Running and Enhancing your own Galaxy

Daniel Blankenberg
The Galaxy Team
http://UseGalaxy.org

Overview

Where and How you can use and build Galaxy

- public website
- local instance
- on the cloud
- tool shed/contributing tools

Exercise: Installing Galaxy and adding Tools

Overview

Where and How you can use and build Galaxy

- public website
- local instance
- on the cloud
- tool shed/contributing tools

Exercise: Installing Galaxy and adding Tools

Galaxy main site (http://usegalaxy.org)

Public web site, anybody can use

~500 new users per month, ~100 TB of user data, ~130,000 analysis jobs per month, every month is our busiest month ever...

Will continue to be maintained and enhanced, but with limits and quotas

Centralized solution cannot scale to meet data analysis demands

Overview

Where and How you can use and build Galaxy

- public website
- local instance
- on the cloud
- tool shed/contributing tools

Exercise: Installing Galaxy and adding Tools

Local Galaxy instances (http://getgalaxy.org)

Galaxy is designed for local installation and customization

- Just download and run, completely self-contained
- Easily integrate new tools
- Easy to deploy and manage on nearly any (unix) system
- Run jobs on existing compute clusters

Especially useful for sensitive data

- can secure data and abide by regulations

Scale up on existing resources

Move intensive processing (tool execution) to other hosts

Frees up the application server to serve requests and manage jobs

Utilize existing resources

Supports any scheduler that supports DRMAA (most of them)

Running a Production Server

Use a real database server: PostgreSQL, MySQL

Run on compute cluster resources

External Authentication: LDAP, Kerberos, OpenID

Load balancing; proxy support

https://bitbucket.org/galaxy/galaxy-central/wiki/Config/ProductionServer

Lack IT knowledge or resources?

No problem, just use the Cloud

Overview

Where and How you can use and build Galaxy

- public website
- local instance
- on the cloud
- tool shed/contributing tools

Exercise: Installing Galaxy and adding Tools

Cloud Computing

network accessible compute resources that can be rapidly acquired, configured, and released on demand

Infrastructure as a service

Compute resources provided and configured on demand (compute nodes, storage, network)

Public commercial: Amazon Web Services, Rackspace, ...
Build your own: Eucalyptus, Nimbus, OpenStack, ...

When to use the cloud?

