

GAL

Genome Annotator *Light*

Version 1.1

User Guide

Authors:

Arijit Panda
Narendrakumar M. Chaudhari
Sucheta Tripathy*

Contact Email: arijpanda@csiriicb.res.in/ tsucheta@gmail.com

Developed at:

Computational Genomics Lab,
Structural Biology and Bioinformatics Division,
CSIR-Indian Institute of Chemical Biology,
Kolkata, India.

*Principal Investigator

Table of Contents

Introduction	1
Getting Started	1
System Requirements.....	1
Quick Start	2
Additional useful Commands	3
GAL User Interface (GUI).....	5
GAL Homepage	5
GAL Data Upload Options	6
GAL Sample Data.....	8
GAL Genome Browser	9
Gene Sequence Page.....	12
BLAST Page	13
BAM Alignment Upload.....	14
GAL Command Line Options	15
How to run GAL in command line mode?.....	15
Accessing host directory	15
Log File	15
Running the programs.....	15
Setting up the configuration file	15
Data Format	17
Sample organism data upload using command line mode:.....	17
List of Reference Genomes.....	18
For AUGUSTUS Annotations:	18
For GeneMark Annotations:	19

Introduction

GAL is a software package for analyzing and visualizing a genome or a group of genomes. GAL is implemented inside Docker. Docker technology is becoming popular throughout the bioinformatics community due to its features, ease with dependencies and more efficient usage of the underlying system and resources. Docker allows deploying an application in a sandbox (called container) to run on the host operating system locally. Docker needs to be installed on the host system (Linux in this case) to proceed with GAL.

Getting Started

GAL can be installed and initiated through Docker. Docker is available in two editions: Community Edition (CE) and Enterprise Edition (EE). Docker CE and EE are available on multiple platforms, on cloud and on-premises.

- Docker website: <https://www.docker.com/>
- Docker Documentation for beginners: <https://docker-curriculum.com/>
- Docker CE and EE are available at:
<https://docs.docker.com/engine/installation/#supported-platforms>

[If your installation is successful, upon doing a 'docker run hello-world' (or if permission issues are there run a 'sudo docker run hello-world' you will get a message on your prompt as: Hello from Docker]

System Requirements

GAL can be installed on the following operating systems:

- CentOS 7.1/7.2 & RHEL 7.0/7.1/7.2/7.3 (YUM-based systems)
- Ubuntu 16.04 LTS or higher

Quick Start

1. GAL can be downloaded and installed using following Docker command:

```
docker pull cglabiicb/gal
```

This will fetch the latest version with 'latest' tag.

For specific version, use the version number for example:

```
docker pull cglabiicb/gal:1.1
```

Depending on the network speed, the entire package gets downloaded and installed around 8 minutes (for 100 mbps network speed).

2. To run GAL use the following command:

```
docker run -it -p 8080:80 cglabiicb/gal
```

This will initiate GAL at port 8080 of local server or localhost. User may use another port to initiate another instance.

[To manipulate Docker utilities refer to [Docker Documentation](#)]

3. While the GAL instance is running inside Docker container, GAL User Interface (UI) can be accessed through a web browser at following URL:

<http://localhost:port/>

In this case, it is

<http://localhost:8080/>

It can also be:

[http:// <IP address of the host computer>:8080](http://<IP address of the host computer>:8080)

4. GAL can now be used to upload your data through the browser.

Additional useful Commands

List docker images

To find the pulled docker images in the system user can use the following commands:

```
docker images
```

This will list images as follows,

REPOSITORY	TAG	IMAGE ID	CREATED	VIRTUAL SIZE
cglabiicb/gal	1.0	cc7be8e0f7d9	2 hours ago	5.7 GB
cglabiicb/gal	latest	cc7be8e0f7d9	2 hours ago	5.7 GB
hello-world	latest	95f1e6dc264a	23 months ago	1.848 kB

Set instance name

Docker by default allocates a random name and id for the running instance. User can change the instance name by adding '--name' option in the command line. It will help the user to track an instance later.

Example:

```
docker run --name=test -it -p 8080:80 cglabiicb/gal
```

Here 'test' is the name of the running instance.

Find docker instances

To find all the available Docker instances use the following commands

```
docker ps -a
```

This is the output example of the above command.

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
969ab10373bc	cglabiicb/gal:0.2	"/bin/sh -c 'service "	26 hours ago	Up 26 hours	0.0.0.0:8080->80/tcp	hopeful_visvesvaraya
476d22340d5f	cglabiicb/gal:0.3	"/bin/sh -c 'service "	47 hours ago	Up 47 hours	0.0.0.0:7070->80/tcp	mad_pare
a5efe47e6bdb	cglabiicb/gal:0.2	"/bin/sh -c 'service "	2 days ago	Exited 2 days ago		trusting_curie

Exit docker instance

To exit from a running Docker instance, use `exit` command.

To exit from Docker command line, use **CTRL+p** followed by **CTRL+q**

Re-enter running instance

To re-enter into a running instance, use the following command

```
docker exec -it <Container_id/Name> bash
```

Example :

```
docker exec -it test bash
```

Here '**test**' is the name of the running instance.

