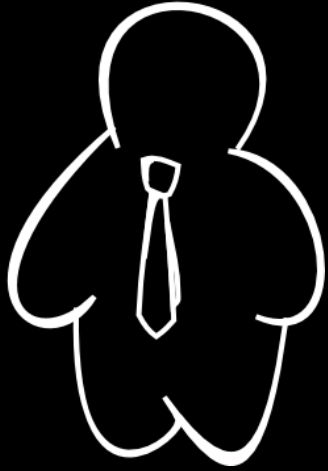


CloudMan



CloudMan – cloud clusters for everyone

Enis Afgan

usecloudman.org

This is accessibility!

← → ↻ <https://usegalaxy.org> ☆ ☰

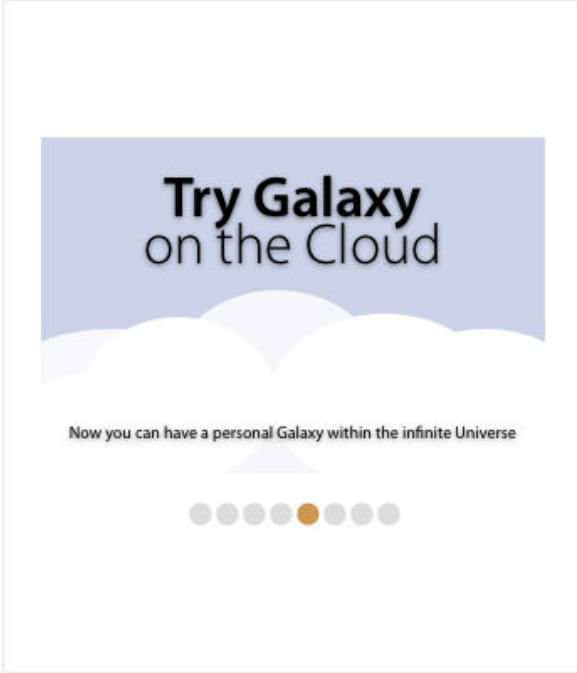
Galaxy Analyze Data Workflow Shared Data Visualization Cloud Help User Using 0 bytes

Tools

search tools

- Get Data
- Send Data
- ENCODE Tools
- Lift-Over
- Text Manipulation
- Convert Formats
- FASTA manipulation
- Filter and Sort
- Join, Subtract and Group
- Extract Features
- Fetch Sequences
- Fetch Alignments
- Get Genomic Scores
- Operate on Genomic Intervals
- Statistics
- Graph/Display Data
- Regional Variation
- Multiple regression
- Multivariate Analysis
- Evolution
- Motif Tools
- Multiple Alignments
- Metagenomic analyses
- Genome Diversity
- Phenotype Association
- NGS TOOLBOX BETA
- NGS: QC and manipulation
- NGS: Mapping
- NGS: SAM Tools
- NGS: GATK Tools (beta)

Galaxy is an open source, web-based platform for data intensive biomedical research. If you are new to Galaxy [start here](#) or consult our [help resources](#).



Try Galaxy on the Cloud





Now you can have a personal Galaxy within the infinite Universe

Tweets Follow

Peter Cock @pjacock 23h
If you #usegalaxy and want to do "samtools idxstats" my wrapper is now available to install from the Galaxy Tool Shed
toolshed.g2.bx.psu.edu/view/peterjc/s..
Retweeted by Galaxy Project
Expand

Galaxy Project @galaxyproject 20 Nov
GalaxyAdmins Meetup in 10 minutes: GCC2013 followup & usegalaxy.org moves to TACC bit.ly/GOyDmB Start connecting now.

Galaxy Project @galaxyproject 20 Nov
Tweet to @galaxyproject

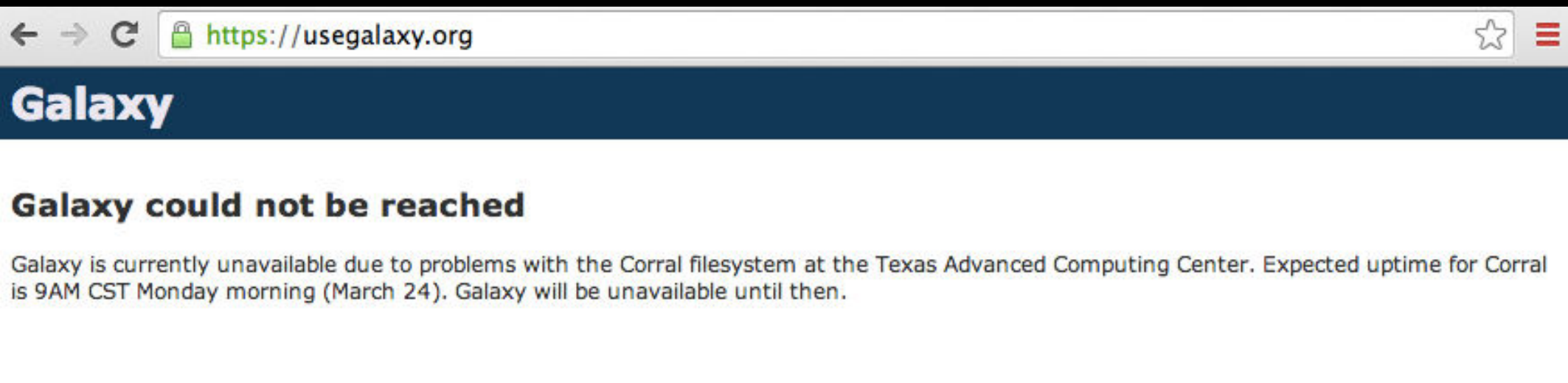
PENNSSTATE  **EMORY UNIVERSITY**  **TACC**  **iPlant Collaborative** 

The [Galaxy Team](#) is a part of the [Center for Comparative Genomics and Bioinformatics](#) at [Penn State](#), and the [Biology and Mathematics & Computer Science](#) departments at [Emory University](#).




This instance of Galaxy is utilizing infrastructure generously provided by the [iPlant Collaborative](#) at the [Texas Advanced Computing Center](#), with support from the [National Science Foundation](#).

History Unnamed history 0 bytes
Your history is empty. Click 'Get Data' on the left pane to start

But only sometimes...