Limited informatics expertise or infrastructure

Extended or particular resource needs

Cannot upload data to a shared resource

Need for customization

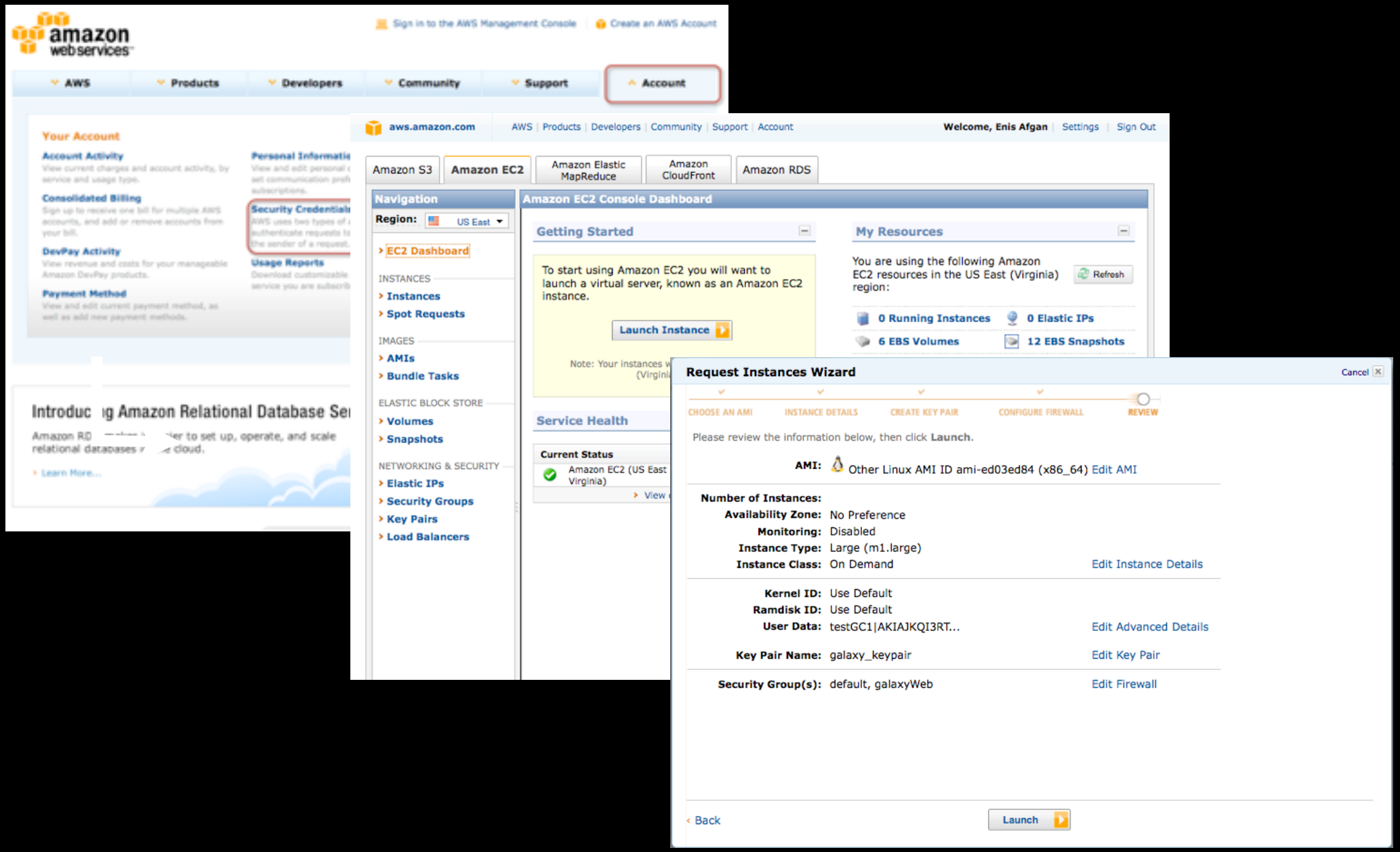
Have oscillating data volume

Deploying Galaxy on the AWS Cloud

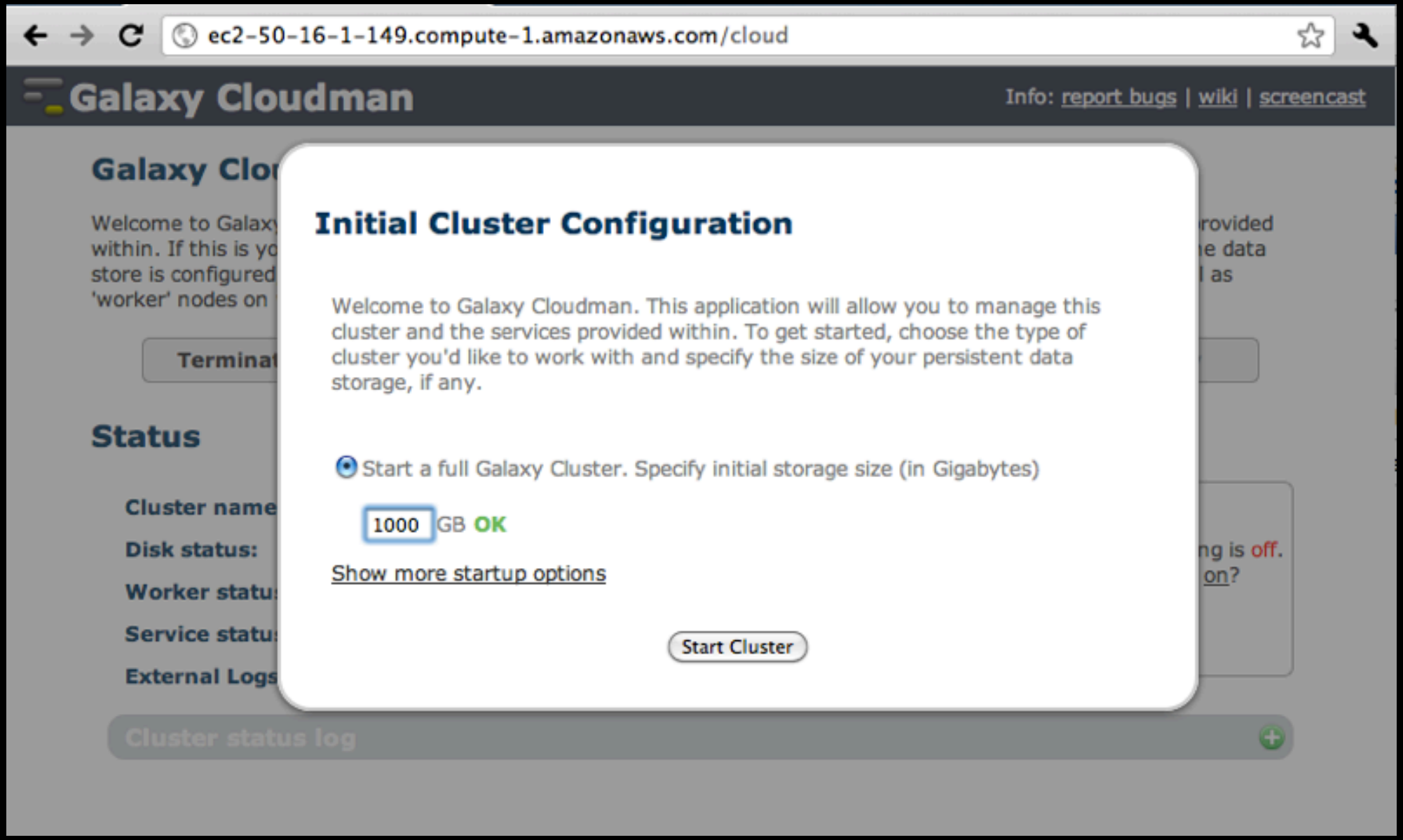
http://usegalaxy.org/cloud

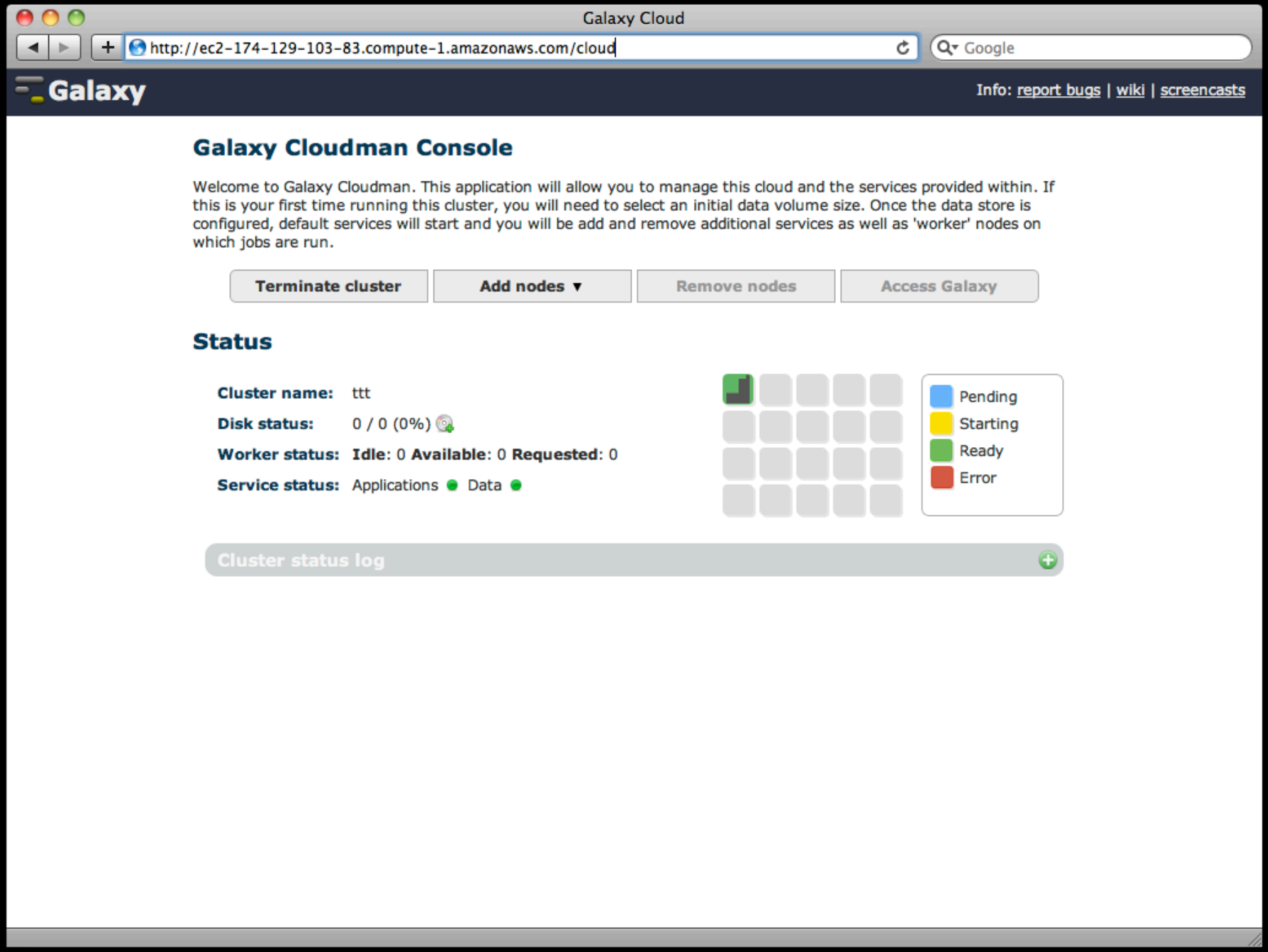
1. Open an AWS account (only once)
2. Use the AWS Management Console to start a master EC2 instance
3. Use the Galaxy CloudMan web interface on the master instance to manage the cluster