Restart Docker instance

To start the stooped instances, use the following command:

```
docker start -i <Container Id/Name>
```

Example:

```
docker start -i test
```

Here '**test**' is the name of the running instance.

Successful GAL Start

On successful docker GAL instance start, the following message will appear.

A terminal window with a black background and green text. It shows two lines of output: '* Starting MySQL database server mysqld' and '* Starting Apache httpd web server apache2'. To the right of the second line is '[OK]'. There is a third line with a single asterisk '*'.

[OK] indicates successful initiation.

Delete Docker Image

To delete any Docker Image, use the following command:

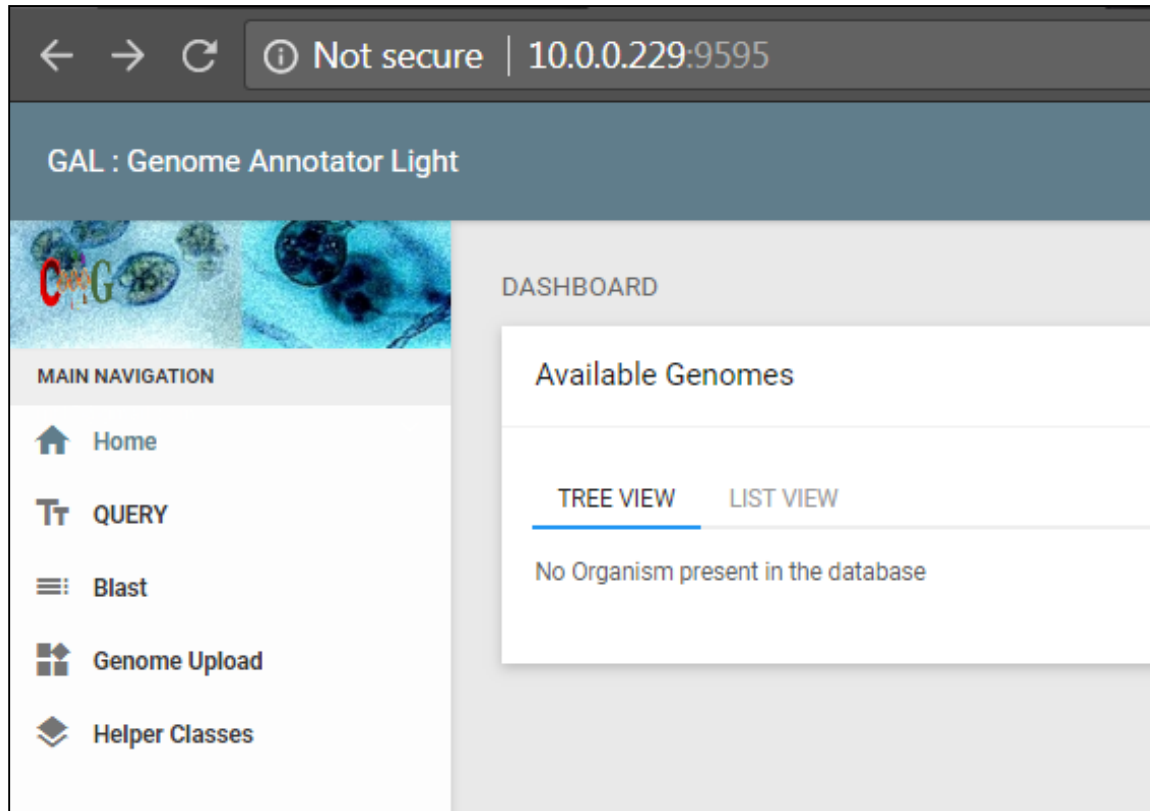
```
docker rmi <image id/name>
```

For force deleting of image, add `--force` option with the command.

GAL User Interface (GUI)

GAL GUI is must for data visualization, and it includes several web pages like,

GAL Homepage



- GUI for GAL can be loaded inside a web browser for Genome Upload, Genome Browsing; downstream analyses like Blast Searches, Annotation Query and Sequence Retrieval along with analyses of all the annotated proteins using various EMBOSS tools.
- The Homepage will list the genomes only after they are processed. Until then there will be no data available in the list view or tree view.

It approximately took 28 minutes to process ~5 Mb *E.coli* genome for Genbank Annotation as input on standard Ubuntu Desktop having 4 CPUs and 4 Gb of RAM. The same genome at various annotation levels took proportionate time. E.g. Product Annotation (31 minutes), Minimal Annotation (30 minutes), and No Annotation (175 minutes using GeneMark annotator + NCBI BLAST).

GAL: An Integrated Virtual Machine for Genome Analysis and Visualization

- The Navigation panel to the left will help the user to access various features like:
 - **Genome Upload:** Upload options at any stage of the annotation process.
 - **QUERY:** Gene search using gene name, primary annotation, genomic locus or HMMPFAM/Signalp/tmhmm annotations.
 - **BLAST:** Sequence search using NCBI BLAST for protein or gene sequence within the uploaded dataset.
 - **Help:** Help and documentation.

GAL Data Upload Options

Genome Upload

[Genbank](#) [Product Annotation](#) [Minimal Annotation](#) [No Annotation](#) [Upload Status](#)


Organism Name

Organism Version

Genbank File

No file selected.