The image shows a screenshot of a web browser window. The address bar at the top displays the URL <https://usegalaxy.org>. Below the address bar, the word "Galaxy" is written in a large, bold, white font on a dark blue background. The main content area of the browser shows a message in bold black text: "Galaxy could not be reached". Below this, a paragraph of smaller black text explains the issue: "Galaxy is currently unavailable due to problems with the Corral filesystem at the Texas Advanced Computing Center. Expected uptime for Corral is 9AM CST Monday morning (March 24). Galaxy will be unavailable until then."

← → ↻  <https://usegalaxy.org>  

Galaxy

Galaxy could not be reached

Galaxy is currently unavailable due to problems with the Corral filesystem at the Texas Advanced Computing Center. Expected uptime for Corral is 9AM CST Monday morning (March 24). Galaxy will be unavailable until then.

So, there are alternatives



Public Galaxy Servers and counting

Publicly Accessible Galaxy Servers

The Galaxy Project's public server ([UseGalaxy.org](#), *Main*) can meet many needs, but it is not suitable for everything (see [Choices](#) for why) and cannot possibly scale to meet the entire world's needs.

Fortunately the Galaxy [Community](#) is helping out by [installing Galaxy](#) at their institutions and then making those installations either publicly available or open to their organizations or community.

This page lists such public or semi-public Galaxy servers.

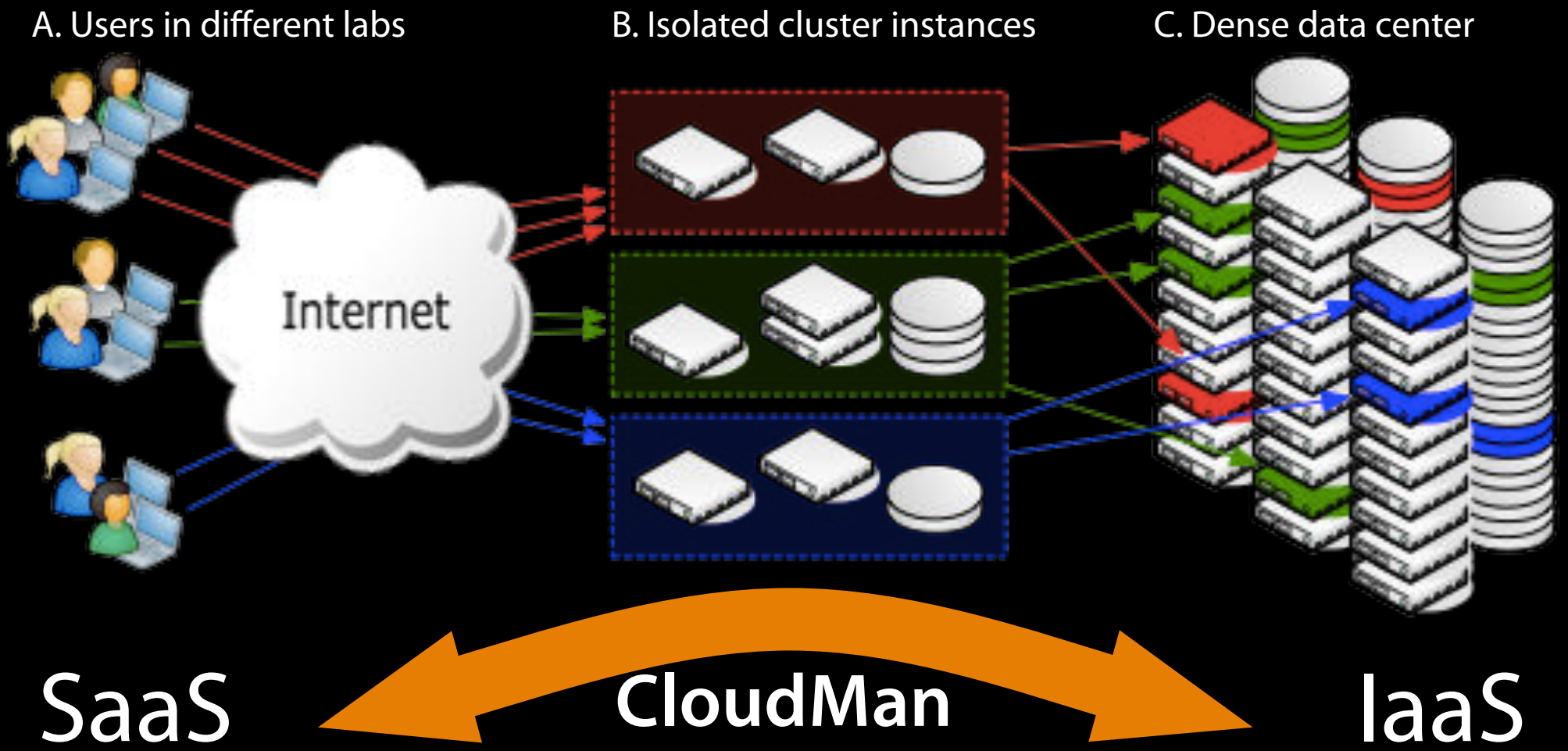
To add your public Galaxy server to this list, please either just add it (hey, *it's a wiki*), or contact Galaxy Outreach <outreach AT galaxyproject DOT org>

Contents

1. [General Purpose Servers](#)
 1. [Andromeda](#)
 2. [Biomina](#)
 3. [CBiB Galaxy](#)
 4. [DBCLS Galaxy](#)
 5. [Galaxy Main](#)
 6. [Galaxy Test](#)
 7. [GeneNetwork](#)
 8. [Genboree](#)
 9. [Genomics Virtual Lab](#)

**BUT WHAT IF YOU WANT YOUR OWN,
QUICKLY**

The big picture



What is CloudMan?

A **cloud manager** that orchestrates all of the steps required to provision, manage, and share a compute platform on a cloud infrastructure, all through a web browser.

More specifically...

CloudMan allows one to **create a compute cluster in the cloud**, use pre-configured applications, or add one's own. And then share it all.

Where is it used?

