EDGE bioinformatics

EDGE Documentation

Release Notes develop

EDGE Development Team

Aug 09, 2023

Contents

1 EDGE ABCs **1**
- 1.1 About EDGE Bioinformatics . . . 1
- 1.2 Bioinformatics overview . . . 1
- 1.3 Computational Environment . . . 4

2 Introduction **5**
- 2.1 What is EDGE? . . . 5
- 2.2 Why create EDGE? . . . 5

3 System requirements **7**
- 3.1 Hardware Requirements . . . 7
- 3.2 Ubuntu 18.04 . . . 7
- 3.3 CentOS 7 . . . 8

4 Installation **10**
- 4.1 EDGE Installation . . . 10
- 4.2 Configure SELinux on CentOS . . . 27
- 4.3 EDGE Docker image . . . 28

5 Graphic User Interface (GUI) **29**
- 5.1 User Login . . . 29
- 5.2 Upload Files . . . 30
- 5.3 Initiating an analysis job . . . 31
- 5.4 Choosing processes/analyses . . . 36
- 5.5 Submission of a job . . . 44
- 5.6 Checking the status of an analysis job . . . 44
- 5.7 Monitoring the Resource Usage . . . 46
- 5.8 Management of Jobs . . . 46
- 5.9 Project List Table . . . 47
- 5.10 Other Methods of Accessing EDGE . . . 48

6 Command Line Interface (CLI) **51**
- 6.1 Configuration File . . . 52
- 6.2 Test Run . . . 54
- 6.3 Descriptions of each module . . . 56
- 6.4 Other command-line utility scripts . . . 63

7 Output **64**
7.1 Example Output . . . 65

8 Databases **66**
8.1 EDGE provided databases . . . 66
8.2 Building bwa index . . . 69
8.3 SNP database genomes . . . 69
8.4 Ebola Reference Genomes . . . 76

9 Third Party Tools **77**
9.1 Assembly . . . 77
9.2 Annotation . . . 78
9.3 Alignment . . . 80
9.4 Taxonomy Classification . . . 81
9.5 Phylogeny . . . 82
9.6 Specialty Genes . . . 83
9.7 Metagenome . . . 83
9.8 Visualization and Graphic User Interface . . . 84
9.9 Utility . . . 85
9.10 Amplicon Analysis . . . 89
9.11 RNA-Seq Analysis . . . 89

10 FAQs and Troubleshooting **90**
10.1 FAQs . . . 90
10.2 Troubleshooting . . . 92
10.3 Discussions / Bugs Reporting . . . 94

11 Copyright **96**

12 Contact Us and Citation **97**
12.1 Citation . . . 97

CHAPTER 1

EDGE ABCs

A quick About EDGE, overview of the Bioinformatic workflows, and the Computational environment

1.1 About EDGE Bioinformatics

EDGE bioinformatics was **developed to help biologists process Next Generation Sequencing data** (in the form of **raw FASTQ** files), even if they have little to no bioinformatics expertise. EDGE is a **highly integrated and interactive web-based platform** that is capable of running many of the standard analyses that biologists require for viral, bacterial/archaeal, and metagenomic samples. EDGE provides the following analytical workflows: **pre-processing, assembly and annotation, reference-based analysis, taxonomy classification, phylogenetic analysis, Gene Family Analysis, PCR analysis, Qiime2 amplicon data analysis, targeted sequencing adjudication and RNA-Seq analysis**. EDGE provides an intuitive web-based interface for user input, allows users to visualize and interact with selected results (e.g. JBrowse genome browser), and generates a final detailed PDF report. Results in the form of tables, text files, graphic files, and PDFs can be downloaded. A user management system allows tracking of an individual's EDGE runs, along with the ability to share, post publicly, delete, or archive their results.

While EDGE was intentionally designed to be as simple as possible for the user, there is still no single 'tool' or algorithm that fits all use-cases in the bioinformatics field. Our intent is to provide a detailed panoramic view of your sample from various analytical standpoints, but users are encouraged to have some knowledge of how each tool/algorithm workflow functions, and some insight into how the results should best be interpreted.

1.2 Bioinformatics overview

1.2.1 Inputs:

The input to the EDGE workflows begins with one or more **illumina FASTQ files** for a single sample. (There is currently limited capability of incorporating PacBio and Oxford Nanopore data into the Assembly module) The user can also enter SRA/ENA accessions to allow processing of publically available datasets. Comparison among samples is not yet supported but development is underway to accommodate such a function for assembly and taxonomy profile comparisons.

1.2.2 Workflows:

Pre-Processing

Assessment of quality control is performed by FAQCS. Users can optionally find and remove adapters from Oxford Nanopore reads using Porechop. In addition, users can optionally stitch paired-end(PE) reads using fastq-join and use joined PE reads for downstream analysis. The host removal step requires the input of one or more reference genomes as FASTA. Several common references are available for selection. Trimmed and host-screened FASTQ files are used for input to the other workflows.

Assembly and Annotation

We provide the IDBA, Spades, MegaHit for illumina reads, LRASM includes miniasm and wtdbg2 alogrithm and (meta)flye for PacBio/Nanopore reads, and Unicycler for bacteria genomes hybird assembly. These assembly tools are to accommodate a range of sample types and data sizes. When the user selects to perform an assembly, all subsequent workflows can execute analysis with either the reads, the contigs, or both (default). For annotation, Prokka and RATT are provided for ab initio or transfer annoation from close-related reference genome. Start from version 2.4, EDGE use antiSMASH v4.1.0 for the rapid genome-wide identification, annotation and analysis of secondary metabolite biosynthesis gene clusters in bacterial and fungal genomes. In addition, the assembled contigs can be binned by Maxbin2 and assessed the quality of binning result by CheckM .

Reference-Based Analysis

For comparative reference-based analysis with reads and/or contigs, users must input one or more references (as FASTA or multi-FASTA if there are more than one replicon) and/or select from a drop-down list of RefSeq complete genomes. Results include lists of missing regions (gaps), inserted regions (with input contigs if assembly was performed), SNPs (and coding sequence changesa with genbank information), as well as genome coverage plots and interactive access via JBrowse. There is an option to output consensus Fasta file from the mapping result.

Taxonomy Classification

For taxonomy classification with reads, multiple tools are used and the results are summarized in heat map and radar plots. Individual tool results are also presented with taxonomy dendograms and Krona plots. Contig classification occurs by assigning taxonomies to all possible portions of contigs. For each contig, the longest and best match (using minimap2) is kept for any region within the contig and the region covered is assigned to the taxonomy of the hit. The next best match to a region of the contig not covered by prior hits is then assigned to that taxonomy. The contig results can be viewed by length of assembly coverage per taxa or by number of contigs per taxa.

Phylogenetic Analysis

For phylogenetic analysis, the user must select datasets from near neighbor isolates for which the user desires a phylogeny. A minimum of two additional datasets are required to draw a tree. At least one dataset must be an assembly or complete genome. RefSeq genomes (Bacteria, Archaea, Viruses) are available from a dropdown menu, SRA and FASTA entries are allowed, and previously built databases for some select groups of bacteria are provided. This workflow (see PhaME) is a whole genome SNP-based analysis that uses one reference assembly to which both reads and contigs are mapped. Because this analysis is based on read alignments and/or contig alignments to the reference genome(s), we **strongly recommend only selecting genomes that can be adequately aligned at the nucleotide level (i.e. ~90% identity or better)**. The number of 'core' nucleotides able to be aligned among all genomes, and the number of SNPs within the core, are what determine the resolution of the phylogenetic tree. Output phylogenies are presented along with text files outlining the SNPs discovered.

Gene Family Analysis

For specialty gene analysis, the user selects read-based analysis and/or ORF(contig)-based analysis.

For read-based analysis antibiotic resistance genes and virulence genes are detected using Huttenhower lab's progam ShortBRED. The antibiotic resistance gene database was generated by the developers of ShortBRED using genes from ARDB and Resfams. The virulence genes database was generated by the developers of EDGE using VFDB.

For ORF-based analysis, antibiotic resistance genes are detected using CARD's (Comprehensive Antibiotic Resistance Database) program RGI (Resistance Gene Identifier). RGI uses CARD's custom database of antibiotic resistance genes. The virulence genes are detected using ShortBRED with a database generated by the developers of EDGE using VFDB.

Primer Analysis

For primer analysis, if the user would like to validate known PCR primers in silico, a FASTA file of primer sequences must be input. New primers can be generated from an assembly as well.

Qiime2 analysis

QIIME2 is an open-source bioinformatics pipeline for performing microbiome analysis from raw DNA sequencing data. EDGE implementation is based on Qiime 2 core 2023.5 and includes demultiplexing and quality control/filtering, feature table construction, taxonomic assignment, and phylogenetic reconstruction, and diversity analyses and visualizations. Currently, EDGE suports three amplicon types, 16s using GreenGenes database, 16s/18s using SILVA database, and Fungal ITS.

DETEQT (TargetedNGS) analysis

DETEQT is a pipeline for diagnostic targeted sequencing adjudication.

This tool been designed to be robust enough to handle a range of assay designs. Therefore, no major assumptions of input reads are made except that they represent amplicons from a multiplexed targeted amplification reaction and that the **reference is comprised of only target regions** in the assay, instead of whole genomes. The idea is to survey the reads and delineate whether each reference sequence, or target, is present or absent.

PiReT analysis

EDGE integrated PiReT (Pipeline for Reference based Transcriptomics) which is an open-source bioinformatics pipeline for performing RNA-Seq analysis. The workflow written mostly in Python on a popular workflow manager package luigi (developed by spotify). It allow users to find differentially expressed transcripts (genes, sRNAs), discover novel non coding RNAs, co-expressed genes and pathways from raw fastq, reference sequence, and experimental design files.

All commands and tool parameters are recorded in log files to make sure the results are repeatable and traceable. The main output is an integrated interactive web page that includes summaries of all the workflows run and features tables, graphical plots, and links to genome (if assembled, or of a selected reference) browsers and to access unprocessed results and log files. Most of these summaries, including plots and tables are included within a final PDF report.

1.2.3 Limitations

Pre-processing

For host removal/screening, not all genomes are available from a drop-down list, however users can provide their own genome fasta file as host input.

Assembly and Taxonomy Classification

EDGE has been primarily designed to **analyze microbial (bacterial, archaeal, viral) isolates or (shotgun) metagenome samples**. Due to the complexity and computational resources required for eukaryotic genome assembly, and the fact that the most taxonomy classification tools do not support eukaryotic classification (except Metaphlan2), EDGE does not fully support eukaryotic samples. The combination of large NGS data files and complex metagenomes may also run into computational memory constraints.

Reference-based analysis

We recommend only aligning against (a limited number of) most closely related genome(s) (default on GUI limit up to 200 fragments). If this is unknown, the Taxonomy Classification module is recommended as an alternative. If the user

selects too many references, this may affect runtimes or require more computational resources than may be available on the user's system.

Phylogenetic Analysis

Because this pipeline provides SNP-based trees derived from whole genome (and contig) alignments or read mapping, **we recommend selecting genomes within the same species or at least within the same genus**.

1.3 Computational Environment

1.3.1 EDGE source code, images, and webservers

EDGE was designed to be installed and implemented from within any institute that provides sequencing services or that produces or hosts NGS data. When installed locally, EDGE can access the raw FASTQ files from within the institute, thereby providing immediate access by the biologist for analysis. EDGE is available in a variety of packages to fit various institute needs. **EDGE source code** can be obtained via our GitHub page. To simplify installation, a Docker image can also be obtained. An **online version of EDGE** is currently available at https://edgebioinformatics.org/.

CHAPTER 2

Introduction

2.1 What is EDGE?

EDGE is a highly adaptable bioinformatics platform that allows laboratories to quickly analyze and interpret genomic sequence data. The bioinformatics platform allows users to address a wide range of use cases including assay validation and the characterization of novel biological threats, clinical samples, and complex environmental samples. EDGE is designed to:

- Align to real world use cases
- Make use of open source (free) software tools
- Run analyses on small, relatively inexpensive hardware
- Provide remote assistance from bioinformatics specialists

2.2 Why create EDGE?

EDGE bioinformatics was **developed to help biologists process Next Generation Sequencing data** (in the form of **raw FASTQ** files), even if they have little to no bioinformatics expertise. EDGE is a **highly integrated and interactive web-based platform** that is capable of running many of the standard analyses that biologists require for viral, bacterial/archaeal, and metagenomic samples. EDGE provides the following analytical workflows: **quality trimming and host removal, assembly and annotation, comparisons against known references, taxonomy classification of reads and contigs, whole genome SNP-based phylogenetic analysis, and PCR analysis**. EDGE provides an intuitive web-based interface for user input, allows users to visualize and interact with selected results (e.g. JBrowse genome browser), and generates a final detailed PDF report. Results in the form of tables, text files, graphic files, and PDFs can be downloaded. A user management system allows tracking of an individual's EDGE runs, along with the ability to share, post publicly, delete, or archive their results.

While the design of EDGE was intentionally done to be as simple as possible for the user, there is still no single 'tool' or algorithm that fits all use-cases in the bioinformatics field. Our intent is to provide a detailed panoramic view of your sample from various analytical standpoints, but users are encouraged to have some insight into how each tool or workflow functions, and how the results should best be interpreted.

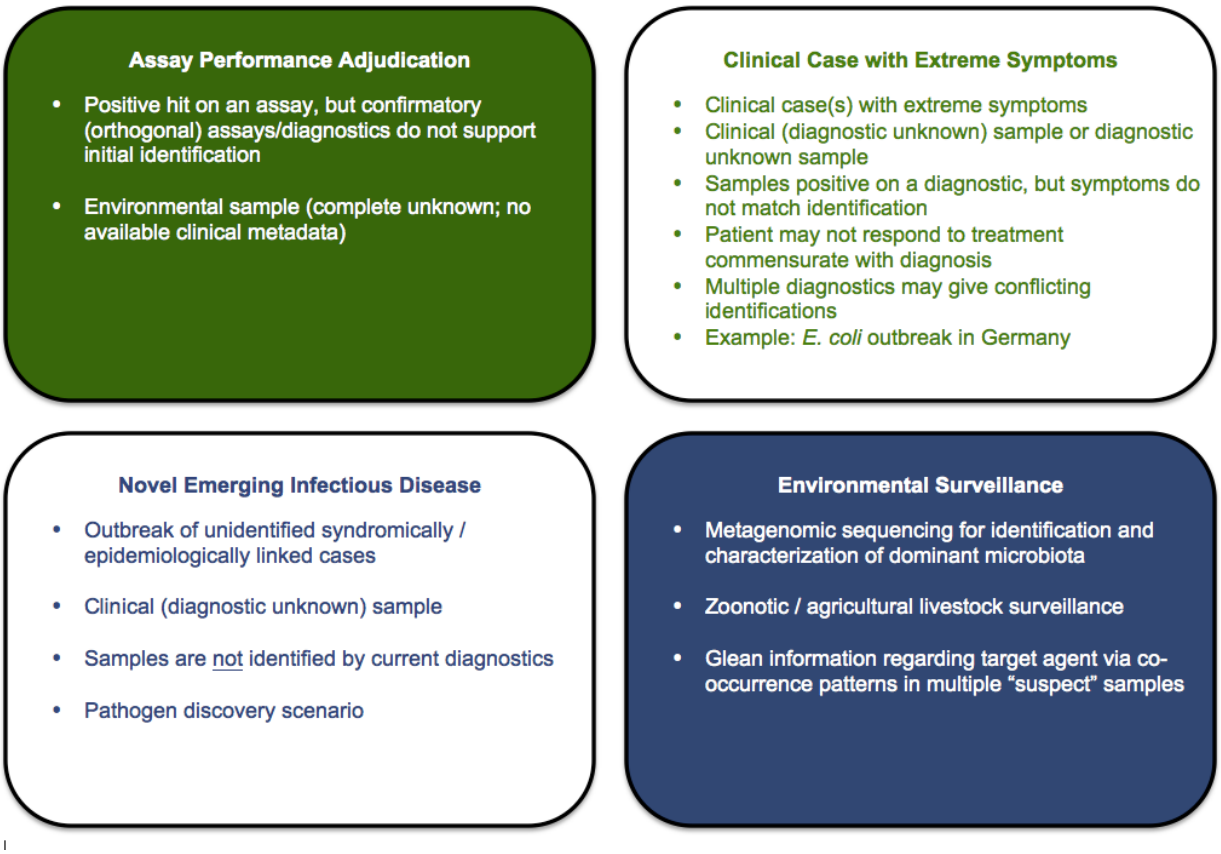


Fig. 1: Four common Use Cases guided initial EDGE Bioinformatic Software development.

CHAPTER 3

System requirements

NOTE: There is an online version of EDGE, found on https://edgebioinformatics.org/ is hosted at Texas Advanced Computing Center (TACC) with 128G memory and 44 CPUs.

The current version of the EDGE pipeline has been extensively tested on a Linux Server with Ubuntu 18.04 and CentOS 6.5/7 operating system and will work on 64bit Linux environments.

3.1 Hardware Requirements

Due to the involvement of several high memory and high cpu consuming steps Minimum requirement: 24GB memory, at least 8 computing CPUs and 1 TB disk space. A higher computer spec is strongly recommended: 256GB memory, 64 computing CPUs and > 4 TB disk space. Please ensure that your system has the essential software packages installed properly before running the installing script. The following should be installed by a system administrator (requires sudo).

Note: If your system OS is neither Ubuntu 18.04 or CentOS 7.0, it may have differnt packages/libraries name and the newer complier on newer OS may fail on compling some of thirdparty bioinformatics tools. We would suggest to use EDGE Docker container.

3.2 Ubuntu 18.04



1. Install build essential libraries and dependancies:

```
sudo apt-get update

sudo apt-get install -y build-essential libreadline-gplv2-dev libx11-dev \
    libxt-dev libgsl-dev libfreetype6-dev libncurses5-dev gfortran \
    inkscape libwww-perl libxml-libxml-perl libperlio-gzip-perl  \
    zlib1g-dev zip unzip libjson-perl libpng-dev cpanminus default-jre \
    firefox wget curl csh liblapack-dev libblas-dev libatlas-base-dev \
    libcairo2-dev libssh2-1-dev libssl-dev libcurl4-openssl-dev bzip2 \
    bioperl rsync libbz2-dev liblzma-dev time libterm-readkey-perl \
    liblwp-protocol-https-perl gnuplot libjson-xs-perl libio-socket-ip-perl \
    vim php sendmail mysql-client mysql-server libgfortran3  texinfo \
    openssh-server openssh-client openjdk-11-jdk texlive git gawk \
    texlive-fonts-extra libboost-all-dev cron less libxml2-dev \
    libcgi-pm-perl libxml-simple-perl libxml-dom-perl locales \
    libspreadsheet-parseexcel-perl libspreadsheet-writeexcel-perl ghostscript
```

2. Install Apache2 for EDGE UI:

```
sudo apt-get install apache2
sudo a2enmod cgid proxy proxy_http headers rewrite
```

3. Install packages for user management system:

```
sudo apt-get install sendmail mysql-client mysql-server

cd /usr/share
wget https://archive.apache.org/dist/tomcat/tomcat-7/v7.0.109/bin/apache-tomcat-7.
↪0.109.tar.gz
tar xzf apache-tomcat-7.0.109.tar.gz
rm apache-tomcat-7.0.109.tar.gz
mv apache-tomcat-7.0.109 tomcat7
echo "export CATALINA_HOME=\"/usr/share/tomcat7\"" >> /etc/profile
```

4. Change the image conversion policy:

```
sed -i.bak 's/rights=\"none\" pattern=\"PDF\"/rights=\"read|write\" pattern=\"PDF\
↪"/' /etc/ImageMagick-6/policy.xml
```

3.3 CentOS 7

1. Install libraries and dependencies by yum:

```
# add epel reporsitory
sudo yum -y install epel-release

sudo yum install -y libX11-devel readline-devel libXt-devel ncurses-devel␣
↪inkscape \
    expat expat-devel freetype freetype-devel zlib zlib-devel perl-App-cpanminus \
    perl-Test-Most blas-devel atlas-devel lapack-devel libpng12 libpng12-devel \
    perl-XML-Simple perl-JSON csh gcc gcc-c++ make binutils gd gsl-devel git␣
↪graphviz \
    java-1.7.0-openjdk perl-Archive-Zip perl-CGI curl perl-CGI-Session \
    perl-CPAN-Meta-YAML perl-DBI perl-Data-Dumper perl-GD perl-IO-Compress \
    perl-Module-Build perl-XML-LibXML perl-XML-Parser perl-XML-SAX perl-XML-SAX-
↪Writer \
```

(continues on next page)

(continued from previous page)

```
    perl-XML-Twig perl-XML-Writer perl-YAML perl-PerlIO-gzip libstdc++-static \
    cairo-devel openssl-devel openssl-static libssh2-devel libcurl-devel \
    wget rsync bzip2 bzip2-devel xz-devel time zip unzip which perl-CPAN \
    perl-LWP-Protocol-https cronie gnuplot gdb perl-JSON-XS perl-IO-Socket-IP \
    texlive texinfo libgfortran.x86_64 java-1.7.0-openjdk-devel boost-devel \
    libxml2-devel libXScrnSaver gtk3 perl-XML-DOM
```

2. Update perl tools:

```
sudo cpanm App::cpanoutdated
cpan-outdated -p | sudo cpanm
```

3. Install perl modules by cpanm:

```
sudo cpanm -f Bio::Perl Bio::SearchIO::hmmer3 Net::Ping File::Which
sudo cpanm Graph Time::Piece Hash::Merge PerlIO::gzip Heap::Simple::XS File::Next
sudo cpanm Algorithm::Munkres Archive::Tar Array::Compare Clone Convert::Binary::C
sudo cpanm HTML::Template HTML::TableExtract List::MoreUtils PostScript::TextBlock
sudo cpanm SOAP::Lite SVG SVG::Graph Set::Scalar Sort::Naturally␣
↪Spreadsheet::ParseExcel
sudo cpanm CGI::Simple GraphViz XML::Parser::PerlSAX XML::Simple Term::ReadKey
sudo cpanm Spreadsheet::WriteExcel
```

4. Install package for httpd for EDGE UI:

```
sudo yum -y install httpd
sudo systemctl enable httpd && sudo systemctl start httpd
```

5. Install packages for user management system:

```
sudo yum -y install sendmail mariadb-server mariadb php phpMyAdmin tomcat
sudo systemctl enable tomcat && sudo systemctl start tomcat
```

6. Configure firewall for ssh, http, https, and smtp:

```
sudo firewall-cmd --permanent --add-service=ssh
sudo firewall-cmd --permanent --add-service=http
sudo firewall-cmd --permanent --add-service=https
sudo firewall-cmd --permanent --add-service=smtp
sudo firewall-cmd --reload
```

7. Disable SELinux:

```
As root edit /etc/selinux/config and set SELINUX=disabled

    Restart the server to make the change
```

Warning: This is for development version of EDGE. Stable version (v2.3) is here.

CHAPTER 4

Installation

Note: These instructions assumes Ubuntu 18 and CentOS 7

4.1 EDGE Installation

Note: A base install is ~12GB for the code base and ~500GB for the databases. It should run as normal user. (not root)

1. Please ensure that your system has the *essential software building packages* (page 7). installed properly before proceeding following installation.
2. Download the codebase, databases and third party tools.:

```
## Codebase is ~207Mb and contains all the scripts and HTML needed to make EDGE␣
↪run
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_main.tgz

## Third party tools is ~1.5Gb and contains the underlying programs needed to do␣
↪the analysis
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_thirdParty_
↪softwares.tgz

## Pipeline database is ~17Gb and contains the other databases needed for EDGE
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_pipeline_
↪databases.tgz

## BWA index is ~41Gb and contains the databases for bwa taxonomic identification␣
↪pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_bwa_index.tgz
```

(continues on next page)

(continued from previous page)

```
## HOST genomes BWA index is ~41Gb for Host removal, including human, bacteria,␣
↪phiX, viruses, invertebrate vectors of human pathogens
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_HostIndex.tgz

## NCBI Genomes is ~21Gb and contain the full genomes for prokaryotes and some␣
↪viruses
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_NCBI_genomes.tgz

## GOTTCHA database is ~16Gb and contains the custom databases for the GOTTCHA␣
↪taxonomic identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_GOTTCHA_db.tgz

## NT database is ~25Gb and contains the NCBI nt database for contig␣
↪identification
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_nt_20160426.tgz

## ShortBRED database is ~27Mb and contains the databases used by ShortBRED for␣
↪virulence factors and read based antibiotic resistance analysis
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_ShortBRED_
↪Database.tgz

## Diamond database is ~16Gb and contains the databases from RefSeq for protein␣
↪based taxonomic identification
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_diamond_db.tgz

## MetaPhlAn4 database is 14Gb file contains the databases used for the␣
↪MetaPhlAn4 taxonomic identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_metaphlan4DB.tgz

## GOTTCHA2 databases is 38Gb file and contains the custom databases for the␣
↪GOTTCHA2 taxonomic identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_GOTTCHA2_db_20190729.
↪tgz

## Kraken2 database is 39Gb file contains the databases used for the Kraken2␣
↪taxonomic identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_Kraken2_db_20211216.
↪tgz

## Centrifuge database is 20G file contains the databases used for the Centrifuge␣
↪taxonomic identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_Centrifuge_db_
↪20200329.tgz

## PanGIA database is 35G file for PanGIA taxonomic identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_PanGIA_db.tgz

## MICCR database is 48GB contains the databases used for the contig taxonomic␣
↪identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_ContigTax_db_20190114.
↪tgz

## CheckM database is 275MB contains the databases used for the Metagenome Binned␣
↪contig quality assessment.
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_checkM_db_20190213.tgz
```

(continues on next page)

(continued from previous page)

```
## Qiime2 database is 1.4GB contains 16s,18s and ITS db.
wget -c  https://ref-db.edgebioinformatics.org/EDGE/dev/edge_qiime2_db_20230719.
↪tgz

## AntiSmash database is 3.2GB contains pfam resfam tigrfam can clusterblast db␣
↪for antismash version 6
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_AntiSmash6.tgz

(Optional)
## Other Host bwa index ~18Gb for host removal, including pig, sheep, cow, monkey,
↪ hamster. and goat.
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_otherHostIndex.tgz

## For machine with < 32Gb memory, we suggest to use the smaller BWA index (~
↪14Gb) and contains the databases for bwa taxonomic identification pipeline
wget -c https://ref-db.edgebioinformatics.org/EDGE/dev/edge_dev_bwa_mini_index.tgz
```

Warning: Be patient; the database files are huge.