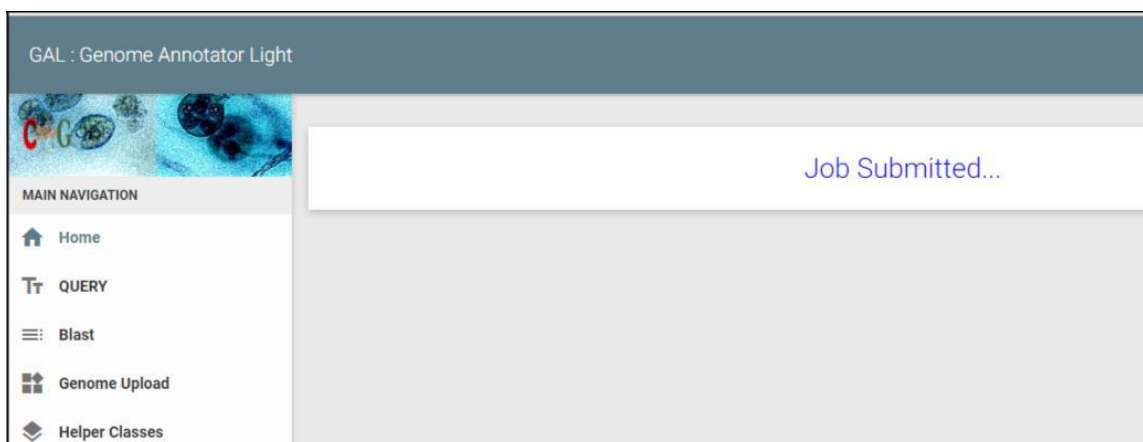
Please provide genbank file of your genome

[\[Different Image \]](#)

The user can provide data in four ways, viz. type1: Genbank Annotation, type2: Only Genome Fasta files, type3: Genome fasta and gff files; type 4: Genome Fasta, gff files and product files

- **Genbank Annotation:** This allows data input through NCBI annotated Genbank file (GBFF).
- **Product Annotation:** This allows genome FASTA, GFF (genome feature file) and product information file.
- **Minimal Annotation:** This allows the basic annotation information provided by the user where user provides genome FASTA (FNA) file and GFF file.
- **No Annotation:** This allows data through only genome FASTA (FNA) file with annotation options using AUGUSTUS or Genmark for eukaryotic and prokaryotic genomes using related reference genomes, respectively.

Once data files are uploaded, you will see the screen changes to:




You can also check the status of submitted job from **Upload status** tab of upload page, which looks like follows:

GAL: An Integrated Virtual Machine for Genome Analysis and Visualization

Genbank Product Annotation Minimal Annotation No Annotation <u>Upload Status</u>					
Show 10 entries			Refresh		Search:
Organism name	Version	Time	Status	Stage	
Halomicronema excentricum str. Lakshadweep	1	Submission: 2019-02-26 07:13:39 Start: 2019-02-26 07:13:40 End: 2019-02-26 07:31:56	Complete	Functional Annotation	
Hassallia byssoidea VB512170	1	Submission: 2019-02-26 07:33:24 Start: 2019-02-26 07:33:24 End: 2019-02-26 08:10:58	Complete	Comparative Genomics	
Lyngbya confervoides BDU141951	1	Submission: 2019-02-26 08:12:21 Start: 2019-02-26 08:12:21 End: 2019-02-26 08:35:05	Complete	Comparative Genomics	
Mastigocladus laminosus UU774	1	Submission: 2019-02-26 08:36:15 Start: 2019-02-26 08:36:15 End:	Running	Functional Annotation	
Organism name	Version	Time	Status	Stage	

GAL Sample Data

GAL : Genome Annotator Light



MAIN NAVIGATION

- Home
- QUERY
- Blast
- Genome Upload
- Helper Classes

DASHBOARD

Available Genomes

TREE VIEW LIST VIEW

- Chroococcales
 - Microcystis
 - Microcystis aeruginosa NIES-843 (V1)
 - Microcystis aeruginosa NIES-2549 (V1)
 - Microcystis aeruginosa NIES-2481 (V1)
 - Microcystis aeruginosa PCC 9432 (V1)
 - Microcystis aeruginosa PCC 9717 (V1)
 - Microcystis aeruginosa PCC 9443 (V1)
 - Microcystis aeruginosa PCC 7941 (V1)
 - Microcystis aeruginosa PCC 9807 (V1)
 - Microcystis aeruginosa PCC 9808 (V1)
 - Gloeocapsa
- Nostocales
- Oscillatoriales
- Synechococcales

