Reproducible Proteomic Workflows using Extensions to the Galaxy Framework

James Johnson[1]; John Chilton[1*]; Pratik Jagtap[1]; Ben Lynch[1]; Tim Griffin[2]

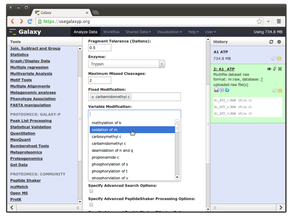
[1]University of Minnesota Supcomputing Institute; [2]University of Minnesota

Introduction to Galaxy

- The Galaxy framework (http://galaxyproject.org/) is a popular and widely cited platform in the genomics community.
  - Open source, web-based.
  - Build complex, reproducible bioinformatics workflows.
- As part of the Galaxy-P project we have:
  - Integrated and developed of tools for proteomic analysis.
  - Built complex workflows for protein identification, quantification, and proteogenomic analysis.

Galaxy-P

- Try out our public server at https://usegalaxyp.org
- Find instructions for building your own Galaxy-P at http://getgalaxyp.org



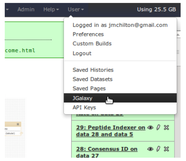
Galaxy-P in the Cloud

- Create your own Galaxy-P cluster on Amazon's cloud infrastructure https://biocloudcentral.msi.umn.edu.
- Launches with CloudMan - an easy-to-use interface for managing your new virtual cluster.

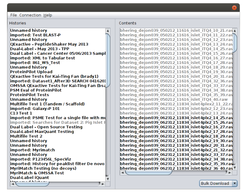
Handling Large Numbers of Files - Challenges

- From Wikipedia, "[Galaxy design choices make] it relatively easy to build typical analyses, but more difficult to build complex workflows..."
- However "typical" in proteomics is more complicated than "typical" in genomics.
- The large number of files resulting from the common use of MudPIT/fractionation means that typical workflows in proteomics are structurally more complex than genomics.
- Galaxy was not geared toward handling large numbers of files.

Handling Large Numbers of Files - JGalaxy



We developed a Java Web Start rich client application called JGalaxy targeting the Galaxy API to allow easy batch file downloads and uploads. This makes it easier to get a large number of files in and out of Galaxy quickly.



Handling Large Numbers of Files - Tool Extensions

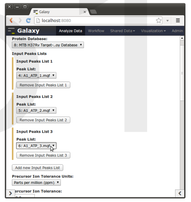


Figure: Traditional multiple input selection.

- Previously Galaxy required a lot of clicking and tracking to provide multiple inputs to a tool.
- Our Galaxy tool extensions allow easy multiple selection.
- **Convenient for jobs requiring a few inputs, necessary for jobs requiring dozens or hundreds.**
- Used in many Galaxy-P proteomics tool wrappers (e.g. MaxQuant, PeptideShaker, Scaffold, etc...).
- Incorporated upstream in core Galaxy framework.

Figure: Our multiple input extension.

Handling Large Numbers of Files - Multiple File Datasets

Figure: Traditional Galaxy workflows would not allow this workflow for two inputs to be used with three inputs.

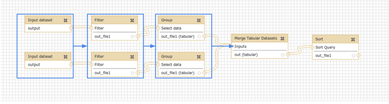


Figure: Key innovation here it to group multiple files into a single dataset resulting in a flatter workflow.

Galaxy cannot create reusable workflows with variable number of input datasets. We have created the concept of multiple file datasets to address this deficiency.

- Key Idea: Group any number of files into one Galaxy "dataset".
- This allows workflows to operate over any number of inputs and produce any number of outputs.
- Built on initial work from Adapt (https://bitbucket.org/glormph/adapt), but extended to transparently work with arbitrary tools and data types.
- Available as part of our galaxy-extras project at https://bitbucket.org/msiappdev/galaxy-extras.
- Additional benefits include implicit "sample tracking" through complex workflows and keeping the Galaxy history user interface manageable and responsive by reducing the number of datasets used and created in such workflows.

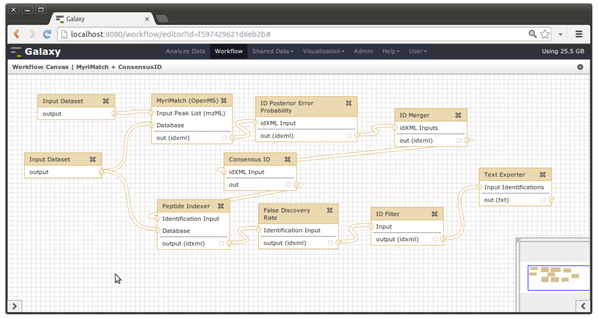


Figure: OpenMS ConsensusID workflow that can operate over any number of peak lists, would not be possible in Galaxy without multiple file dataset extensions. Extension also works seamlessly, the ConsensusID tool does not need to know if it was given multiple input datasets or a single multiple file dataset.

Cross Platform Job Execution - Challenges

Perhaps driven by proprietary data formats, proteomics has a Windows-centric culture that genomics does not. As an upshot, previously there was no easy way to run Windows applications in the Galaxy framework.

The number of important proteomics applications tied to the Windows platform necessitated the creation of the LWR.

Cross Platform Job Execution with the LWR

The LWR is an application that can be installed on any server (Windows or *nix), allowing remote Galaxy instances can submit standard Galaxy jobs to that host.

- Learn how to install the LWR and configuring Galaxy clients at https://lwr.readthedocs.org.
- Check out the LWR source code at https://bit.ly/galaxy-lwr-code.
- Easy to secure and maintain. Runner code has been included in core Galaxy framework.
- LWR allows Galaxy-P to utilize the applications highlighted below (each implemented as standard Galaxy tools).

Tool Highlight - msconvert

- msconvert (part of ProteWizard) is a tool for converting between various mass spec file formats.
- The LWR allows Galaxy-P to utilize the Windows version of msconvert required for converting vendor formats.
- Galaxy-P provides access to many msconvert filtering options:
  - Peak picking (centroiding)
  - Filtering by activation (CID, HCD, etc...), m/z, intensity, MS level, etc...
  - Advanced ETD filtering, deisotoping, and denoising

Tool Highlight - MaxQuant

- A popular quantitative proteomics package for Thermo Fisher instruments developed at the Max Planck Institute.
- Workflow to the right demonstrates combining MaxQuant for SILAC analysis with iQuant for iTRAQ analysis for dually labeled samples.

Tool Highlight - ProteinPilot

- ProteinPilot is a commercial tool developed by AB SCIEX.
- http://www.absciex.com/products/software/proteinpilot-software
- LWR is used for running ProteinPilot as well as supporting tools to extract data from proprietary ProteinPilot outputs.
- The "Minnesota Two Step" workflow for proteogenomics and metaproteomics using ProteinPilot in Galaxy-P is shown to the right.

*chilton@msi.umn.edu