3. Unpack main archive:

```
tar -xvzf edge_dev_main.tgz
```

Note: The main directory, edge_dev, will be created.

Create a link from edge to that directory:

```
ln -sf edge_dev edge
```

4. Unpack the third party software into main directory (edge):

```
tar -xvzf edge_dev_thirdParty_softwares.tgz -C edge/
```

Note: You should see a thirdParty directory inside the edge directory.

Note: If you are updating from old version, you should still expand the new thirdParty tgz file into the existing thirdParty directory.

5 Unpack the databases:

```
# unpack databases
tar -xvzf edge_dev_pipeline_databases.tgz
tar -xvzf edge_dev_GOTTCHA_db.tgz
tar -xzvf edge_dev_bwa_index.tgz
tar -xvzf edge_dev_NCBI_genomes.tar.gz
tar -xzvf edge_dev_amplicons_db.tgz
tar -xzvf edge_dev_nt_20160426.tgz
tar -xvzf edge_dev_ShortBRED_Database.tgz
tar -xvzf edge_dev_HostIndex.tgz
```

(continues on next page)

(continued from previous page)

```
tar -xvzf edge_dev_diamond_db.tgz
tar -xvzf edge_dev_metaphlan4DB.tgz
tar -xvzf edge_GOTTCHA2_db_20190729.tgz
tar -xvzf edge_Kraken2_db_20211216.tgz
tar -xvzf edge_ContigTax_db_20190114.tgz
tar -xvzf edge_checkM_db_20190213.tgz
tar -xvzf edge_qiime2_db_20230719.tgz
tar -xvzf edge_dev_AntiSmash6.tgz
```

Note: At this point, you should see a database directory and the edge directory.

6. Create the symlink from edge to the database directory:

```
ln -s `pwd`/database edge/database
```

Note: This will keep the database directory outside of the edge install location. Should you need to reinstall the code base you will not need to redownload/install the databases.

7. Installing pipeline:

```
cd edge
./INSTALL.sh
```

Note: When installing JBrowse, it may require internet connection.

Note: If the machine is shared and used with others, the system installed tools version may not be compatible with EDGE. In this case, we would suggest to use force option `./INSTALL.sh force` to install all list tools locally.

It will install the following depended *tools* (page 77).

- Assembly
  - idba
  - spades
  - megahit
  - long_read_assembly
  - racon
  - unicycler
- Annotation
  - prokka
  - RATT
  - tRNAscan
  - barrnap

    - BLAST+
    - blastall
    - phageFinder
    - glimmer
    - aragorn
    - prodigal
    - tbl2asn
    - ShortBRED
    - antismash
- Alignment
    - hmmer
    - infernal
    - bowtie2
    - bwa
    - mummer
    - RAPSearch2
    - diamond
    - minimap2
- Taxonomy
    - kraken2
    - metaphlan
    - kronatools
    - gottcha
    - gottcha2
    - centrifuge
    - miccr
    - pangia
- Phylogeny
    - FastTree
    - RAxML
- Metagenome
    - MaxBin
    - checkM
- Utility
    - FaQCs
    - bedtools

    - R
    - GNU_parallel
    - tabix
    - JBrowse
    - bokeh
    - primer3
    - samtools
    - bcftools
    - sratoolkit
    - ea-utils
    - omics-pathway-viewer
    - NanoPlot
    - Porechop
    - seqtk
    - Rpackages
    - Chromium
- Perl_Modules
    - perl_parallel_forkmanager
    - perl_excel_writer
    - perl_archive_zip
    - perl_string_approx
    - perl_pdf_api2
    - perl_html_template
    - perl_html_parser
    - perl_JSON
    - perl_bio_phylo
    - perl_xml_twig
    - perl_cgi_session
    - perl_email_valid
    - perl_mailtools
- Python_Packages
    - Mambaforge
- Pipeline_Tools
    - DETEQT
    - reference-based_assembly
    - PyPiReT

- qiime2

8. Restart the Terminal Session to allow $EDGE_HOME to be exported.

Note: After running INSTALL.sh successfully, the binaries and related scripts will be stored in the ./bin and ./scripts directory. It also writes EDGE_HOME environment variable into .bashrc or .bash_profile.

4.1.1 Testing the EDGE Installation

After installing the packages above, it is highly recommended to test the installation:

```
> cd $EDGE_HOME/testData
> ./runAllTest.sh
```

```
Working Dir: /panfs/biopan01/edge-dev-master/edge-dev-test/testData
EDGE_HOME Dir: /panfs/biopan01/edge-dev-master/edge-dev-test/
[00:00:00]  [5 %]     Test Assembly ......    [OK]
[00:05:32]  [10 %]    Test Contigs2Reference ......   [OK]
[00:14:28]  [15 %]    Test ContigsAnnotation ......   [OK]
[00:21:14]  [20 %]    Test ContigsTaxonomy ......     [OK]
[00:32:34]  [25 %]    Test DETEQt ......      [OK]
[00:34:02]  [30 %]    Test HostRemoval ...... [OK]
[00:35:10]  [35 %]    Test JoinPE ......      [OK]
[00:35:25]  [40 %]    Test NanoEDGE ......    [OK]
[00:45:24]  [45 %]    Test PCRassay ......    [OK]
[00:52:17]  [50 %]    Test PhageFinder ...... [OK]
[00:53:35]  [55 %]    Test PhylogeneticAnalysis ......        [OK]
[00:54:56]  [60 %]    Test QC ......  [OK]
[00:55:19]  [65 %]    Test Qiime ......       [OK]
[01:18:57]  [70 %]    Test Reads2Contigs ......       [OK]
[01:21:21]  [75 %]    Test Reads2Reference ......     [OK]
[01:26:44]  [80 %]    Test ReadsTaxonomy ......       [OK]
[01:41:06]  [85 %]    Test Report ......      [OK]
[01:45:55]  [90 %]    Test SpecialtyGenesProfiling ......     [Failed]
[02:00:09]  [95 %]    Test SRADownload ...... [OK]
[02:01:11]  [100 %]   Test VariantAnalysis ......     [OK]

 19/20 test(s) passed

 Total Running Time: 02:07:02
```

There are 20 module/unit tests which took around 2 hours07 mins in our testing environments. (64 cores 2.30GHz, 512GB ram with CentOS-7.1.1503). You will see test output on the terminal indicating test successes and failures. The **Specialty Genes Profiling test** will fail in this stage since it requires virulence database imported and configured. You can test it again after database created and configured. Some tests may fail due to missing external applications/modules/packages or failed installation. These will be noted separately in the $EDGE_HOME/testData/runXXXXTest/TestOutput/error.log or log files in each modules. If these are related to features of EDGE that you are not using, this is acceptable. Otherwise, you'll want to ensure that you have the EDGE installed correctly. If the output doesn't indicate any failures, you are now ready to use EDGE through command line. To take advantage of the user friendly GUI, please follow the section below to configure the EDGE Web server.

4.1.2 Apache Web Server Configuration

Note: The following system service/tools confiruration will require **sudo** privilege.

1. Modify/Check sample apache configuration file:

```
For Ubuntu

Double check $EDGE_HOME/edge_ui/apache_conf/edge_apache.conf alias directories␣
↪the match EDGE
installation path at line 2,5,6,16,17,29,38,69.

The default is configured as http://localhost/edge_ui/ or http://www.yourdomain.
↪com/edge_ui/

For CentOS

Double check $EDGE_HOME/edge_ui/apache_conf/edge_httpd.conf alias directories the␣
↪match EDGE
installation path at line 2,5,6,16,17,29,38,70.

The default is configured as http://localhost/edge_ui/ or http://www.yourdomain.
↪com/edge_ui/
```

2. Confirm apache/httpd user and groups are edge:

```
For Ubuntu

The user and group can be edited at /etc/apache2/envvars and the variables are␣
↪APACHE_RUN_USER and APACHE_RUN_GROUP.

For CentOS

The User and Group on lines 66 and 67 in $EDGE_HOME/edge_ui/apache_conf/centos_
↪httpd.conf should be edge

## Make APACHE_RUN_USER have Permission to write
> sudo chown -R xxxxx $EDGE_HOME/edge_ui  $EDGE_HOME/edge_ui/JBrowse/data
↪#(xxxxx is the APACHE_RUN_USER value)

> sudo chgrp -R xxxxx $EDGE_HOME/edge_ui  $EDGE_HOME/edge_ui/JBrowse/data
↪#(xxxxx is the APACHE_RUN_GROUP value)
```

3. (Optional) If users are behind a corporate proxy for internet:

```
Please add proxy info into $EDGE_HOME/edge_ui/apache_conf/edge_apache.conf or
↪$EDGE_HOME/edge_ui/apache_conf/edge_httpd.conf

# Add following proxy env
SetEnv http_proxy http://yourproxy:port
SetEnv https_proxy http://yourproxy:port
SetEnv ftp_proxy http://yourproxy:port
```

4. Copy configuration files to the appropriate directories:

```
For Ubuntu

> sudo cp $EDGE_HOME/edge_ui/apache_conf/edge_apache.conf /etc/apache2/conf-
↪available/
> sudo ln -s /etc/apache2/conf-available/edge_apache.conf /etc/apache2/conf-
↪enabled/
> sudo cp $EDGE_HOME/edge_ui/apache_conf/pangia-vis.conf /etc/apache2/conf-
↪available/
> sudo ln -s /etc/apache2/conf-available/pangia-vis.conf /etc/apache2/conf-
↪enabled/

For CentOS

> sudo cp $EDGE_HOME/edge_ui/apache_conf/edge_httpd.conf /etc/httpd/conf.d/
> sudo cp -f $EDGE_HOME/edge_ui/apache_conf/centos_httpd.conf /etc/httpd/conf/
↪httpd.conf
> sudo cp $EDGE_HOME/edge_ui/apache_conf/pangia-vis.conf /etc/httpd/conf.d/
```

5. (Optional) HTTPS / SSL configuration:

```
i. Please add redirect conditions into $EDGE_HOME/edge_ui/apache_conf/edge_apache.
↪conf or $EDGE_HOME/edge_ui/apache_conf/edge_httpd.conf

# Add redirect to https
RewriteEngine on
RewriteCond %{HTTPS} !=on
RewriteRule ^(.*) https://%{SERVER_NAME}$1 [R,L]

ii. Use pangia-vis-https.conf instead of pangia-vis.conf

For Ubuntu
> sudo cp $EDGE_HOME/edge_ui/apache_conf/pangia-vis-https.conf /etc/apache2/conf-
↪available/pangia-vis.conf

For CentOS
> sudo cp $EDGE_HOME/edge_ui/apache_conf/pangia-vis-https.conf /etc/httpd/conf.d/

iii. Add SSL configuration::

see edge_ssl.conf using letsencrypt (https://letsencrypt.org/) as an example. ␣
↪Please modify it as your environments and

copy modified $EDGE_HOME/edge_ui/apache_conf/edge_ssl.conf to /etc/httpd/conf.d/␣
↪for CentOS or /etc/apache2/conf-enabled/ for Ubuntu.
```

6. Restart the apache2/httpd to activate the new configuration:

```
For Ubuntu

> sudo service apache2 restart

For CentOS

> sudo systemctl restart httpd
```

4.1.3 User Management System Installation: MySQL

Note: Setup two temporary environmental variables:

```
UN=username
PW=password
```

These will be used when setting up the user management system

Note: If you were using the user management system and are updating from old EDGE version to this version. You only need to run the commands below and continue to install tomcat.:

```
cd $EDGE_HOME/userManagement
mysql -u $UN -p userManagement
mysql> source update_userManagement_db.sql
```

Note: For MySQL 5.7 (Ubuntu 18.04), please append following content to /etc/mysql/my.cnf

```
[mysqld]
show_compatibility_56 = on
sql-mode=""
```

1. Start mysql (if it is not already running):

   ```
   For Ubuntu

   > sudo service mysql start

   For CentOS

   > sudo systemctl start mariadb.service && sudo systemctl enable mariadb.service
   ```

2. Secure mysql:

   Note: The root password here is for the mysql root and not the system root.

   ```
   > sudo mysql_secure_installation
   ```

   1. Enter root password (likely none)
   2. Set root password? Yes
   3. Enter new root password.
   4. Re-enter new root password.
   5. Remove anonymous users? Yes
   6. Disallow root login remotely? Yes
   7. Remove test database and access to it? Yes
   8. Reload privilege table now? Yes

3. Create database: userManagement:

```
> cd $EDGE_HOME/userManagement
> mysql -p -u root

mysql> create database userManagement;
mysql> use userManagement;
```

4. Load userManagement_schema.sql:

```
mysql> source userManagement_schema.sql;
```

5. Load userManagement_constrains.sql:

```
mysql> source userManagement_constrains.sql;
```

6. Create an user account and grant all privileges to user:

Note: This is the database user (not an individual EDGE user account).

Replace with the appropriate values:

```
username: yourDBUsername
password: yourDBPassword
```

```
mysql> CREATE USER 'yourDBUsername'@'localhost' IDENTIFIED BY
↪'yourDBPassword';
mysql> GRANT ALL PRIVILEGES ON userManagement.* to 'yourDBUsername'@
↪'localhost';
mysql> exit;
```

If you need to allow remote access mysql, you will need to change localhost to % and comment out bind_address=127.0.0.1 at /etc/mysql/mysql.conf.d/mysqld.cnf

mysql> CREATE USER 'yourDBUsername'@'%' IDENTIFIED BY 'yourDBPassword';
mysql> GRANT ALL PRIVILEGES ON userManagement.* to 'yourDBUsername'@'%';
mysql> exit;

4.1.4 User Management System Installation: Tomcat

Note: If you were using the user management system and are updating from old EDGE version to this version. You only need continue from step 6.

---

1. Configure tomcat basic auth to secure /user/admin/register web service:

Warning: Run this code only once!

Note: The username and password here should be the same as the database user.

Update the values for the username and password accordingly before running the code.

This adds the following to /usr/share/tomcat/conf/tomcat-users.xml or /usr/share/tomcat7/conf/tomcat-users.xml:

```
<role rolename="admin"/>
<user username="yourAdminName" password="yourAdminPassword" roles="admin"/
↪>
```

```
For Ubuntu

sudo sed -i 's@</tomcat-users>@<role rolename="admin"/>\n<user username="'
↪"${UN}"'" password="'"${PW}"'" roles="admin"/>\n</tomcat-users>@g' /usr/
↪share/tomcat7/conf/tomcat-users.xml

For CentOS

sudo sed -i 's@<!-- <role rolename="admin"/> -->@<!-- <role rolename=
↪"admin"/> -->\n<role rolename="admin"/>\n<user username="'"${UN}"'"␣
↪password="'"${PW}"'" roles="admin"/>@g' /usr/share/tomcat/conf/tomcat-
↪users.xml
```

2. Update inactive timeout to a more reasonable number 4320 min (3 days) from default (30mins) in /usr/share/tomcat7/conf/web.xml or /etc/tomcat/web.xml

---

Note: This is modifying the following code:

```
<!--  <session-config>
        <session-timeout>30</session-timeout>
</session-config> -->
```

```
For Ubuntu

sudo sed -i 's@<session-timeout>.*</session-timeout>@<session-timeout>4320
↪</session-timeout>@g' /usr/share/tomcat7/conf/web.xml

For CentOS

sudo sed -i 's@<session-timeout>.*</session-timeout>@<session-timeout>4320
↪</session-timeout>@g' /usr/share/tomcat/conf/web.xml
```

3. Add memory constrains to Java:

> **Warning:** Run this code only once!

Note: This will add the following line to the appropriate file:

```
JAVA_OPTS=" -Xms256M -Xmx1024M -XX:PermSize=256m -XX:MaxPermSize=512m"
```

```
For Ubuntu

sudo sed -i 's@#JAVA_OPTS@JAVA_OPTS="-Xms256m -Xmx1024m -XX:PermSize=256m␣
↪-XX:MaxPermSize=512m"\n#JAVA_OPTS@g' /usr/share/tomcat7/bin/catalina.sh
```

(continues on next page)

(continued from previous page)

```
For CentOS

sudo sed -i 's@#JAVA_OPTS@JAVA_OPTS="-Xms256m -Xmx1024m -XX:PermSize=256m␣
↪-XX:MaxPermSize=512m"\n#JAVA_OPTS@g' /usr/share/tomcat/conf/tomcat.conf
```

4. Restart tomcat server:

```
For Ubuntu
sudo /usr/share/tomcat7/bin/startup.sh

For CentOS7
sudo systemctl restart tomcat
```

5. Copy database connector clients to appropriate lib directory:

```
For Ubuntu

sudo cp mysql-connector-java-6.0.6-bin.jar /usr/share/tomcat7/lib/
sudo chmod 744 /usr/share/tomcat7/lib/mysql-connector-java-6.0.6-bin.jar

For CentOS

sudo cp mariadb-java-client-1.2.0.jar /usr/share/tomcat/lib/
sudo chmod 744 /usr/share/tomcat/lib/mariadb-java-client-1.2.0.jar
```

6. Centos Only: Update the MySQL database driver to be used:

```
sed -i 's@driverClassName=.*$@driverClassName="org.mariadb.jdbc.Driver"@' $EDGE_
↪HOME/userManagement/userManagementWS.xml
```

7. Deploy userManagement to tomcat server:

Note: For CentOS the userManagementWS.xml should have:

```
driverClassName="org.mariadb.jdbc.Driver"
```

Please check and confirm this before deploying userManagement.

```
For Ubuntu

sudo rm -rf /usr/share/tomcat7/webapps/userManagementWS
sudo cp userManagementWS.war /usr/share/tomcat7/webapps/
sudo rm -rf /usr/share/tomcat7/webapps/userManagement
sudo cp userManagement.war /usr/share/tomcat7/webapps/
sudo chmod 755 /usr/share/tomcat7/webapps/*war
sudo cp userManagementWS.xml /usr/share/tomcat7/conf/Catalina/localhost/
sudo chmod 744 /usr/share/tomcat7/conf/Catalina/localhost/
↪userManagementWS.xml

For CentOS

sudo rm -rf /var/lib/tomcat/webapps/userManagementWS
sudo cp userManagementWS.war /var/lib/tomcat/webapps/
sudo rm -rf /var/lib/tomcat/webapps/userManagement
sudo cp userManagement.war /var/lib/tomcat/webapps/
```

(continues on next page)

(continued from previous page)

```
sudo chmod 755 /var/lib/tomcat/webapps/*war
sudo cp userManagementWS.xml /etc/tomcat/Catalina/localhost/
sudo chmod 744 /etc/tomcat/Catalina/localhost/userManagementWS.xml
```

8. Modify the username/password in userManagementWS.xml:

```
For Ubuntu

sudo sed -i 's@username=.*$@username="'"${UN}"'"@' /usr/share/tomcat7/conf/
↪Catalina/localhost/userManagementWS.xml
sudo sed -i 's@password=.*$@password="'"${PW}"'"@' /usr/share/tomcat7/conf/
↪Catalina/localhost/userManagementWS.xml

For CentOS

sudo sed -i 's@username=.*$@username="'"${UN}"'"@' /etc/tomcat/Catalina/localhost/
↪userManagementWS.xml
sudo sed -i 's@password=.*$@password="'"${PW}"'"@' /etc/tomcat/Catalina/localhost/
↪userManagementWS.xml
```

9. Update sys.properties in the userManagement deployment:

Note: Tomcat should automatically unarchive the .war files.

The default configuration is to have the user management system on localhost with email notifications turned off.

For "Forgot your password" reset function, the 'email_notification' should be on and a valid email address for 'email_sender'

Modify the user management sys.properties if you want to change the default behavior. (make sure port match with tomcat server)

You will need to copy the sys.properties files to the directory of the userManagement deployment.

```
For Ubuntu

sudo cp $EDGE_HOME/userManagement/sys.properties /usr/share/tomcat7/
↪webapps/userManagement/WEB-INF/classes/sys.properties
sudo chmod 744 /usr/share/tomcat7/webapps/userManagement/WEB-INF/classes/
↪sys.properties

For CentOS

sudo cp $EDGE_HOME/userManagement/sys.properties /usr/share/tomcat/
↪webapps/userManagement/WEB-INF/classes/sys.properties
sudo chmod 744 /usr/share/tomcat/webapps/userManagement/WEB-INF/classes/
↪sys.properties
```

10. Restart tomcat server:

```
For Ubuntu
sudo /usr/share/tomcat7/bin/shutdonw.sh
sudo /usr/share/tomcat7/bin/startup.sh
```

(continues on next page)

(continued from previous page)

```
For CentOS7
sudo systemctl restart tomcat
```

11. Setup admin user:

Note: The script createAdminAccount.pl creates an admin user account for EDGE userManagement.

Update email (-e), First Name (-fn), and Last Name (-ln) appropriately.

It will ask tomcat service username and password (tomcat-users.xml:) before creating EDGE user account (email).

If "HTTP Status 401" error shows, please make sure the tomcat username and password in the first step match with what entered here.

If "HTTP Status 403" error shows, please make sure the tomcat rolename in the first step match with /var/lib/tomcat/webapps/userManagementWS/WEB-INF/web.xml and where the web.xml file existed or not.

If "HTTP Status 500" error shows, please make sure the port (default: 8080) for tomcat service are matched in tomcat server.xml, $EDGE_HOME/edge_ui/sys.properties and $EDGE_HOME/userManagement/sys.properties (need to redo step 9).

If "Fatal Exception: Could not create resource factory instance during transaction connect" error shows, please make suer the tomcat userManagementWS.xml is in /etc/tomcat/Catalina/localhost/ or /usr/share/tomcat7/conf/Catalina/localhost/ and corret.

If you want to use userManagment as a remote service instead of localhost, please modify the userManagementWS.xml file to allow access from remote ip address, and corresponding mysql address.

Should this script fail, the userManagement is not set up correctly.

```
perl createAdminAccount.pl -e <email> -fn <first name> -ln <last name>
```

12. Enable userManagement in EDGE sys.properties:

Note: See *EDGE Configuration* (page 26) below

```
> sed -i 's@user_management=.*$@user_management=1@g' $EDGE_HOME/edge_ui/
↪sys.properties
> sed -i 's@edge_user_management_url=.*$@edge_user_management_url=http://
↪localhost/userManagement@g' $EDGE_HOME/edge_ui/sys.properties
```

13. Optional: configure social (facebook,google,windows live, Linkedin) login function:

- modify $EDGE_HOME/edge_ui/javascript/social.js, change apps id you created on each social media.

Note: This allow users to use their social media account to login EDGE. You need to register your EDGE's domain on each social media to get apps id. e.g.: A FACEBOOK app needs to be created and configured for the domain and website set up by EDGE. see https://developers.facebook.com/ and StackOverflow Q&A

Google+

Windows

LinkedIn

14. Optional: configure sendmail to use SMTP to email out of local domain:
    - edit /usr/share/tomcat7/conf/Catalina/localhost/userManagementWS.xml (Ubuntu) or /etc/tomcat/Catalina/localhost/userManagementWS.xml (CentOS)

      email_host=<ip or host name>
    - edit /etc/mail/sendmail.cf and edit this line:

      # "Smart" relay host (may be null) DS
    - and append the correct server right next to DS (no spaces);

      # "Smart" relay host (may be null) DSmail.yourdomain.com
    - Then, restart the sendmail service

      > sudo service sendmail restart

4.1.5 MYSQL Databases CREATION

Note: This requires that MySQL is installed and running.

Note: EDGE provides Virulence Factors, Metadata, and Pathogen sql dump files which will be used for Speciality Gene Profling module, Sample MetaData module and Pathogen Detection module, respectively. You will need configure the database info in the $EDGE_HOME/edge_ui/sys.properties. See *EDGE Configuration* (page 26) below

1. Change directory into database:

```
cd $EDGE_HOME/SQLdbfile
```

2. Run install script for databases and Grant privilege database user to have access to the databases:

```
mysql -u root -p

mysql> source virulence_db.sql ;
mysql> GRANT ALL PRIVILEGES ON virulenceFactors.* to 'yourDBUsername'@'localhost';

mysql> create database edgeDB;
mysql> use edgeDB;
mysql> source edge_db.sql ;
mysql> GRANT ALL PRIVILEGES ON edgeDB.* to 'yourDBUsername'@'localhost';

mysql> create database pathogens ;
mysql> use pathogens;
mysql> source pathogen_db.sql ;
mysql> GRANT ALL PRIVILEGES ON pathogens.* to 'yourDBUsername'@'localhost';
mysql> exit;
```

3. Configure Virulence, Metadata and Pathogen Database information:

```
Edit $EDGE_HOME/edge_ui/sys.properties with the appropriate database username and␣
↪password.

# Virluence Factors database
VFDB_dbhost = localhost
VFDB_dbport = 3306
VFDB_dbname = virulenceFactors
VFDB_dbuser = edge_user
VFDB_dbpasswd = edge_user_password

##configure edge pathogen detection 1: with 0: without
edge_pathogen_detection=0
pathogen_dbhost=localhost
pathogen_dbname=pathogens
pathogen_dbuser=edge_user
pathogen_dbpasswd=edge_user_password

##configure edge sample metadata option 1: with 0: without
edge_sample_metadata=0
edge_dbhost=localhost
edge_dbname=edgeDB
edge_dbuser=edge_user
edge_dbpasswd=edge_user_password
```

4.1.6 EDGE configuration

Note: EDGE system configuration file is $EDGE_HOME/edge_ui/sys.proprties. You can edit this file to turn on/off EDGE functions/modules here. (on=1, off=0);

1. Add EDGE GUI admin info:

#According to User Management system installation step 11:

```
edgeui_admin=admin@my.com
edgeui_admin_password=admin
```

2. Turn on user management system:

   Note: This assumes localhost is the domain. Update the domain as necessary. If user management system is not in the same domain with EDGE.:

   ```
   edge_user_management_url=http://www.someother.com/userManagement
   ```

   ```
   # If you have User Management system enabled.
   user_management=1
   edge_user_management_url=http://localhost/userManagement
   ```

3. Turn on upload function:

   ```
   user_upload=1
   user_upload_maxFileSize='5gb'
   ```

4. Turn on project intermediate files clean up:

```
#Clean up old bam/sam/fastq/gz files (based on file age) in project directories
edgeui_proj_store_days=10
```

5. Set up the archive directory:

```
#The archive space is for offload the main computational disk space
edgeui_archive=/path/to/archive_SPACE
```

7. Adjust number of CPUs assigned to EDGE and number of job able to run simultaneously:

```
edgeui_tol_cpu=4
max_num_jobs=2
```

8. Turn on/off Social Login function:

```
#If you have User Management system installation step 18 done.
user_social_login=0
```

9. Turn on job submission for SGE/UGE cluster environment:

Note: make sure the user/apache user running EDGE is a cluster user.