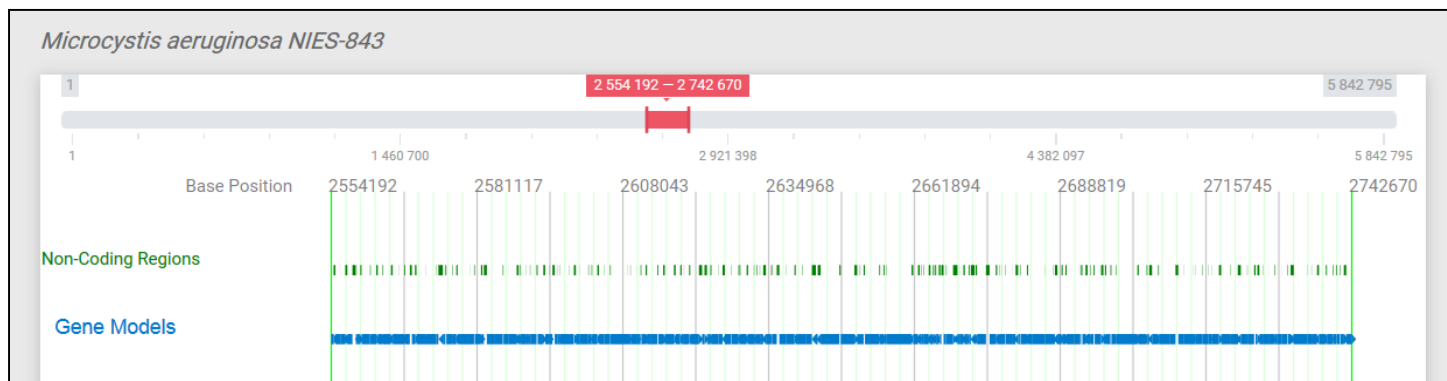
Clicking the genome name will direct the browser to Genome Summary Page for respective organism where organism details and links to the Scaffold-wise Genome browser links are provided.

From genome browser, each coding and non-coding regions can be visualized in details with exon-intron boundaries along with sequence download links and analysis options.

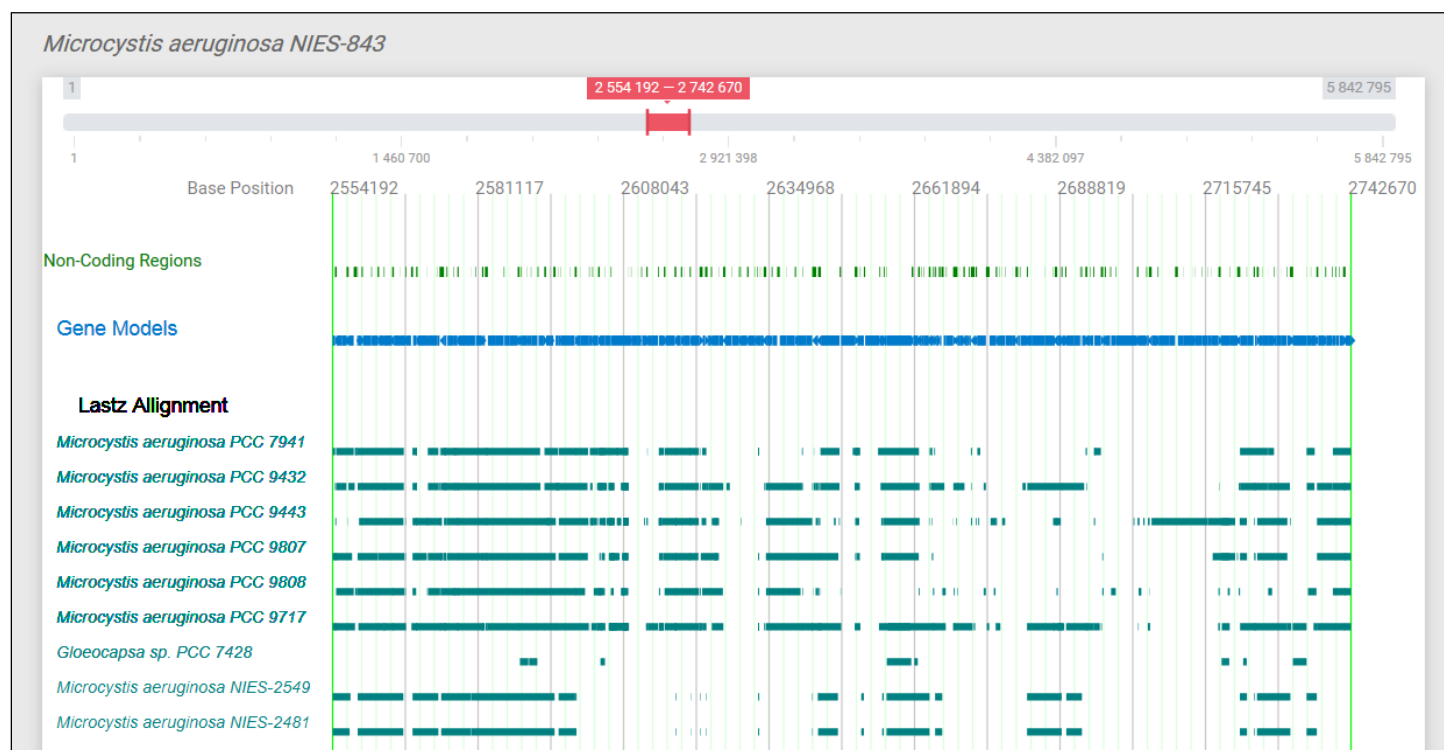
GAL Genome Browser

GAL Genome browser can visualize **coding** and **non-coding** regions in selected locus range of selected genome, as shown in the following image.