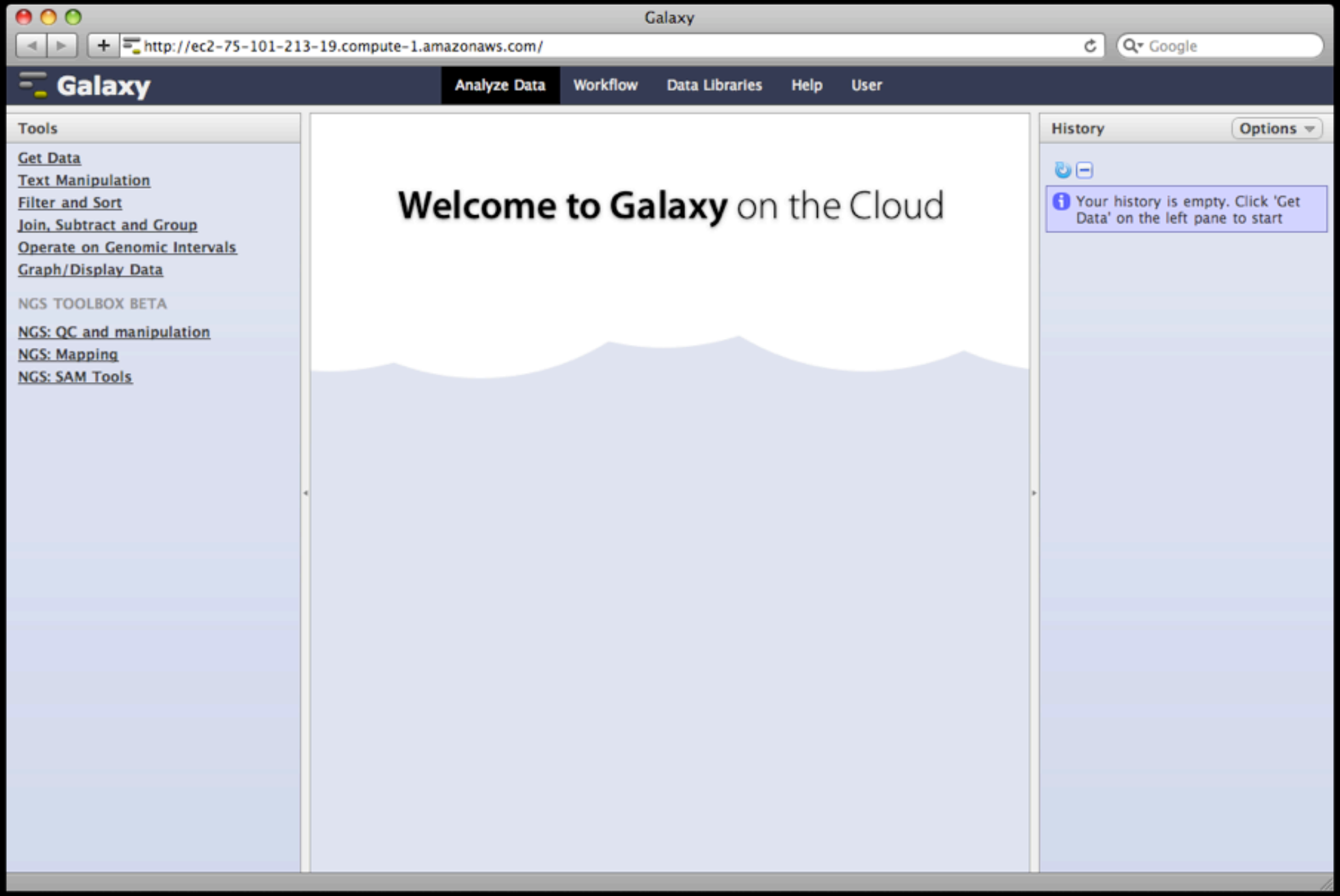
2. Start an EC2 Instance

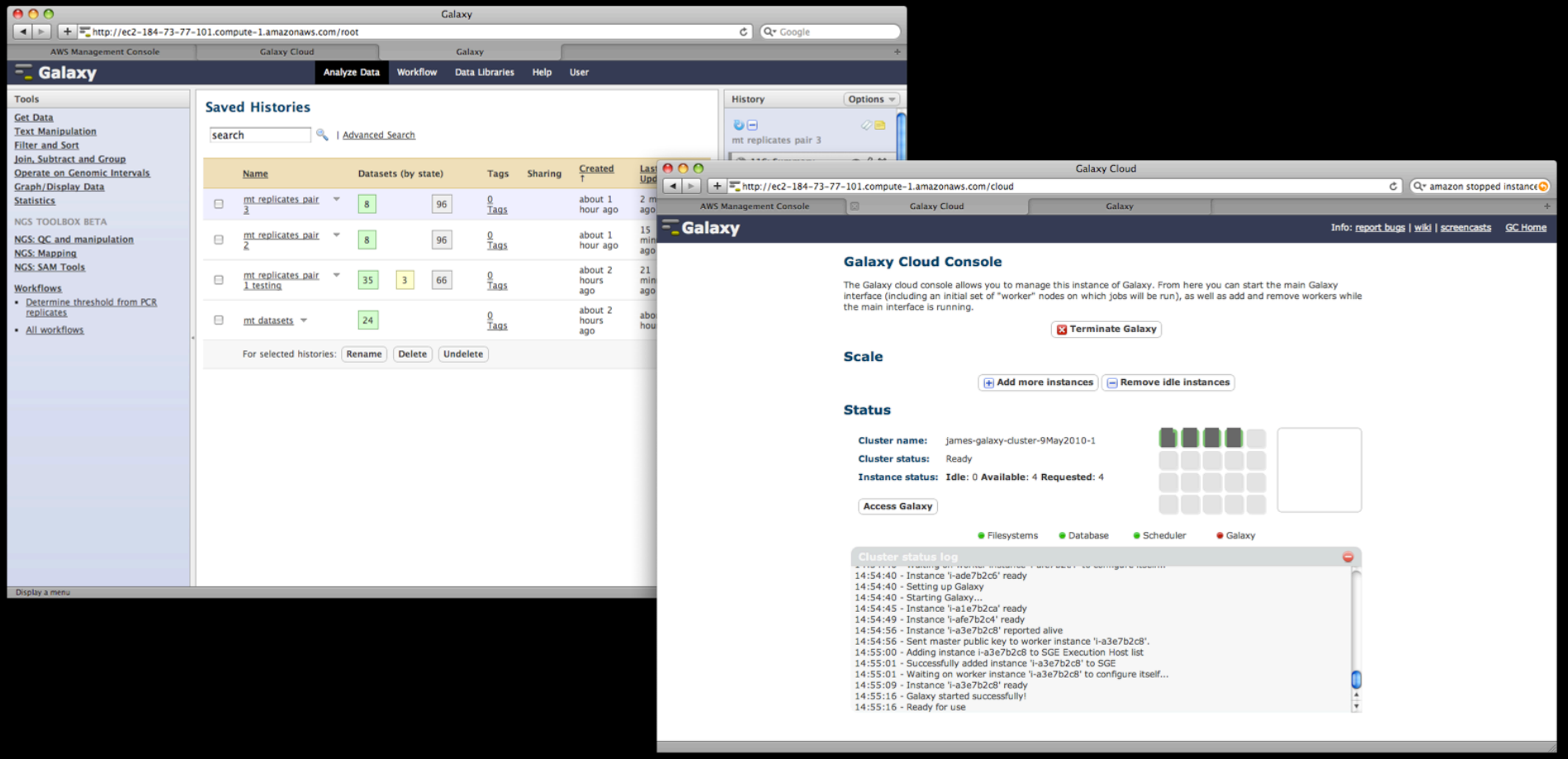


3. Configure Your Cluster

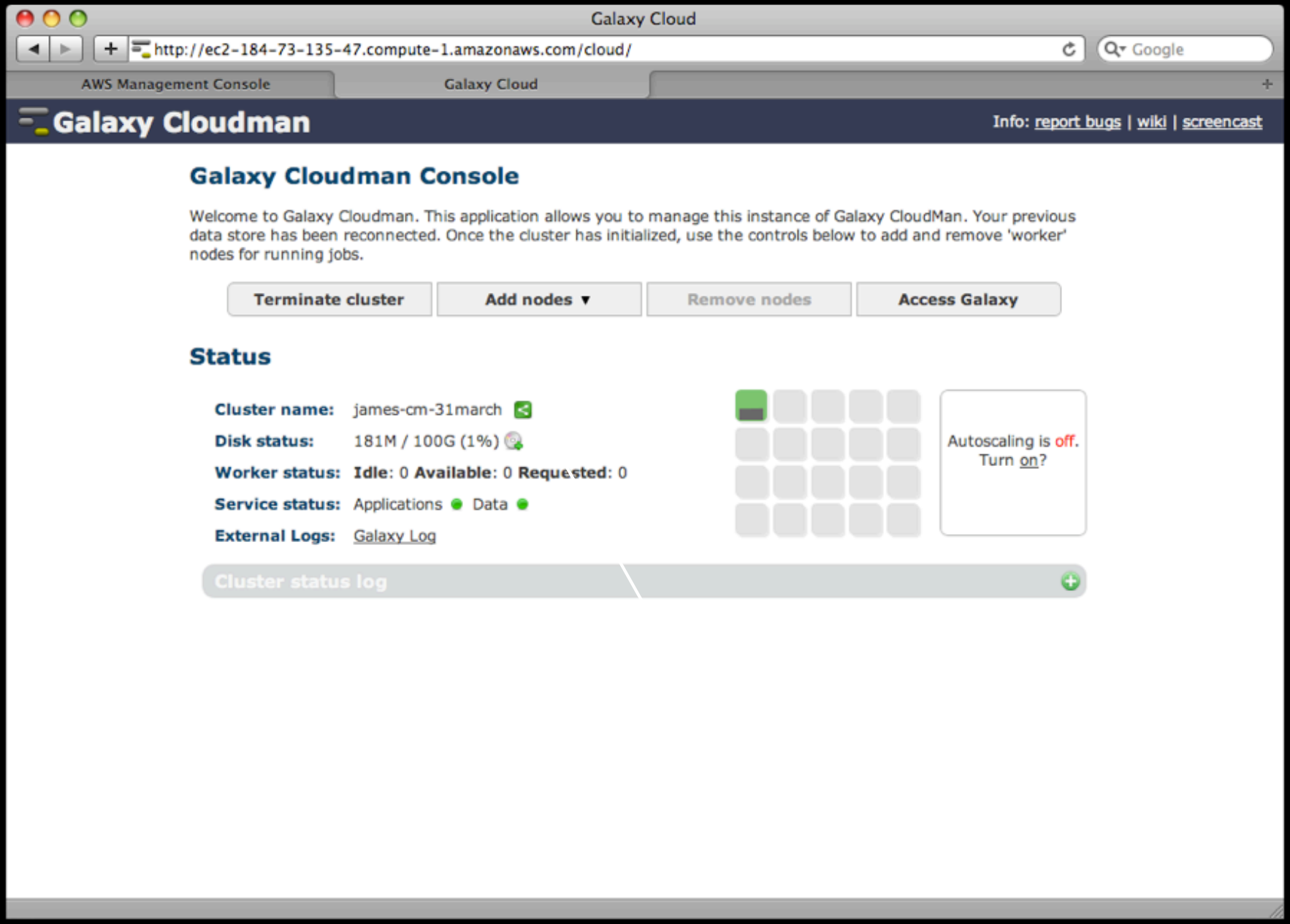


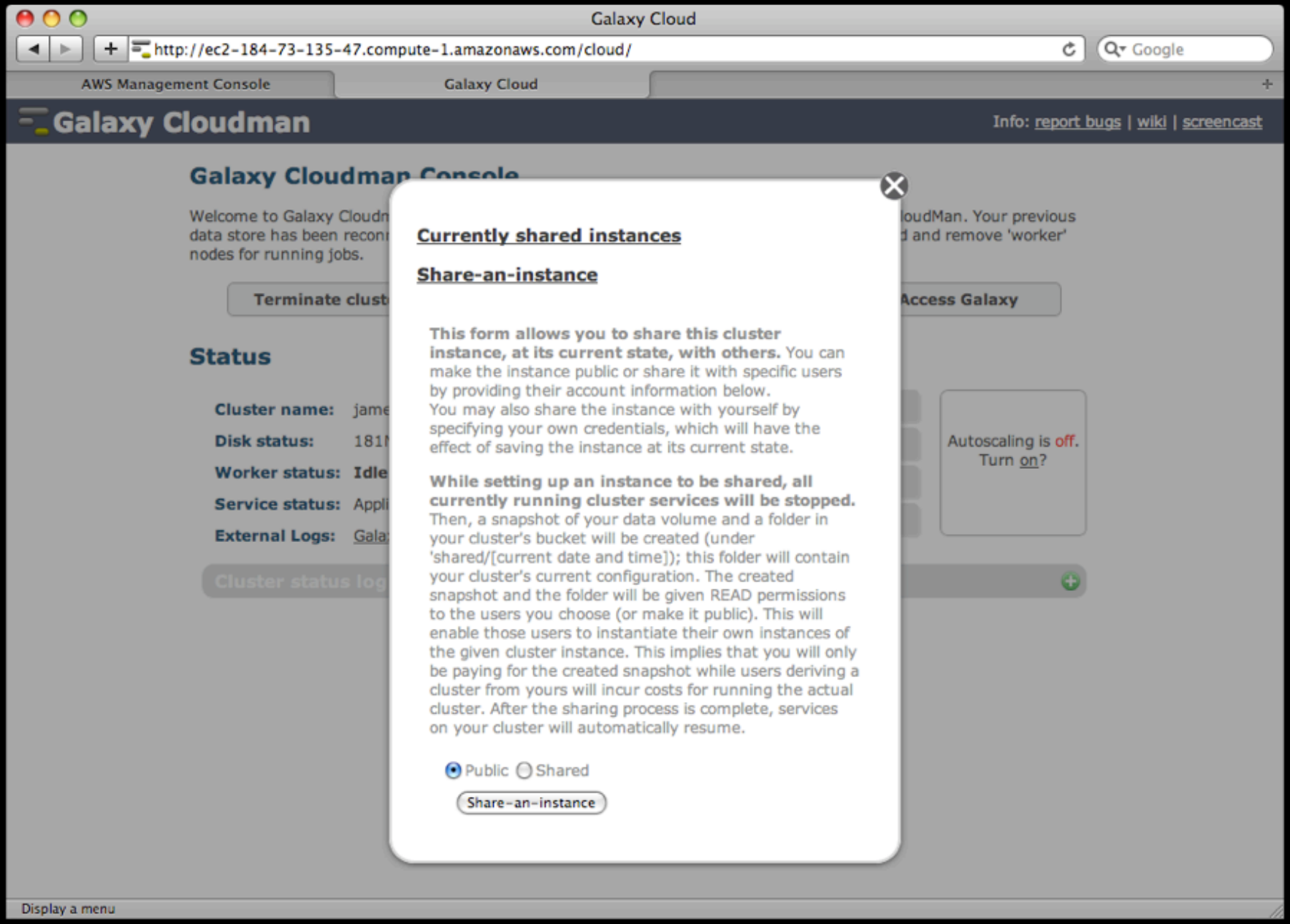

Galaxy Cloud
http://ec2-174-129-103-83.compute-1.amazonaws.com/cloud
Google
Galaxy
Info: report bugs | wiki | screencasts
Galaxy Cloudman Console
Welcome to Galaxy Cloudman. This application will allow you to manage this cloud and the services provided within. If this is your first time running this cluster, you will need to select an initial data volume size. Once the data store is configured, default services will start and you will be add and remove additional services as well as 'worker' nodes on which jobs are run.
Terminate cluster
Add nodes ▾
Remove nodes
Access Galaxy
Status
Cluster name: ttt
Disk status: 0 / 0 (0%)
Worker status: Idle: 0 Available: 0 Requested: 0
Service status: Applications Data
Pending
Starting
Ready
Error
Cluster status log


