communication with a ballistic server (local or remote) into the modelling toolkit. ballistic is based on the Galaxy workflow engine and uses its powerful user management, tool handling and workflow environment.

To this end, we extend Galaxy to understand molecular file formats such as PDB or MOL2, and by tools for structural data in the context of molecular modelling and computer-aided drug design. Automated file format detection inside Galaxy uses the python interface of BALL. Structures can be downloaded, visualized and manipulated in BALLView and, through an entry in their context menu, directly uploaded to the server. A browser window embedded into BALLView is then used to interact with ballistic in the usual fashion to create or run tools and workflows. The results of these workflows can then be directly downloaded into the BALLView instance through the click of a button, where they are displayed in 3D and can be further manipulated or stored.

To provide a collection of useful tools, we further extended the BALL library. These tools will be integrated into BALL version 1.5, which is in development at the time of writing. Besides convenience tools like molecular file converters or a connected component splitter, we currently account for four main application areas: NMR shift prediction (Dehof et al., 2013, 2011a), ligand optimal bond order assignment (Dehof et al., 2011b), pose clustering (Hildebrandt et al., 2013) and docking (Kohlbacher, 2012).

Furthermore, various tools for computer-aided drug design provide functionality to set up ligand-based and structure-based virtual screening workflows. The ligand-based part comprises quantitative structure-activity relationship (QSAR) tools for reading and preprocessing of QSAR datasets, model generation and validation, feature selection and activity prediction. The structure-based part offers tools to read, check and prepare protein structures and virtual compound libraries, for pocket detection, receptor grid generation and protein-ligand docking. Tools for target and antitarget rescoring of initial docking poses allow customized improvement of the docking outcome. Post-docking analysis tools like a ScoreAnalyzer or an RMSDcalculator enable the examination of final docking results and their clustering. All the tools support a standard set of parameters, e.g. for file input, output and help texts. All the tools also support different execution environments: the same tool can be used for standard command line usage as well as for integration into a workflow toolkit. Each tool is additionally able to export its own configuration file used to tell the workflow package about its existence and mode of operation.

3 EXAMPLE

Workflow systems such as Galaxy offer great benefits for many fields of science. Using a system such as ballistic, these benefits also apply to applications in structural-based drug discovery. A typical docking workflow using CADDsuite (Kohlbacher, 2012) tools is shown in Supplementary Figure S1, where it is described in detail.

Finally, addition of new tools using BALL is simple and is described in our documentation (http://ball-trac.bioinf.uni-sb.de/wiki/ballaxy).

ACKNOWLEDGEMENTS

The authors wish to thank the BALL development team. They gratefully acknowledge the use of software from OpenEye Scientific Software, Inc.

Funding: A.H. acknowledges financial support from the Intel Visual Computing Institute (IVCI) of Saarland University and the ‘Schwerpunkt Rechnergestützte Forschungsmethoden’ of Johannes-Gutenberg University Mainz, A.H. and H.P.L. financial support from DFG (BIZ4:1-4). O.K. acknowledges financial support from DFG core facilities (grant KO 2313/6-1), the European Commissions Seventh Framework Programme (FP7/2007-2013) under grant agreement no. 283481 (SCI-BUS). J.K. and O.K. would like to thank the German Federal Ministry of Education and Research (BMBF) for the opportunity to do research in the MoSGrid project (reference 01IG09006).

Conflict of interest: none declared.

REFERENCES