`qconf -suserl` to check cluster user list

Edit the sge_bin, sge_root and sge_cell corresonding to your cluster environment

---

```
#Configure cluster system  1: with 0: without
cluster=1

## sge environment configuration
sge_bin=/cm/shared/apps/sge/2011.11p1/bin/linux-x64
sge_root=/cm/shared/apps/sge/2011.11p1
sge_cell=default

## edge job submission configuration
cluster_job_notify=edge@yourdomain.com
cluster_job_prefix=EDGE_pipeline_
cluster_qsub_options=
cluster_job_resource=h_vmem=6G -pe smp <CPU> -binding linear:<CPU/2>
cluster_job_max_cpu=64
```

4.2 Configure SELinux on CentOS

Warning: This is not complete.

1. Install semanage (if not already installed):

```
> sudo yum install -y policycoreutils-python setroubleshoot
```

2. Allow httpd to access $EDGE_HOME, the databases, and read/write to the EDGE_input/EDGE_output:

```
> sudo semanage fcontext -a -t httpd_sys_content_t "$EDGE_HOME(/.*)?"
> sudo semanage fcontext -a -t httpd_sys_content_t "$EDGE_HOME/database(/.*)?"
> sudo semanage fcontext -a -t httpd_sys_content_t "$EDGE_HOME/edge_ui/EDGE_
↪input(/.*)?"
> sudo semanage fcontext -a -t httpd_sys_content_t "$EDGE_HOME/edge_ui/EDGE_
↪output(/.*)?"
```

3. Allow httpd to execute cgi-scripts in $EDGE_HOME/edge_ui/cgi-bin/:

```
> sudo semanage boolean -m --on httpd_enable_cgi
> sudo semanage fcontext -a -t httpd_sys_script_exec_t "$EDGE_HOME/edge_ui/cgi-
↪bin(/.*)?"
```

4. Allow httpd to connect to mysql database:

```
> sudo semanage boolean -m --on httpd_can_network_connect_db
```

5. Optional: Allow httpd to work with nfs and send mail:

```
> sudo semanage boolean -m --on httpd_use_nfs
> sudo semanage boolean -m --on httpd_can_sendmail
```

6. REQUIRED: Apply the rules:

```
> sudo restorecon -R $EDGE_HOME
> sudo restorecon -R $EDGE_HOME/database/
> sudo restorecon -R $EDGE_HOME/edge_ui/EDGE_input/
> sudo restorecon -R $EDGE_HOME/edge_ui/EDGE_output/
```

4.3 EDGE Docker image

EDGE has a lot of dependencies and can (but doesn't have to) be very challenging to install. The EDGE docker gets around the difficulty of installation by providing a functioning EDGE full install on top of offical Ubuntu Base Image (18.04.2). You can find the image and usage at docker hub. We would recommend to use Docker container for easy update in the future.

CHAPTER 5

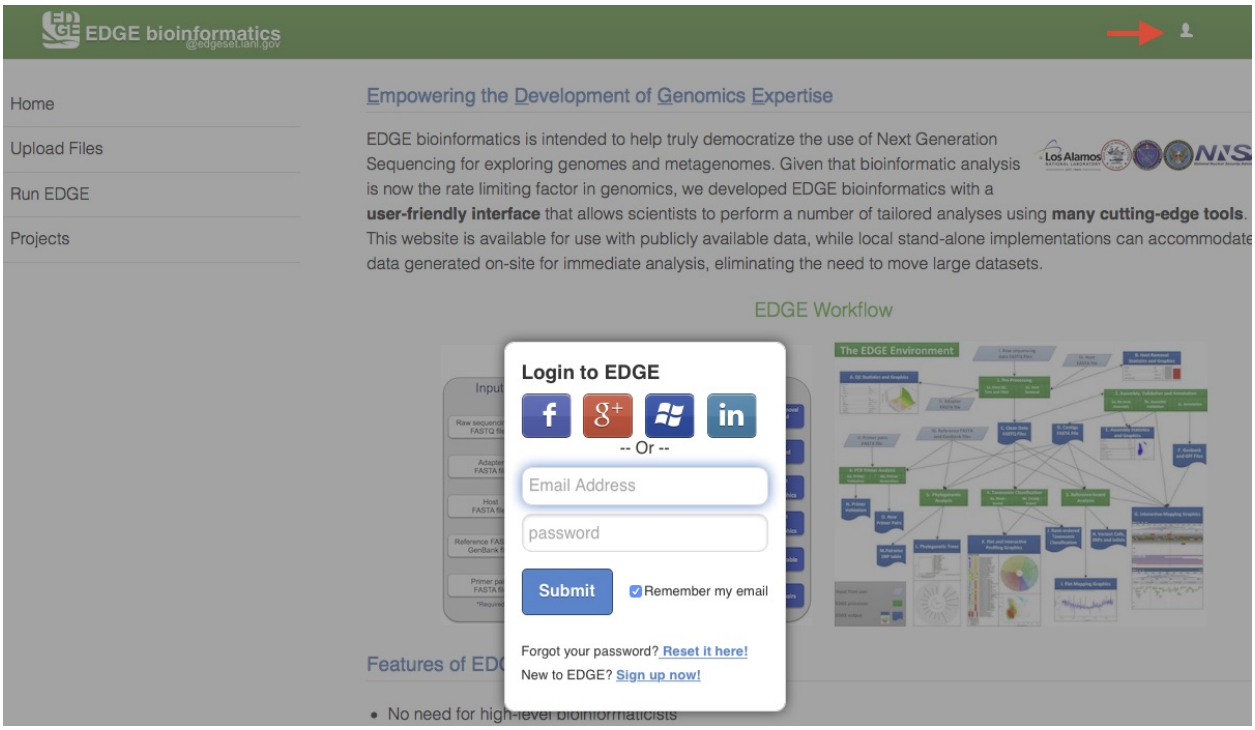
Graphic User Interface (GUI)

The User Interface was mainly implemented in JQuery Mobile, CSS, javascript and perl CGI. It is a HTML5-based user interface system designed to make responsive web sites and apps that are accessible on all smartphone, tablet and desktop devices. (see *How to make an app icon on the mobile device* (page 91))

See GUI page

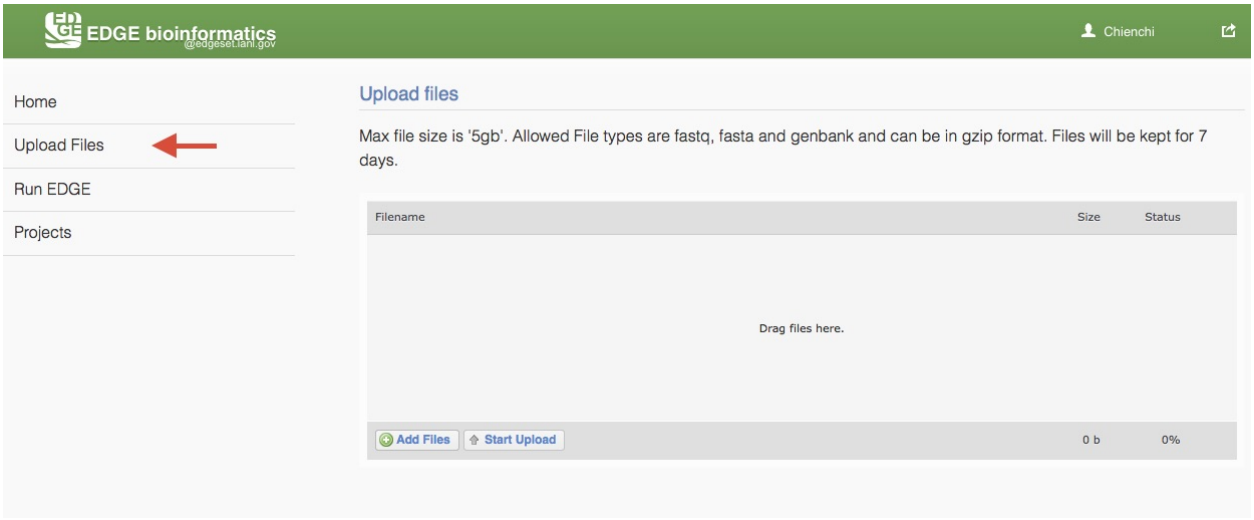
5.1 User Login

A user management system has been implemented to provide a level of privacy/security for a user's submitted projects. When this system is activated, any user can view projects that have been made public, but other projects can only be accessed by logging into the system using a registered local EDGE account or via an existing social media account (Facebook, Google+, Windows, or LinkedIn). The users can then run new jobs and view their own previously run projects or those that have been shared with them. Click on the upper-right user icon will pop up an user login window.



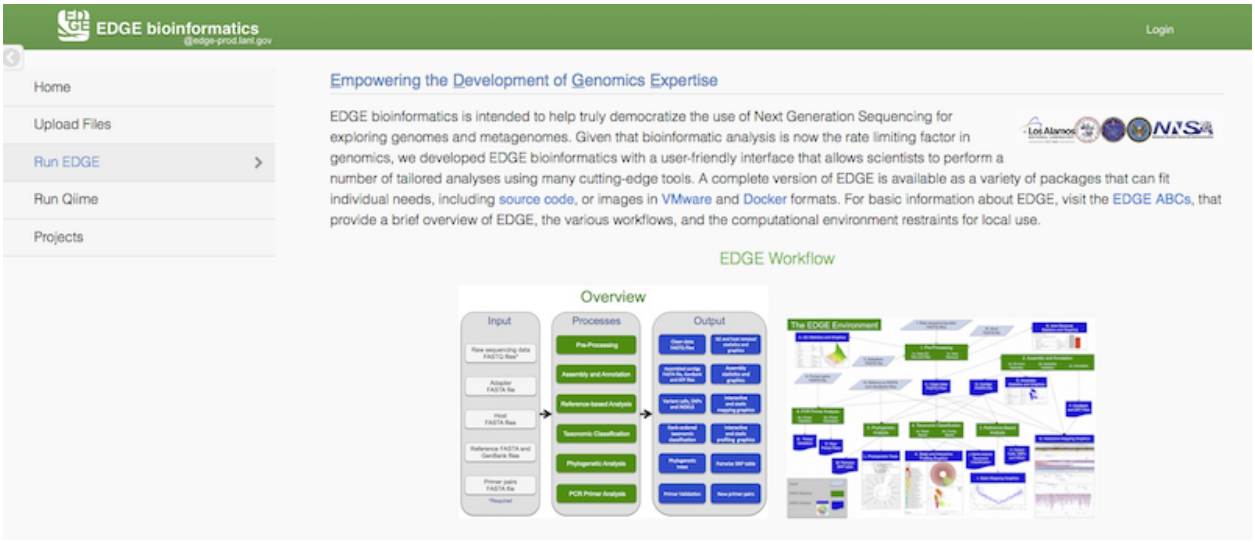
5.2 Upload Files

EDGE supports input from NCBI Sequence Reads Archive (SRA) and select files from the EDGE server. To analyze users' own data, EDGE allows user to upload fastq, fasta and genbank (which can be in gzip format) and text (txt). Max file size is '5gb' and files will be kept for 7 days. Choose "Upload files" from the navigation bar on the left side of the screen. Add users files by clicking "Add Files" buttion or drag files to the upload feature window. Then, click "Start Upload" button to upload files to EDGE server.



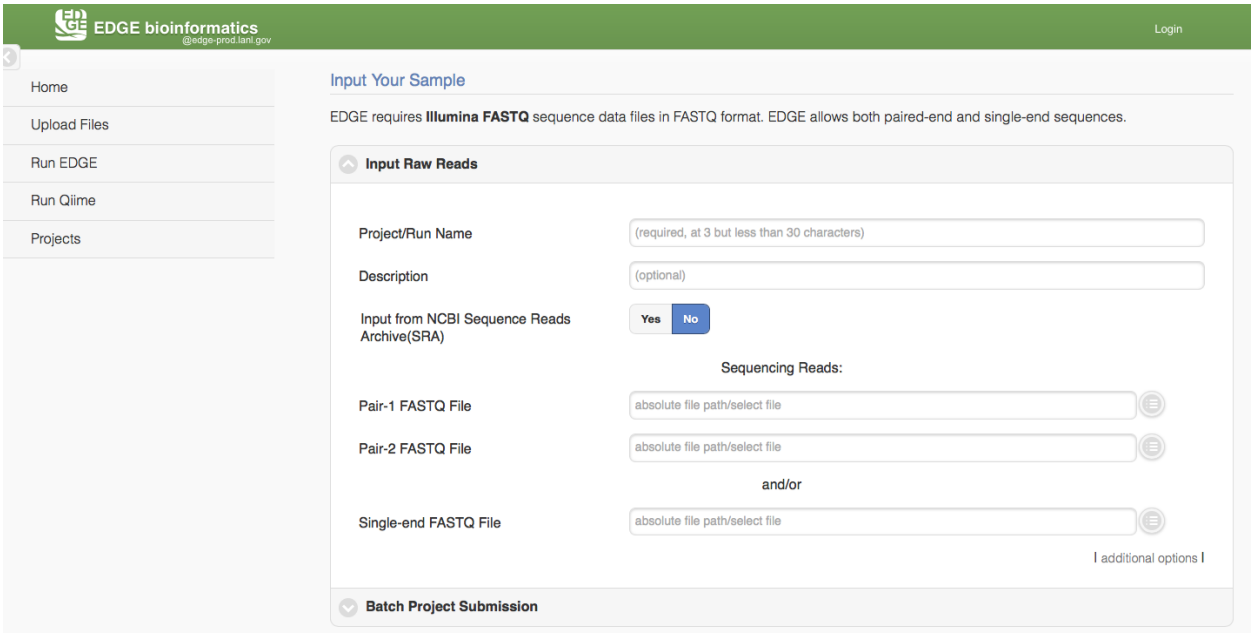
5.3 Initiating an analysis job

Choose “Run EDGE” or “Run Qiime” from the navigation bar on the left side of the screen.



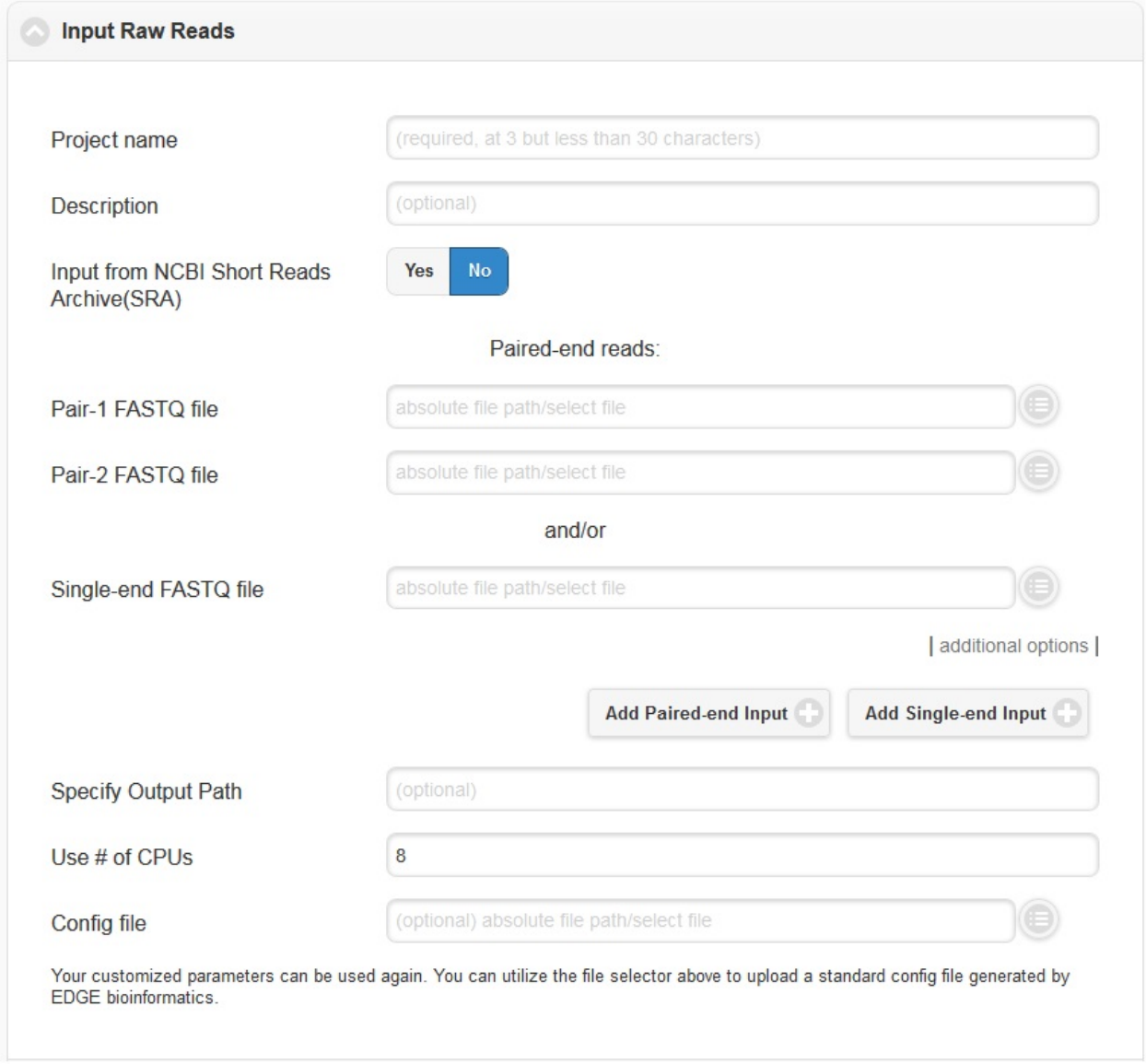
5.3.1 Run EDGE

Click “Run EDGE” will cause a section to appear called “Input Raw Reads.” Here, you may browse the EDGE Input Directory and select FASTQ files containing the reads to be analyzed. EDGE supports gzip compressed fastq files. At minimum, EDGE will accept two FASTQ files containing paired reads and/or one FASTQ file containing single reads as initial input. Alternatively, rather than providing files through the EDGE Input Directory, you may decide to use as input reads from the Sequence Read Archive (SRA). In this case, select the “yes” option next to “Input from NCBI Sequence Reads Archive” and a field will appear where you can type in an SRA accession number.



In addition to the input read files, you have to specify a project name. The project name is restricted to only alphanumerical characters and underscores and requires a minimum of three characters. For example, a project name of “E.

coli. Project" is not acceptable, but a project name of "E_coli_project" could be used instead. In the "Description" fields you may enter free text that describes your project. If you would like, you may use as input more reads files than the minimum of 2 paired read files or one file of single reads. To do so, click "additional options" to expose more fields, including two buttons for "Add Paired-end Input" and "Add Single-end Input".

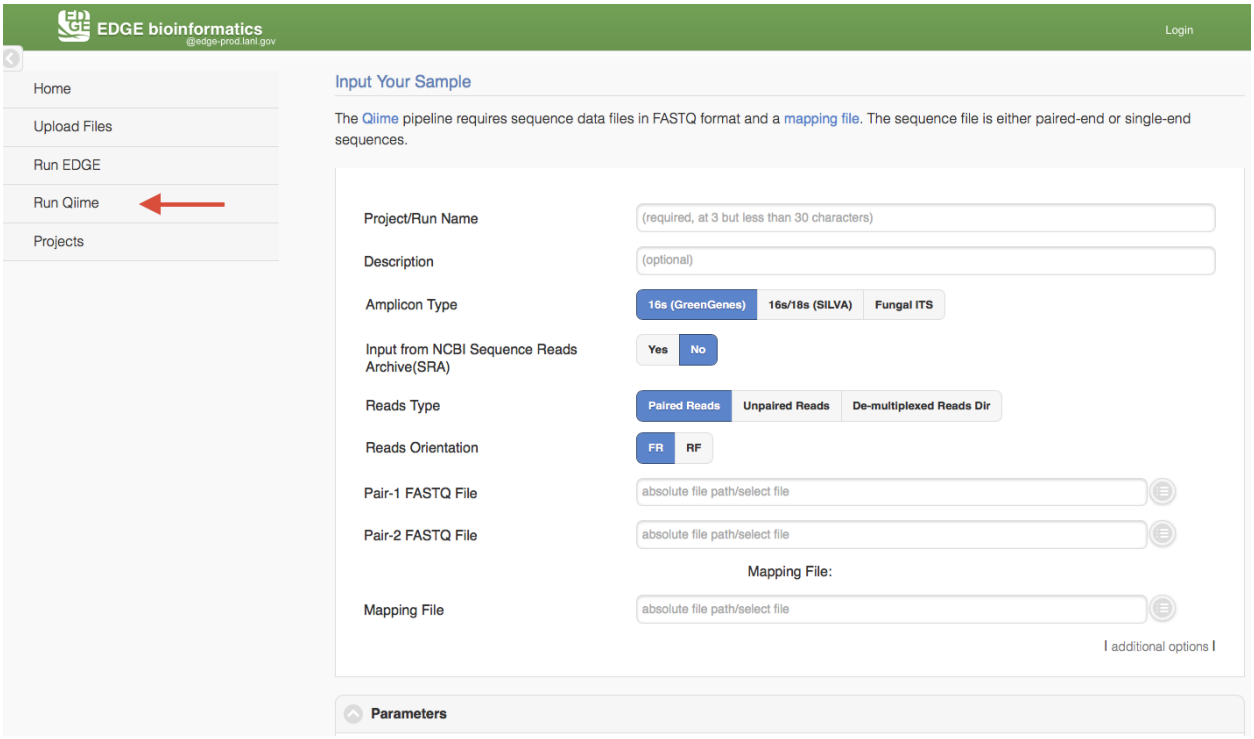


In the "additional options", there are several more options, for output path, number of CPUs, and config file. In most cases, you can ignore these options, but they are described briefly below.

5.3.2 Run Qiime

Click "Run Qiime2" will cause a section to appear for Qiime input and parameters. Currently, EDGE suports four amplicon types, 16s using GreenGenes database, 16s/18s using SILVA database, 16s V3-V4 341F/805R (SILVA) and Fungal ITS. Similar to "Run EDGE", input can be browse the EDGE Input Directory based on the reads type. The Qiime pipeline support one Reads Type in a run, paired-reads, single end reads, or de-multiplexed reads directory. There is also a mapping file input requirment which is adapted from QIIME Metadata mapping file. This mapping file contains all of the information about the samples necessary to perform the data analysis. It is in tab-delimited format

or EXCEL (.xlsx) file. In general, the header for this mapping file starts with a pound (#) character, and generally requires a "SampleID", "BarcodeSequence", and a "Description". (Cap does matter) Users can put more meaningful metadata fields for analysis between #SampleID and Description column, ex BodySite, age, date, location, etc.



Mapping File Example:

#SampleID	BarcodeSequence	SampleType	Description
Sample1	ACATACCGTCTA	Stool	MiSeq,metagenome
Sample2	ACCATGCGTCTA	Blood	MiSeq,clinical
Control1	AGCCATCGTCTA	Control	Negative
Control2	CGTCTAACCATG	Control	Spike-in Control

When the reads type is "De-multiplexed Reads Directory ", the mapping file needs a 'Files' column with **FASTQ** filenames for each sampleID. It can be paired-end or single-end FASTQ file where paired-end FASTQ files are comma-separated.