Galaxy
http://ec2-75-101-213-19.compute-1.amazonaws.com/
Google
Galaxy
Analyze Data
Workflow
Data Libraries
Help
User
Tools
Get Data
Text Manipulation
Filter and Sort
Join, Subtract and Group
Operate on Genomic Intervals
Graph/Display Data
NGS TOOLBOX BETA
NGS: QC and manipulation
NGS: Mapping
NGS: SAM Tools
Welcome to Galaxy on the Cloud
History
Options
Your history is empty. Click 'Get Data' on the left pane to start



Can use like any other Galaxy instance, with additional compute nodes acquired and released (*automatically*) in response to usage


Galaxy Cloud
http://ec2-184-73-135-47.compute-1.amazonaws.com/cloud/
Google
AWS Management Console
Galaxy Cloud
Galaxy Cloudman
Info: report bugs | wiki | screencast
Galaxy Cloudman Console
Welcome to Galaxy Cloudman. This application allows you to manage this instance of Galaxy CloudMan. Your previous data store has been reconnected. Once the cluster has initialized, use the controls below to add and remove 'worker' nodes for running jobs.
Terminate cluster
Add nodes ▾
Remove nodes
Access Galaxy
Status
Cluster name: james-cm-31march
Disk status: 181M / 100G (1%)
Worker status: Idle: 0 Available: 0 Requested: 0
Service status: Applications Data
External Logs: Galaxy Log
Autoscaling is off.
Turn on?
Cluster status log


Galaxy Cloud
http://ec2-184-73-135-47.compute-1.amazonaws.com/cloud/
Google
AWS Management Console
Galaxy Cloud
Galaxy Cloudman
Info: report bugs | wiki | screencast
Galaxy Cloudman Console
Welcome to Galaxy Cloudm
data store has been recon
nodes for running jobs.
loudMan. Your previous
d and remove 'worker'
Terminate clust
Access Galaxy
Status
Cluster name:
Disk status:
Worker status: Idle
Service status:
External Logs:
Autoscaling is off.
Turn on?
Currently shared instances
Share-an-instance
This form allows you to share this cluster instance, at its current state, with others. You can make the instance public or share it with specific users by providing their account information below.
You may also share the instance with yourself by specifying your own credentials, which will have the effect of saving the instance at its current state.
While setting up an instance to be shared, all currently running cluster services will be stopped. Then, a snapshot of your data volume and a folder in your cluster's bucket will be created (under 'shared/[current date and time]); this folder will contain your cluster's current configuration. The created snapshot and the folder will be given READ permissions to the users you choose (or make it public). This will enable those users to instantiate their own instances of the given cluster instance. This implies that you will only be paying for the created snapshot while users deriving a cluster from yours will incur costs for running the actual cluster. After the sharing process is complete, services on your cluster will automatically resume.
Public
Shared
Share-an-instance
Display a menu

Automation

Cloud instances include all tools available in main Galaxy and more

Tool installation and configuration, image creation, etc, all completely automated and extensible

Same automation approach can be used for configuring tool dependencies for a local Galaxy

VM image with tools (not data) also available, currently at http://usegalaxy.org/vm

Overview

Where and How you can use and build Galaxy

- public website
- local instance
- on the cloud
- tool shed/contributing tools

Exercise: Installing Galaxy and adding Tools

The Problem

You have written a Python script to analyze genomic data and you want to share it with command-line averse colleagues

The Galaxy Solution

Solution: Integrate the script as a new Tool into your own Galaxy server

Steps:

- Obtain and install Galaxy source code (GetGalaxy.org)
- Write an XML file describing the inputs and outputs and how to execute the script
- Instruct Galaxy to load the tool

Adding your Own

Write or download a command-line executable

Determine number and kind of

- Input and Output Datasets
- Input Parameters

Construct a descriptive tool configuration XML file

- Write a wrapper script, only if required

Tool Configuration

Tool Action - Default tool action should be adequate (Upload tool uses custom tool action)

Tool Command

Inputs

- Action - Used by datasource tools
- Parameters

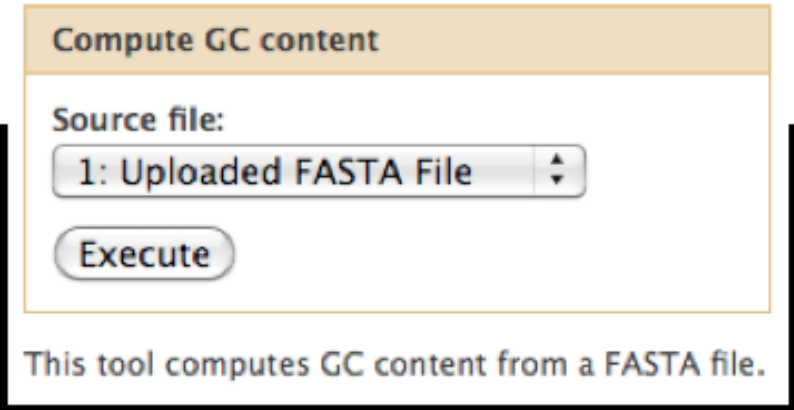
Outputs

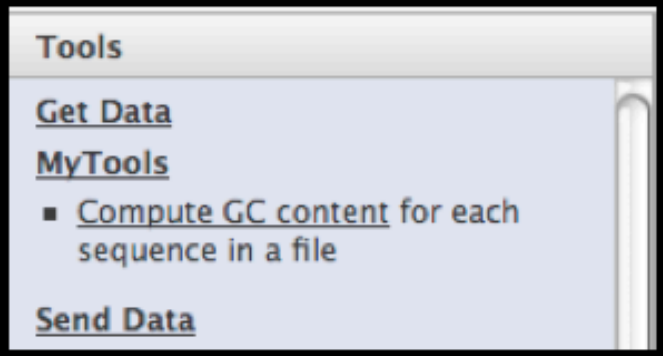
Help

Tests

A Basic Tool

```
<tool id="fa_gc_content_1" name="Compute GC content">
  <description>for each sequence in a file</description>
  <command interpreter="perl">toolExample.pl $input $output</command>
  <inputs>
    <param format="fasta" name="input" type="data" label="Source file"
  </inputs>
  <outputs>
    <data format="tabular" name="output" />
  </outputs>

  <tests>
    <test>
      <param name="input" value="fa_gc_content_input.fa"/>
      <output name="out_file1" file="fa_gc_content_output.txt"/>
    </test>
  </tests>

  <help>
This tool computes GC content from a FASTA file.
  </help>

</tool>
```






```
<section name="MyTools" id="mTools">
    <tool file="myTools/toolExample.xml" />
</section>
```

tool_conf.xml

Cluster

Cluster intervals of: 6: UCSC Main on Human: knownGene

max distance between intervals: 1 (bp)

min number of intervals per cluster: 2

Return type: Merge clusters into single intervals

Execute

TIP: If your query does not appear in the pulldown menu -> it is not in interval format. Use "edit attributes" to set chromosome, start, end, and strand columns

---

Screencasts!

See Galaxy Interval Operation Screencasts (right click to open this link in another window).

Syntax

- **Maximum distance** is greatest distance in base pairs allowed between intervals that will be considered "clustered". **Negative** values for distance are allowed, and are useful for clustering intervals that overlap.
- **Minimum intervals per cluster** allow a threshold to be set on the minimum number of intervals to be considered a cluster. Any area with less than this minimum will not be included in the ouput.
- **Merge clusters into single intervals** outputs intervals that span the entire cluster.
- **Find cluster intervals; preserve comments and order** filters out non-cluster intervals while maintaining the original ordering and comments in the file.
- **Find cluster intervals; output grouped by clusters** filters out non-cluster intervals, but outputs the cluster intervals so that they are grouped together. Comments and original ordering in the file are lost.

