Implementation of a standalone Galaxy instance for the creation of re-usable bioinformatics processing pipelines

Daniel J. Shea

August 16th, 2014

1 Introduction

Using yeast (Schizosaccharomyces Pombe and Saccharomyces Cervisiae) as our model organism, our research is conducted into understanding the biological processes underlying the formation, function, and epigenetic inheritance of silent chromatin domains. Three major areas of research are related to RNAi-mediated assembly of heterochromatin, SIR-mediated assembly of silent chromatin, and the regulation of recombination within the ribosomal DNA repeats. In order to improve the efficiency of data analysis via re-usable bioinformatics pipelines and to encourage collaboration and software re-use within the lab, we have chosen to implement a local instance of the Galaxy Project's open source web-based platform.

1.1 Heterochromatin's role in the silencing of genes

The physical organization of DNA in eukaryotes plays a role in the levels of genetic expression by restricting the physical access of transcriptional machinery to various regions of an organism's genome. The organization of DNA is structured around histones, octamers composed of highly conserved proteins, H2A, H2B, H3, and H4. These proteins form heterodimers H2A/H2B and H3/H4, respectively, that then further form tetramers. In turn, these tetramers then combine to form an octamer that is referred to as a histone. DNA winds around these histones forming a nucleosome, with small segments of DNA, referred to as linker DNA, between them. Structured around this basic organization of DNA are regions of tightly or loosely condensed chromatin, referred to as heterochromatin and euchromatin respectively. Heterochromatin plays a functional role in gene silencing through the physical restriction of access to DNA by transcriptional machinery involved in transcribing DNA into messenger RNA (mRNA).

1.2 Benefits derived from the implementation of Galaxy within the lab

Galaxy is an open, web-based platform for the analysis of data intensive biological research. It provides an interactive and readily accessible web interface to many of the commonly used bioinformatics tools available today. Computational pipelines may be constructed from within the web interface and then shared between users. Workflows are accessible via the workflow editor, which allows for the visualization and editing of processing pipelines. These workflows may then be abstracted for application to multiple data sets, and the workflows themselves can be shared between Galaxy users through the Galaxy web interface. This allows researchers within our environment to compose analytical workflows that may then be re-used or re-purposed by other collaborators in the lab with minimal effort.

2 Computational server architectural overview

Our computational server, chromosome.med.harvard.edu, serves as the platform on which we run our Galaxy instance. It should be noted however, that a local galaxy instance can be run on hardware with relatively modest specifications.

Depending upon the data sets and types of analyses performed, you may be constrained by the available amount RAM on your server when dealing with particularly large data sets or application of computationally intensive analysis such as attempting to perform de novo assembly. The table below details the hardware configuration for chromosome.

2.1 Hardware Configuration

chromosome.med.harvard.edu	
CPU:	(2) Intel(R) Xeon(R) CPU X5660 @ 2.80GHz, 6 Physical cores (12 Hyper-threaded)
Memory:	(18) 4GB DDR3-800 DIMMs for a total of 72GB
Disk:	(8) 1.817 TB SATA Drives configured as a RAID-6 array, providing 10.908TB of usable space
OS:	Ubuntu 12.04.4 LTS

3 Galaxy Architectural Overview

Galaxy provides default web server and local database storage for standalone instances. Depending on the use case, it is advised to reconfigure the instance to make use of more robust data storage and/or web server solutions. For our particular instance, we are currently utilizing a postgresql database as our back-end storage. Given the relatively small number of users within our lab utilization of nginx or apache as a web server, however, should the internal webserver prove to pose a bottleneck at a later date, the existing configuration may be migrated to nginx to take advantage of caching functionality to serve the static portions of the Galaxy web site. By default, sqlite internal database files are utilized for the storage of data and computational workflows. This proved promblematic as our anticipated data sets were determined to quickly tax a small flat file database system such as sqlite. Therefore, we opted to create a postgresql database wherein to store all of our data. Details on how to accomplish this are detailed below in the database subsection.

User authentication and role based authentication are handled by the Galaxy server itself. Larger installations with access to a centralized user identification management system such as LDAP or kerberos may be able to take advantage of identity caching through configuration of login caching and forwarding via apache or nginx. Attempts to tie our local instance to existing LDAP authentication mechanisms proved to break the existing administrative functionality of the server. These issues were raised with the development team of the Galaxy Project and it is currently being assessed as a possible feature request by Galaxy development team.

Below, we outline the necessary steps undertaken to install and configure our Galaxy standalone instance using the native python based web server and a postgresql back-end database for the storage of our data.

3.1 Galaxy and postgresql installation procedures

3.2 Galaxy standalone instance configuration

4 Galaxy workflow creation

4.1 Example workflow

5 Conclusions