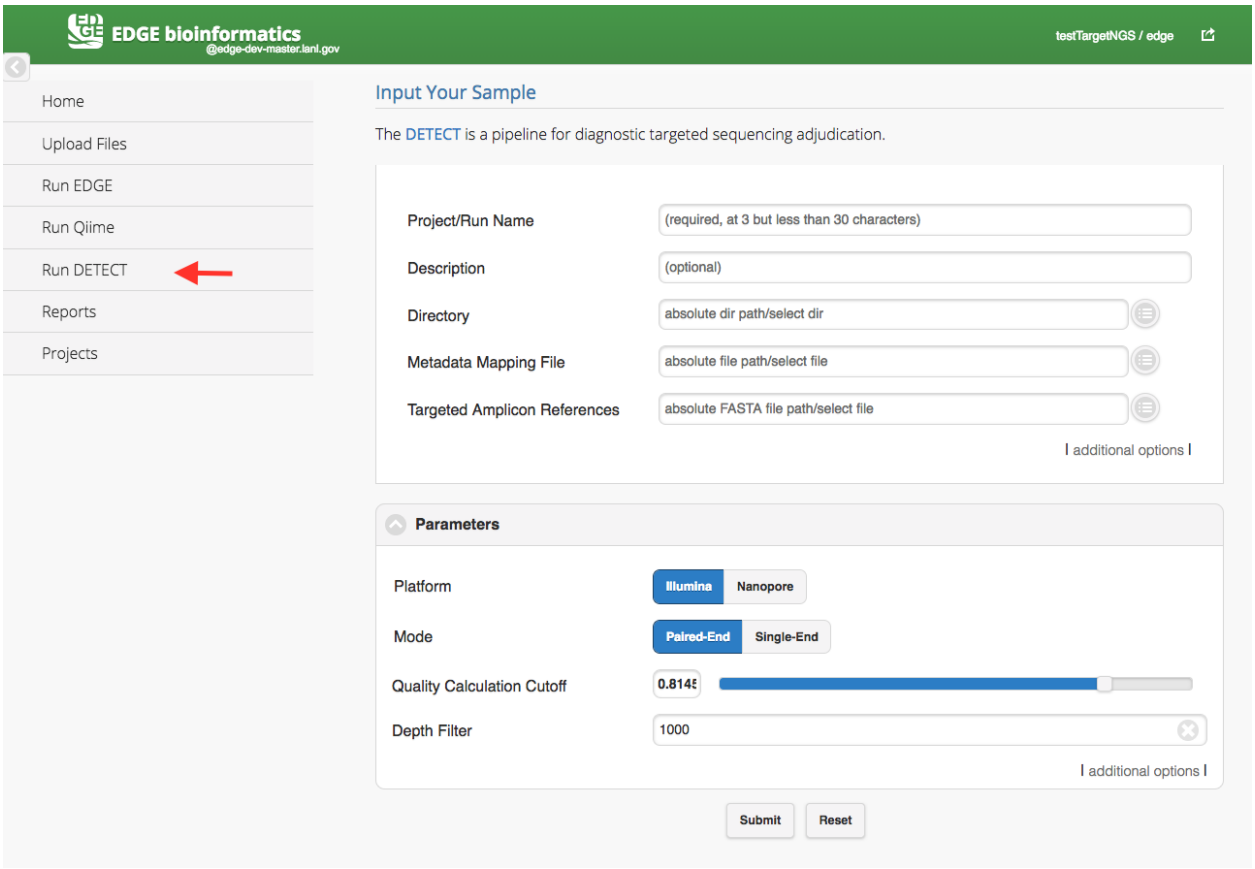
#SampleID	Files	SampleType	Date	Age	Description
Sample1	S1.R1.fastq,S1.R2.fastq	Stool	2019-05-10	60	MiSeq,metagenome
Sample2	S2.R1.fastq,S2.R2.fastq	Blood	2019-08-10	70	MiSeq,clinical
Control1	C1.R1.fastq,C1.R2.fastq	Control	2019-09-01	45	Negative
Control2	C2.R1.fastq,C2.R2.fastq	Control	2019-08-10	50	Spike-in Control

Note: example metadata: https://data.qiime2.org/2020.2/tutorials/moving-pictures/sample_metadata.tsv

While QIIME 2 does not enforce standards for what types of metadata to collect, the MIMARKS standard provides recommendations for microbiome studies and may be helpful in determining what information to collect in your study. If you plan to deposit your data in a data archive (e.g. ENA or Qiita), it is also important to determine the types of metadata expected by the archive, as each archive may have its own requirements.

5.3.3 Run DETEQT

Click "Run DETEQT" will cause a section to appear for DETEQT input and parameters. The DETEQT is a pipeline for diagnostic targeted sequencing adjudication. You may find more information from here. The DETEQT pipeline required user to select a directory, a metadata mapping file and a targeted amplicon references. The metadata mapping file is a tab-delimited file or excel file which header or first row includes #SampleID and Files. (Cap does matter) In the Files column, the paired-end fastq files are separated by a comma(,) and all the fastq files should be located in the input directory. The reference is comprised of only target regions in FASTA format in the assay.

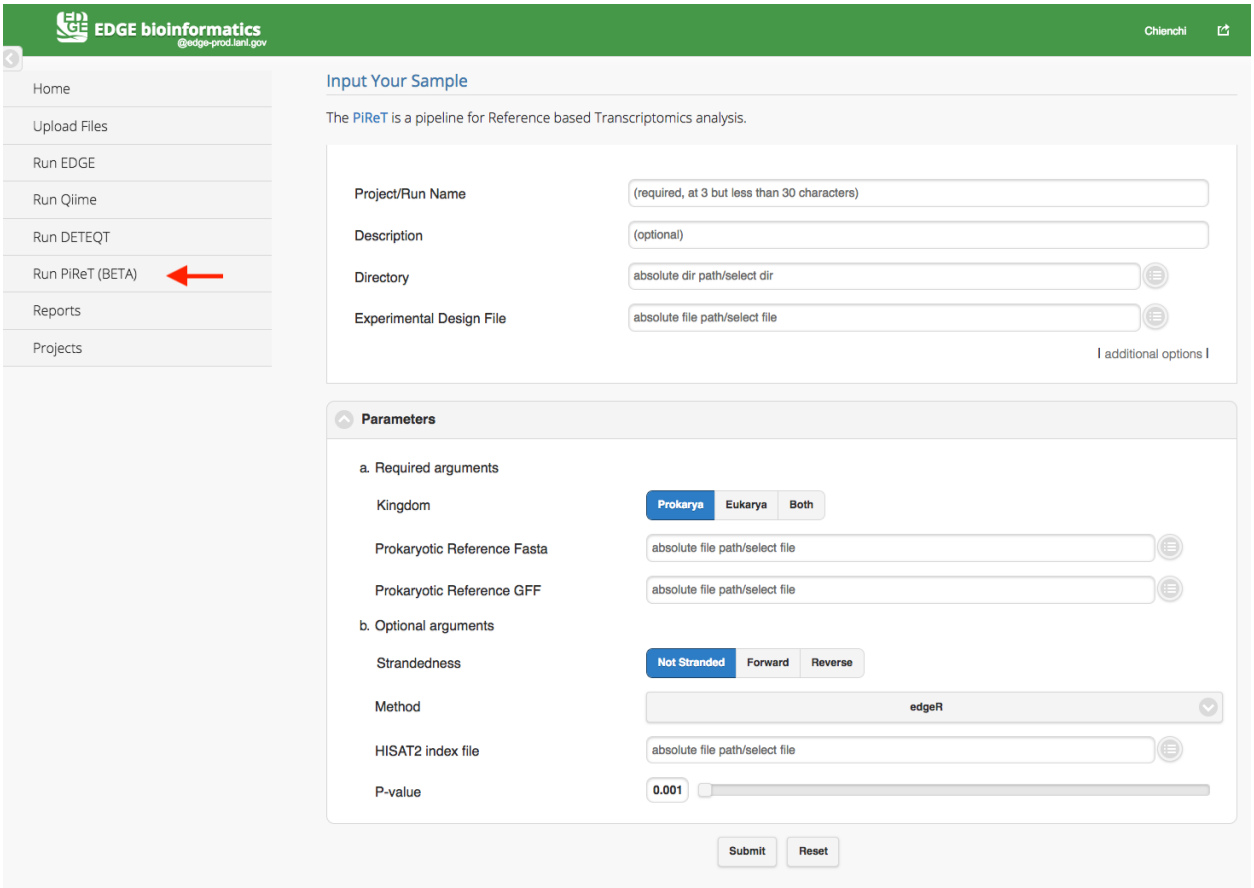


Metadata Mapping File example:

#SampleID	Files
Dengue	sample.1.fq,sample.2.fq
Flu	flu.1.fq,flu.2.fq
Ebola	ebola.1.fq,ebola.2.fq
MERS	mers.1.fq,mers.2.fq
SARS	sars.1.fq,sars.2.fq
Zika	zika.1.fq,zika.2.fq
Rota	rota.1.fq,rota.2.fq
HIV	hiv.1.fq,hiv.2.fq
Hanta	hanta.1.fq,hanta.2.fq
HCV	hcv.1.fq,hcv.2.fq

5.3.4 Run PiReT

Click "Run PiReT" will cause a section to appear for PiReT input and parameters. The PiReT is a pipeline for Reference based Transcriptomics analysis. You may find more information from PiReT github. The PiReT pipeline required user to select a directory, a experimental design file and references FASTA and GFF files in the parameters section. The experimental file is a tab-delimited file or excel file which header or first row includes #SampleID, Files, and Group. (Cap does matter) In the Files column, the paired-end fastq files are separated by a colon(:) and all the fastq files should be located in the input directory. The feature ID in the reference GFF files should be uniqe within the scope of the GFF file.



Experimental Design File example:

#SampleID	Files	Group
samp1	samp1_R1.fastq:samp1_R2.fastq	liver
samp2	samp2_R1.fastq:samp2_R2.fastq	spleen
samp3	samp3_R1.fastq:samp3_R2.fastq	spleen
samp4	samp4_R1.fastq:samp4_R2.fastq	liver
samp5	samp5_R1.fastq:samp5_R2.fastq	liver
samp6	samp6_R1.fastq:samp6_R2.fastq	spleen

5.3.5 Number of CPUs

Additionally, you may specify the number of CPUs to be used. The default and minimum value is one-fourth of total number of server CPUs. You may adjust this value if you wish. Assuming your hardware has 64 CPUs, the default is

16 and the maximum you should choose is 62 CPUs. Otherwise, if the jobs currently in progress use the maximum number of CPUs, the new submitted job will be queued (and colored in grey. Color-coding see *Checking the status of an analysis job* (page 44)). For instance, if you have only one job running, you may choose 62 CPUs. However, if you are planning to run 6 different jobs simultaneously, you should divide the computing resources (in this case, 10 CPUs per each job, totaling 60 CPUs for 6 jobs).

5.3.6 Config file

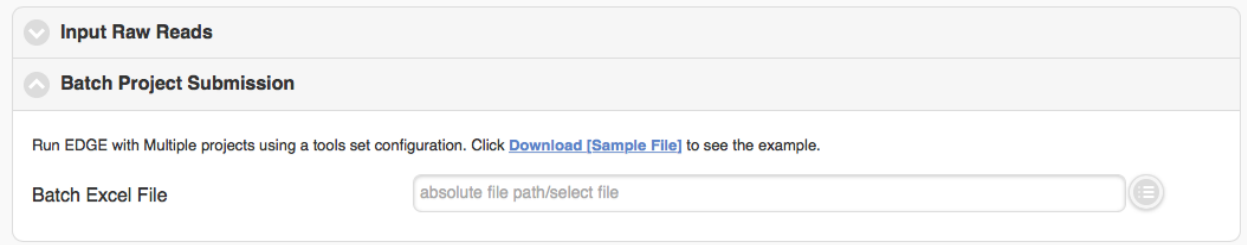
Below the "Use # of CPUs" field is a field where you may select a configuration file. A configuration file is automatically generated for each job when you click "Submit." This field could be used if you wanted to restart a job that hadn't finished for some reason (e.g. due to power interruption, etc.). This option ensures that your submission will be run exactly the same way as previously, with all the same options.

See also:

Example of config file (page 52)

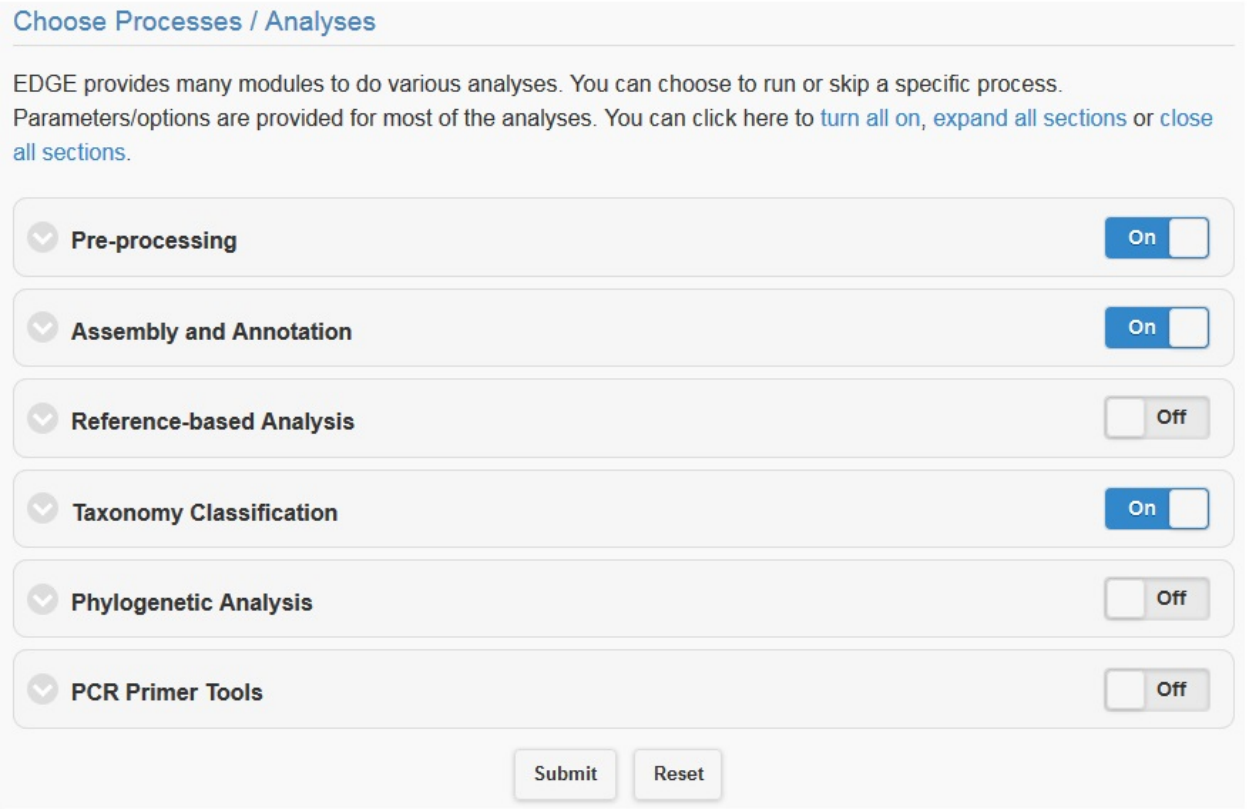
5.3.7 Batch project submission

The "Batch project submission" section is toggled off by default. Clicking on it will open it up and toggle off the "Input Sequence" section at the same time. When you have many samples in "EDGE Input Directory" and would like to run them with the same configuration, instead of submitting several times, you can compile a Excel file with project name, fastq inputs and optional project descriptions (you can download the example excel file and fill it with your own data) and submit through the "Batch project submission" section



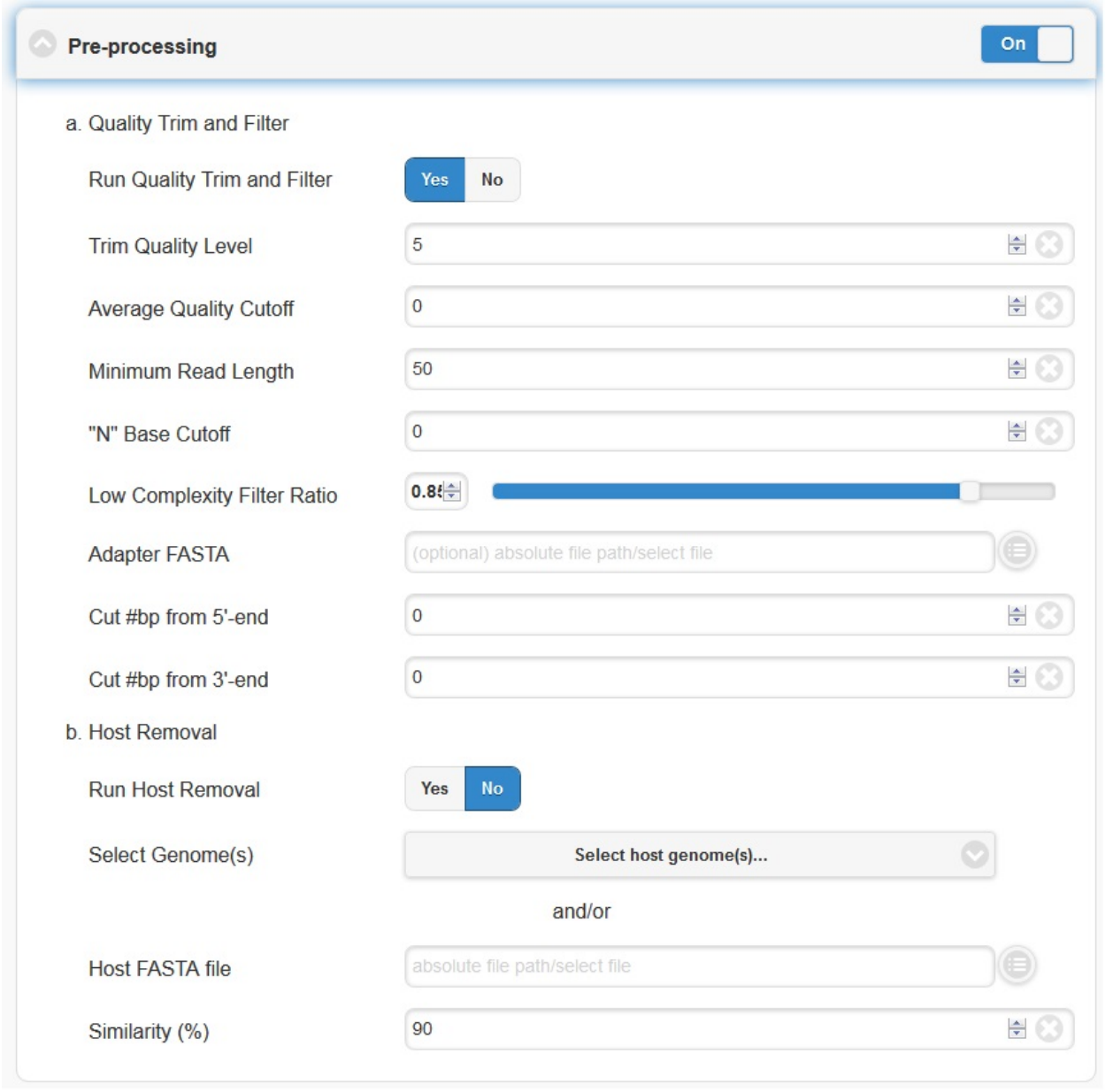
5.4 Choosing processes/analyses

Once you have selected the input files and assigned a project name and description, you may either click "Submit" to submit an analysis job using the default parameters, or you may change various parameters prior to submitting the job. The default settings include quality filter and trimming, assembly, annotation, and community profiling. Therefore, if you choose to use default parameters, the analysis will provide an assessment of what organism(s) your sample is composed of, but will not include host removal, primer design, etc. Below the "Input Your Sample" section is a section called "Choose Processes / Analyses". It is in this section that you may modify parameters if you would like to use settings other than the default settings for your analysis (discussed in detail below).



5.4.1 Pre-processing

Pre-processing is by default on, but can be turned off via the toggle switch on the right hand side. The default parameters should be sufficient for most cases. However, if your experiment involves specialized adapter sequences that need to be trimmed, you may do so in the Quality Trim and Filter subsection. There are two options for adapter trimming. You may either supply a FASTA file containing the adapter sequences to be trimmed, or you may specify N number of bases to be trimmed from either end of each read.

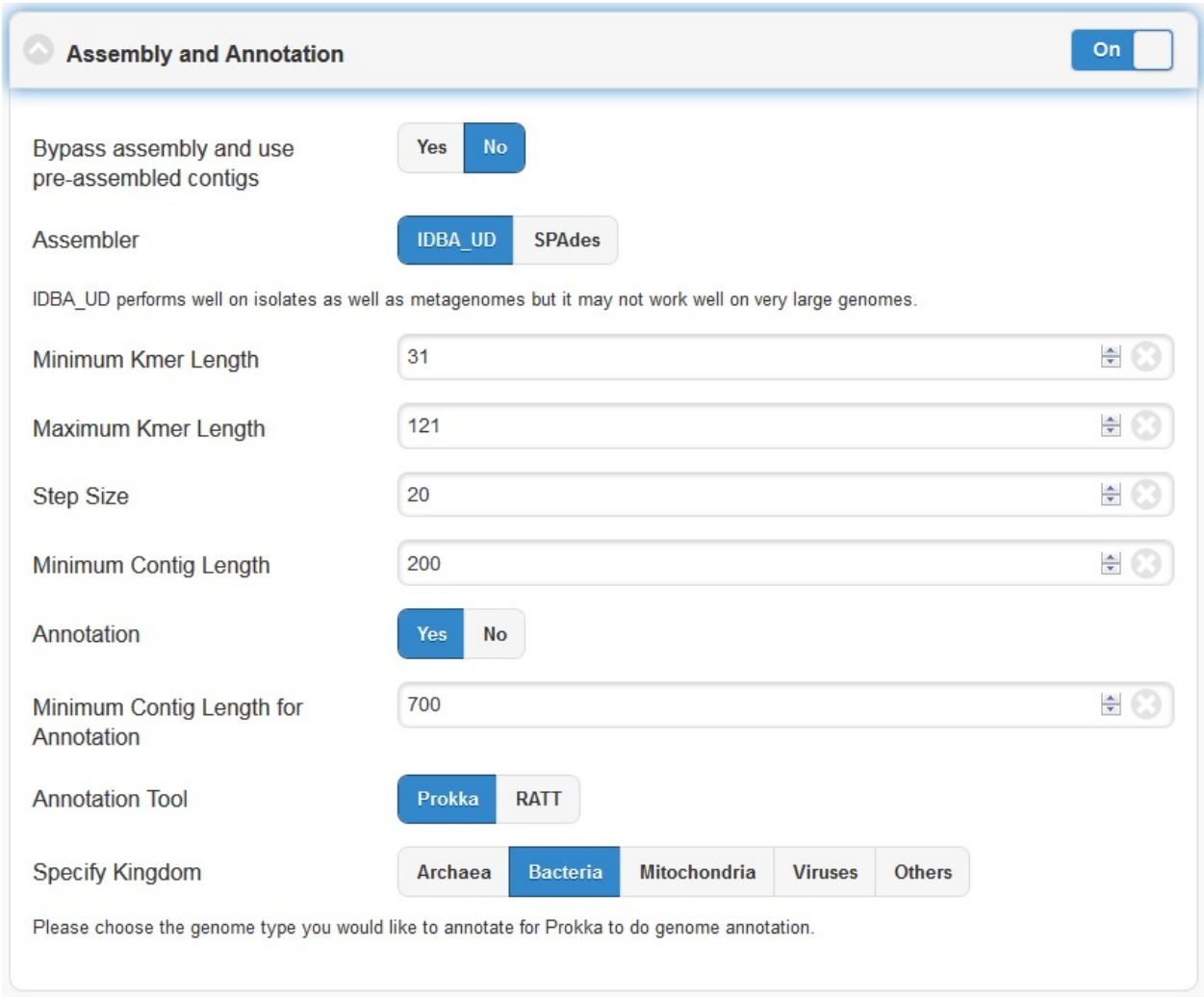


Note: Trim Quality Level can be used to trim reads from both ends with defined quality. "N" base cutoff can be used to filter reads which have more than this number of continuous base "N". Low complexity is defined by the fraction of mono-/di-nucleotide sequence. Ref: FaQCs.

The host removal subsection allows you to subtract host-derived reads from your dataset, which can be useful for metagenomic (complex) samples such as clinical samples (blood, tissue), or environmental samples like insects. In order to enable host removal, within the "Host Removal" subsection of the "Choose Processes / Analyses" section, switch the toggle box to "On" and select either from the pre-build host list (Human , Invertebrate Vectors of Human Pathogens , PhiX , RefSeq Bacteria and RefSeq Viruses .) or the appropriate host FASTA file for your experiment from the navigation field. The Similarity (%) can be varied if desired, but the default is 90 and we would not recommend using a value less than 90.

5.4.2 Assembly And Annotation

The Assembly option by default is turned on. It can be turned off via the toggle button. EDGE performs iterative kmers de novo assembly by IDBA-UD . It performs well on isolates as well as metagenomes but it may not work well on very large genomes. By default, it starts from kmer=31 and iterative step by adding 20 to maximum kmer=121. When the maximum k value is larger than the input average reads length, it will automatically adjust the maximum value to average reads length minus 1. User can set the minimum cutoff value on the final contigs. By default, it will filter out all contigs with size smaller than 200 bp.

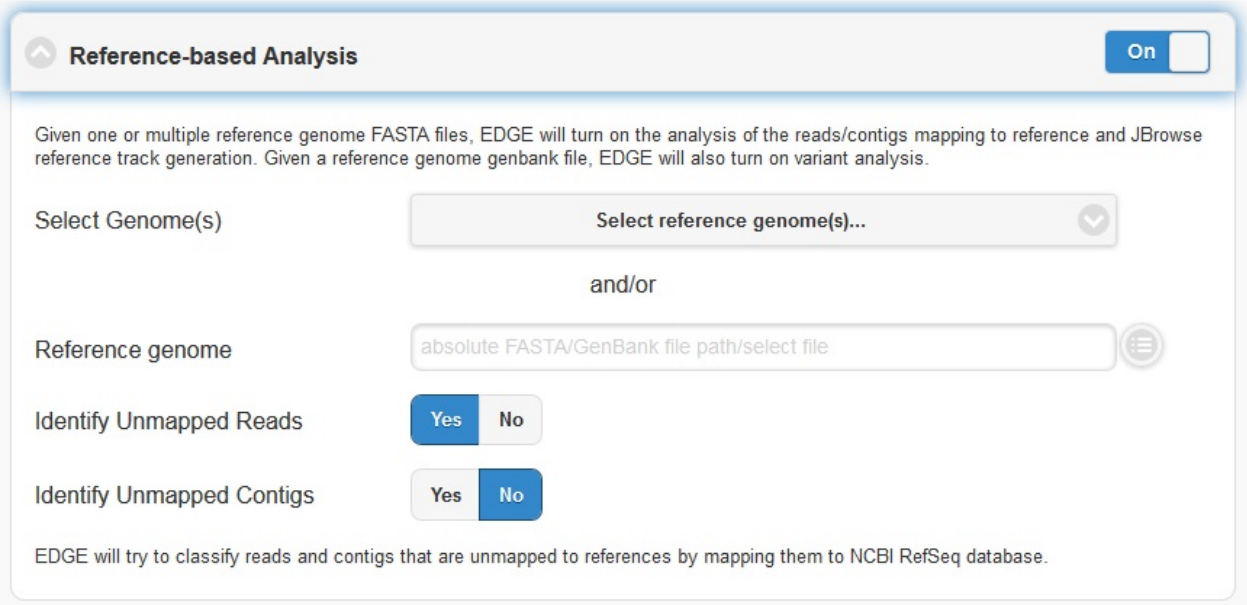


The Annotation module will be performed only if the assembly option is turned on and reads were successfully assembled. EDGE has the option of using Prokka or RATT to do genome annotation. For most cases, Prokka is the appropriate tool to use, however, if your input is a viral genome with attached reference annotation (GenBank file), RATT is the preferred method. If for some reason the assembly fails (ex: run out of Memory), EDGE will bypass any modules requiring a contigs file including the annotation analysis.