Example

cluster.xml

```
<tool id="gops_cluster_1" name="Cluster">
  <description>[[Cluster]] the intervals of a query</description>
  <command interpreter="python2.4">
    gops_cluster.py $input1 $output -1 $input1_chromCol,$input1_startC
                   -d $distance -m $minregions -o $returntype
  </command>
  <inputs>
    <param format="interval" name="input1" type="data">
      <label>Cluster intervals of</label>
    </param>
    <param name="distance" size="5" type="integer" value="1" help="(bp
      <label>max distance between intervals</label>
    </param>
    <param name="minregions" size="5" type="integer" value="2">
      <label>min number of intervals per cluster</label>
    </param>
    <param name="returntype" type="select" label="Return type">
      <option value="1">Merge clusters into single intervals</option>
      <option value="2">Find cluster intervals; preserve comments and
      <option value="3">Find cluster intervals; output grouped by clus
      <option value="4">Find the smallest interval in each cluster</op
      <option value="5">Find the largest interval in each cluster</opt
    </param>
  </inputs>
  <help>

.. class:: infomark

**TIP:** If your query does not appear in the pulldown menu -> it is n

-----

**Screencasts!**

See Galaxy Interval Operation Screencasts_ (right click to open this l

.. _Screencasts: http://www.bx.psu.edu/cgi-bin/trac.cgi/wiki/GopsDesc

-----

**Syntax**

- **Maximum distance** is greatest distance in base pairs allowed betw
- **Minimum intervals per cluster** allow a threshold to be set on the
- **Merge clusters into single intervals** outputs intervals that span
- **Find cluster intervals; preserve comments and order** filters out
- **Find cluster intervals; output grouped by clusters** filters out n
```

Line: 87 Column: 8 XML Soft Tabs: 2

Input Parameter types

Basic

- Text
- Integer
- Float
- Select
  - Static
  - Dynamic
- Boolean
- Genome build
- Data column
- Data
- Hidden
- Base URL
- File
- Drill down
- Grouping
  - Conditional
  - Repeat
- Config Files

Datasets and Datatypes

All datasets are associated with a Datatype

- File format
- Type of Data: genomic intervals, sequence, alignment
- Hierarchical structure useful for inputs
- Automatic conversion possible
- Metadata

datatypes_conf.xml and lib/galaxy/datatypes

https://bitbucket.org/galaxy/galaxy-central/wiki/AddingDatatypes

Adding your Own Display Application

Define An XML configuration which describes how and where to present the data to the External Web Application

- Static
- Dynamic - display options can be loaded from a file

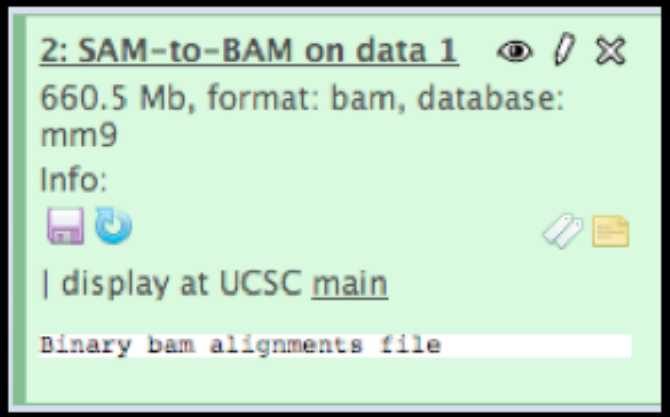
Inform Galaxy about the new display by adding to the appropriate datatype in datatypes_conf.xml

https://bitbucket.org/galaxy/galaxy-central/wiki/ExternalDisplayApplications/Tutorial

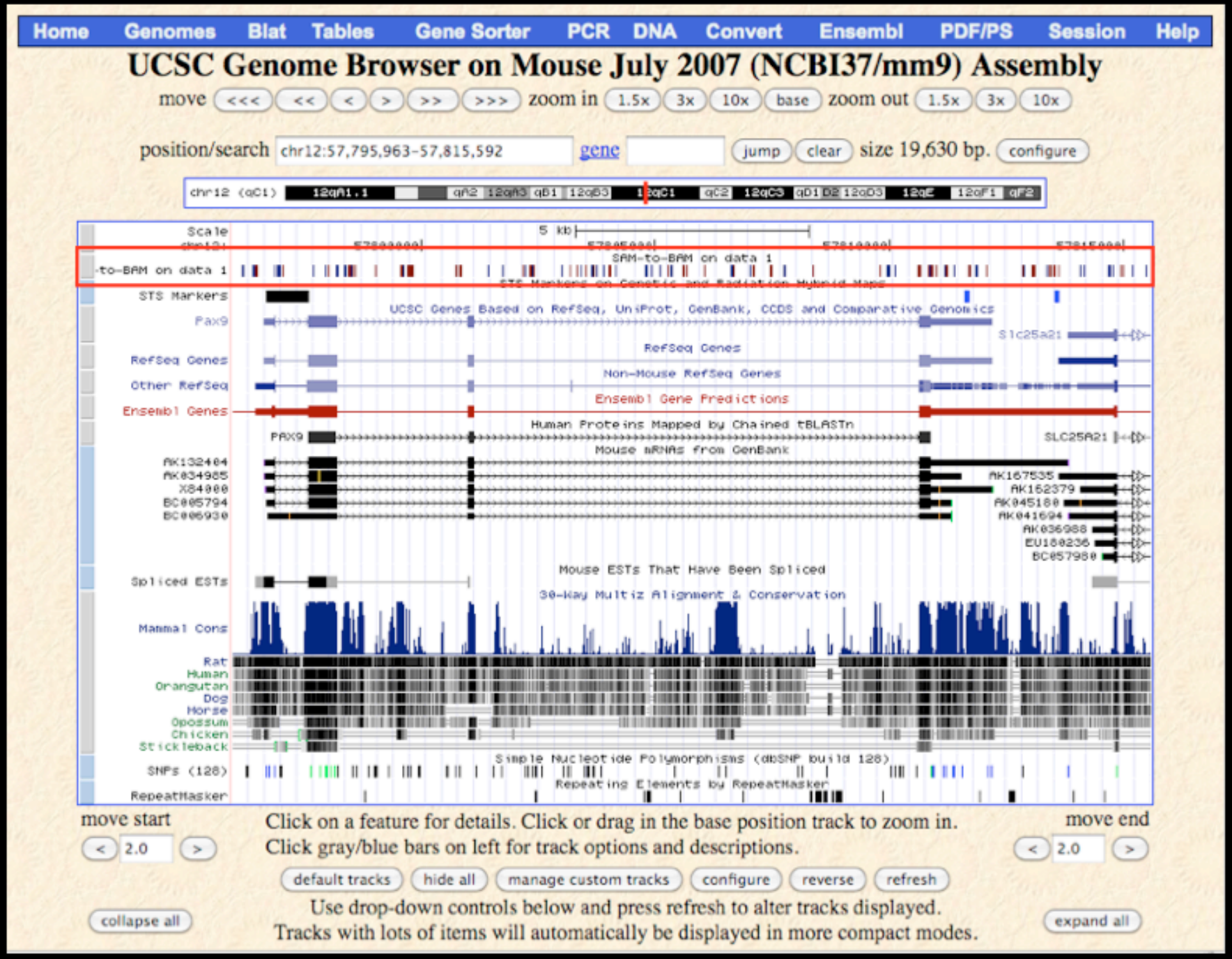
Static External Display Application

```
<display id="ucsc_bam" version="1.0.0" name="display at UCSC">
    <link id="main" name="main">
        <url>http://genome.ucsc.edu/cgi-bin/hgTracks?db=${qp($bam_file.dbkey)}&amp;hgt.customText=${qp($track.url)}</url>
        <param type="data" name="bam_file" url="galaxy.bam" strip_https="True" />
        <param type="data" name="bai_file" url="galaxy.bam.bai" metadata="bam_index" strip_https="True" />
        <param type="template" name="track" viewable="True" strip_https="True">
            track type=bam name="${bam_file.name}" bigDataUrl=${bam_file.url} db=${bam_file.dbkey}
        </param>
    </link>
</display>
```

```
<datatype extension="bam" type="galaxy.datatypes.binary:Bam"
    mimetype="application/octet-stream" display_in_upload="true">
    <display file="ucsc/bam.xml" />
</datatype>
```



