# Automated Transfer of Workflows From Galaxy to Yabi and Command Line Tools

David Molik, Ying Jin, Molly Hammell
Cold Spring Harbor Laboratory, Cold Spring Harbor, NY 11724

## Abstract

The web-based bioinformatics platform Galaxy gained great popularity as a tool for allowing access to powerful compute clusters and sophisticated bioinformatics software with user-friendly point-and-click interfaces[1]. In contrast, command line tools will always be more efficient for those who are comfortable working within a Unix-like environment. Likewise, biologists who understand the computing environment are more likely to understand what is and is not computationally efficient or possible[2]. The obstacle lies in training users with little programming experience to be comfortable with the command line. We have explored alternate frameworks to Galaxy that would allow users to design their workflows in a simplified web browser environment, and then automatically transfer these workflows into pipelines suitable for running at the command line.

Yabi is an alternate web-based bioinformatics platform designed by the Center for Comparative Genomics at Murdoch University[3]. It fulfills many of the same operations as Galaxy, and is interoperable with the same tools. Yabi provides a similar user experience, providing a graphical user interface to bioinformatics software that can be executed either locally or remotely. Moreover, because Galaxy and Yabi both use simple xml formatted interface configuration files, many of the tool interfaces designed for Galaxy can be automatically transferred to the Yabi format. However, Yabi additionally offers a command line tool, yabish, where users can design their workflows with the web-based graphical interface, and then automatically transfer that workflow into a pipeline suitable for running at the command line. This can be an intermediate step between offering a graphical user interface to the end user and having users do all of their analysis on the command line, while providing the benefits of logging, saved workflows, and remote data. For the benefit of server administrators, Yabi accesses data and submits jobs to computational clusters as the end user and not as the daemonized user; meaning data management is more secure and job submission has more equality.

We implemented Yabi as an analogous software application to Galaxy and provide contrasting benefits of both platforms. Moreover, we present tools that automatically parse and reformat tool configuration files for use in either the Galaxy or Yabi format.

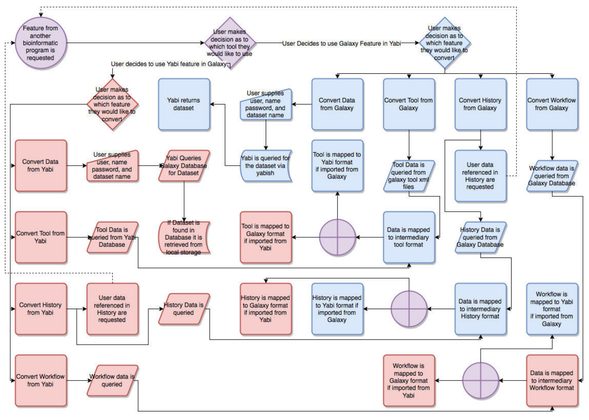



**Figure 1: Import/Export Maps between Galaxy & Yabi**. While converting data from Galaxy to Yabi is more pertinent to the task of exporting a Galaxy defined workflow, many other export and import scenarios are possible, including the export of a Galaxy history into Yabi.

## Users design their workflows in Galaxy, execute at the terminal

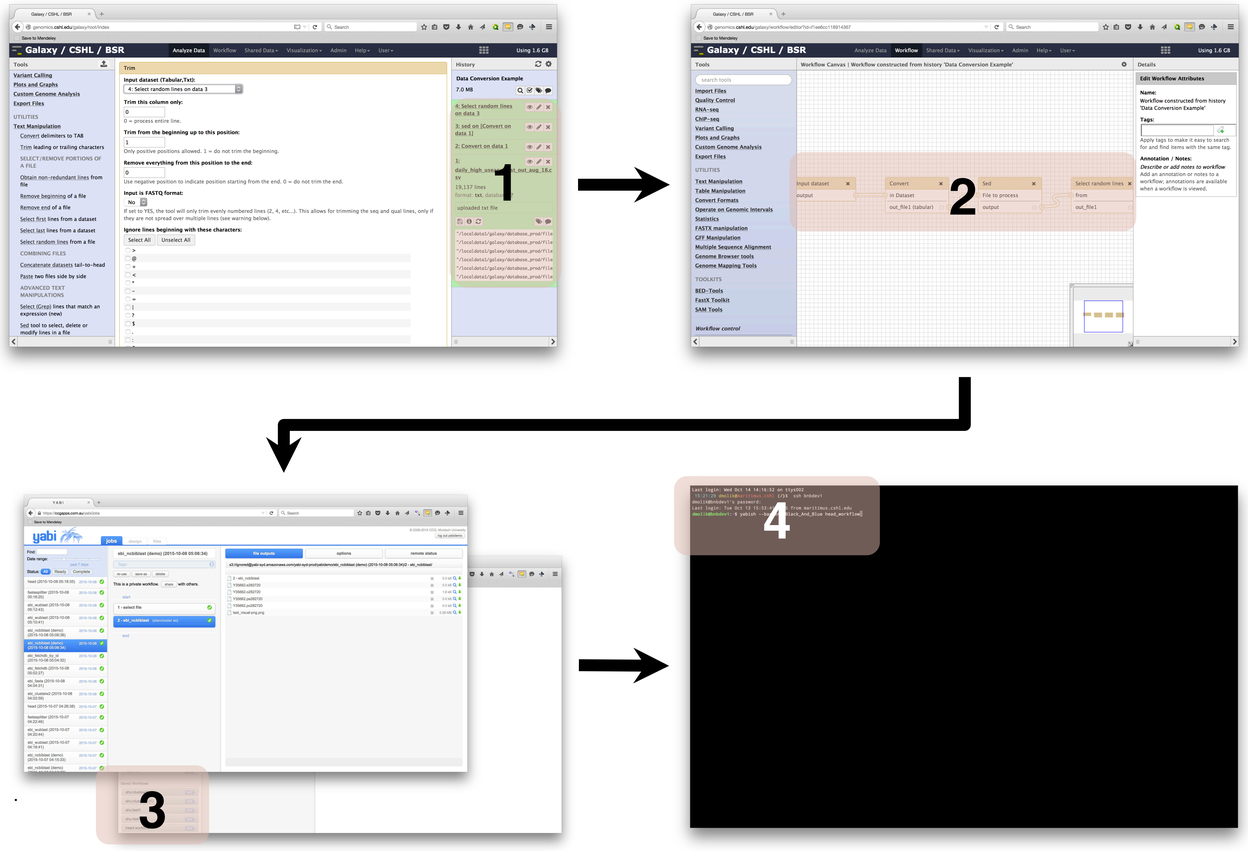
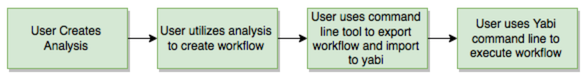