5.4.3 Reference-based Analysis

The reference-based analysis section allows you to map reads/contigs to the provided references, which can be useful for known isolated species such as cultured samples, to get the coverage information and validate the assembled contigs. In order to enable reference-based analysis, switch the toggle box to “On” and select either from the pre-

build Reference list (*Ebola virus genomes* (page 76) , E.coli 55989 , E.coli O104H4 , E.coli O127H6 and E.coli K12 MG1655 .) or the appropriate FASTA/GenBank file for your experiment from the navigation field.



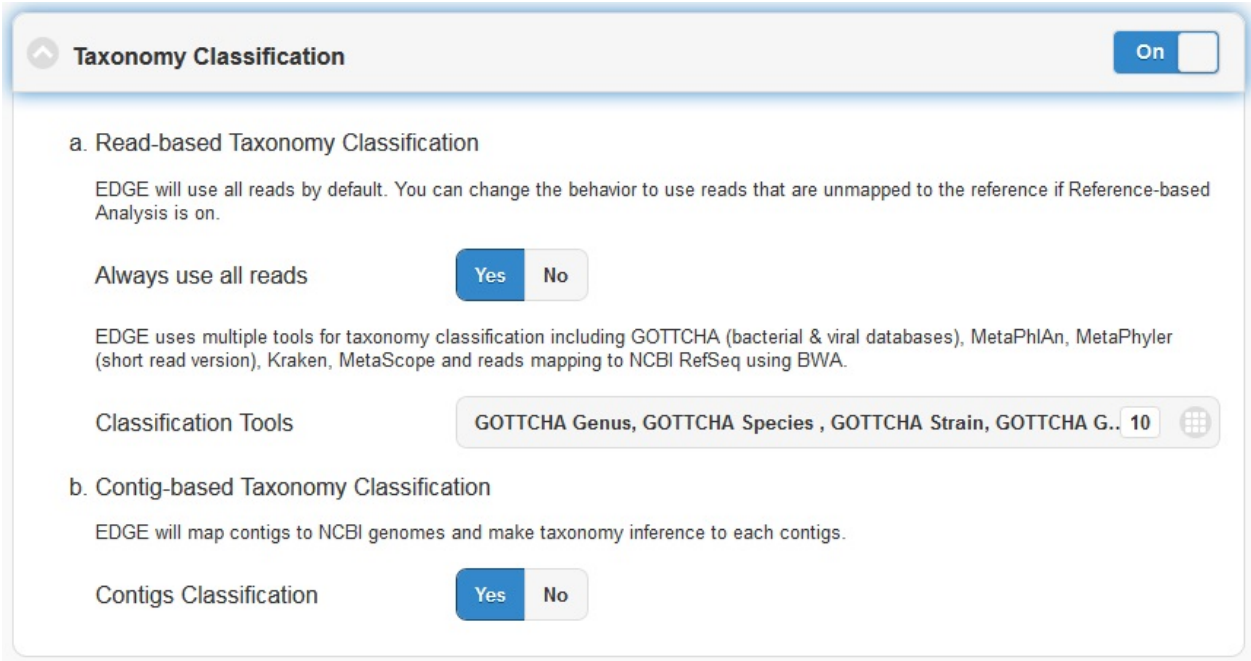
Given a reference genome fasta file, EDGE will turn on the analysis of the reads/contigs mapping to reference and JBrowse reference track generation. If a GenBank file is provided, EDGE will also turn on variant analysis.

Note: If there are more than one sequence in the reference genome fasta (mulit >), the fasta header mush have unique id for each sequence which is defined in the beginning non space words. ex: >unique_id any other annotation

---

5.4.4 Taxonomy Classification

Taxonomic profiling is performed via the "Taxonomy Classification" feature. This is a useful feature not only for complex samples, but also for purified microbial samples (to detect contamination). In the "Community profiling" subsection in the "Choose Processes / Analyses section," community profiling can be turned on or off via the toggle button.

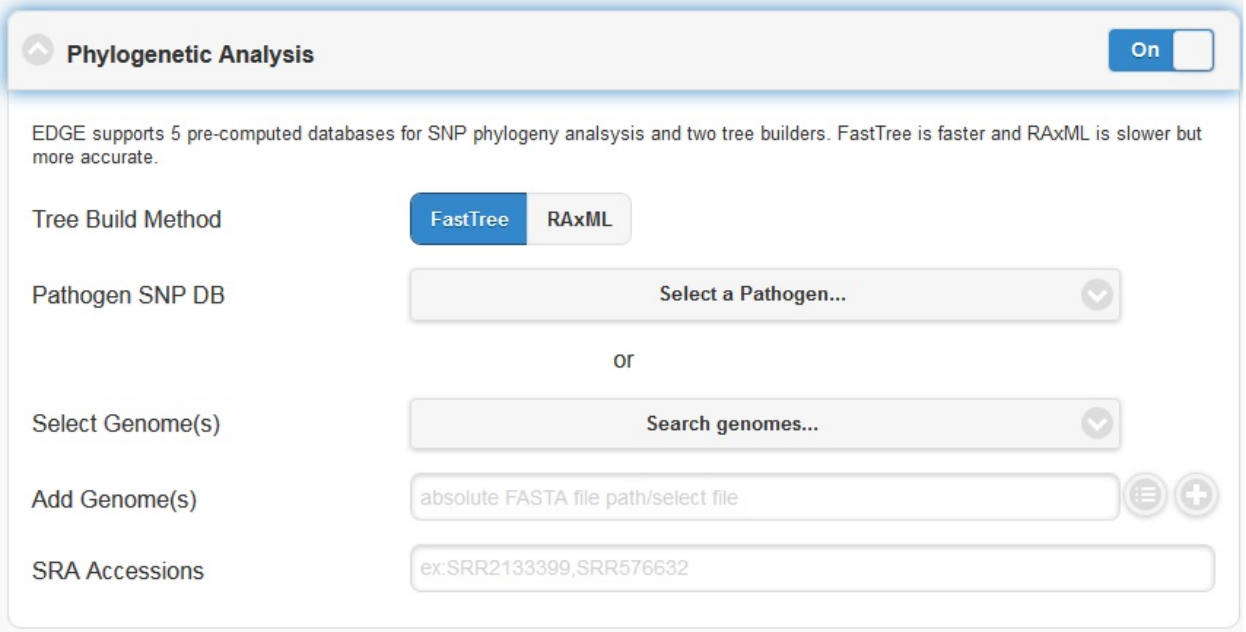


There is an option to "Always use all reads" or not. If "Always use all reads" is not selected, then only those reads that do not map to the user-supplied reference will be shown in downstream analyses (i.e. the results will only include what is different from the reference). Additionally, the user can use different profiling tools with checkbox selection menu. EDGE uses multiple tools for taxonomy classification including GOTTCHA (bacterial & viral databases) , MetaPhlAn , Kraken and reads mapping to NCBI RefSeq using BWA .

Turning on the "Contig-Based Taxonomy Classification" section will initiate mapping contigs against NCBI databases for taxonomy and functional annotations.

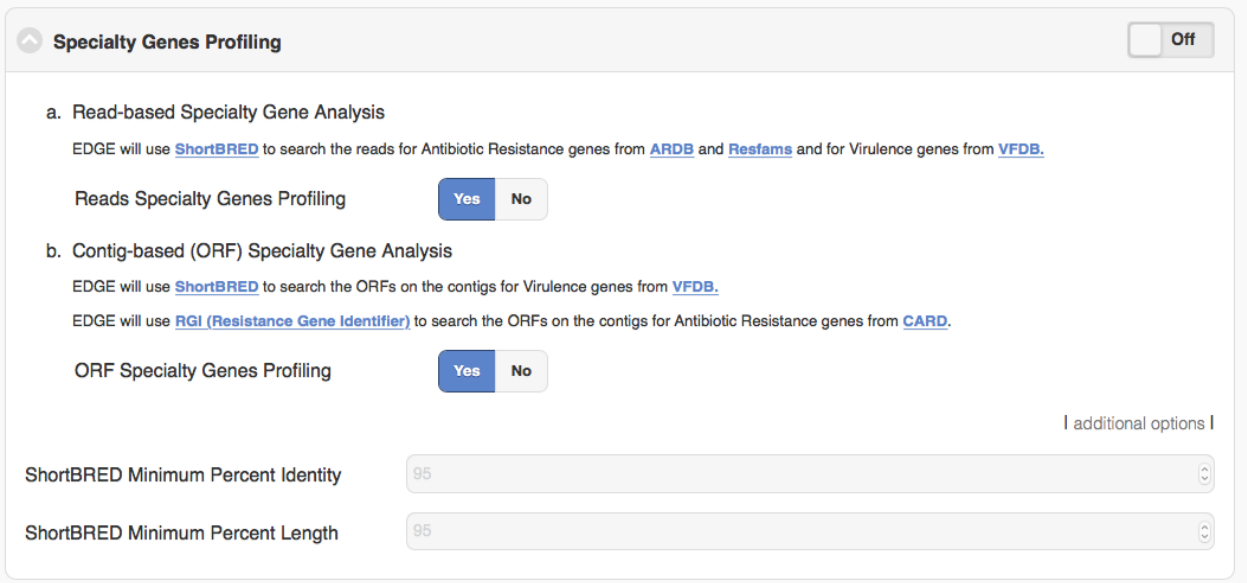
5.4.5 Phylogenomic Analysis

EDGE supports 5 pre-computed pathogen databases (*E.coli, Yersinia, Francisella, Brucella, Bacillus* (page 69)) for SNP phylogeny analysis. You can also choose to build your own database by first selecting a build method (either FastTree or RAxML), then selecting a pathogen from the "Search Genomes" search function. You can also add FASTA files or SRA Accessions.



5.4.6 Specialty Genes Profiling

For specialty gene analysis, the user selects read-based analysis and/or ORF(contig)-based analysis.

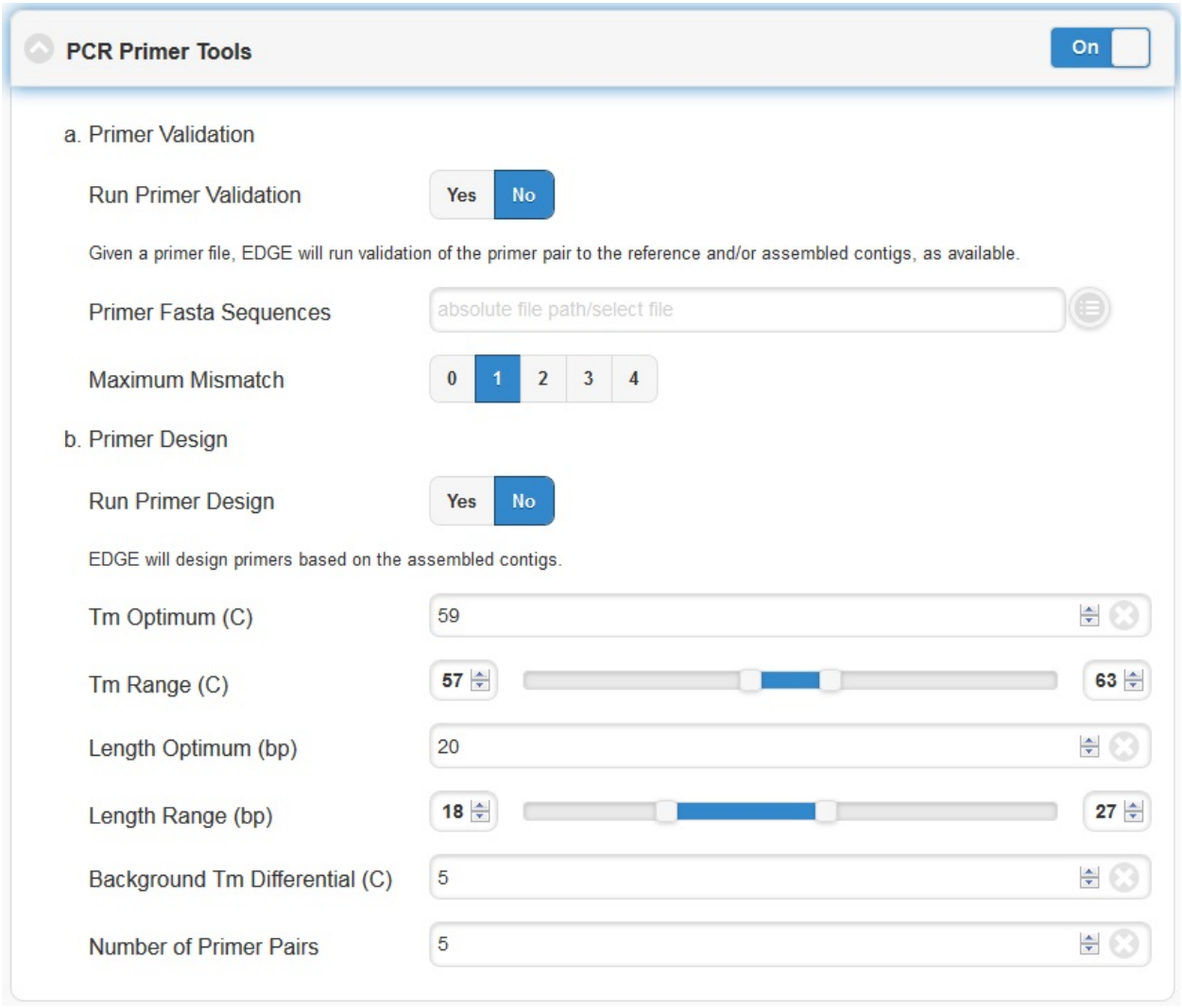


For read-based analysis antibiotic resistance genes and virulence genes are detected using Huttenhower lab's progam ShortBRED. The antibiotic resistance gene database was generated by the developers of ShortBRED using genes from ARDB and Resfams. The virulence genes database was generated by the developers of EDGE using VFDB.

For ORF-based analysis, antibiotic resistance genes are detected using CARD's (Comprehensive Antibiotic Resistance Database) program RGI (Resistance Gene Identifier). RGI uses CARD's custom database of antibiotic resistance genes. The virulence genes are detected using ShortBRED with a database generated by the developers of EDGE using VFDB.

5.4.7 PCR Primer Tools

EDGE includes PCR-related tools for use by those who want to use PCR data for their projects.



- **Primer Validation**

  The "Primer Validation" tool can be used to verify whether and where given primer sequences would align to the genome of the sequenced organism. Prior to initiating the analysis, primer sequences in FASTA format must be deposited in the folder on the desktop in the directory entitled "EDGE Input Directory."

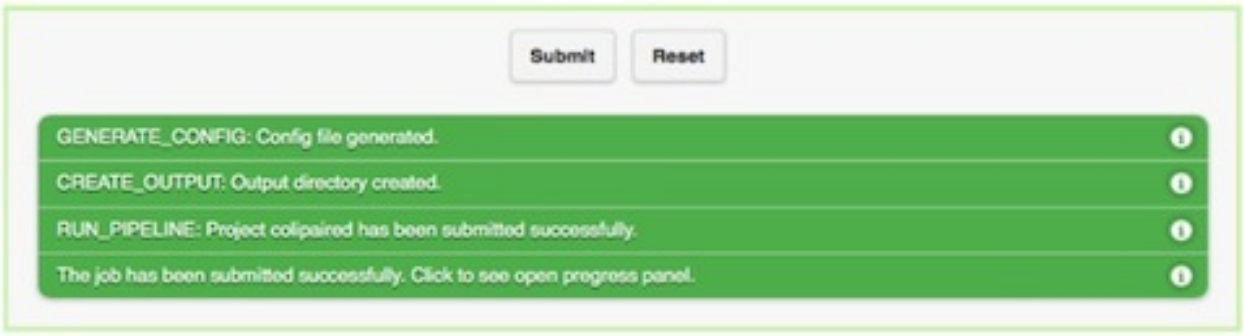
  In order to initiate primer validation, within the "Primer Validation" subsection switch the "Run Primer Validation" toggle button to "On". Then, within the "Primer FASTA Sequences" navigation field, select your file containing the primer sequences of interest. Next, in the "Maximum Mismatch" field, choose the maximum number of mismatches you wish to allow per primer sequence. The available options are 0, 1, 2, 3, or 4.

- **Primer Design**

  If you would like to design new primers that will differentiate a sequenced microorganism from all other bacteria and viruses in NCBI, you can do so using the "Primer Design" tool. To initiate primer design switch the "Run Primer Design" toggle button to "On". There are default settings supplied for Melting Temperature, Primer Length, Tm Differential, and Number of Primer Pairs, but you can change these settings if desired.

5.5 Submission of a job

When you have selected the appropriate input files and desired analysis options, and you are ready to submit the analysis job, click on the "Submit" button at the bottom of the page. Immediately you will see indicators of successful job submission and job status below the submit button, in green. If there is something wrong with the input, it will stop the submission and show the message in red, highlighting the sections with issues.

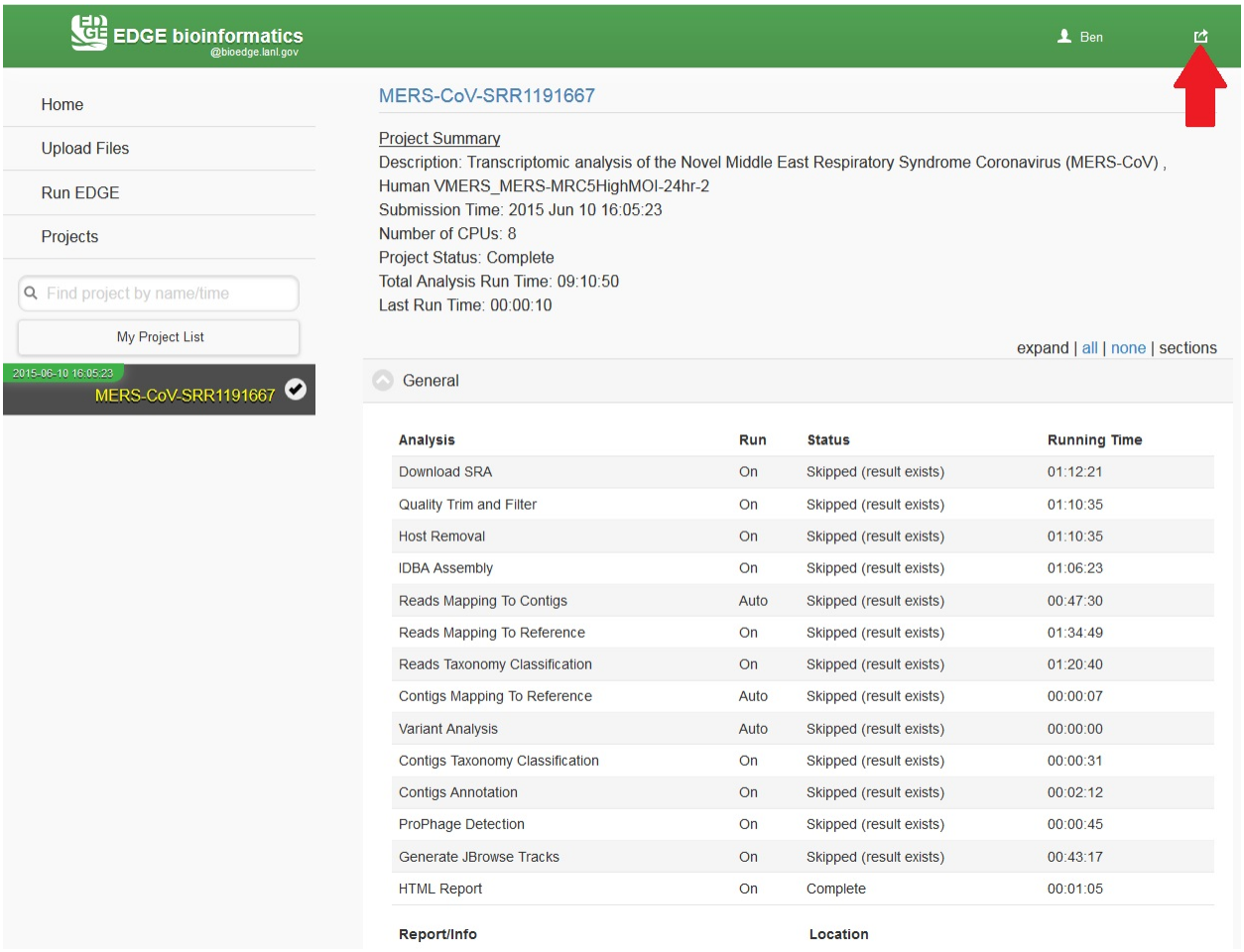


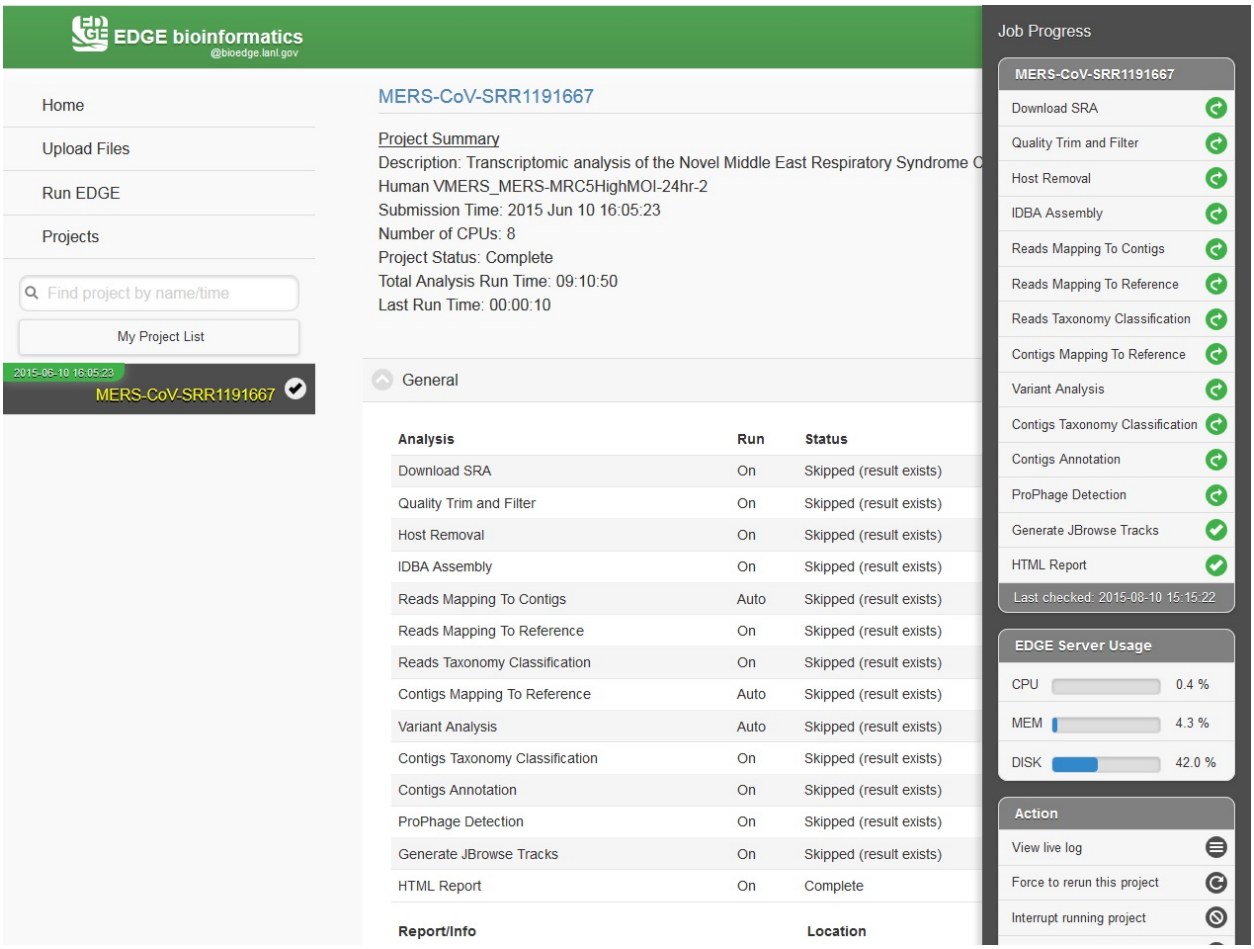
5.6 Checking the status of an analysis job

Once an analysis job has been submitted, it will become visible in the left navigation bar. There is a grey, red, orange, green color-coding system that indicates job status as follow:

Status	Not yet begun	Error	In progress (running)	Completed
Color	Grey	Red	Orange	Green

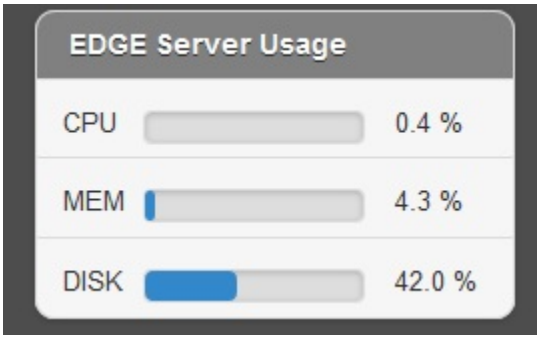
While the job is in progress, clicking on the project in the left navigation bar will allow you to see which individual steps have been completed or are in progress, and results that have already been produced. Clicking the job progress widget at top right opens up a more concise view of progress.