SINGLE GENOME BROWSER MODE



MULTI GENOME BROWSER MODE



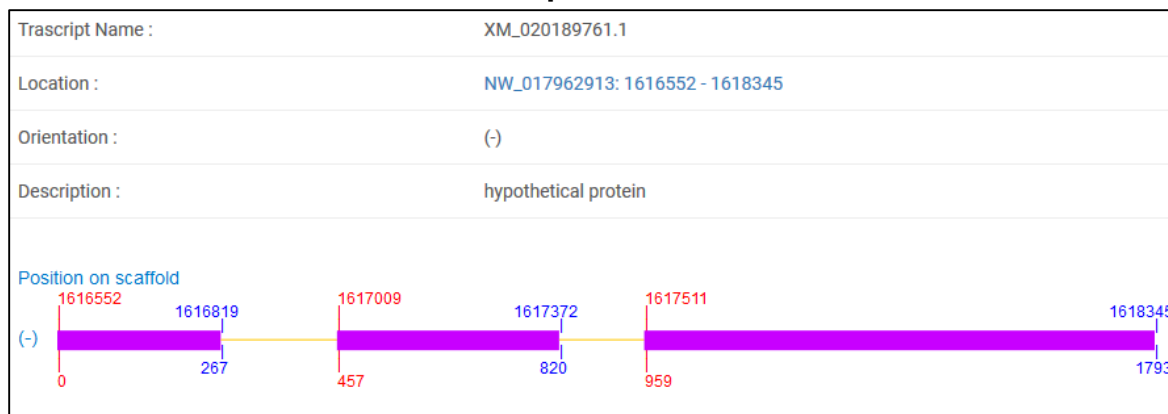
Additionally, GAL can automatically visualize respective regions from multiple taxonomically related species (if present in given dataset) based on LastZ Alignments.

Each highlighted region links to the individual gene details page with annotation details, gene analysis options and sequence download options.

Gene Details Page

All the annotated genes, transcript or proteins can be analyzed separately into Gene details page,

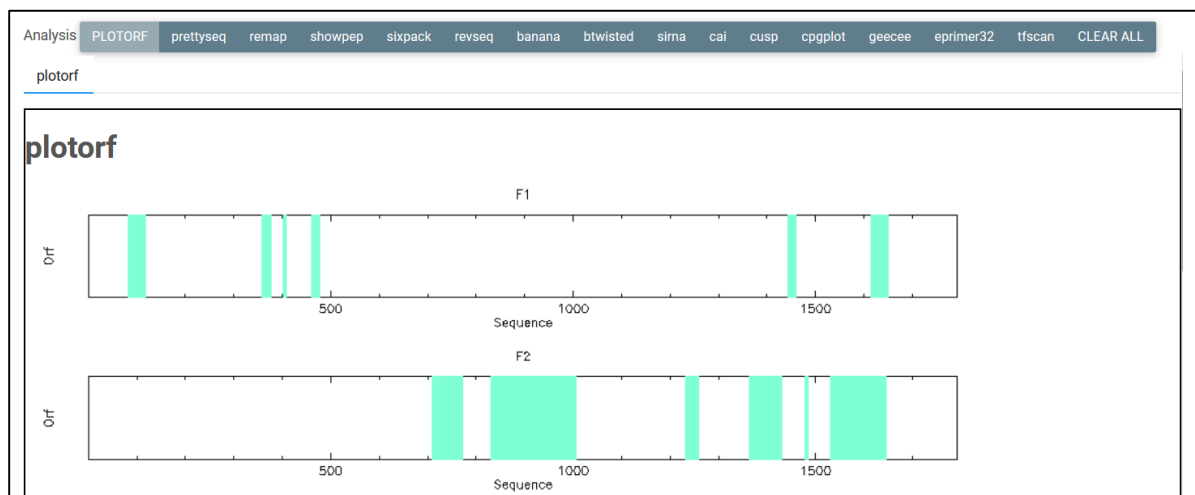
Exon Intron Boundaries for transcripts:



Annotation summary tables for various methods are also displayed on the same page for more details.

EMBOSS TOOLKIT

The protein analysis supported by various EMBOSS tools is available at each gene details page. The outputs can be visualized on the same page by just clicking the name of the package. All the outputs can be downloaded as image or text format wherever suitable.



The above screenshot shows various EMBOSS tools incorporated into the GAL analysis. The example output for the given transcript by plotorf tool is shown here. All the adjacent tabs with the name of these tools can generate the

standard outputs. These tools include **banana**, **cpgplot**, **epimer32**, **sixpack**, **showpep**, **tfscan** etc.

Gene Sequence Page

The gene sequence page provides the option for retrieving nucleotide sequences of the genomic region as well as protein sequence of the translated gene. The green highlighted sequence indicates the exons for easy understanding and reporting.