- Galaxy project
- Genomics Virtual Laboratory (GVL)
 - Endocrine Genomics Virtual Laboratory (endoVL)
 - Neuroimaging Virtual Lab
 - Human Communications Services VL
- NBIC's SARA cloud
- JCVI, MSI, Harvard Medical School

Deploying a CloudMan Platform

1. **An account** on the supported cloud
2. **Start a master instance** via BioCloudCentral.org or the cloud web console
3. **Use the CloudMan web interface** on the master instance to manage the platform

Starting an Instance

Launcher choices

usegalaxy.org/cloudlaunch

launch.genome.edu.au

biocloudcentral.org

Cloud dashboard / console

Script it via API

The image displays three overlapping screenshots of the BioCloudCentral web interface for launching a Galaxy instance. The top-most screenshot shows the 'Launch a Galaxy' page with a 'Start an Instance' button. The middle screenshot shows the 'Launch Instance' form with fields for Key ID (AKIAJIMDUA2LXX), Secret Key, Instance Name, Flavor (m1.small), Instance Count (1), Instance Boot Source (Boot from image), Image Name (Select Image), Cluster Name, Password, Instance Type (Large), and a 'Submit' button. The bottom screenshot shows the 'GVL Launcher' form with fields for Access key, Secret key, Cluster name (Create a new cluster or Recreate an existing cluster), Password, Instance type (Large (4 ECUs / 7.5GB RAM)), and a 'Start an Instance' button.

Configuring a cluster

← → ↻ 115.146.85.64/cloud# ☆ ☰

CloudMan from Galaxy [Admin](#) | [Report bugs](#) | [Wiki](#) | [Screencast](#)

Initial CloudMan Platform Configuration

Welcome to CloudMan. This application will allow you to manage this cluster platform and the services provided within. To get started, choose the type of platform you'd like to work with and provide the associated value, if any.

Galaxy Cluster: Galaxy application, available tools, reference datasets, SGE job manager, and a data volume. Specify the initial storage type:

Volume - Default (10 GB) Volume - Custom: GB

Transient Storage

Share-an-Instance Cluster: derive your cluster form someone else's cluster. *Note that this form field works only for instances that were shared after July 1, 2013! For instances shared before that date, please use [CloudLaunch](#) and provide the share string there.*
Specify the provided cluster share-string (for example, cm-0011923649e9271f17c4f83ba6846db0/shared/2013-07-01--21-00):

Cluster share-string

Data Cluster: a persistent data volume and SGE. Specify the initial storage size (in Gigabytes):

GB

Test Cluster: SGE only. No persistent storage is created.

[Hide extra options](#)

Manage Your Cluster

CloudMan Console

Welcome to [CloudMan](#). 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.

Terminate cluster

Add nodes ▼

Remove nodes ▼

Access Galaxy

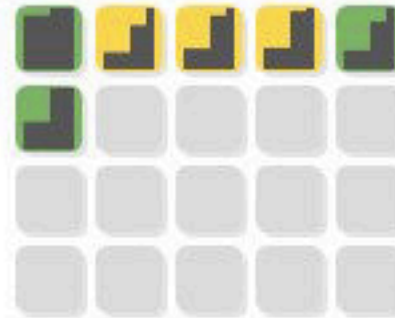
Status

Cluster name: ghem

Disk status: 0 / 0 (0%)

Worker status: Idle: 4 Available: 2 Requested: 5

Service status: Applications ● Data ●



Autoscaling is **off**.
Turn on?

Cluster status log



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](#)
- [NCBI BLAST+](#)
- [NGS: QC and manipulation](#)
- [NGS: Picard \(beta\)](#)
- [NGS: Mapping](#)
- [NGS: Indel Analysis](#)
- [NGS: RNA Analysis](#)
- [NGS: SAM Tools](#)
- [NGS: GATK Tools \(beta\)](#)