EDGE bioinformatics
@bioedge.lanl.gov
Ben
Home
Upload Files
Run EDGE
Projects
Find project by name/time
My Project List
2015-06-10 16:05:23
MERS-CoV-SRR1191667
MERS-CoV-SRR1191667
Project Summary
Description: Transcriptomic analysis of the Novel Middle East Respiratory Syndrome Coronavirus (MERS-CoV) , Human VMERS_MERS-MRC5HighMOI-24hr-2
Submission Time: 2015 Jun 10 16:05:23
Number of CPUs: 8
Project Status: Complete
Total Analysis Run Time: 09:10:50
Last Run Time: 00:00:10
expand | all | none | sections
General
Analysis Run Status Running Time
Download SRA On Skipped (result exists) 01:12:21
Quality Trim and Filter On Skipped (result exists) 01:10:35
Host Removal On Skipped (result exists) 01:10:35
IDBA Assembly On Skipped (result exists) 01:06:23
Reads Mapping To Contigs Auto Skipped (result exists) 00:47:30
Reads Mapping To Reference On Skipped (result exists) 01:34:49
Reads Taxonomy Classification On Skipped (result exists) 01:20:40
Contigs Mapping To Reference Auto Skipped (result exists) 00:00:07
Variant Analysis Auto Skipped (result exists) 00:00:00
Contigs Taxonomy Classification On Skipped (result exists) 00:00:31
Contigs Annotation On Skipped (result exists) 00:02:12
ProPhage Detection On Skipped (result exists) 00:00:45
Generate JBrowse Tracks On Skipped (result exists) 00:43:17
HTML Report On Complete 00:01:05
Report/Info
Location



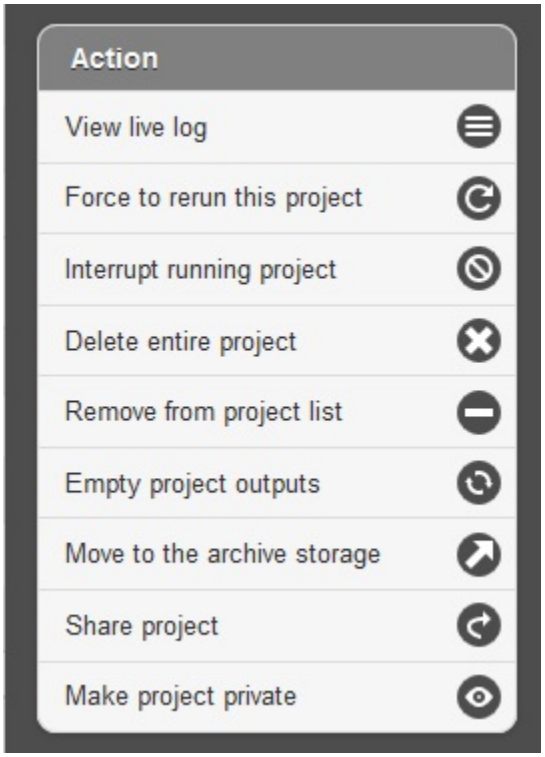
5.7 Monitoring the Resource Usage

In the job project sidebar, you can see there is an "EDGE Server Usage" widget that dynamically monitors the server resource usage for %CPU, %MEMORY and %DISK space. If there is not enough available disk space, you may consider deleting or archiving the submitted job with the Action tool described below.



5.8 Management of Jobs

Below the resource monitor is the "Action" tool, used for managing jobs in progress or existing projects.

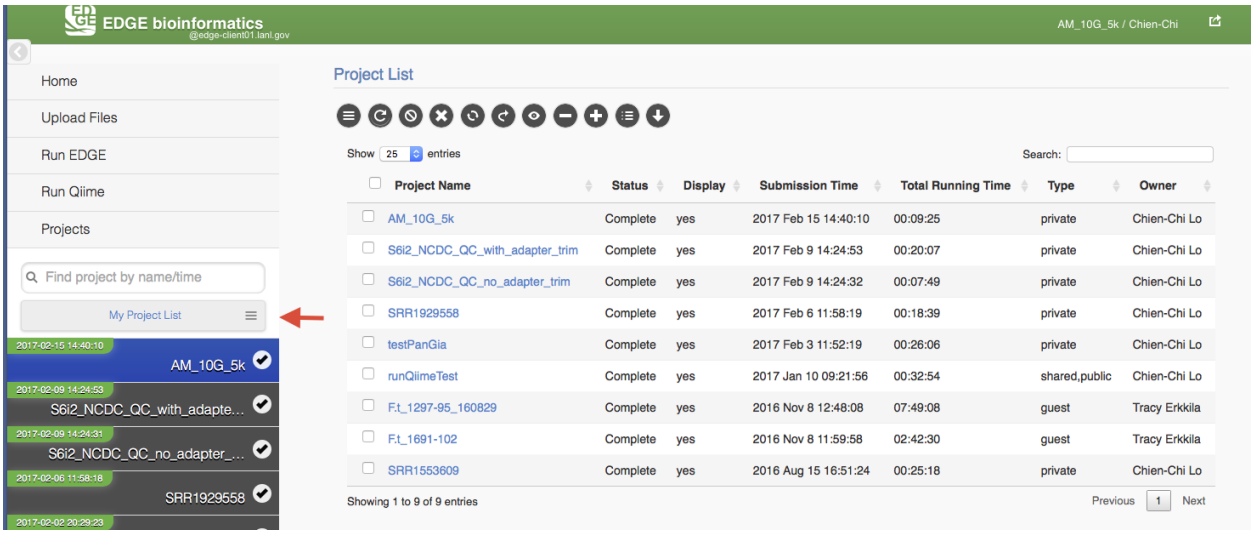


The available actions are:

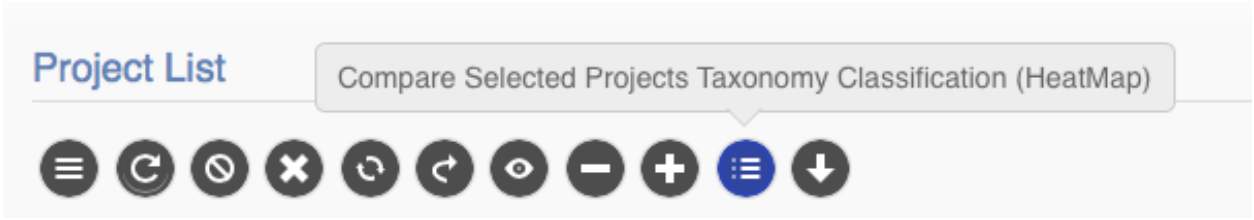
- **View live log** A terminal-like screen showing all the command lines and progress log information. This is useful for troubleshooting or if you want to repeat certain functions through command line at edge server.
- **Force to rerun this project** Rerun a project with the same inputs and configuration. No additional input needs.
- **Interrupt running project** Immediately stop a running project.
- **Delete entire project** Delete the entire output directory of the project.
- **Remove from project list** Keep the output but remove project name from the project list
- **Empty project outputs** Clean all the results but keep the config file. User can use this function to do a clean rerun.
- **Move to an archive directory** For performance reasons, the output directory will be put in local storage. User can use this function to move projects from local storage to a slower but larger network storage, which are configured when the edge server is installed.
- **Share Project** Allow guests and other users to view the project.
- **Make project Private/Public** Restrict access to viewing the project to only yourself. Or open it everyone.

5.9 Project List Table

When you click "My Project List", all your projects or projects shared to you will show in a table. It lists the projects status, submission time, running time, type and owner. User can select one or more jobs from the checkbox in the project table and perform actions similar to "Action" Widget described in the previous section. The action will apply to all checked projects.



When mouse over the action buttons on the project list page, it will show a pop up info for the action buttons. There is a special action button for multiple projects, "Compare Selected Projects Taxonomy Classification (HeatMap)" which will draw heatmaps of taxonomy profiling results for multiple projects using MetaComp.



5.10 Other Methods of Accessing EDGE

5.10.1 Internal Python Web Server

EDGE includes a simple web server for single-user applications or other testing. It is not robust enough for production usage, but it is simple enough that it can be run on practically any system.

To run gui, type:

```
$EDGE_HOME/start_edge_ui.sh
```

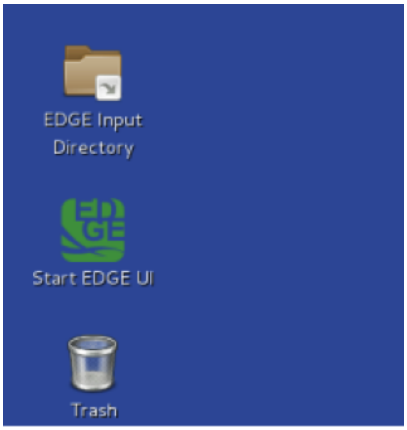
This will start a localhost and the GUI html page will be opened by your default browser.

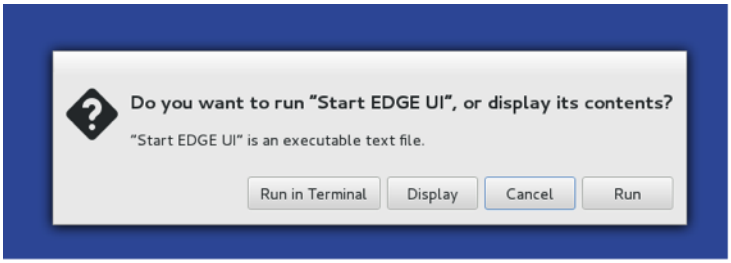
5.10.2 Apache Web Server

The preferred installation of EDGE uses Apache 2 (See *Testing the EDGE Installation* (page 16)), and serves the application as a proper system service. A sample httpd.conf (or apache2.conf, depending on your operating system) is provided in the root directory of your installation. If this configuration is used, EDGE will be available on any IP or hostname registered to the machine, on ports 80 and 8080.

You can access EDGE by opening either the desktop link (below), or your browser, and entering http://localhost:80 in the address bar.

Note: If the desktop environment is available, after installation, a "Start EDGE UI" icon should be on the desktop. Click on the green icon and choose "Run in Terminal." Results should be the same as those obtained by the above method to start the GUI.

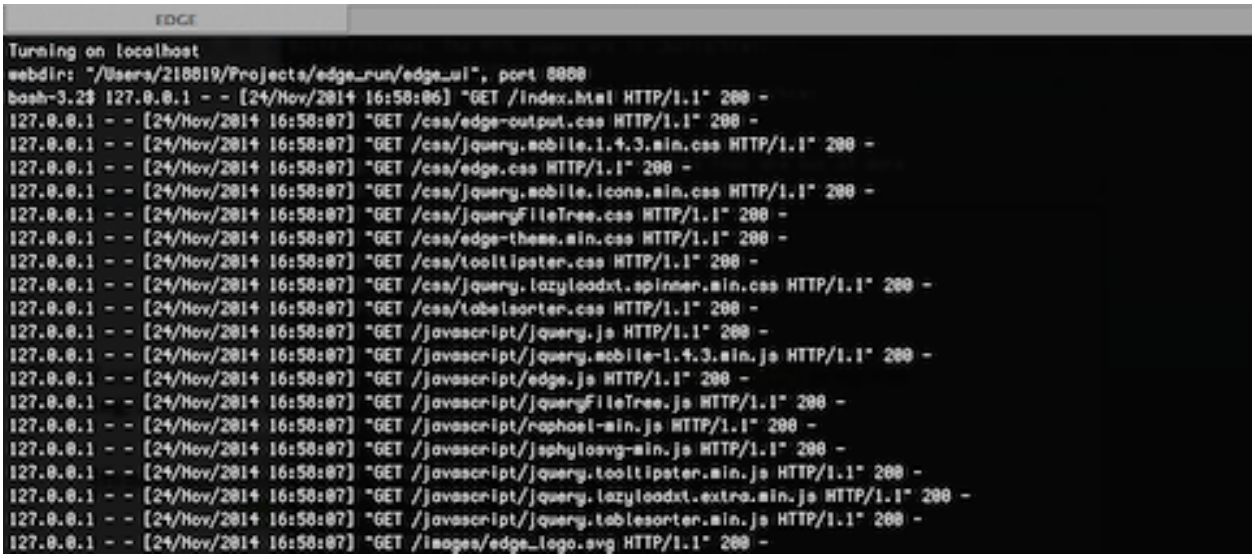




The URL address is 127.0.0.1:8080/index.html. It may not be that powerful,as it is hosted by Apache HTTP Server, but it works. With system administrator help, the Apache HTTP Server is the suggested method to host the gui interface.

Note: You may need to configure the edge_wwwroot and input and output in the edge_ui/edge_config.tmpl file while configuring the Apache HTTP Server and link to external drive or network drive if needed.

A Terminal window will display messages and errors as you run EDGE. Under normal operating conditions you can minimize this window. Should an error/problem arise, you may maximize this window to view the error.



Warning: IMPORTANT: Do not close this window!

The Browser window is the window in which you will interact with EDGE.

CHAPTER 6

Command Line Interface (CLI)

The command line usage is as followings:

```
Usage: perl runPipeline.pl [options] -c config.txt -p 'reads1.fastq reads2.fastq' -o
↪out_directory
Version 1.1
Input File:
        -u              Unpaired reads, Single end reads in fastq

        -p              Paired reads in two fastq files and separate by space in quote

        -c              Config File
Output:
        -o              Output directory.

Options:
        -ref            Reference genome file in fasta

        -primer         A pair of Primers sequences in strict fasta format

        -cpu            number of CPUs (default: 8)

        -version        print verison
```

A config file (example in the below section, the *Graphic User Interface (GUI)* (page 29) will generate config automatically), reads Files in fastq format, and a output directory are required when run by command line. Based on the configuration file, if all modules are turned on, EDGE will run the following steps. Each step contains at least one command line scripts/programs.

1. Data QC
2. Host Removal QC
3. *De novo* Assembling
4. Reads Mapping To Contig
5. Reads Mapping To Reference Genomes

6. Taxonomy Classification on All Reads or unMapped to Reference Reads
7. Map Contigs To Reference Genomes
8. Variant Analysis
9. Contigs Taxonomy Classification
10. Contigs Annotation
11. ProPhage detection
12. PCR Assay Validation
13. PCR Assay Adjudication
14. Phylogenetic Analysis
15. Generate JBrowse Tracks
16. HTML report

6.1 Configuration File

The config file is a text file with the following information. If you are going to do host removal, you need to *build host index* (page 69) for it and change the fasta file path in the config file.

```
[Count Fastq]
DoCountFastq=auto

[Quality Trim and Filter]
## boolean, 1=yes, 0=no
DoQC=1
##Targets quality level for trimming
q=5
##Trimmed sequence length will have at least minimum length
min_L=50
##Average quality cutoff
avg_q=0
##"N" base cutoff.  Trimmed read has more than this number of continuous base "N"␣
↪will be discarded.
n=1
##Low complexity filter ratio, Maximum fraction of mono-/di-nucleotide sequence
lc=0.85
## Trim reads with adapters or contamination sequences
adapter=/PATH/adapter.fasta
## phiX filter, boolean, 1=yes, 0=no
phiX=0
## Cut # bp from 5 end before quality trimming/filtering
5end=0
## Cut # bp from 3 end before quality trimming/filtering
3end=0

[Host Removal]
## boolean, 1=yes, 0=no
DoHostRemoval=1
## Use more Host=  to remove multiple host reads
Host=/PATH/all_chromosome.fasta
similarity=90
```

(continues on next page)

(continued from previous page)

```
[Assembly]
## boolean, 1=yes, 0=no
DoAssembly=1
##Bypass assembly and use pre-assembled contigs
assembledContigs=
minContigSize=200
## spades or idba_ud
assembler=idba_ud
idbaOptions="--pre_correction  --mink 31"
## for spades
singleCellMode=
pacbioFile=
nanoporeFile=

[Reads Mapping To Contigs]
# Reads mapping to contigs
DoReadsMappingContigs=auto

[Reads Mapping To Reference]
# Reads mapping to reference
DoReadsMappingReference=0
bowtieOptions=
# reference genbank or fasta file
reference=
MapUnmappedReads=0

[Reads Taxonomy Classification]
## boolean, 1=yes, 0=no
DoReadsTaxonomy=1
## If reference genome exists, only use unmapped reads to do Taxonomy Classification.␣
↪Turn on AllReads=1 will use all reads instead.
AllReads=0
enabledTools=gottcha-genDB-b,gottcha-speDB-b,gottcha-strDB-b,gottcha-genDB-v,gottcha-
↪speDB-v,gottcha-strDB-v,metaphlan,bwa,kraken_mini

[Contigs Mapping To Reference]
# Contig mapping to reference
DoContigMapping=auto
## identity cutoff
identity=85
MapUnmappedContigs=0

[Variant Analysis]
DoVariantAnalysis=auto

[Contigs Taxonomy Classification]
DoContigsTaxonomy=1

[Contigs Annotation]
## boolean, 1=yes, 0=no
DoAnnotation=1
# kingdom: Archaea Bacteria Mitochondria Viruses
kingdom=Bacteria
contig_size_cut_for_annotation=700
## support tools: Prokka or RATT
annotateProgram=Prokka
```

(continues on next page)

(continued from previous page)

```
annotateSourceGBK=

[ProPhage Detection]
DoProPhageDetection=1

[Phylogenetic Analysis]
DoSNPtree=1
## Availabe choices are Ecoli, Yersinia, Francisella, Brucella, Bacillus
SNPdbName=Ecoli
## FastTree or RAxML
treeMaker=FastTree
## SRA accessions ByrRun, ByExp, BySample, ByStudy
SNP_SRA_ids=

[Primer Validation]
DoPrimerValidation=1
maxMismatch=1
primer=

[Primer Adjudication]
## boolean, 1=yes, 0=no
DoPrimerDesign=0
## desired primer tm
tm_opt=59
tm_min=57
tm_max=63
## desired primer length
len_opt=18
len_min=20
len_max=27
## reject primer having Tm < tm_diff difference with background Tm
tm_diff=5
## display # top results for each target
top=5

[Generate JBrowse Tracks]
DoJBrowse=1

[HTML Report]
DoHTMLReport=1
```

6.2 Test Run

EDGE provides an example data set which is an E. coli MiSeq dataset and has been subsampled to ~10x fold coverage reads.

In the EDGE home directory,

```
cd testData
sh runTest.sh
```

See *Output* (page 64)

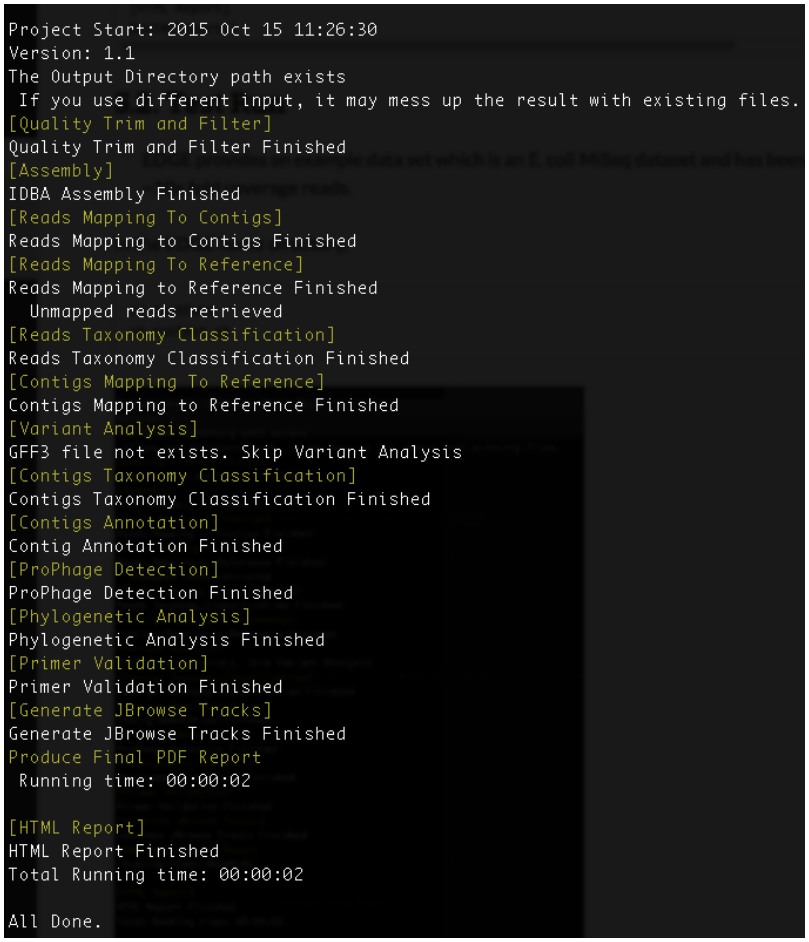


Fig. 1: Snapshot from the terminal.

6.3 Descriptions of each module

Each module comes with default parameters and user can see the optional parameters by entering the program name with –h or -help flag without any other arguments.

1. **Data QC**

- Required step? **No**
- Command example

```
perl $EDGE_HOME/scripts/illumina_fastq_QC.pl  -p 'Ecoli_10x.1.fastq Ecoli_10x.2.
↪fastq'  -q 5 -min_L 50 -avg_q 5 -n 0 -lc 0.85 -d QcReads -t 10
```

- What it does
  - Quality control
  - Read filtering
  - Read trimming
- Expected input
  - Paired-end/Single-end reads in FASTQ format
- Expected output
  - QC.1.trimmed.fastq
  - QC.2.trimmed.fastq
  - QC.unpaired.trimmed.fastq
  - QC.stats.txt
  - QC_qc_report.pdf

2. **Host Removal QC**

- Required step? **No**
- Command example

```
perl $EDGE_HOME/scripts/host_reads_removal_by_mapping.pl  -p 'QC.1.trimmed.fastq␣
↪QC.2.trimmed.fastq' -u QC.unpaired.trimmed.fastq -ref human_chromosomes.fasta -
↪o QcReads -cpu 10
```

- What it does
  - Read filtering
- Expected input
  - Paired-end/Single-end reads in FASTQ format
- Expected output
  - host_clean.1.fastq
  - host_clean.2.fastq
  - host_clean.mapping.log
  - host_clean.unpaired.fastq
  - host_clean.stats.txt

3. **IDBA Assembling**

- Required step? **No**
- Command example

```
fq2fa --merge host_clean.1.fastq  host_clean.2.fastq  pairedForAssembly.fasta
idba_ud  --num_threads 10 -o AssemblyBasedAnalysis/idba --pre_correction␣
↪pairedForAssembly.fasta
```

- What it does
  - Iterative kmers de novo Assembly, it performs well on isolates as well as metagenomes. It may not work well on very large genomes.
- Expected input
  - Paired-end/Single-end reads in FASTA format
- Expected output
  - contig.fa
  - scaffold.fa (input paired end)

4. **Reads Mapping To Contig**

- Required step? **No**
- Command example

```
perl $EDGE_HOME/scripts/runReadsToContig.pl  -p 'host_clean.1.fastq host_clean.2.
↪fastq' -d AssemblyBasedAnalysis/readsMappingToContig -pre readsToContigs  -ref␣
↪AssemblyBasedAnalysis/contigs.fa
```

- What it does
  - Mapping reads to assembled contigs
- Expected input
  - Paired-end/Single-end reads in FASTQ format
  - Assembled Contigs in Fasta format
  - Output Directory
  - Output prefix
- Expected output
  - readsToContigs.alnstats.txt
  - readsToContigs_coverage.table
  - readsToContigs_plots.pdf
  - readsToContigs.sort.bam
  - readsToContigs.sort.bam.bai

5. **Reads Mapping To Reference Genomes**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/runReadsToGenome.pl  -p 'host_clean.1.fastq host_clean.2.
↪fastq'  -d ReadsBasedAnalysis -pre readsToRef -ref Reference.fna
```

- What it does
  - Mapping reads to reference genomes
  - SNPs/Indels calling
- Expected input
  - Paired-end/Single-end reads in FASTQ format
  - Reference genomes in Fasta format
  - Output Directory
  - Output prefix
- Expected output
  - readsToRef.alnstats.txt
  - readsToRef_plots.pdf
  - readsToRef_refID.coverage
  - readsToRef_refID.gap.coords
  - readsToRef_refID.window_size_coverage
  - readsToRef.ref_windows_gc.txt
  - readsToRef.raw.bcf
  - readsToRef.sort.bam
  - readsToRef.sort.bam.bai
  - readsToRef.vcf

6. **Taxonomy Classification on All Reads or unMapped to Reference Reads**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/microbial_profiling/microbial_profiling_configure.pl
↪$EDGE_HOME/scripts/microbial_profiling/microbial_profiling.settings.tmpl␣
↪gottcha-speDB-b > microbial_profiling.settings.ini
perl $EDGE_HOME/scripts/microbial_profiling/microbial_profiling.pl -o  Taxonomy -
↪s microbial_profiling.settings.ini -c 10 UnmappedReads.fastq
```

- What it does
  - Taxonomy Classification using multiple tools, including BWA mapping to NCBI Refseq, metaphlan, kraken, GOTTCHA.
  - Unify varies output format and generate reports
- Expected input
  - Reads in FASTQ format
  - Configuration text file (generated by microbial_profiling_configure.pl)
- Expected output

  - Summary EXCEL and text files.
  - Heatmaps tools comparison
  - Radarchart tools comparison
  - Krona and tree-style plots for each tool.