Predicted Gene Model(green marked regions are exons, white marked are introns)

Length: 1793 bases

```
ccctgctttgttggcgaccactttaccggagggtctacttggattagtagttgttcaggggtcagggtcgatccagttgtcgaatgattctgatccggagaaggttgttagga
gttagttgagttaacacgagctgcatcagataaggctgctgttaagtaggtagttgttcttgttgaatcataaggataaccgctggttagctggtcaataaactag
gtaattgacttatgaggggacttagttgactattttaaatgttggcggtttgttggtagaggaaaaatggttttaagttcgaatagagttataagttactaagtttag
attataggagttaaacggtccttgatgattgacagggtttgttgaagtgtagttaaccagttgaaagatgtcttgaacagtaaggtagattccttagtttcagaaccttg
taccgtagctatttaatgggtatagttttgatttaagttaaaaggagggttgtagctgcttcattgttcttagttgacatagaaggatctgcgggatttttagccgagttg
gaataccgaggaaaacttttagtatttccggacaccgtggagttaggggataaaaaatatttcaatgtactctgtgcttcgacgttgttctttaaattactggttgatagga
ggaaagatcgttgaaggtagtggttaaggactttcctaactatggaggataaaagtgttttcttctaactttgttagtaggtccaagaaattgaactgatttgatt
ttgattaagatttatctgtagttgcttttttagggtaaaattgtggaaatggtaagttgttgcgttgagggttgttgttagagttgttggagttgttag
agttgttggagttgttggagttgttagagttgttggagttgttggagttattgtagaataatctttatgtttagtttagtttcagttgttggatattttgagagattact
acttttttaattatagttttatgatgttgagtagttaggtttgttatctgagtaagccattgctatttggattttggaatactttcgtctcagaagtggagttgacctttt
agaattgactgtatcgtttaccaaccggttagctggaagttatgccaaaccatttgccttctcagtaggtaaaagctaacgcttttgaattacagacatctattatact
ttctcttttaagttagtagaaatcgtagtgaagataaggatttttaatatctgctttttagatgattggttttagttcctttttgtttcttttagttcctttgattcta
gttccagtttcttaaatgttaggtgagtcctatgattatttggtagtgccttactattactattattactattacttttgggtccttttagttaaagtaaaactag
atgagttactacttctaatactactgttataggtatgggtagttatagctatagaattacttaagggttagttactttcaggttaatgtttatagggtttcgattttc
aaagaccaaccggctataaagttacaaaaggagggttggaattgtctaccacagaaaaatgtcaagaaacaatgacctttaccttagtttaacctaggttagttggtc
tcgattgggttgtagtactcttttatatttgattcgtaggtttatttaatttactaccgataattttagtatcttaattagttatttgcattttaagatttgtttcgagag
ggaaagtagaaaagttt
```

Predicted protein sequence

Length: 489 amino acids

```
WDETTAGEMASQMNLINKSPVQLGQQLRLGLFQQSSINSIVLDVVYSDDNSSIKQNNKLVFLFGDQLDQLFDPLTEYSPESD KIYKPPNKP LSFYQNSRLISIFNDSN
LISSICQELLTVQNTFTINLVNLFQNFVIPLRIKVLHGIKLPISKLNIFPPTIDEVTRINCFDALKSAQPYGSFEI IKACGTSIPYFYKAYMRHEAATRNFDQLS
SFLDNFHHQIPERIDTSYFTKRRIETIIHGSNLTKLKLILNRLINEKISHLNTFTINNHNKSLMMKKLISKYNNSSIQTIDSGNDKLPYESRVFTPTGKILTELANGW
PIDLQYGVNRRVISIFDCENLMSVDNMKDEITIIIFSDHILFLKIIDENYYNQIKKKQRKSRKL RSSPITNIPKLKVS GWADISNVFPSTYNDGVFLQFFVTGN GIKLDPN
QPELTQHMRKYKLSDPNKLNDGYKIIELINKAKILNKSPFHLFK
```

BLAST Page

As the genomes are available in the database after processing the genomes uploaded by the user, any nucleotide or protein sequences can be BLASTed against the available genomes. The selection of any of the genomes or all the is possible from the checkboxes near organism names. The genomes are shown as tree view for the blast options.

Local Blast

Copy and Paste your sequence

```

>sequence
ttttgagagattactacttttttaattatagttttatgatgttgagtagttaggt
ttgttatctgagtaagccattgctatttgattttggaatactttcgtctcagaa
gtgaggttgacctttttagaattgacttgatcggtttaccaaccggttagctg
gaagttatgccaaaccatttgccttctcagtataggtaaaagctaacgcttt
tqaattacaqacatctattatactttctactttaatqgtaqtaqaaatcqcta

```

Select Blast Program NCBI-BLASTN

Select Database

- ☐ Enterobacterales
 - ☐ Escherichia
- ☒ Saccharomycetales
 - ☒ Ascoidea
 - ☒ Ascoidea rubescens DSM 1968 (V1)
 - ☐ Ascoidea rubescens DSM 1968 (V2)
 - ☐ Ascoidea rubescens DSM 1968 (V3)

Evalue(E):

Cutoff Value(S):