Welcome to Galaxy on the Cloud

managed by CloudMan

History



0 bytes

i Your history is empty. Click 'Get Data' on the left pane to start

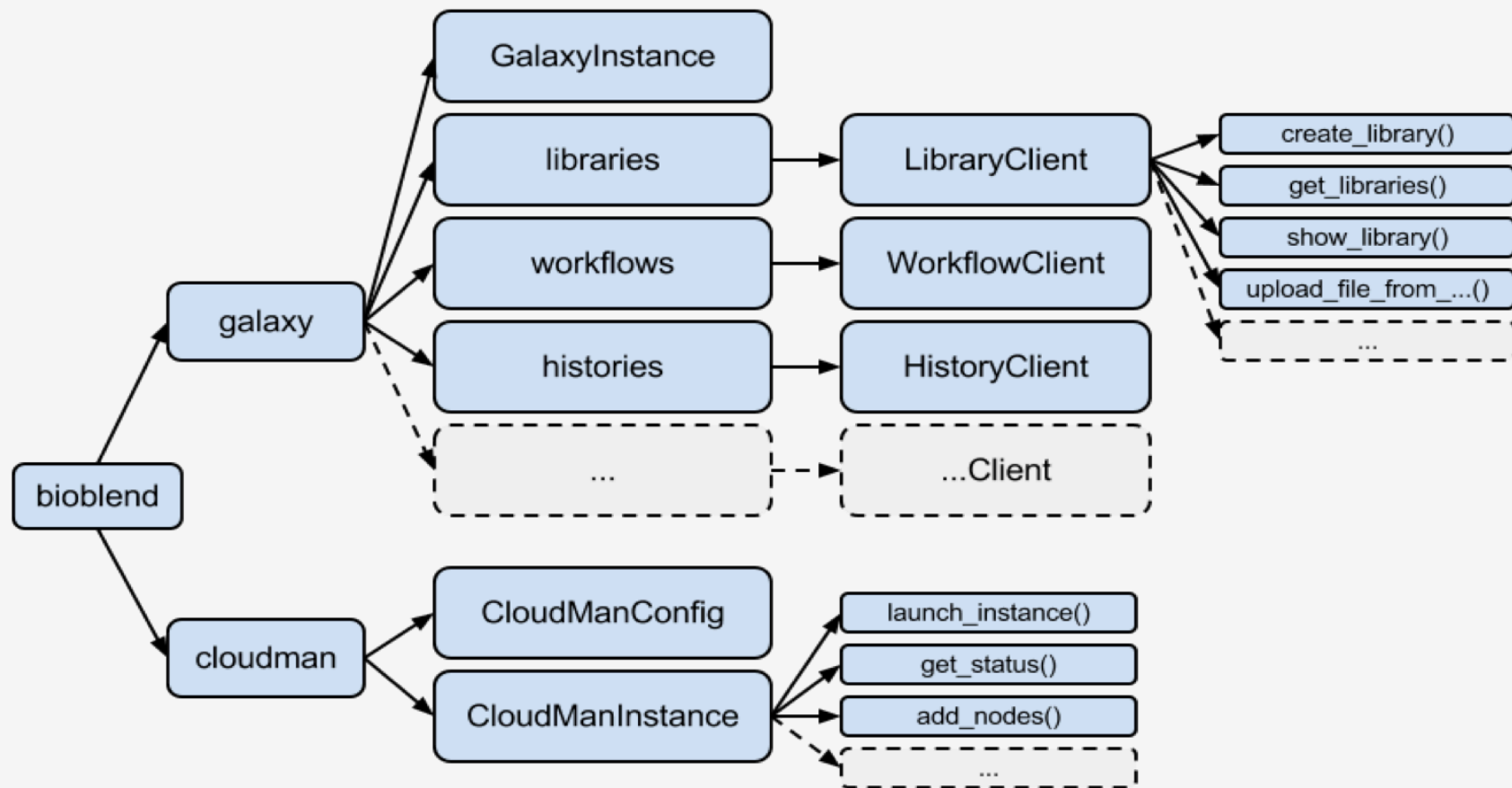
Remote desktop

- Use VNC server on the instance
- In-browser access via noVNC
 - Just point your browser to `<inst IP>:6080`
- Early days for the feature...

Beyond GUI

- API interface
- Goal is to enable creation of automated and scalable pipelines while hiding infrastructure details
- CloudMan as an infrastructure manager
 - Galaxy as a workflow execution engine

BioBlend is a Python library which wraps the Galaxy API and the CloudMan API



http://bioblend.readthedocs.org/

The screenshot shows the BioBlend documentation page. The left sidebar contains a 'Table Of Contents' section, which is highlighted with a blue box. The main content area includes a 'BioBlend' title, an 'About' section, and three bullet points with code examples for interacting with CloudMan and Galaxy.

bioblend.readthedocs.org/en/latest/

BioBlend 0.2.3-dev documentation »

Project Versions
latest

RTD Search
 Go
Full-text doc search.

Table Of Contents

- BioBlend
 - About
 - Installation
 - Usage
 - Development
 - API Documentation
 - CloudMan API
 - Galaxy API
 - Configuration
 - Testing
 - Getting help
 - Related documentation
 - Indices and tables

Next topic
API documentation for interacti

This Page
Show Source
Show on GitHub
Edit on GitHub

BioBlend

About

BioBlend is a Python (2.6 or higher) library for interacting with [CloudMan](#) and [Galaxy's API](#).

Conceptually, it makes it possible to script and automate the process of cloud infrastructure p via Galaxy. In reality, it makes it possible to do things like this:

- Create a CloudMan compute cluster, via an API and directly from your local machine:

```
from bioblend.cloudman import CloudManConfig
from bioblend.cloudman import CloudManInstance
cfg = CloudManConfig('<your cloud access key>', '<your cloud secret key>')
cmi = CloudManInstance.launch_instance(cfg)
cmi.get_status()
```

- Reconnect to an existing CloudMan instance and manipulate it:

```
from bioblend.cloudman import CloudManInstance
cmi = CloudManInstance("<instance IP>", "<password>")
cmi.add_nodes(3)
cluster_status = cmi.get_status()
cmi.remove_nodes(2)
```

- Interact with Galaxy via a straightforward API:

```
from bioblend.galaxy import GalaxyInstance
gi = GalaxyInstance('<Galaxy IP>', key='your API key')
libs = gi.libraries.get_libraries()
gi.workflows.show_workflow('workflow ID')
gi.workflows.run_workflow('workflow ID', input_dataset_map)
```

Beyond GUI #2

Command line access (with sudo access)

```
$ ssh ubuntu@<instance IP address>  
$ sudo -s
```

Run jobs on a cluster

```
$ qsub job_script_from_any_sge_cluster.sh  
$ qstat -f
```

Use Galaxy tools and reference genome data

```
$ cd ~/gvl_commandline_utilities  
$ sh run_all.sh  
$ module avail
```

CloudMan Platform Features

- A complete solution for instantiating, running and scaling cloud resources
 - Get a scalable **compute cluster**: SGE, Hadoop, HTCondor
 - Get an automatically **configured Galaxy application**
 - Scope of tools and reference datasets exceed Galaxy Main
- Deployment on **AWS, OpenStack, Eucalyptus, and OpenNebula** clouds
 - **Automated** configuration for machine image, tools, and data
 - **Wizard-guided startup**: requires no computational expertise, no infrastructure, no software
- **Self-contained** deployment
- Ability to **re-launch** clusters after periods of inactivity
- **Elastic resource scaling**: manual or automatic
 - On AWS, support for **Spot** instances
- **Dynamic persistent storage**
- **Use any S3 bucket** as a local file system
 - Use managed NFS / Gluster as a file system
 - Use archive URL to download arbitrary file system
- **Share** your instance: including all customizations (data, tools & configurations)
 - Easily replicate the EXACT environment

Value Added Features

Customizing, Sharing, Scaling

Customize Your Instance

- Each CloudMan **instance is self-contained**, meaning that it can be built upon
- If a tool is missing, simply install it
 - Readily integrates with the Tool Shed
- **Install your tool** and make it available
 - With all the configurations and sample data

Share Your Instance

- **Share entire (Galaxy) CloudMan platform**
 - Even the customized ones (including data and/or tools)
 - Fully automated solution
- Publish a self-contained analysis
 - Analyses in progress or complete



Instance sharing

CloudMan from Galaxy Admin | Report bugs | Wiki | Screenshot

Initial CloudMan Platform Configuration

Welcome to CloudMan. This application will allow you to manage this cluster platform and the services provided within. To get started, choose the type of platform you'd like to work with and provide the associated value, if any.

Galaxy Cluster: Galaxy application, available tools, reference datasets, SGE job manager, and a data volume. Specify the initial storage type:

Volume - Default (10 GB) Volume - Custom: GB

Transient Storage

Share-an-Instance Cluster: derive your cluster from someone else's cluster. *Note that this form field works only for instances that were shared after July 1, 2013! For instances shared before that date, please use [CloudLaunch](#) and provide the share string there.* Specify the provided cluster share-string (for example, cm-0011923649e9271f17c4f83ba6846db0/shared/2013-07-01--21-00):

Cluster share-string

Data Cluster: a persistent data volume and SGE. Specify the initial storage size (in Gigabytes):

GB

Test Cluster: SGE only. No persistent storage is created.

[Hide extra options](#)

Currently shared instances

Share-an-instance

This form allows you to share this cluster instance, at instance public or share it with specific users by providing the You may also share the instance with yourself by specifying y saving the instance at its current state.

