# Galaxy CloudMan

Enis Afgan & Dannon Baker GCC 2012, Chicago

# The big picture



### When to use the cloud?

- Don't have informatics expertise or the infrastructure to run and maintain
- Have variable or particular resource needs
- Cannot upload data to a shared resource
- Need for customization
- Have oscillating data volume
- Want to test or share a tool, quickly & safely
- Want to make your analysis readily available to others
- Want fast access to AWS public datasets

### Deploying an instance

- 1. Get an account on the supported cloud
- 2. Start a master instance via the cloud web console or CloudLaunch
- 3. Use CloudMan's web interface on the master instance to manage the platform
- 4. Use or modify Galaxy

### Available functionality

- Start/launch through a web browser or the command line
- Choose between four cluster types
- Terminate/restart
- Scale
- Auto-scale
- Spot support
- Persist changes
- Share-an-instance (customized one too)
- Expand the file system
- Customize (via CLI or the Tool Shed): tools, data, references
- Access through ssh
- Mount an S3 bucket -> data library
- Control the Galaxy process
- API (via Blend)
- Build your own image (via mi-deployment)

# Workshop plan

- Launch an instance
- Go through (most of) the 'Galaxy 101' tutorial while using the following CloudMan features :
  - Auto-scaling
  - Using an S3 bucket as a data source
  - Accessing an instance over ssh
  - Customizing an instance
  - Controlling Galaxy
  - Persisting changes to an instance (maybe)
  - Sharing-an-instance
  - API (time permitting)

### **GENTLEMAN, START YOUR BROWSERS**



### Start an Instance

- 1. Go to the URL on the sheet handed out at the beginning of the workshop and login
- 2. Create a security group (use your username as the name of the security group)
- 3. Create a key pair (*use your username as the key name and save it locally*)
- 4. Start an instance (ami-78a00411), composing the user data in the process and using your username as the cluster name
- 5. Wait for the instance to start

For exact details, see Step 2 on wiki.g2.bx.psu.edu/CloudMan

### **Configure Your Cluster**

#### CloudMan from Galaxy

Admin | Report bugs | Wiki | Screencast



### Manage Your Cluster

#### CloudMan from Galaxy

Admin | Report bugs | Wiki | Screencast

#### CloudMan Console

Welcome to <u>CloudMan</u>. This application allows you to manage this instance cloud cluster and the services provided within. Your previous data store has been reconnected. Once the cluster has initialized, use the controls below to manage services provided by the application.

ļ	Terminate cluster		Add nodes ▼	Remove nodes v	Access Galaxy	
Sta	tus					
CI	uster name:	ghem				
Di	sk status:	0/0(0%)			Autoscaling is off.	
w	orker status: Idle: 4 Available: 2 Requested: 5				Turn on?	
Se	rvice status:	Applications	🔵 Data 🎯			

-	Ga	la)	cy

Tools search tools Get Data Send Data **ENCODE Tools** Lift-Over **Text Manipulation Filter and Sort** Join, Subtract and Group **Convert Formats Extract Features Fetch Sequences** Fetch Alignments **Get Genomic Scores Operate on Genomic Intervals** Statistics Wavelet Analysis Graph/Display Data **Regional Variation** Multiple regression **Multivariate Analysis** Evolution Motif Tools **Multiple Alignments** Metagenomic analyses

**FASTA** manipulation

NGS: Picard (beta) NGS: Mapping NGS: Indel Analysis NGS: RNA Analysis NGS: SAM Tools

NGS: QC and manipulation

NGS: GATK Tools (beta)

NCBI BLAST+

#### Welcome to Galaxy on the Cloud managed by CloudMan

Visualization -

Help 🔻

User 🔻

His	story	٥
0		
00	lytes	
O	Data' on the left pane to start	t

Using 0%

>

Register as a new user (use gcc2012 as the password because you'll share it with your neighbor later)

Shared Data -

Analyze Data

츕

Workflow

# Workshop plan

- Launch an instance
- Go through (most of) the 'Galaxy 101' tutorial while using the following CloudMan features :
  - Auto-scaling
  - Using an S3 bucket as a data source
  - Accessing an instance over ssh
  - Customizing an instance
  - Controlling Galaxy
  - Persisting changes to an instance (maybe)
  - Sharing-an-instance
  - API (time permitting)

### Auto-scaling

Autoscaling is off. Turn on?

#### **Autoscaling Configuration**

Autoscaling attempts to automate the elasticity offered by cloud computing for this particular cluster. Once turned on, autoscaling takes over the control over the size of your cluster.

Autoscaling is simple, just specify the cluster size limits you want to want to work within and use your cluster as you normally do. The cluster will not automatically shrink to less than the minimum number of worker nodes you specify and it will never grow larger than the maximum number of worker nodes you specify.

While respecting the set limits, if there are more jobs than the cluster can comfortably process at a given time autoscaling will automatically add compute nodes; if there are cluster nodes sitting idle at the end of an hour autoscaling will terminate those nodes reducing the size of the cluster and your cost.