**Figure 2: Example screen shots from Galaxy workflow design to Yabish shell script execution**.

## Modular Design




As seen in Figure 2, above, the modular design is intended to allow flexible design of analysis pipelines that easily flows back and forth between the interactive Galaxy web server and the powerful Yabish shell implementations.

(1.) The user runs an exploratory, interactive data analysis through the Galaxy web server
(2.) The user decides to turn the analysis they designed into a workflow using the Galaxy editor.
(3.) The user exports the Galaxy workflow, with automated conversion to Yabi format.
(4.) Yabi workflows can be run at the command line as a single pipeline with Yabish.

This enables the use of a Graphical User Interface for the analysis and design of the workflow, but the command line for the actual running of the newly created pipeline on additional samples. This scenario gently introduces advanced Galaxy users to the terminal where pipelines run much more efficiently for large sample sets.

## Data Mapping

Both Yabi and Galaxy use databases to store data on tools that have been integrated, histories that users have completed, workflows that users have saved and information on datasets. In order to transfer this data from one system to the other these databases need to be mapped and contingencies must be in place for data that is stored in one system but not the other.

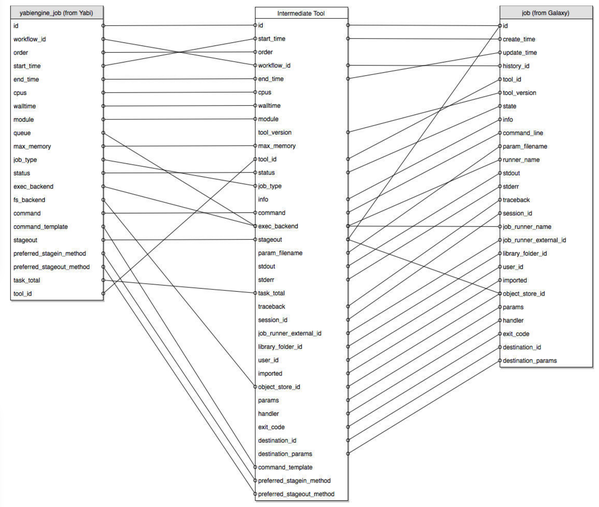



**Figure 3: Yabi Jobs Table mapped to Galaxy Jobs Table.** Showing the complexity of mapping data between the two programs. Many tables like these are required to be mapped.

## Availability

This Galaxy/Yabi conversion tool will be made available open source in part through the Galaxy Toolshed, and in full at CPAN and the Github software repository as soon as beta testing is complete. While this tool was designed and configured for the CSHL Compute Environment, it may be adopted for other environments. Extensive documentation will be provided on how to do so.

## Acknowledgements

We thank Tamas Szabo, Todd Heywood, Gerald McCloskey, Oliver Tam, Osama El Demerdash, and Darian DeFalco for helpful discussions and help with compute cluster implementation design. This work was performed with assistance from the CSHL Bioinformatics Shared Resource, which is funded, in part, by an NIH/NCI Cancer Center Support Grant 5P30CA045508.

## References

1. Goecks, Jeremy, Anton Nekrutenko, and James Taylor. 2010. “Galaxy: A Comprehensive Approach for Supporting Accessible, Reproducible, and Transparent Computational Research in the Life Sciences.” *Genome Biology* 11 (8): R86. http://genomebiology.com/2010/11/8/R86.
2. Dudley, Joel T, and Atul J Butte. 2009. “A Quick Guide for Developing Effective Bioinformatics Programming Skills.” *PLoS Computational Biology* 5 (12): e1000589. doi:10.1371/journal.pcbi.1000589.
3. Hunter, Adam A, Andrew B Macgregor, Tamas O Szabo, Crispin A Wellington, and Matthew I Bellgard. 2012. “Yabi: An Online Research Environment for Grid, High Performance and Cloud Computing.” *Source Code for Biology and Medicine* 7 (1): 1. http://www.scfbm.org/content/7/1/1.