While setting up an instance to be shared, all currentl Then, a snapshot of your data volume and a folder in your cl

Name	Instance ID
Exome sequencing pipeline	cm-b53c6f12

Public Shared

Specific user permissions:

Both fields must be provided for each of the users. These numbers can be obtained from the bottom of the [AWS Identifiers](#) section.

AWS account numbers:

AWS canonical user IDs:

Scaling the infrastructure with the computation

The screenshot displays the Galaxy Cloudman console interface. A modal dialog box titled "Autoscaling Configuration" is open, providing information about the autoscaling feature. The dialog text states: "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 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."

The background console shows the "Galaxy Cloudman Console" header and a "Status" section with the following details:

- Cluster name: share-an-instance demo
- Disk status: 84M / 10G (1%)
- Worker status: Idle: 0 Available: 0 Requested: 0
- Service status: Applications (green dot) Data (green dot)
- External Logs: [Galaxy Log](#)

Below the status section, there is a "Cluster status log" button. To the right, a grid of 15 worker nodes is visible, with a tooltip indicating "Autoscaling is on. Turn off? Min nodes: 0 Max nodes: 15 Adjust limits?".

Exercising elasticity with Auto-Scaling

Fixed cluster size

5 nodes

Computation time: 9 hrs

Computation cost: \$20

20 nodes

Computation time: 6 hrs

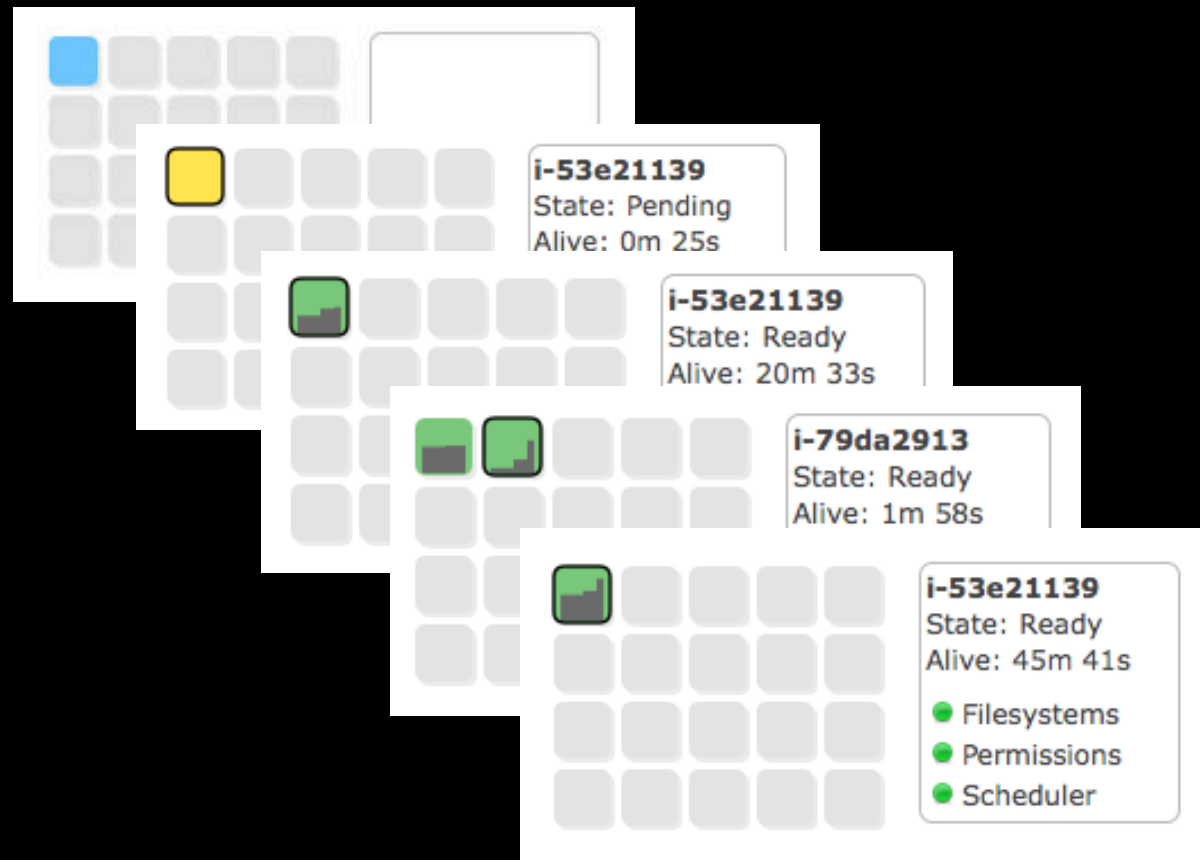
Computation cost: \$50

Dynamic cluster size

1 to 16 nodes

Computation time: 6 hrs

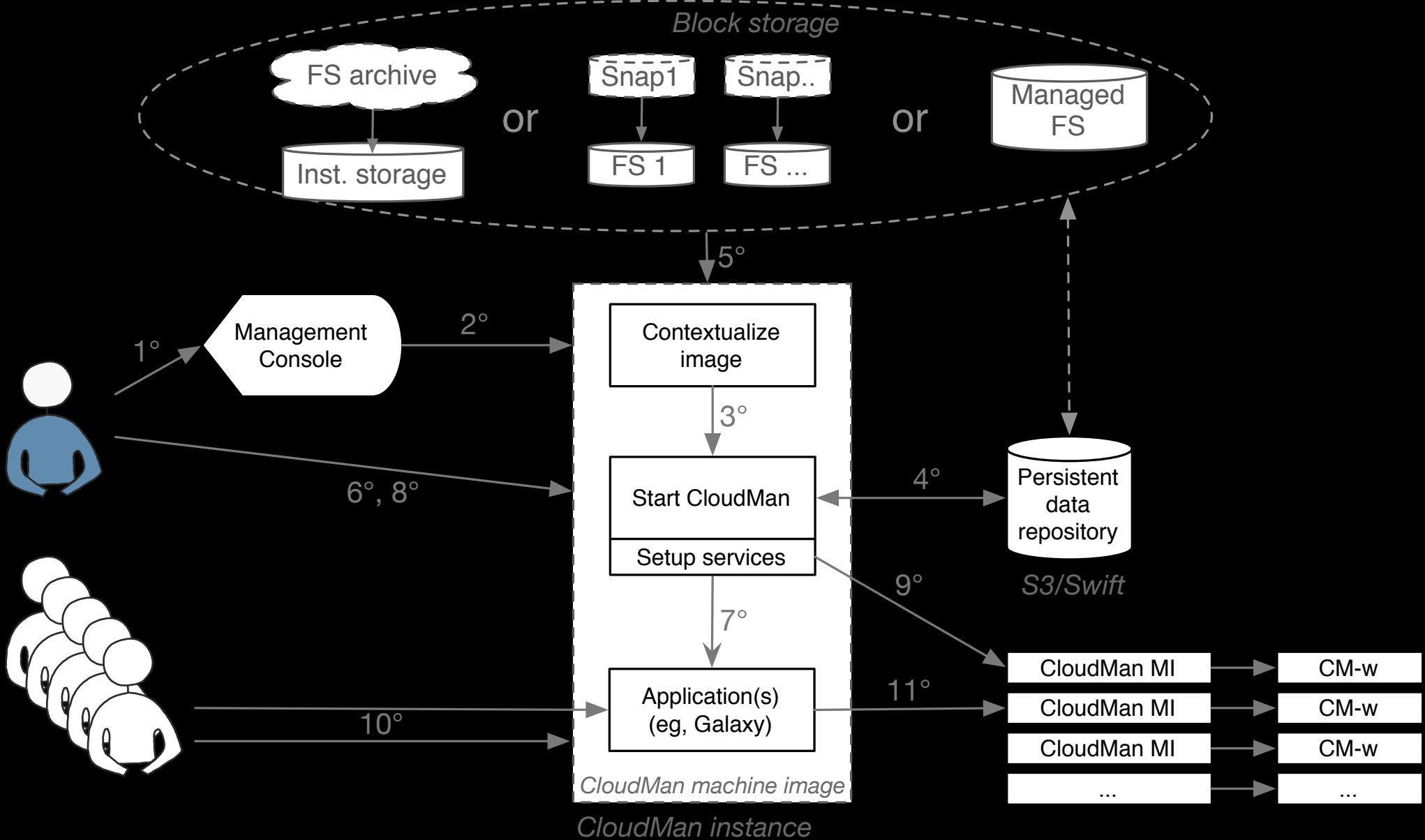
Computation cost: \$20



Underlying architecture

Deploying, coding, extending

System architecture



Data incarnations

Swift container

https://swift.rc.nectar.org.au:8888/v1/AUTH_377/cloudman-os/galaxyFS-2.13.tar.gz

↓
wget & untar

Volume
snapshots

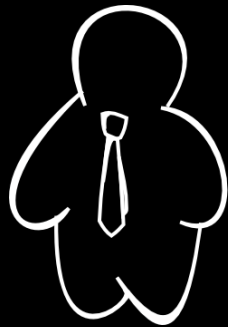
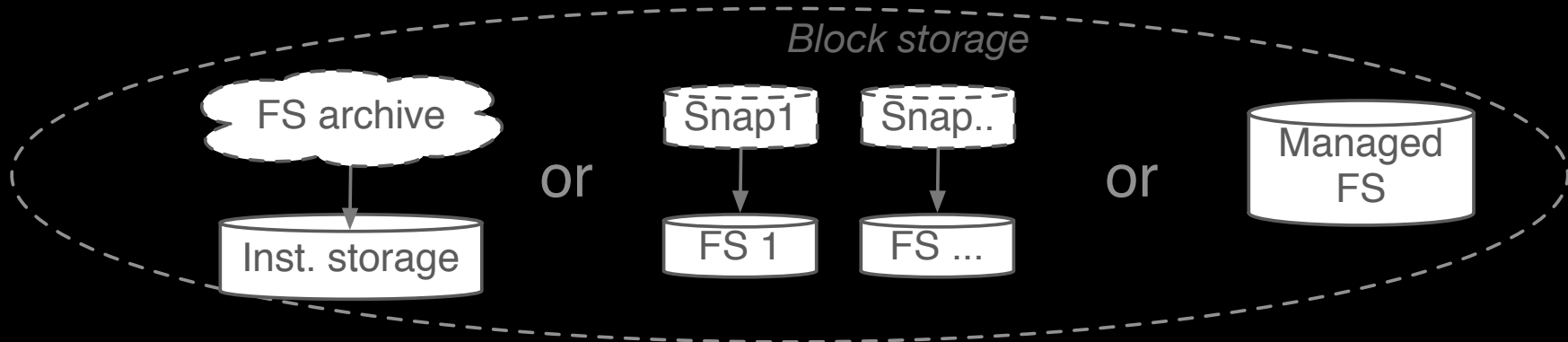
snap-a734hin1

↓
vol-82ojids2

Managed
Gluster
file system

115.146.85.193:/gvl-vol-replicated

↓
mount & nfs share



CloudMan

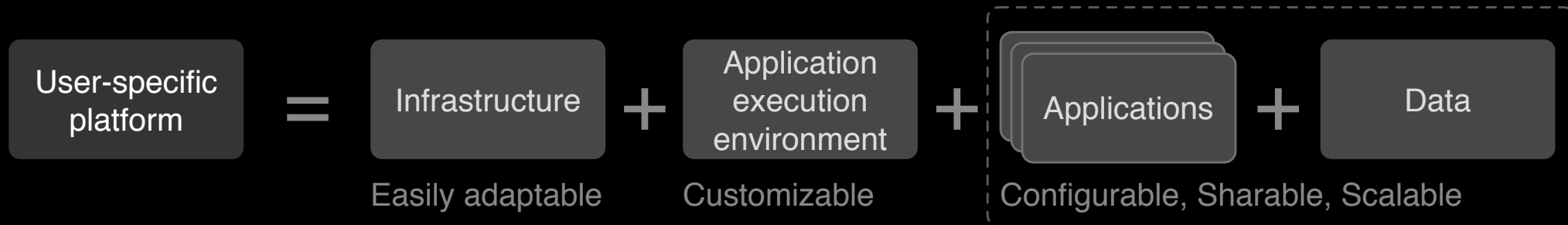
CloudMan infrastructure requirements

- Customizable machine image
- Support for instance *user data*
- Persistent object store
- Data volumes and (shareable) volume snapshots
- Resource metadata (ie, tags)

Building the components

- Leverage CloudBioLinux build framework
- A number of **flavors** exist
 - Core CloudMan image
 - Base Galaxy image
 - Full CloudBioLinux image
- Semi-automated process
 - See Ron Horst's talk tomorrow