Once turned on, the cluster size limits respected by autoscaling can be adjusted or autoscaling can be turned off.

u Aaximu	m number of no	des to maint
3	ок	
Type of	Nodes(s):	
Same as	Master	\$

### Using an S3 bucket as a data source

A Coudentral 3/ Coudent	H AWS Management Core	tole 📧 🤗 ANS Management C	ansole # 10			
← → C # ③ 75.101.217.82/cloud/root/admin#		Destantion of the Contractor Street and a				2 4
CloudMan from Galaxy					Admin I	Report Jours   Wild   Screencest
	Available file system Add a new file system Add a new file system adding 53 buckets as a d dard owned by the user of system under deal (book dard owned by the user of system under deal (book dard owned by the user of system under deal (book dard owned by the user of data file system (book name in the user of data file system) • Bucket name in the data file system (book data file system) • Bucket controls Inter / System controls file system (book data file system) • Bucket controls the advectors of system (book data file system) • Base controls • Base controls the advectors • Base of the output the data • Status data file system • Status control used to come set - 5 (path to close • Status control used to come set - 5 (path to close • Status control used to come set - 5 (path to close • Status control used to come set - 5 (path to close • Status control the system control • Status control the system control • Status control the system control • Status control the system control		Emake R eventable as a Nock P Punctionality. These buckets d, the file system will be even (intercepted by Doubting, Curr Start Restact Update, DB Start Restact Update, DB	The system. Currently, may be public or private lable on the underlying control running a Galaxy gated a AWE canadel Access to:	° CC2012	

### Accessing an instance over ssh

- \$ chmod 400 GCC2012.pem
- \$ ssh -i GCC2012.pem ubuntu@ec2-107-21-161-225.compute-1.amazonaws.com

- Once logged in, you have full system access to your instance, including sudo; use it as any other system
- galaxy user exists on the system and should be used when manipulating Galaxy (sudo su galaxy)

### Customizing an instance

- Edit Galaxy's configuration
  - \$ sudo su galaxy
  - \$ cd /mnt/galaxyTools/galaxy-central
  - \$ vi universe\_wsgi.ini

allow\_library\_path\_paste = True

- Add a new tool via the command line
- Add a new tool via the Tool Shed

### Controlling Galaxy

- Restart Galaxy so the config change takes effect
- Add an admin user
  - Use the email you registered with
- Within Galaxy, create a Data Library, using S3 bucket path as the data source (/mnt/gcc2012)
  - This will import SNPs.bed dataset to the Data Library
  - Import that dataset into a history

### Galaxy 101

- usegalaxy.org/galaxy101/
- Follow through the end of Step 2
  - Instead of downloading SNPs from UCSC, import them from the created Data Library

### Sharing-an-Instance

- Share the entire Galaxy CloudMan platform
  - Includes all of user data and even the customizations
- Publish a self-contained analysis
- Make a note of the share-string and send it to your neighbor



Name	Instance ID	References
Exome sequencing pipeline	cm-b53c6f1223f966914df347687f6fc818/shared/2011-10-0714-00	Pipeline descriptio

# Use CloudLaunch to start a new, derived, instance

- http://bit.ly/Pgv2yf
- CloudLaunch alleviates all instance startup/ setup steps (and is fully interchangeable with the web console!)
- Choose Share-an-instance type and enter your neighbor's share-string
- After Galaxy starts up, login using your neighbor's email address & see all the data

### API (via Blend)

### blend

- A python library for interacting with Galaxy's API
- And CloudMan

• Documentation available at http://blend.readthedocs.org/

blend/ bcc/ init .py cloudman/ init .py launch.py v galaxy/ datasets/ init .py histories/ init .py libraries/ init\_.py users/ \_init\_\_.py workflows/ init\_.py init\_.py client.py init .py config.py build/ dist/ docs/ build/ static/ templates/ api docs/ v examples/ biocloudcentral basic us cloudman basic usage sce example1.py galaxy-upload\_to\_workflo conf.py index.rst Makefile Test tests/ init .py bcc\_tests.py cloudman tests.py Distribute README.rst setup.pv

Request compute infrastructure

Manipulate compute infrastructure

Upload data and run analyses

#### Docs and examples included

### Requesting an Instance

\$ [sudo] pip install blend-lib

\$ python

from blend.cloudman.launch import CloudManLaunch
cml = CloudManLaunch(access\_key, secret\_key)
c = cml.launch('GCC2012\_CloudMan\_Workshop',
 img\_id,'m1.large', 'gcc2012')
cml.get\_status()

Just an example, we'll skip this step

### Working with an Instance

from blend.cloudman import CloudMan
cm = CloudMan("instance IP", "gcc2012")
# cm.initialize(type="Galaxy")
cm.get\_status()
cm.add\_nodes(1, 't1.micro')

### Working with Galaxy's API

from blend.galaxy import GalaxyInstance

Come to the Galaxy API workshop WS6 @ 11am and 3:30pm (in this room)

# Workshop plan

- Launch an instance
- Go through (most of) the 'Galaxy 101' tutorial while using the following CloudMan features :
  - Auto-scaling
  - Using an S3 bucket as a data source
  - Accessing an instance over ssh
  - Customizing an instance
  - Controlling Galaxy
  - Persisting changes to an instance (maybe)
  - Sharing-an-instance
  - API (time permitting)