7. **Map Contigs To Reference Genomes**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/nucmer_genome_coverage.pl  -e 1 -i 85 -p contigsToRef␣
↪Reference.fna contigs.fa
```

- What it does
  - Mapping assembled contigs to reference genomes
  - SNPs/Indels calling
- Expected input
  - Reference genome in Fasta Format
  - Assembled contigs in Fasta Format
  - Output prefix
- Expected output
  - contigsToRef_avg_coverage.table
  - contigsToRef.delta
  - contigsToRef_query_unUsed.fasta
  - contigsToRef.snps
  - contigsToRef.coords
  - contigsToRef.log
  - contigsToRef_query_novel_region_coord.txt
  - contigsToRef_ref_zero_cov_coord.txt

8. **Variant Analysis**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/SNP_analysis.pl -genbank Reference.gbk -SNP contigsToRef.
↪snps -format nucmer
perl $EDGE_HOME/scripts/gap_analysis.pl -genbank Reference.gbk -gap  contigsToRef_
↪ref_zero_cov_coord.txt
```

- What it does
  - Analyze variants and gaps regions using annotation file.
- Expected input
  - Reference in GenBank format
  - SNPs/INDELs/Gaps files from "Map Contigs To Reference Genomes"

- Expected output
  - contigsToRef.SNPs_report.txt
  - contigsToRef.Indels_report.txt
  - GapVSReference.report.txt

9. **Contigs Taxonomy Classification**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/contig_classifier_by_bwa/contig_classifier_by_bwa.pl --db
↪$EDGE_HOME/database/bwa_index/NCBI-Bacteria-Virus.fna --threads 10 --prefix␣
↪OuputCT --input contigs.fa
```

- What it does
  - Taxonomy Classification on contigs using BWA mapping to NCBI Refseq
- Expected input
  - Contigs in Fasta format
  - NCBI Refseq genomes bwa index
  - Output prefix
- Expected output
  - prefix.assembly_class.csv
  - prefix.assembly_class.top.csv
  - prefix.ctg_class.csv
  - prefix.ctg_class.LCA.csv
  - prefix.ctg_class.top.csv
  - prefix.unclassified.fasta

10. **Contig Annotation**

- Required step? **No**
- Command example:

```
prokka --force --prefix PROKKA --outdir Annotation contigs.fa
```

- What it does
  - The rapid annotation of prokaryotic genomes.
- Expected input
  - Assembled Contigs in Fasta format
  - Output Directory
  - Output prefix
- Expected output
  - It produces GFF3, GBK and SQN files that are ready for editing in Sequin and ultimately submitted to Genbank/DDJB/ENA.

11. **ProPhage detection**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/phageFinder_prepare.pl -o Prophage -p Assembly Annotation/
↪PROKKA.gff Annotation/PROKKA.fna
$EDGE_HOME/thirdParty/phage_finder_v2.1/bin/phage_finder_v2.1.sh Assembly
```

- What it does
  - Identify and classify prophages within prokaryotic genomes.
- Expected input
  - Annotated Contigs GenBank file
  - Output Directory
  - Output prefix
- Expected output
  - phageFinder_summary.txt

12. **PCR Assay Validation**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/pcrValidation/validate_primers.pl -ref contigs.fa -primer␣
↪primers.fa -mismatch 1 -output AssayCheck
```

- What it does
  - In silico PCR primer validation by sequence alignment.
- Expected input
  - Assembled Contigs/Reference in Fasta format
  - Output Directory
  - Output prefix
- Expected output
  - pcrContigValidation.log
  - pcrContigValidation.bam

13. **PCR Assay Adjudication**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/pcrAdjudication/pcrUniquePrimer.pl --input contigs.fa  --
↪gff3 PCR.Adjudication.primers.gff3
```

- What it does
  - Design unique primer pairs for input contigs.
- Expected input

  - Assembled Contigs in Fasta format
  - Output gff3 file name
- Expected output
  - PCR.Adjudication.primers.gff3
  - PCR.Adjudication.primers.txt

14. **Phylogenetic Analysis**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/prepare_SNP_phylogeny.pl -o output/SNP_Phylogeny/Ecoli -
↪tree FastTree -db Ecoli -n output -cpu 10 -p QC.1.trimmed.fastq QC.2.trimmed.
↪fastq -c contigs.fa -s QC.unpaired.trimmed.fastq
perl $EDGE_HOME/scripts/SNPphy/runSNPphylogeny.pl output/SNP_Phylogeny/Ecoli/
↪SNPphy.ctrl
```

- What it does
  - Perform SNP identification against selected pre-built SNPdb or selected genomes
  - Build SNP based multiple sequence alignment for all and CDS regions
  - Generate Tree file in newick/PhyloXML format
- Expected input
  - SNPdb path or genomesList
  - Fastq reads files
  - Contig files
- Expected output
  - SNP based phylogentic multiple sequence alignment
  - SNP based phylogentic tree in newick/PhyloXML format.
  - SNP information table

15. **Generate JBrowse Tracks**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/edge2jbrowse_converter.pl --in-ref-fa Reference.fna --in-
↪ref-gff3 Reference.gff --proj_outdir EDGE_project_dir
```

- What it does
  - Convert several EDGE outputs into JBrowse tracks for visualization for contigs and reference, respectively.
- Expected input
  - EDGE project output Directory
- Expected output
  - EDGE post-processed files for JBrowse tracks in the JBrowse directory.
  - Tracks configuration files in the JBrowse directory.

16. **HTML Report**

- Required step? **No**
- Command example:

```
perl $EDGE_HOME/scripts/munger/outputMunger_w_temp.pl EDGE_project_dir
```

- What it does
  - Generate statistical numbers and plots in an interactive html report page.
- Expected input
  - EDGE project output Directory
- Expected output
  - report.html

6.4 Other command-line utility scripts

1. To extract certain taxa fasta from contig classification result:

```
cd /home/edge_install/edge_ui/EDGE_output/41/AssemblyBasedAnalysis/Taxonomy
perl /home/edge_install/scripts/contig_classifier_by_bwa/extract_fasta_by_taxa.pl␣
↪-fasta ../contigs.fa -csv ProjectName.ctg_class.top.csv -taxa "Enterobacter␣
↪cloacae" > Ecloacae.contigs.fa
```

2. To extract unmapped/mapped reads fastq from the bam file:

```
cd /home/edge_install/edge_ui/EDGE_output/41/AssemblyBasedAnalysis/
↪readsMappingToContig
# extract unmapped reads
perl /home/edge_install/scripts/bam_to_fastq.pl -unmapped readsToContigs.sort.bam
# extract mapped reads
perl /home/edge_install/scripts/bam_to_fastq.pl -mapped readsToContigs.sort.bam
```

3. To extract mapped reads fastq of a specific contig/reference from the bam file:

```
cd /home/edge_install/edge_ui/EDGE_output/41/AssemblyBasedAnalysis/
↪readsMappingToContig
perl /home/edge_install/scripts/bam_to_fastq.pl -id ProjectName_00001 -mapped␣
↪readsToContigs.sort.bam
```

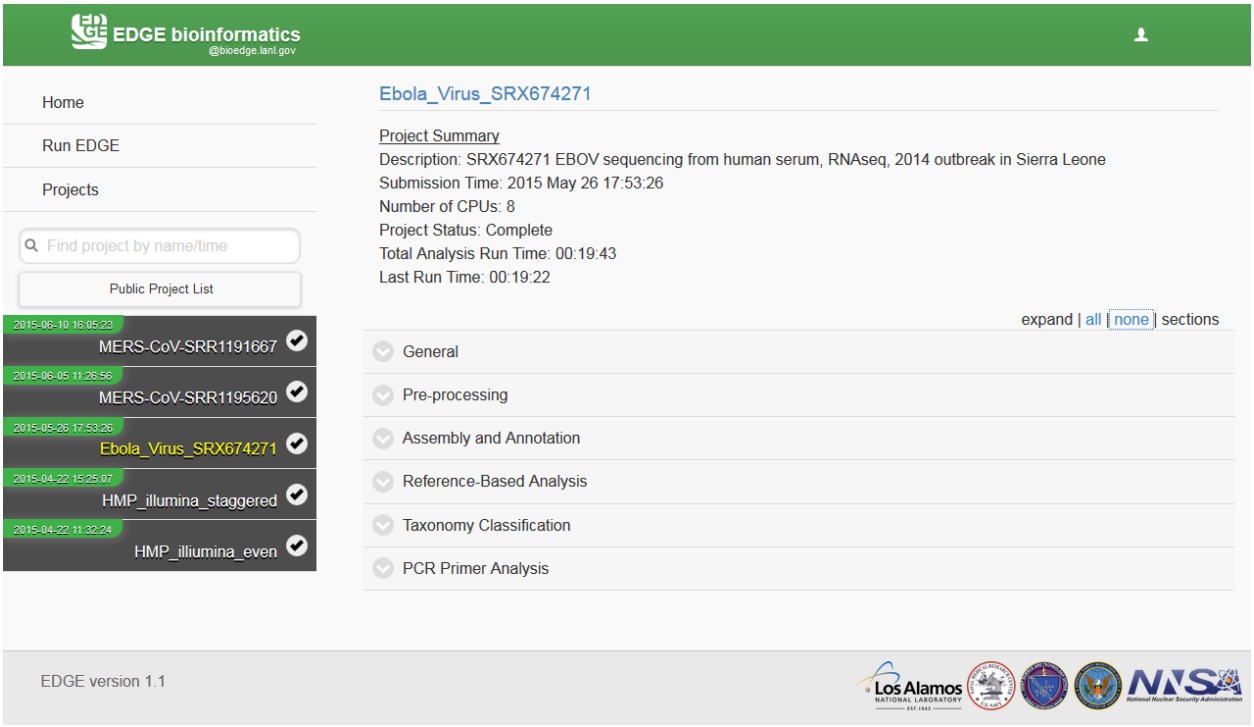
CHAPTER 7

Output

The output directory structure contains ten major sub-directories when all modules are turned on. In addition to the main directories, EDGE will generate a final report in portable document file format (pdf), process log and error log file in the project main directory.

- AssayCheck
- AssemblyBasedAnalysis
- HostRemoval
- HTML_Report
- JBrowse
- QcReads
- ReadsBasedAnalysis
- ReferenceBasedAnalysis
- Reference
- SNP_Phylogeny

In the graphic user interface, EDGE generates an interactive output webpage which includes summary statistics and taxonomic information, etc. The easiest way to interact with the results is through the web interface. If a project run finished through the command line, user can open the report html file in the HTML_report subdirectory off-line. When a project run is finished, user can click on the project id from the menu and it will generate the interactive html report on the fly. User can browse the data structure by clicking the project link and visualize the result by JBrowse links, download the pdf files, etc.



7.1 Example Output

See http://lanl-bioinformatics.github.io/EDGE/example_output/report.html

Note: The example link is just an example of graphic output. The JBrowse and links are not accessible in the example links.

CHAPTER 8

Databases

8.1 EDGE provided databases

8.1.1 Taxnomomy Database Info Table

https://lanl-bioinformatics.github.io/EDGE/docs/taxonomyDBtable.html

8.1.2 NCBI Refseq

EDGE prebuilt blast db and bwa_index of NCBI RefSeq genomes.

- Bacteria: NCBI all complete bacteria download method
  - Version: NCBI 2017 Oct 3
  - 245 Archaea + 7917 Bacteria genomes
- Virus: NCBI Virus
  - Version: NCBI 2017 Oct 3
  - 7458 complete genomes + Neighbor Nucleotoides (118039 seuqences)

see $EDGE_HOME/database/bwa_index/id_mapping.txt for all gi/accession to genome name lookup table.

8.1.3 Krona taxonomy

- paper: http://www.ncbi.nlm.nih.gov/pubmed/?term=21961884
- website: http://sourceforge.net/p/krona/home/krona/

Update Krona taxonomy db

Go to the folder in the KronaTools installation and run (internet required)

```
cd $EDGE_HOME/thirdParty/KronaTools-2.7/
./updateTaxonomy.sh
./updateAccessions.sh
```

This uses about 16 GB of disk space and an additional 16 GB of scratch space during installation (at the time of this writing; it is always growing) and takes minutes or up to an hour to run.

8.1.4 Metaphlan4 database

MetaPhlAn 4 relies on ~5.1M unique clade-specific marker genes identified from ~1M microbial genomes (~236,600 references and 771,500 metagenomic assembled genomes) spanning 26,970 species-level genome bins (SGBs, http://segatalab.cibio.unitn.it/data/Pasolli_et_al.html), 4,992 of them taxonomically unidentified at the species level.

- paper: http://www.ncbi.nlm.nih.gov/pubmed/?term=36823356
- website: http://huttenhower.sph.harvard.edu/metaphlan4

8.1.5 Human Genome

The bwa index is prebuilt in the EDGE. The human hs_ref_GRCh38 sequences from NCBI ftp site.

- website https://ftp.ncbi.nlm.nih.gov/genomes/all/GCF/000/001/405/GCF_000001405.26_GRCh38/

8.1.6 Kraken2 DB

Kraken2 is a system for assigning taxonomic labels to short DNA sequences, usually obtained through metagenomic studies. Kraken2 database in EDGE is a pre-built database constructed from Refseq bacteria, archaea, and viral libraries and the GRCh38 human genome and UniVec_Core in RefSeq (as of Dec, 2021).

- Kraken1 paper: http://www.ncbi.nlm.nih.gov/pubmed/?term=24580807
- website: http://ccb.jhu.edu/software/kraken2/

8.1.7 Centrifuge DB

Centrifuge is a very rapid and memory-efficient system for the classification of DNA sequences from microbial samples, with better sensitivity than and comparable accuracy to other leading systems. The database includes human genome, prokaryotic genomes, and viral genomes including 106 SARS-CoV-2 complete genomes. (as of Mar 29, 2020)

- Centrifuge paper: https://pubmed.ncbi.nlm.nih.gov/27852649/
- website: https://ccb.jhu.edu/software/centrifuge/

8.1.8 GOTTCHA DB

A novel, annotation-independent and signature-based metagenomic taxonomic profiling tool.

- website: http://lanl-bioinformatics.github.io/GOTTCHA/
- ftp: https://edge-dl.lanl.gov/gottcha/

- version: v20150825

8.1.9 SNPdb

SNP database based on whole genome comparison. Current available db are *Ecoli, Yersinia, Francisella, Brucella, Bacillus* (page 69) .

8.1.10 Invertebrate Vectors of Human Pathogens

The bwa index is prebuilt in the EDGE.

- paper: http://www.ncbi.nlm.nih.gov/pubmed/?term=22135296
- website: https://www.vectorbase.org

version: 2014 July 24

8.1.11 NCBI Nucleotide database (NT) database

- website: ftp://ftp.ncbi.nih.gov/blast/db/
- version: 2016 April 26

8.1.12 VFDB

A Microbial database of virulence factors

- paper: http://www.ncbi.nlm.nih.gov/pubmed/?term=26578559
- website: http://www.mgc.ac.cn/VFs/main.htm
- version: 20160818

8.1.13 ARDB

Antibiotic Resistance Genes Database

- website: http://ardb.cbcb.umd.edu/index.html
- version: 1.1

8.1.14 CARD

The Comprehensive Antibiotic Resistance Database

- website: https://card.mcmaster.ca/
- Version: 3.0.7

8.1.15 Amplicon: 16s/18s/ITS

For QIIME (Quantitative insights into Microbial Ecology) analysis (scikit-learn=0.24.1)

- Greengenes OTUs (16s)
  - website: http://greengenes.secondgenome.com/
  - version: 2022_10
- SILVA OTUs (16S/18S)
  - website: http://www.arb-silva.de/download/archive/qiime/
  - version: 138
- UNITE OTUs (Fungal ITS)
  - website: https://unite.ut.ee/repository.php
  - version: 16.10.2022

8.2 Building bwa index

Here take human genome as example.

1. Download the human hs_ref_GRCh38 sequences from NCBI ftp site:

```
wget ftp://ftp.ncbi.nlm.nih.gov/genomes/all/GCA/000/001/405/GCA_000001405.28_
↪GRCh38.p13/GCA_000001405.28_GRCh38.p13_genomic.fna.gz
```

2. Use the installed bwa to build the index:

```
$EDGE_HOME/bin/bwa index -p human_ref_GRCh38 GCA_000001405.28_GRCh38.p13_genomic.
↪fna.gz
```

   Now, you can configure the config file with "host=/path/to/human_ref_GRCh38" for host removal step.

8.3 SNP database genomes

SNP database was pre-built from the below genomes.

8.3.1 Ecoli Genomes

Name	Description	URL
Ecoli_042	Escherichia coli 042, complete genome	http://www.ncbi.nlm.nih.g
Ecoli_11128	Escherichia coli O111:H- str. 11128, complete genome	http://www.ncbi.nlm.nih.g
Ecoli_11368	Escherichia coli O26:H11 str. 11368 chromosome, complete genome	http://www.ncbi.nlm.nih.g
Ecoli_12009	Escherichia coli O103:H2 str. 12009, complete genome	http://www.ncbi.nlm.nih.g
Ecoli_2009EL2050	Escherichia coli O104:H4 str. 2009EL-2050 chromosome, complete genome	http://www.ncbi.nlm.nih.g
Ecoli_2009EL2071	Escherichia coli O104:H4 str. 2009EL-2071 chromosome, complete genome	http://www.ncbi.nlm.nih.g
Ecoli_2011C3493	Escherichia coli O104:H4 str. 2011C-3493 chromosome, complete genome	http://www.ncbi.nlm.nih.g
Ecoli_536	Escherichia coli 536, complete genome	http://www.ncbi.nlm.nih.g

Cor

Table 1 – continued from previous page

Name	Description	URL
Ecoli_55989	Escherichia coli 55989 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_ABU_83972	Escherichia coli ABU 83972 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_APEC_O1	Escherichia coli APEC O1 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_ATCC_8739	Escherichia coli ATCC 8739 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_BL21_DE3	Escherichia coli BL21(DE3) chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_BW2952	Escherichia coli BW2952 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_CB9615	Escherichia coli O55:H7 str. CB9615 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_CE10	Escherichia coli O7:K1 str. CE10 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_CFT073	Escherichia coli CFT073 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_DH1	Escherichia coli DH1, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_Di14	Escherichia coli str. 'clone D i14' chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_Di2	Escherichia coli str. 'clone D i2' chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_E2348_69	Escherichia coli O127:H6 str. E2348/69 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_E24377A	Escherichia coli E24377A chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_EC4115	Escherichia coli O157:H7 str. EC4115 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_ED1a	Escherichia coli ED1a chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_EDL933	Escherichia coli O157:H7 str. EDL933 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_ETEC_H10407	Escherichia coli ETEC H10407, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_HS	Escherichia coli HS, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_IAI1	Escherichia coli IAI1 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_IAI39	Escherichia coli IAI39 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_IHE3034	Escherichia coli IHE3034 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_K12_DH10B	Escherichia coli str. K-12 substr. DH10B chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_K12_MG1655	Escherichia coli str. K-12 substr. MG1655 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_K12_W3110	Escherichia coli str. K-12 substr. W3110, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_KO11FL	Escherichia coli KO11FL chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_LF82	Escherichia coli LF82, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_NA114	Escherichia coli NA114 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_NRG_857C	Escherichia coli O83:H1 str. NRG 857C chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_P12b	Escherichia coli P12b chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_REL606	Escherichia coli B str. REL606 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_RM12579	Escherichia coli O55:H7 str. RM12579 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_S88	Escherichia coli S88 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_SE11	Escherichia coli O157:H7 str. Sakai chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_SE15	Escherichia coli SE11 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_SMS35	Escherichia coli SE15, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_Sakai	Escherichia coli SMS-3-5 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_TW14359	Escherichia coli O157:H7 str. TW14359 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_UM146	Escherichia coli UM146 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_UMN026	Escherichia coli UMN026 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_UMNK88	Escherichia coli UMNK88 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_UTI89	Escherichia coli UTI89 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_W	Escherichia coli W chromosome, complete genome	http://www.ncbi.nlm.nih.go
Ecoli_Xuzhou21	Escherichia coli Xuzhou21 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Sboydii_CDC_3083_94	Shigella boydii CDC 3083-94 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Sboydii_Sb227	Shigella boydii Sb227 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Sdysenteriae_Sd197	Shigella dysenteriae Sd197, complete genome	http://www.ncbi.nlm.nih.go
Sflexneri_2002017	Shigella flexneri 2002017 chromosome, complete genome	http://www.ncbi.nlm.nih.go
Sflexneri_2a_2457T	Shigella flexneri 2a str. 2457T, complete genome	http://www.ncbi.nlm.nih.go

Con

Table 1 – continued from previous page

Name	Description	URL
Sflexneri_2a_301	Shigella flexneri 2a str. 301 chromosome, complete genome	http://www.ncbi.nlm.nih.g
Sflexneri_5_8401	Shigella flexneri 5 str. 8401 chromosome, complete genome	http://www.ncbi.nlm.nih.g
Ssonnei_53G	Shigella sonnei 53G, complete genome	http://www.ncbi.nlm.nih.g
Ssonnei_Ss046	Shigella sonnei Ss046 chromosome, complete genome	http://www.ncbi.nlm.nih.g

8.3.2 Yersinia Genomes

Name	Description	URL
Ypestis_A1122	Yersinia pestis A1122 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/384137007
Ypestis_Angola	Yersinia pestis Angola chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/162418099
Ypestis_Antiqua	Yersinia pestis Antiqua chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/108805998
Ypestis_CO92	Yersinia pestis CO92 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/16120353
Ypestis_D106004	Yersinia pestis D106004 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/384120592
Ypestis_D182038	Yersinia pestis D182038 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/384124469
Ypestis_KIM_10	Yersinia pestis KIM 10 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/22123922
Ypestis_Medievalis_Harbin_35	Yersinia pestis biovar Medievalis str. Harbin 35 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/384412706
Ypestis_Microtus_91001	Yersinia pestis biovar Microtus str. 91001 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/45439865
Ypestis_Nepal516	Yersinia pestis Nepal516 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/108810166
Ypestis_Pestoides_F	Yersinia pestis Pestoides F chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/145597324
Ypestis_Z176003	Yersinia pestis Z176003 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/294502110
Ypseudotuberculosis_IP_31758	Yersinia pseudotuberculosis IP 31758 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/153946813
Ypseudotuberculosis_IP_32953	Yersinia pseudotuberculosis IP 32953 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/51594359
Ypseudotuberculosis_PB1	Yersinia pseudotuberculosis PB1/+ chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/186893344
Ypseudotuberculosis_YPIII	Yersinia pseudotuberculosis YPIII chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/170022262

8.3.3 Francisella Genomes

Name	Description	URL
Fnovicida_U112	Francisella novicida U112 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/118496615
Ftularensis_holarctica_F92	Francisella tularensis subsp. holarctica F92 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/423049750
Ftularensis_holarctica_FSC200	Francisella tularensis subsp. holarctica FSC200 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/422937995
Ftularensis_holarctica_FTNF002-00	Francisella tularensis subsp. holarctica FTNF002-00 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/156501369
Ftularensis_holarctica_LVS	Francisella tularensis subsp. holarctica LVS chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/89255449
Ftularensis_holarctica_OSU18	Francisella tularensis subsp. holarctica OSU18 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/115313981
Ftularensis_mediasiatica_FSC147	Francisella tularensis subsp. mediasiatica FSC147 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/187930913
Ftularensis_TIGB03	Francisella tularensis TIGB03 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/379716390
Ftularensis_tularensis_FSC198	Francisella tularensis subsp. tularensis FSC198 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/110669657
Ftularensis_tularensis_NE061598	Francisella tularensis subsp. tularensis NE061598 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/385793751
Ftularensis_tularensis_SCHU_S4	Francisella tularensis subsp. tularensis SCHU S4 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/255961454
Ftularensis_tularensis_TI0902	Francisella tularensis subsp. tularensis TI0902 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/379725073
Ftularensis_tularensis_WY96-3418	Francisella tularensis subsp. tularensis WY96-3418 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/134301169

8.3.4 Brucella Genomes

Name	Description	URL
Babortus_1_9941	Brucella abortus bv. 1 str. 9-941	http://www.ncbi.nlm.nih.gov/bioproject/58019
Babortus_A13334	Brucella abortus A13334	http://www.ncbi.nlm.nih.gov/bioproject/83615
Babortus_S19	Brucella abortus S19	http://www.ncbi.nlm.nih.gov/bioproject/58873
Bcanis_ATCC_23365	Brucella canis ATCC 23365	http://www.ncbi.nlm.nih.gov/bioproject/59009
Bcanis_HSK_A52141	Brucella canis HSK A52141	http://www.ncbi.nlm.nih.gov/bioproject/83613
Bceti_TE10759_12	Brucella ceti TE10759-12	http://www.ncbi.nlm.nih.gov/bioproject/229880
Bceti_TE28753_12	Brucella ceti TE28753-12	http://www.ncbi.nlm.nih.gov/bioproject/229879
Bmelitensis_1_16M	Brucella melitensis bv. 1 str. 16M	http://www.ncbi.nlm.nih.gov/bioproject/200008
Bmelitensis_Abortus_2308	Brucella melitensis biovar Abortus 2308	http://www.ncbi.nlm.nih.gov/bioproject/16203
Bmelitensis_ATCC_23457	Brucella melitensis ATCC 23457	http://www.ncbi.nlm.nih.gov/bioproject/59241
Bmelitensis_M28	Brucella melitensis M28	http://www.ncbi.nlm.nih.gov/bioproject/158857
Bmelitensis_M590	Brucella melitensis M5-90	http://www.ncbi.nlm.nih.gov/bioproject/158855
Bmelitensis_NI	Brucella melitensis NI	http://www.ncbi.nlm.nih.gov/bioproject/158853
Bmicroti_CCM_4915	Brucella microti CCM 4915	http://www.ncbi.nlm.nih.gov/bioproject/59319
Bovis_ATCC_25840	Brucella ovis ATCC 25840	http://www.ncbi.nlm.nih.gov/bioproject/58113
Bpinnipedialis_B2_94	Brucella pinnipedialis B2/94	http://www.ncbi.nlm.nih.gov/bioproject/71133
Bsuis_1330	Brucella suis 1330	http://www.ncbi.nlm.nih.gov/bioproject/159871
Bsuis_ATCC_23445	Brucella suis ATCC 23445	http://www.ncbi.nlm.nih.gov/bioproject/59015
Bsuis_VBI22	Brucella suis VBI22	http://www.ncbi.nlm.nih.gov/bioproject/83617