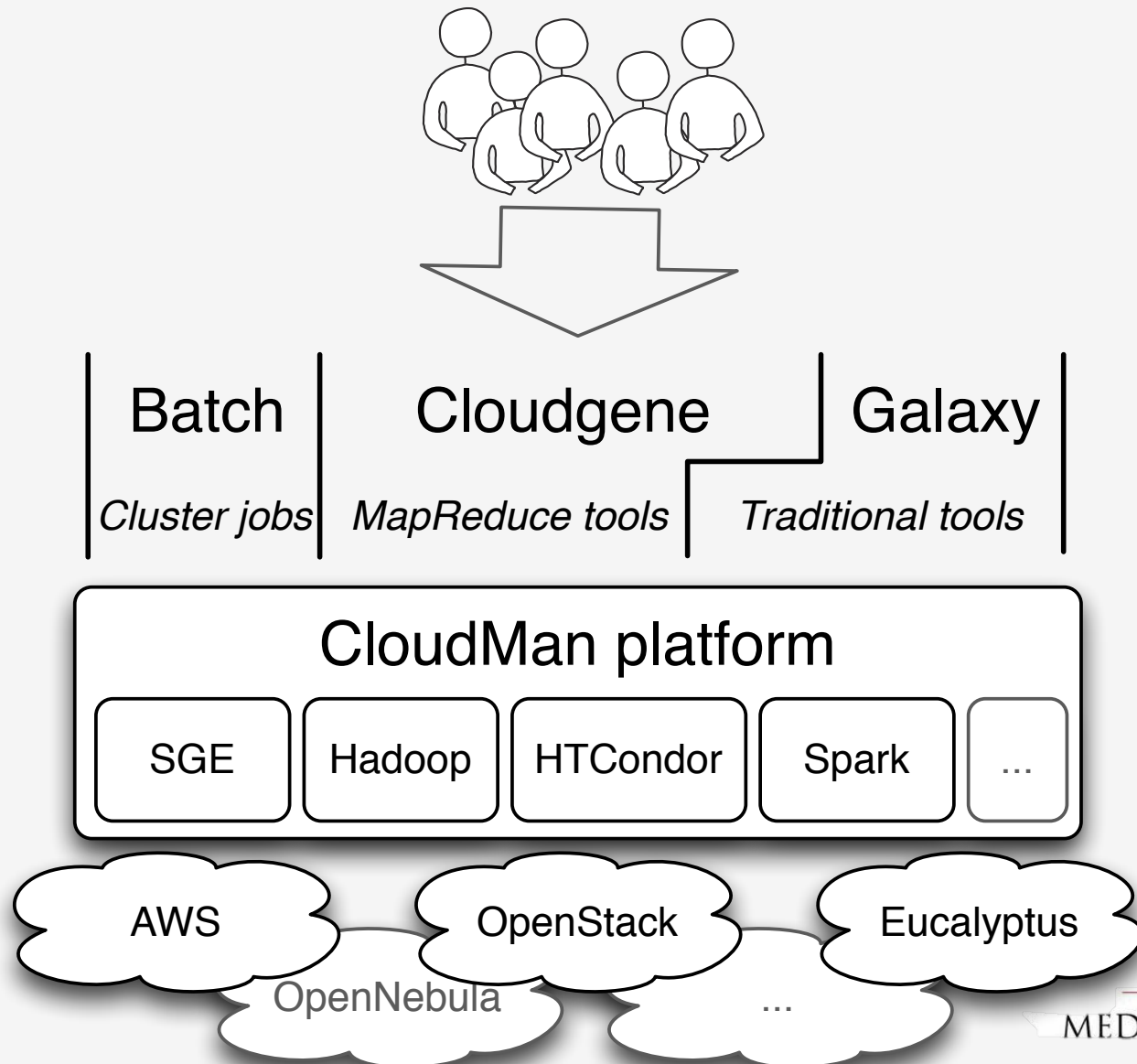
CloudMan-as-a-Platform



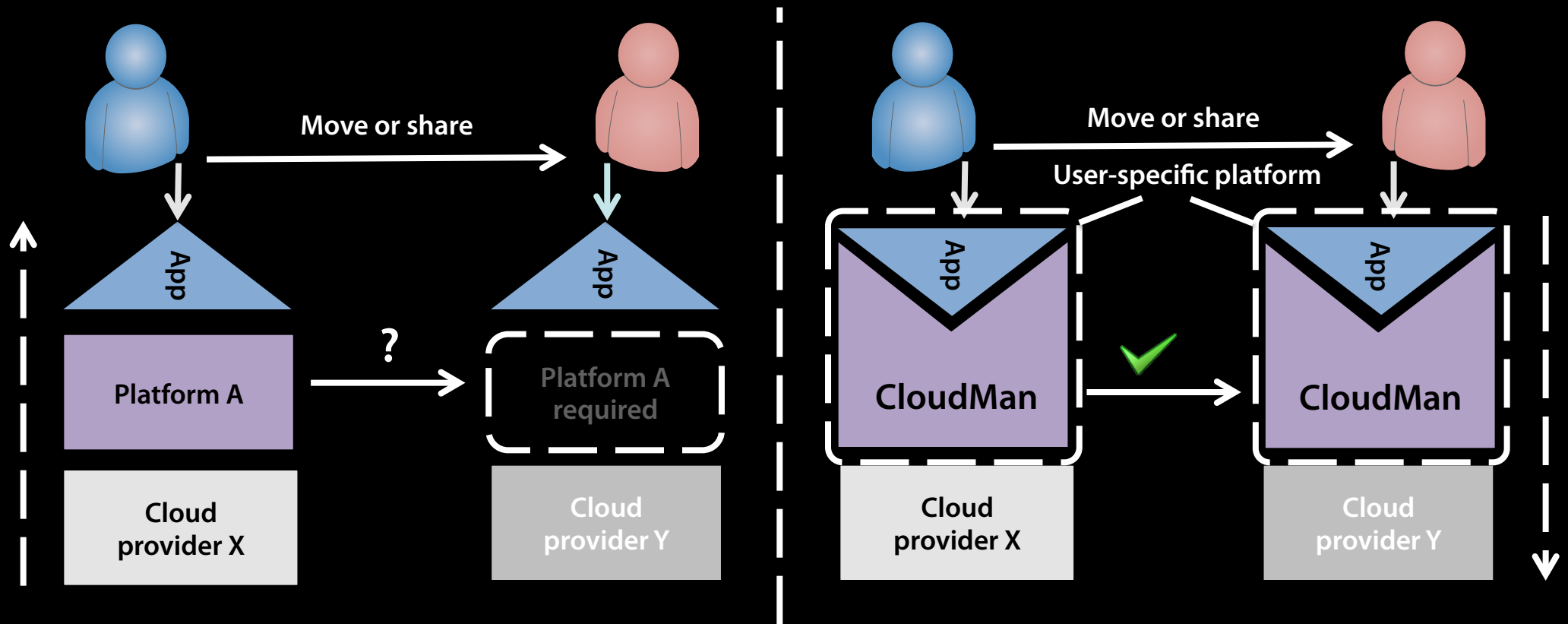
Enable easy creation of **user-specific cloud platforms**