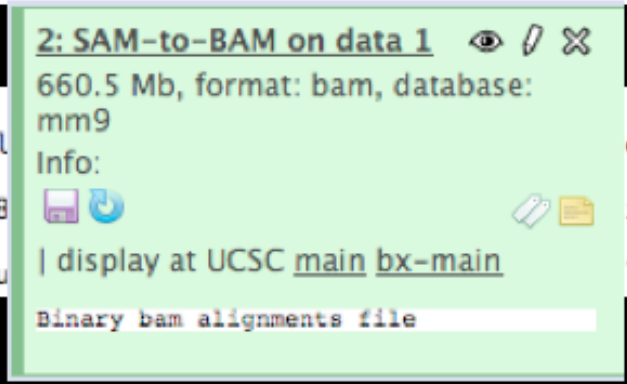
BAM at UCSC


Home Genomes Blat Tables Gene Sorter PCR DNA Convert Ensembl PDF/PS Session Help
UCSC Genome Browser on Mouse July 2007 (NCBI37/mm9) Assembly
move <<< << < > >> >>> zoom in 1.5x 3x 10x base zoom out 1.5x 3x 10x
position/search chr12:57,795,963-57,815,592 gene jump clear size 19,630 bp. configure
SAM-to-BAM on data 1
STS Markers
UCSC Genes Based on RefSeq, UniProt, GenBank, CCDS and Comparative Genomics
Pax9
Slc25a21
RefSeq Genes
Non-Mouse RefSeq Genes
Other RefSeq
Ensembl Gene Predictions
Ensembl Genes
Human Proteins Mapped by Chained tBLASTn
Mouse mRNAs from GenBank
Mouse ESTs That Have Been Spliced
Spliced ESTs
30-Way Multiz Alignment & Conservation
Mammal Cons
Rat
Human
Orangutan
Dog
Horse
Opossum
Chicken
Stickleback
Simple Nucleotide Polymorphisms (dbSNP build 126)
SNPs (128)
Repeating Elements by RepeatMasker
RepeatMasker
move start
move end
Click on a feature for details. Click or drag in the base position track to zoom in.
Click gray/blue bars on left for track options and descriptions.
default tracks hide all manage custom tracks configure reverse refresh
Use drop-down controls below and press refresh to alter tracks displayed.
Tracks with lots of items will automatically be displayed in more compact modes.
collapse all
expand all

Dynamic External Display Application

```
<display id="ucsc_bam" version="1.0.0" name="display at UCSC">
    <!-- Load links from file: one line to one link -->
    <dynamic_links from_file="tool-data/shared/ucsc/ucsc_build_sites.txt" skip_startswith="#" id="0" name="0">

        <!-- Define parameters by column from file, allow splitting on builds -->
        <dynamic_param name="site_id" value="0"/>
        <dynamic_param name="ucsc_link" value="1"/>
        <dynamic_param name="builds" value="2" split="True" separator="," />

        <!-- Filter out some of the links based upon matching site_id to a Galaxy application configuration parameter and b
        <filter>${site_id in $APP.config.ucsc_display_sites}</filter>
        <filter>${dataset.dbkey in $builds}</filter>

        <!-- We define url and params as normal, but values defined in dynamic_param are available by specified name -->
        <url>${ucsc_link}db=${qp($bam_file.dbkey)}&amp;hgt.customText=${qp($track.url)}</url>
        <param type="data" name="bam_file" url="galaxy_${DATASET_HASH}.bam" strip_https="True" />
        <param type="data" name="bai_file" url="galaxy_${DATASET_HASH}.bam.bai" metadata="bam_index" strip_https="True" />
        <param type="template" name="track" viewable="True" strip_https="True">
            track type=bam name="${bam_file.name}" bigDataUrl=${bam_file.url} db=${bam_file.dbkey}
        </param>

    </dynamic_links>
</display>
```

```
#Harvested from http://genome.ucsc.edu/cgi-bin/das/dsn
main    http://genome.ucsc.edu/cgi-bin/hgTracks?    anoCar1,ce6,ce4,ce2,rn3,l
#Harvested from http://archaea.ucsc.edu/cgi-bin/das/dsn
archaea http://archaea.ucsc.edu/cgi-bin/hgTracks?   therSibi1,symbTher_IAM148
#Harvested from http://main.genome-browser.bx.psu.edu/cgi-bin/das/dsn
bx-main http://main.genome-browser.bx.psu.edu/cgi-bin/hgTracks? oviAri1,eriEu
```

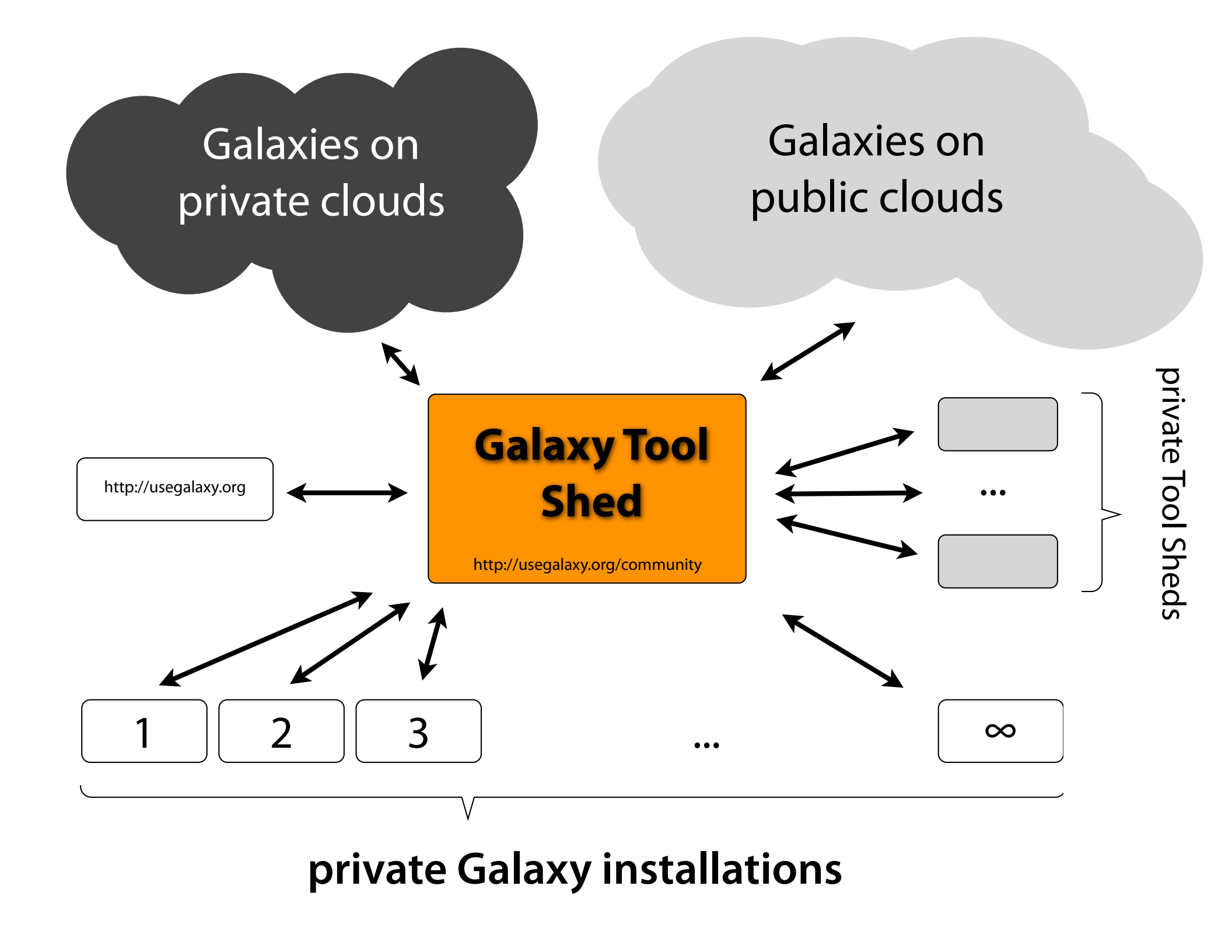


You added a tool, now what?