Substitution matrix BLOSUM62

Maximum Alignments(B): 10

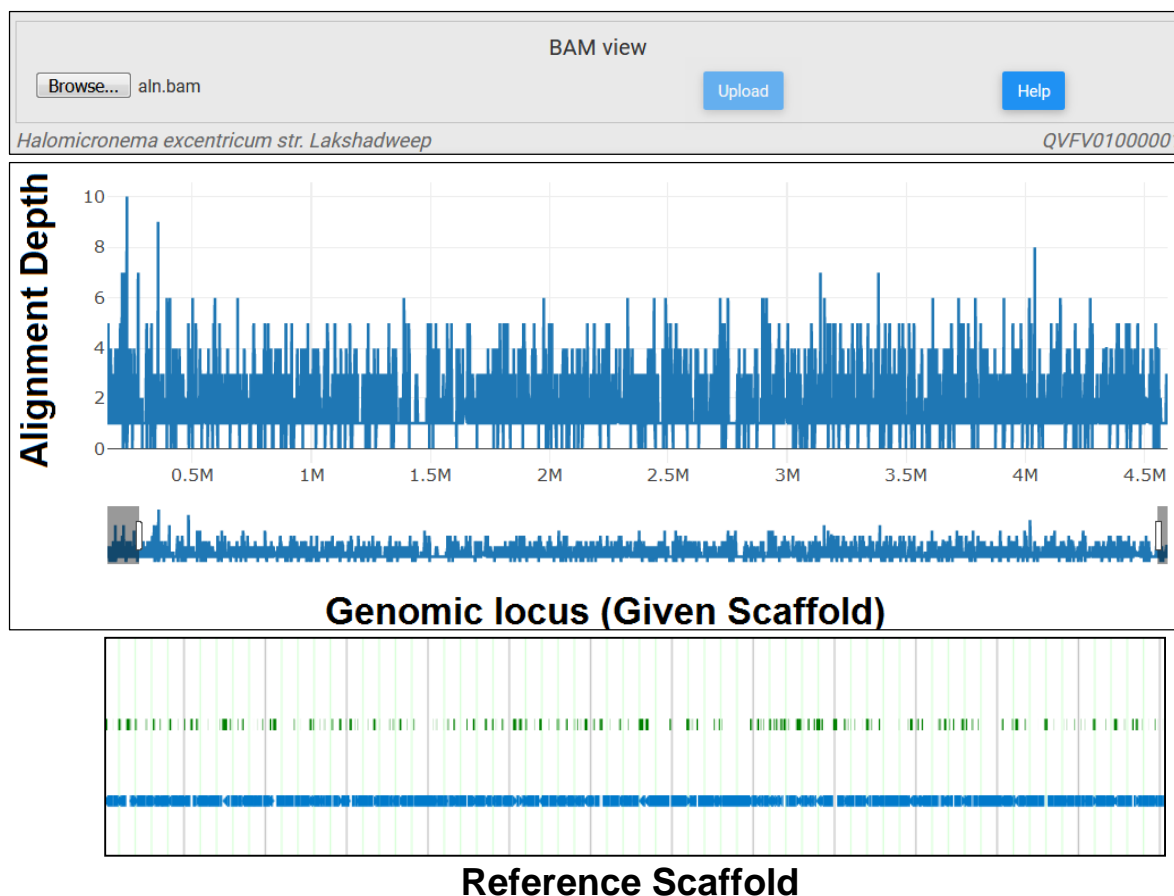
Set up Filter Option YES

The screenshot of the BLAST page showing various option for sequence input and parameter as well as genome selection.

BAM Alignment Upload

User can also upload BAM alignment files run against the given scaffold into the genome browser page which will visualize the depth of the alignment against the given scaffold loci.

The BAM visualization will look like this:



GAL Command Line Options

How to run GAL in command line mode?

GAL can easily be run from a web browser. Optionally, for users familiar with Docker command line and Ubuntu Terminal can run GAL through command line.

Accessing host directory

The host directory can be accessed through the following command:

```
docker run -it -v [host_directory_path]:[GAL_file  
system_path] -p 8080:80 cglabiicb/gal:[GAL version]
```

Example:

```
docker run -it -v /home/arijit/test:/usr/GAL_data -p  
8080:80 cglabiicb/gal
```

After running the above command, the host operating directory will be available to the GAL file system. In that way user can process data from the host directory. Now you will enter to GAL container.

```
root@container_id:/#
```

Log File

In case you have uploaded your genome for a long time and have not seen any progress or the tracks are still not appearing, check the `/usr/GAL/gal.log` file in your Docker instance. That should have the latest status. If the run is stuck and you want to resubmit, you can delete the file from the `/tmp` directory. The names of the files can be retrieved from the `gal.log` file as described above.

Running the programs

GAL is based on Python. Python 3.6 or above is required to use GAL. The main program for GAL is **gal_manage.py** present at: `/usr/GAL` path.

To run the GAL control script use following command:

```
python3 /usr/GAL/ gal_manage.py --orgconfig=[config_file_path]
```

Setting up the configuration file

User needs to provide configuration file in INI format.