Couple the infrastructure, functional application execution environments, applications, and data into a single unit that can easily be used and manipulated by a user.

Extending the platform



Packaged platform enables reproducibility



Galaxy CloudMan: delivering cloud compute clusters

Enis Afgan¹, Dannon Baker¹, Nate Coraor², Brad Chapman³, Anton Nekrutenko², James Taylor^{3*}

From The 11th Annual Bioinformatics Open Source Conference (BOSC) 2010 Boston, MA, USA. 9-10 July 2010

Abstract
Background: Widespread adoption of high-throughput sequencing has greatly increased the scale and sophistication of computational infrastructure needed to perform genomic research. An alternative to building and maintaining local infrastructure is "cloud computing", which, in principle, offers on demand access to flexible computational infrastructure. However, cloud computing resources are not yet suitable for immediate "as is" use by experimental biologists.

- Project start as part of Galaxy
- Initial version available on AWS

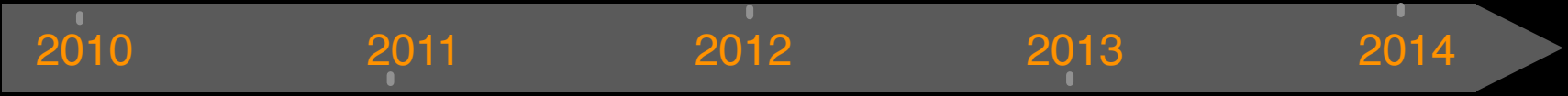
CloudMan as a platform for tool, data, and analysis distribution

Enis Afgan^{1,3,4*}, Brad Chapman² and James Taylor^{3,4}

Abstract
Background: Cloud computing provides an infrastructure that facilitates large scale computational analysis in a scalable, democratized fashion. However, in this context it is difficult to ensure sharing of an analysis environment and associated data in a scalable and precisely reproducible way.

- Support for instance sharing
- AWS spot instance support
- Use AWS S3 as a local file system

- Versioned deployments
- Galaxy & object store integration
- Support for cloud bursting



- Expansion of available tools
- Deployment automation
- Infrastructure scaling: compute and storage

Harnessing cloud computing with Galaxy Cloud

To the Editor:
Continuing evolution of DNA sequencing has transformed modern biology. Lower sequencing costs coupled with novel sequencing-based assays have led to rapid adoption of next-generation sequencing across diverse areas of life sciences research¹⁻⁴. Sequencing has moved out of the genome centers into core facilities and individual laboratories where any investigator can access it for modest and progressively declining cost. Although easy to generate in tremendous quantities, sequence data are still difficult to manage and analyze. Sophisticated informatics techniques and supporting infrastructure are needed to make sense of even conceptually simple sequencing experiments, let alone the more complex analysis techniques being developed. The most pressing challenge facing the sequencing community today is providing the informatics infrastructure and accessible analysis methods needed to make it possible for all investigators to realize the power of high-throughput sequencing to advance their research.

been demonstrated⁵⁻⁸. In general, however, cloud resources are not provided in a form that can be immediately used by a researcher without informatics expertise. Several commercial vendors provide cloud-based sequence analysis services through the web that hide all complexity of the underlying infrastructure. Yet these contain limited sets of analysis tools, and because they are proprietary solutions, users must give up some control over their own data and risk becoming dependent on a single commercial service for continued data access and analysis. All "battle-tested" next-generation sequencing analysis practices (e.g., analysis of human variation exemplified by the 1000 Genome Consortium publication⁹) are open source. One popular open-source platform that has made substantial progress toward making complex analysis available to researchers is Galaxy^{10,11}. Galaxy enables users to perform analysis using nothing more than a web browser. The environment automatically and transparently tracks every detail of the analysis, allows the construction of complex workflows and

- Multi-cloud support: OpenStack, OpenNebula, Eucalyptus
- Support for Hadoop jobs, HTCondor
- API

Support for data-intensive computing with CloudMan

Y. Kowsar¹ and E. Afgan^{1,2}

¹ Victorian Life Sciences Computation Initiative (VLSI), University of Melbourne, Melbourne, Australia
² Centre for Informatics and Computing (CIR), Ruder Bošković Institute (RBI), Zagreb, Croatia
ykowsar@student.unimelb.edu.au, enis.afgan@unimelb.edu.au, irb.hr

Abstract - Infrastructure-as-a-Service (IaaS) compute infrastructure model has showcased its ability to transform how access to compute resources is realized; it delivered on the notion of Infrastructure-as-Code and enabled a new wave of compute adaptability. However, many workloads still execute only in a more structured and traditional cluster computing environment where jobs are handed off to a job manager and possibly executed in parallel. We have been developing CloudMan (usecloudman.org) as a versatile solution for enabling and managing compute clusters in cloud environments via a simple web interface or an API. In this paper, we describe a recent extension of CloudMan to add support for data intensive workloads by incorporating Hadoop and HTCondor job managers and thus complement the previously available Sun Grid Engine (SGE).

the data collection and step into the world of data analysis. As a step in this direction, we have been developing a cloud resource manager called CloudMan [2] that facilitates creation of a compute platform [3] capable of handling a range of workloads, including biological analyses [4]. Cloud computing in general allows compute and storage resources to be requested, provisioned, and utilized to handle the necessary scaling of a computational problem at hand. However, those resources are often provisioned as bare virtual machines and disks without application context or coordination. CloudMan helps in this regard by orchestrating all the steps required to provision a functional compute and application environment based on the flexible cloud resources and

* planned

Acknowledgments