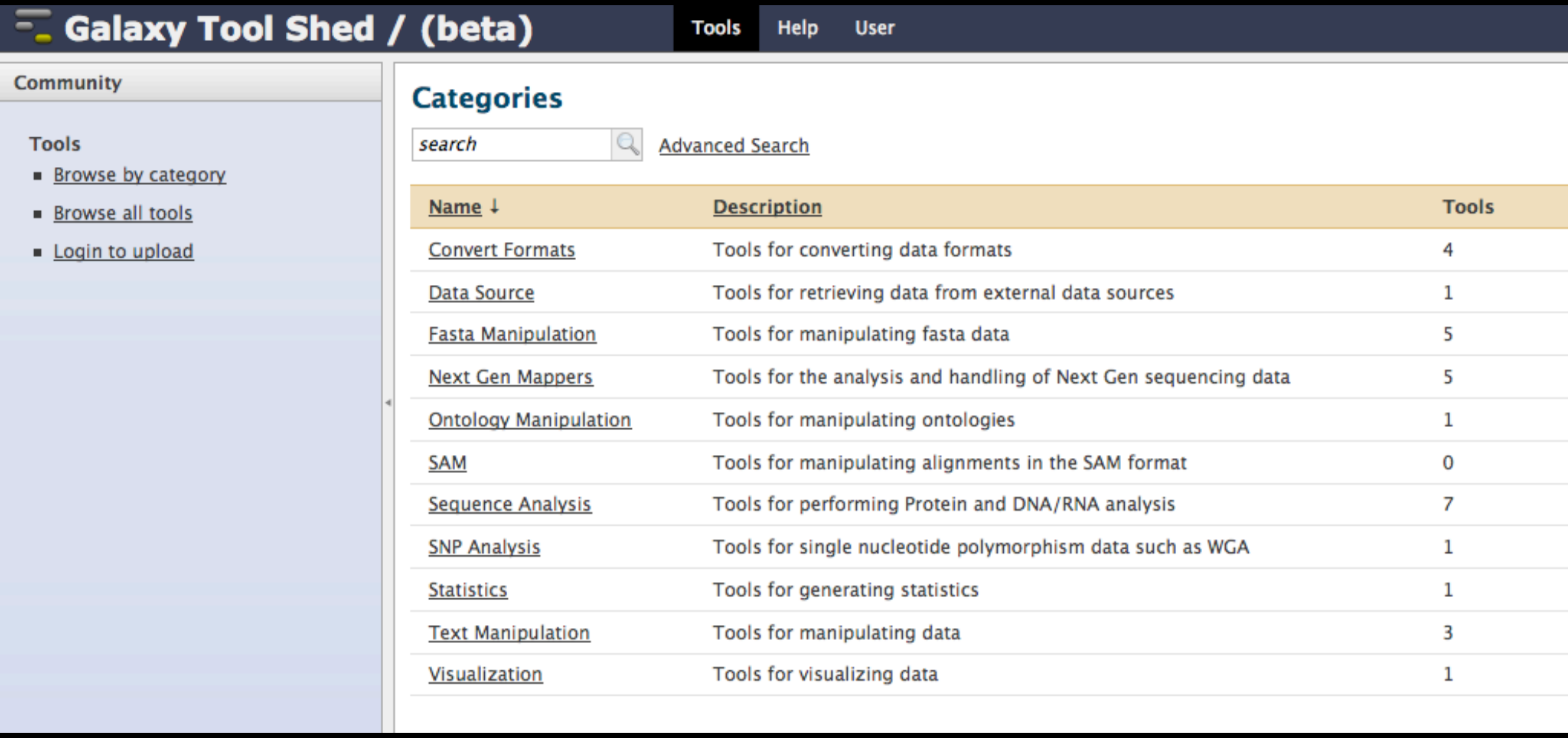
Share it with the community!

Galaxy Tool Shed

- Upload and Download contributed tools
- Rate and provide comments and feedback


Galaxies on private clouds
Galaxies on public clouds
Galaxy Tool Shed
http://usegalaxy.org/community
http://usegalaxy.org
private Tool Sheds
1
2
3
...
∞
private Galaxy installations

Get and Contribute Tools



http://usegalaxy.org/community

Try it now:

http://usegalaxy.org

Develop and deploy:

http://getgalaxy.org

http://galaxyproject.org

Come do cool stuff, contact us at:

http://wiki.g2.bx.psu.edu/News/Galaxy is Hiring

Opportunities for collaboration, positions for postdocs, researchers, software engineers

Overview

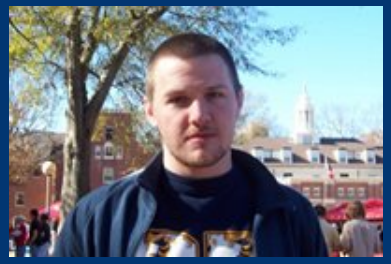
Where and How you can use and build Galaxy

- public website
- local instance
- on the cloud
- tool shed/contributing tools

Exercise: Installing Galaxy and adding Tools

EMORY

Enis Afgan



Dannon Baker

Dave Clements

Jeremy Goecks

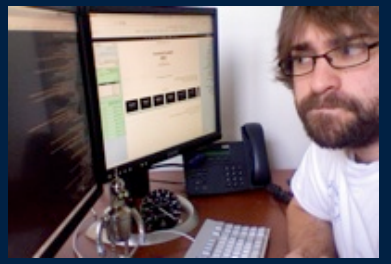
Kanwei Li

James Taylor

PENNSTATE

1855

Dan Blankenberg



Nate Coraor

Jennifer Jackson

Greg von Kuster

Guru Ananda

Anton Nekrutenko

Supported by the **NHGRI** (HG005542, HG004909, HG005133), **NSF** (DBI-0850103), Penn State University, Emory University, and the Pennsylvania Department of Public Health

Download and Install

GetGalaxy.org

Requirements:

- Linux / Mac OS
- Python 2.5 - 2.7
- Mercurial (hg) for downloading (preferred), tar.gz available
- Internet connectivity for setup of dependencies

Follow directions: http://GetGalaxy.org

Adding a Tool

GetGalaxy.org/wiki

Requirements:

- Have or write a Command Line executable
- Determine inputs and outputs of tool
- Write XML description of tool
- Instruct Galaxy to load tool

Follow directions: http://wiki.g2.bx.psu.edu/Admin/Tools/Add Tool Tutorial

Deploying Galaxy on the AWS Cloud

http://usegalaxy.org/cloud

1. Open an AWS account (only once)
2. Use the AWS Management Console to start a master EC2 instance
3. Use the Galaxy CloudMan web interface on the master instance to manage the cluster