INI format:

```
[section]  
name=value
```

Structure of the organism configuration file:

```
[OrganismDetails]  
Organism:  
version:  
source_url:  
  
[SequenceType]  
SequenceType:  
  
[AnnotationInfo]  
Blastp:  
signalp:  
pfam:  
tmhmm:  
  
[filePath]  
GenBank:  
FASTA:  
GFF:  
Product:  
LastZ:  
SignalP:  
pfam:  
TMHMM:  
Interproscan:  
  
[other]  
Program:  
ReferenceGenome:
```

Sample configuration file is present at: /usr/GAL/config/organism_config_format.ini

Data Format

We have defined input data type in four ways,

Data type Name	Input files
Genbank Annotation	Genbank Sequence File
No Annotation	Genome Fasta File
Minimal Annotation	Genome Fasta File, GFF file
Product Annotation	Genome Fasta File, GFF File, Product file

Sample organism data upload using command line mode:

Data Type	Commands to upload Sample genomes
Genbank Annotation	<code>python3 gal_manage.py -org=/usr/GAL/SampleFiles/genbank_annotation/org_config.ini</code>
No Annotation	<code>python3 gal_manage.py -org=/usr/GAL/SampleFiles/no_annotation/org_config.ini</code>
Minimal Annotation	<code>python3 gal_manage.py -org=/usr/GAL/SampleFiles/minimal_annotation/org_config.ini</code>
Product Annotation	<code>python3 gal_manage.py -org=/usr/GAL/SampleFiles/product_annotation/org_config.ini</code>

List of Reference Genomes

For AUGUSTUS Annotations:

Organism Name	Organism code for configuration file
Animals	
<i>Aedes aegypti</i>	aedes
<i>Amphimedon queenslandica</i>	amphimedon
<i>Acyrtosiphon pisum</i>	pea_aphid
<i>Brugia malayi</i>	brugia
<i>Caenorhabditis elegans</i>	caenorhabditis
<i>Drosophila melanogaster</i>	fly
<i>Homo sapiens</i>	human
<i>Nasonia vitripennis</i>	nasonia
<i>Tribolium castaneum</i>	tribolium
<i>Trichinella spiralis</i>	trichinella
Alveolata	
<i>Tetrahymena thermophila</i>	tetrahymena
<i>Toxoplasma gondii</i>	toxoplasma
Plants and Algae	
<i>Arabidopsis thaliana</i>	arabidopsis
<i>Galdieria sulphuraria</i>	galdieria
<i>Solanum lycopersicum</i>	tomato
<i>Zea mays</i>	maize
Fungi	
<i>Aspergillus fumigatus</i>	aspergillus_fumigatus
<i>Aspergillus nidulans</i>	aspergillus_nidulans
<i>Aspergillus oryzae</i>	aspergillus_oryzae
<i>Aspergillus terreus</i>	aspergillus_terreus
<i>Botrytis cinerea</i>	botrytis_cinerea
<i>Candida albicans</i>	candida_albicans
<i>Candida guilliermondii</i>	candida_guilliermondii
<i>Candida tropicalis</i>	candida_tropicalis
<i>Chaetomium globosum</i>	chaetomium_globosum
<i>Coccidioides immitis</i>	coccidioides_immitis
<i>Coprinus cinereus</i>	coprinus
<i>Cryptococcus neoformans</i>	cryptococcus_neoformans_neoformans_B
<i>Debaryomyces hansenii</i>	debaryomyces_hansenii
<i>Encephalitozoon cuniculi</i>	encephalitozoon_cuniculi_GB
<i>Eremothecium gossypii</i>	eremothecium_gossypii
<i>Fusarium graminearum</i>	fusarium_graminearum
<i>Histoplasma capsulatum</i>	histoplasma_capsulatum
<i>Kluyveromyces lactis</i>	kluyveromyces_lactis
<i>Laccaria bicolor</i>	laccaria_bicolor

Organism Name	Organism code for configuration file
<i>Lodderomyces elongisporus</i>	lodderomyces_elongisporus
<i>Magnaporthe grisea</i>	magnaporthe_grisea
<i>Neurospora crassa</i>	neurospora_crassa
<i>Phanerochaete chrysosporium</i>	phanerochaete_chrysosporium
<i>Pichia stipitis</i>	pichia_stipitis
<i>Rhizopus oryzae</i>	rhizopus_oryzae
<i>Saccharomyces cerevisiae</i>	saccharomyces_cerevisiae_S288C
<i>Schizosaccharomyces pombe</i>	schizosaccharomyces_pombe
<i>Ustilago maydis</i>	ustilago_maydis
<i>Yarrowia lipolytica</i>	yarrowia_lipolytica

For GeneMark Annotations:

Organism Name	Organism code for configuration file
<i>Vibrio fischeri</i> ES114	Aliivibrio_fischeri_hmm.mod
<i>Azotobacter vinelandii</i> DJ	Azotobacter_vinelandii_hmm.mod
<i>Bacillus subtilis</i> subsp. <i>subtilis</i> str. 168	Bacillus_subtilis_hmm.mod
<i>Escherichia coli</i> str. K-12 substr. MG1655	Escherichia_coli_hmm.mod
<i>Mycoplasma genitalium</i> G37	Mycoplasma_genitalium_hmm.mod
<i>Pseudomonas fluorescens</i> SBW25	Pseudomonas_fluorescens_hmm.mod
<i>Synechocystis</i> sp. PCC 6803	Synechocystis_sp._PCC_6803_hmm.mod

END OF DOCUMENT