8.3.5 Bacillus Genomes

Name	Description	URL
Banthracis_A0248	Bacillus anthracis str. A0248, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/229599883
Banthracis_Ames	Bacillus anthracis str. 'Ames Ancestor' chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/50196905
Banthracis_Ames_Ancestor	Bacillus anthracis str. Ames chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/30260195
Banthracis_CDC_684	Bacillus anthracis str. CDC 684 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/227812678
Banthracis_H9401	Bacillus anthracis str. H9401 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/386733873
Banthracis_Sterne	Bacillus anthracis str. Sterne chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/49183039
Bcereus_03BB102	Bacillus cereus 03BB102, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/225862057
Bcereus_AH187	Bacillus cereus AH187 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/217957581
Bcereus_AH820	Bacillus cereus AH820 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/218901206
Bcereus_anthracis_CI	Bacillus cereus biovar anthracis str. CI chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/301051741
Bcereus_ATCC_10987	Bacillus cereus ATCC 10987 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/42779081
Bcereus_ATCC_14579	Bacillus cereus ATCC 14579, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/30018278
Bcereus_B4264	Bacillus cereus B4264 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/218230750
Bcereus_E33L	Bacillus cereus E33L chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/52140164
Bcereus_F837_76	Bacillus cereus F837/76 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/376264031
Bcereus_G9842	Bacillus cereus G9842 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/218895141
Bcereus_NC7401	Bacillus cereus NC7401, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/375282101
Bcereus_Q1	Bacillus cereus Q1 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/222093774
Bthuringiensis_AlHakam	Bacillus thuringiensis str. Al Hakam chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/118475778
Bthuringiensis_BMB171	Bacillus thuringiensis BMB171 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/296500838
Bthuringiensis_Bt407	Bacillus thuringiensis Bt407 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/409187965
Bthuringiensis_chinensis_CT43	Bacillus thuringiensis serovar chinensis CT-43 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/384184088
Bthuringiensis_finitimus_YBT020	Bacillus thuringiensis serovar finitimus YBT-020 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/384177910
Bthuringiensis_konkukian_9727	Bacillus thuringiensis serovar konkukian str. 97-27 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/49476684
Bthuringiensis_MC28	Bacillus thuringiensis MC28 chromosome, complete genome	http://www.ncbi.nlm.nih.gov/nuccore/407703236

8.4 Ebola Reference Genomes

Accession	Description	URL
NC_014372	Tai Forest ebolavirus isolate Tai Forest virus H.sapiens-tc/CIV/1994/Pauleoula-CI, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/NC_014372
FJ217162	Cote d'Ivoire ebolavirus, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/FJ217162
FJ968794	Sudan ebolavirus strain Boniface, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/FJ968794
NC_006432	Sudan ebolavirus isolate Sudan virus H.sapiens-tc/UGA/2000/Gulu-808892, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/NC_006432
KJ660348	Zaire ebolavirus isolate H.sapiens-wt/GIN/2014/Gueckedou-C05, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KJ660348
KJ660347	Zaire ebolavirus isolate H.sapiens-wt/GIN/2014/Gueckedou-C07, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KJ660347
KJ660346	Zaire ebolavirus isolate H.sapiens-wt/GIN/2014/Kissidougou-C15, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KJ660346
JN638998	Sudan ebolavirus - Nakisamata, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/JN638998
AY354458	Zaire ebolavirus strain Zaire 1995, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/AY354458
AY729654	Sudan ebolavirus strain Gulu, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/AY729654
EU338380	Sudan ebolavirus isolate EBOV-S-2004 from Sudan, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/EU338380
KM655246	Zaire ebolavirus isolate H.sapiens-tc/COD/1976/Yambuku-Ecran, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KM655246
KC242801	Zaire ebolavirus isolate EBOV/H.sapiens-tc/COD/1976/deRoover, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242801
KC242800	Zaire ebolavirus isolate EBOV/H.sapiens-tc/GAB/2002/Ilembe, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242800
KC242799	Zaire ebolavirus isolate EBOV/H.sapiens-tc/COD/1995/13709 Kikwit, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242799
KC242798	Zaire ebolavirus isolate EBOV/H.sapiens-tc/GAB/1996/1Ikot, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242798
KC242797	Zaire ebolavirus isolate EBOV/H.sapiens-tc/GAB/1996/1Oba, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242797
KC242796	Zaire ebolavirus isolate EBOV/H.sapiens-tc/COD/1995/13625 Kikwit, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242796
KC242795	Zaire ebolavirus isolate EBOV/H.sapiens-tc/GAB/1996/1Mbie, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242795
KC242794	Zaire ebolavirus isolate EBOV/H.sapiens-tc/GAB/1996/2Nza, complete genome.	http://www.ncbi.nlm.nih.gov/nuccore/KC242794

CHAPTER 9

Third Party Tools

9.1 Assembly

- IDBA-UD
  - Citation: Peng, Y., et al. (2012) IDBA-UD: a de novo assembler for single-cell and metagenomic sequencing data with highly uneven depth, Bioinformatics, 28, 1420-1428.
  - Site: http://i.cs.hku.hk/~alse/hkubrg/projects/idba_ud/
  - Version: 1.1.1
  - License: GPLv2
- SPAdes
  - Citation: Prjibelski et al. (2020) Using SPAdes De Novo Assembler. Curr Protoc Bioinformatics. 2020;70(1):e102.
  - Site: https://github.com/ablab/spades
  - Version: 3.15.5
  - License: GPLv2
- MEGAHIT
  - Citation: Li D. et al. (2015) MEGAHIT: an ultra-fast single-node solution for large and complex metagenomics assembly via succinct de Bruijn graph. Bioinformatics. 2015 May 15;31(10):1674-6
  - Site: https://github.com/voutcn/megahit
  - Version: 1.2.9
  - License: GPLv3
- LRASM: Long Read Assembler
  - Citation:
  - Site: https://gitlab.com/chienchi/long_read_assembly

  - Version: 0.1.0
  - License: GPLv3

- RACON
  - Citation: Vaser R et al.(2017) Fast and accurate de novo genome assembly from long uncorrected reads. Genome Res. 2017 May;27(5):737-746.
  - Site: https://github.com/isovic/racon
  - Version: 1.4.13
  - License: MIT

- Unicycler
  - Citation: Wick RR et al.(2017) Unicycler: Resolving bacterial genome assemblies from short and long sequencing reads. PLoS Comput Biol. 2017 Jun 8;13(6):e1005595.
  - Site: https://github.com/rrwick/Unicycler
  - Version: 0.5.0
  - License: GPLv3

9.2 Annotation

- RATT
  - Citation: Otto, T.D., et al. (2011) RATT: Rapid Annotation Transfer Tool, Nucleic acids research, 39, e57.
  - Site: http://ratt.sourceforge.net/
  - Version:
  - License: GPLv3
  - Note: **The original RATT program does not deal with reverse complement strain annotations transfer. We edited the source code to fix it.**

- Prokka
  - Citation: Seemann, T. (2014) Prokka: rapid prokaryotic genome annotation, Bioinformatics, 30,2068-2069.
  - Site: http://www.vicbioinformatics.com/software.prokka.shtml
  - Version: 1.14.5
  - License: GPLv2
  - Note: **The NCBI tool tbl2asn included within PROKKA can have very slow runtimes (up to several hours) while it is dealing with numerous contigs, such as when we input metagenomic data. We modified the code to allow parallel processing using tbl2asn.**

- tRNAscan
  - Citation: Lowe, T.M. and Eddy, S.R. (1997) tRNAscan-SE: a program for improved detection of transfer RNA genes in genomic sequence, Nucleic acids research, 25, 955-964.
  - Site: http://lowelab.ucsc.edu/tRNAscan-SE/
  - Version: 1.3.1
  - License: GPLv2

- Barrnap
  - Citation:
  - Site: http://www.vicbioinformatics.com/software.barrnap.shtml
  - Version: 0.9
  - License: GPLv3
- BLAST+
  - Citation: Camacho, C., et al. (2009) BLAST+: architecture and applications, BMC bioinformatics, 10, 421.
  - Site: ftp://ftp.ncbi.nlm.nih.gov/blast/executables/blast+/2.2.29/
  - Version: 2.10.0
  - License: Public domain
- blastall
  - Citation: Altschul, S.F., et al. (1990) Basic local alignment search tool, Journal of molecular biology, 215, 403-410.
  - Site: ftp://ftp.ncbi.nlm.nih.gov/blast/executables/release/2.2.26/
  - Version: 2.2.26
  - License: Public domain
- Phage_Finder
  - Citation: Fouts, D.E. (2006) Phage_Finder: automated identification and classification of prophage regions in complete bacterial genome sequences, Nucleic acids research, 34, 5839-5851.
  - Site: http://phage-finder.sourceforge.net/
  - Version: 2.1
  - License: GPLv3
- Glimmer
  - Citation: Delcher, A.L., et al. (2007) Identifying bacterial genes and endosymbiont DNA with Glimmer, Bioinformatics, 23, 673-679.
  - Site: http://ccb.jhu.edu/software/glimmer/index.shtml
  - Version: 302b
  - License: Artistic License
- ARAGORN
  - Citation: Laslett, D. and Canback, B. (2004) ARAGORN, a program to detect tRNA genes and tmRNA genes in nucleotide sequences, Nucleic acids research, 32, 11-16.
  - Site: http://mbio-serv2.mbioekol.lu.se/ARAGORN/
  - Version: 1.2.36
  - License: GPLv2
- Prodigal
  - Citation: Hyatt, D., et al. (2010) Prodigal: prokaryotic gene recognition and translation initiation site identification, BMC bioinformatics, 11, 119.

  - Site: http://prodigal.ornl.gov/
  - Version: 2_60
  - License: GPLv3
- tbl2asn
  - Citation:
  - Site: http://www.ncbi.nlm.nih.gov/genbank/tbl2asn2/
  - Version: 25.8 (2022 Jun 13)
  - License: Public Domain

Warning: tbl2asn must be compiled within the past year to function. We attempt to recompile every 6 months or so. Most recent compilation is 27 Feb 2018

- AntiSmash
  - Citation: Kai Blin et al. (2021) antiSMASH 6.0: improving cluster detection and comparison capabilities, Nucleic Acids Research, Volume 49, Issue W1, 2 July 2021, Pages W29–W35
  - Site: https://antismash.secondarymetabolites.org/#!/start
  - Version: 6.1.1
  - License: AGPL-3.0

9.3 Alignment

- HMMER3
  - Citation: Eddy, S.R. (2011) Accelerated Profile HMM Searches, PLoS computational biology, 7, e1002195
  - Site: http://hmmer.janelia.org/
  - Version: 3.1b1
  - License: GPLv3
- Infernal
  - Citation: Nawrocki, E.P. and Eddy, S.R. (2013) Infernal 1.1: 100-fold faster RNA homology searches, Bioinformatics, 29, 2933-2935.
  - Site: http://infernal.janelia.org/
  - Version: 1.1rc4
  - License: GPLv3
- Bowtie 2
  - Citation: Langmead, B. and Salzberg, S.L. (2012) Fast gapped-read alignment with Bowtie 2, Nature methods, 9, 357-359.
  - Site: http://bowtie-bio.sourceforge.net/bowtie2/index.shtml
  - Version: 2.5.1
  - License: GPLv3

- BWA
  - Citation: Li, H. and Durbin, R. (2009) Fast and accurate short read alignment with Burrows-Wheeler transform, Bioinformatics, 25, 1754-1760.
  - Site: http://bio-bwa.sourceforge.net/
  - Version: 0.7.12
  - License: GPLv3
- MUMmer3
  - Citation: Kurtz, S., et al. (2004) Versatile and open software for comparing large genomes, Genome biology, 5, R12.
  - Site: http://mummer.sourceforge.net/
  - Version: 3.23
  - License: GPLv3
- RAPSearch2
  - Citation: Zhao et al. (2012) RAPSearch2: a fast and memory-efficient protein similarity search tool for next-generation sequencing data. Bioinformatics. 2012 Jan 1;28(1):125-6
  - Site: http://omics.informatics.indiana.edu/mg/RAPSearch2/
  - Version: 2.23
  - License: GPL
- minimap2
  - Citation: Li, H. (2018) Minimap2: fast pairwise alignment for nucleotide sequences. Bioinformatics, 34:3094-3100.
  - Site: https://github.com/lh3/minimap2
  - Version: 2.24
  - License: MIT
- diamond
  - Citation: Buchfink, Xie C., D. Huson (2015) Fast and sensitive protein alignment using DIAMOND, Nature Methods 12, 59-60
  - Site: https://github.com/bbuchfink/diamond
  - Version: v0.9.22.123
  - License: GPLv3

9.4 Taxonomy Classification

- Kraken2
  - Citation: Wood, D.E. and Salzberg, S.L. (2014) Kraken: ultrafast metagenomic sequence classification using exact alignments, Genome biology, 15, R46.
  - Site: http://ccb.jhu.edu/software/kraken2/
  - Version: 2.0.7-beta

  - License: MIT

- Metaphlan
  - Citation: Blanco-Míguez A., et al. (2023) Extending and improving metagenomic taxonomic profiling with uncharacterized species using MetaPhlAn 4. Nat Biotechnol. 2023 Feb 23.
  - Site: http://huttenhower.sph.harvard.edu/metaphlan4
  - Version: 4.0.6
  - License: MIT License
- GOTTCHA
  - Citation: Tracey Allen K. Freitas, Po-E Li, Matthew B. Scholz, Patrick S. G. Chain (2015) Accurate Metagenome characterization using a hierarchical suite of unique signatures. Nucleic Acids Research (DOI: 10.1093/nar/gkv180)
  - Site: http://lanl-bioinformatics.github.io/GOTTCHA/
  - Version: 1.0c
  - License: GPLv3
- GOTTCHA2
  - Citation:
  - Site: https://gitlab.com/poeli/GOTTCHA2
  - Version: 2.1.6 BETA
  - License: BSD 3-Clause

9.5 Phylogeny

- FastTree
  - Citation: Morgan N. Price, Paramvir S. Dehal, and Adam P. Arkin. 2009. FastTree: Computing Large Minimum Evolution Trees with Profiles instead of a Distance Matrix. Mol Biol Evol (2009) 26 (7): 1641-1650
  - Site: http://www.microbesonline.org/fasttree/
  - Version: 2.1.9
  - License: GPLv2
- RAxML
  - Citation: Stamatakis,A. 2014. RAxML version 8: A tool for phylogenetic analysis and post-analysis of large phylogenies. Bioinformatics, 30:1312-1313
  - Site: http://sco.h-its.org/exelixis/web/software/raxml/index.html
  - Version: 8.0.26
  - License: GPLv2
- Bio::Phylo
  - Citation: Rutger A Vos, Jason Caravas, Klaas Hartmann, Mark A Jensen and Chase Miller, (2011). Bio::Phylo - phyloinformatic analysis using Perl. BMC Bioinformatics 12:63.
  - Site: http://search.cpan.org/~rvosa/Bio-Phylo/

  - Version: 0.58
  - License: GPLv3

- PhaME
  - Citation: Sanaa Afroz Ahmed, Chien-Chi Lo, Po-E Li, Karen W Davenport, Patrick S.G. Chain. From raw reads to trees: Whole genome SNP phylogenetics across the tree of life. bioRxiv doi: http://dx.doi.org/10.1101/032250
  - Site: https://github.com/LANL-Bioinformatics/PhaME/
  - Version: 1.0
  - License: GPLv3

9.6 Specialty Genes

- ShortBRED
  - Citation: Kaminski J, et al. (2015) High-specificity targeted functional profiling in microbial communities with ShortBRED. PLoS Comput Biol.18;11(12):e1004557.
  - Site: https://huttenhower.sph.harvard.edu/shortbred
  - Version: 0.9.4M
  - License: MIT
- RGI (Resistance Gene Identifier)
  - Citation: McArthur & Wright. (2015) Bioinformatics of antimicrobial resistance in the age of molecular epidemiology. Current Opinion in Microbiology, 27, 45-50.
  - Site: https://card.mcmaster.ca/analyze/rgi
  - Version: 5.2.1
  - License: Apache Software License

9.7 Metagenome

- MaxBin2
  - Citation: Wu YW, et al. (2016) MaxBin 2.0: an automated binning algorithm to recover genomes from multiple metagenomic datasets", Bioinformatics, 32(4): 605-607, 2016.
  - Site: https://downloads.jbei.org/data/microbial_communities/MaxBin/MaxBin.html
  - Version: 2.2.6
  - License: BSD
- CheckM
  - Citation: Parks DH, et al. (2015) CheckM: assessing the quality of microbial genomes recovered from isolates, single cells, and metagenomes. Genome Research, 25: 1043–1055.
  - Site: https://ecogenomics.github.io/CheckM/
  - Version: 1.2.2
  - License: GPLv3

9.8 Visualization and Graphic User Interface

- jsPhyloSVG
  - Citation: Smits SA, Ouverney CC, (2010) jsPhyloSVG: A Javascript Library for Visualizing Interactive and Vector-Based Phylogenetic Trees on the Web. PLoS ONE 5(8): e12267.
  - Site: http://www.jsphylosvg.com
  - Version: 1.55
  - License: GPL
- JBrowse
  - Citation: Skinner, M.E., et al. (2009) JBrowse: a next-generation genome browser, Genome research, 19, 1630-1638.
  - Site: http://jbrowse.org
  - Version: 1.16.8
  - License: Artistic License 2.0/LGPLv.1
- KronaTools
  - Citation: Ondov, B.D., Bergman, N.H. and Phillippy, A.M. (2011) Interactive metagenomic visualization in a Web browser, BMC bioinformatics, 12, 385.
  - Site: http://sourceforge.net/projects/krona/
  - Version: 2.8.1
  - License: BSD
- JQuery
  - Site: http://jquery.com/
  - Version: 1.10.2
  - License: MIT
- JQuery Mobile
  - Site: http://jquerymobile.com
  - Version: 1.4.3
  - License: CC0
- DataTables
  - Site: https://datatables.net
  - Version: 1.10.11
  - License: MIT
- jQuery File Tree
  - Site: http://www.abeautifulsite.net/jquery-file-tree/
  - Version: 1.01
  - License: GPL and MIT
- Raphael - JavaScript Vector Library

  - Site: http://dmitrybaranovskiy.github.io/raphael/
  - Version: 1.4.3
  - License: MIT
- Tooltipster
  - Site: http://iamceege.github.io/tooltipster/
  - Version: 3.2.6
  - License: MIT
- Lazy Load XT
  - Site: http://ressio.github.io/lazy-load-xt/
  - Version: 1.0.6
  - License: MIT
- Plupload
  - Site: http://www.plupload.com
  - Version: 2.1.7
  - License: GPLv2 and OEM
- hello.js
  - Site: http://adodson.com/hello.js/
  - Version: 1.8.1
  - License: MIT
- bokeh
  - Citation: Bokeh Development Team (2014). Bokeh: Python library for interactive visualization
  - Site: https://bokeh.pydata.org/en/latest/
  - Version: 0.12.10
  - License: BSD 3-Clause

9.9 Utility

- Chromium
  - Citation:
  - Site: https://www.chromium.org
  - Version: 95.0.4615.0
  - License: Google-authored portion is released under the BSD license.
- BEDTools
  - Citation: Quinlan, A.R. and Hall, I.M. (2010) BEDTools: a flexible suite of utilities for comparing genomic features, Bioinformatics, 26, 841-842.
  - Site: https://github.com/arq5x/bedtools2
  - Version: 2.19.1

  - License: GPLv2

- Pilon
  - Citation: Walker BJ et al. (2014) Pilon: an integrated tool for comprehensive microbial variant detection and genome assembly improvement. PLoS One. 2014 Nov 19;9(11):e112963.
  - Site: https://github.com/broadinstitute/pilon
  - Version: 1.23
  - License: GPLv2 & MIT
- R
  - Citation: R Core Team (2013). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL http://www.R-project.org/.
  - Site: http://www.r-project.org/
  - Version: 3.6.3
  - License: GPLv2
- R_Packages
  - Custom built direcotry containing all the packages required to install R packages offline
  - The majority of the packages were downloaded automatically using the following R commands.

Function to get dependencies and imports for a given list of packages.

```
getPackages <- function(packs){
        packages <- unlist(
        tools::package_dependencies(packs, available.packages(), which=c(
↪"Depends", "Imports"), recursive=TRUE)
        )
packages <- union(packs, packages)
packages
}
```

Use the function by providing the names of the desired packages.

```
packages <- getPackages(c("packageName", "packageName2"))
# For example
#packages <- getPackages(c("MetaComp","gtable","gridExtra","devtools",
↪"phyloseq","webshot","plotly","shiny","DT","ape", "igraph", "vegan","BH
↪","plogr","dplyr","ade4","codetools","iterators","foreach","gplots"))
```

Download packages to current/desired directory.

```
download.packages(packages, destdir="./", type="source")
```

  - The packages specific to bioconductor ('phyloseq', 'Biobase', 'biomformat', 'rhdf5', 'BiocGenerics', 'Biostrings', 'multtest','S4Vectors','IRanges','XVector','Rhdf5lib','zlibbioc') needed to be manually downloade from the site.
  - stringi defaults to downloading icudt55I.zip from online, the following method, from their documentation, was used to build a custom stringi package to avoid connecting to the internet.:

```
1. File the `git clone https://github.com/gagolews/stringi.git`␣
↪command.
2. Edit the `.Rbuildignore` file and get rid of the `^src/icu55/data`␣
↪line.
3. Run `R CMD build stringi_dir_name`.
```

index the downloaded packages into PACKAGES files.

```
require(tools)
write_PACKAGES('.')
```

- MetaComp: EDGE Taxonomy Assignments Visualization
  - Citation:
  - Site: https://cran.r-project.org/
  - Version: 1.0.2
  - License: BSD 3-Clause
- GNU_parallel
  - Citation: O. Tange (2011): GNU Parallel - The Command-Line Power Tool, ;login: The USENIX Magazine, February 2011:42-47
  - Site: http://www.gnu.org/software/parallel/
  - Version: 20190422
  - License: GPLv3
- tabix
  - Citation:
  - Site: http://sourceforge.net/projects/samtools/files/tabix/
  - Version: 0.2.6
  - License: MIT/Expat License
- Primer3
  - Citation: Untergasser, A., et al. (2012) Primer3–new capabilities and interfaces, Nucleic acids research, 40, e115.
  - Site: http://primer3.sourceforge.net/
  - Version: 2.3.5
  - License: GPLv2
- SAMtools
  - Citation: Li, H., et al. (2009) The Sequence Alignment/Map format and SAMtools, Bioinformatics, 25, 2078-2079.
  - Site: http://www.htslib.org/
  - Version: 1.16.1
  - License: MIT
- FaQCs

  - Citation: Chienchi Lo, PatrickS.G. Chain (2014) Rapid evaluation and Quality Control of Next Generation Sequencing Data with FaQCs. BMC Bioinformatics. 2014 Nov 19;15
  - Site: https://github.com/LANL-Bioinformatics/FaQCs
  - Version: 2.08
  - License: GPLv3
- Seqtk
  - Citation: Heng Li https://github.com/lh3/seqtk
  - Site: https://github.com/lh3/seqtk
  - Version: 1.3
  - License: MIT
- NanoPlot
  - Citation: De Coster W, et al.(2018) NanoPack: visualizing and processing long read sequencing data, Bioinformatics. 2018 Mar 14.
  - Site: https://github.com/wdecoster/NanoPlot
  - Version: 1.40.0
  - License: GPLv3
- Porechop
  - Citation:
  - Site: https://github.com/rrwick/Porechop
  - Version: 0.2.4
  - License: GPLv3
- wigToBigWig
  - Citation: Kent, W.J., et al. (2010) BigWig and BigBed: enabling browsing of large distributed datasets, Bioinformatics, 26, 2204-2207.
  - Site: https://genome.ucsc.edu/goldenPath/help/bigWig.html#Ex3
  - Version: 4
  - License: Free for academic, nonprofit, and personal use. A license is required for commercial usage.
- sratoolkit
  - Citation:
  - Site: https://github.com/ncbi/sra-tools
  - Version: 3.0.0
  - License: Public Domain
- ea-utils
  - Citation: Erik Aronesty (2011) ea-utils : "Command-line tools for processing biological sequencing data"
  - Site: https://code.google.com/archive/p/ea-utils/
  - Version: 1.1.2-537
  - License: MIT License

- Mambaforge (Python 3)
  - Citation:
  - Site: https://github.com/conda-forge/miniforge
  - Version: 22.11.1-4
  - License: 3-clause BSD

9.10 Amplicon Analysis

- QIIME2
  - Citation: Caporaso et al. (2010) QIIME allows analysis of high-throughput community sequencing data. Nat Methods. 2010 May;7(5):335-6
  - Site: http://qiime2.org/
  - Version: 2023.5
  - License: BSD 3-Clause
- DETEQT: Diagnostic targeted sequencing adjudication
  - Citation: Conrad TA et al. (2019) Diagnostic targETEd seQuencing adjudicaTion (DETEQT): Algorithms for Adjudicating Targeted Infectious Disease Next-Generation Sequencing Panels.
  - Site: https://github.com/LANL-Bioinformatics/DETEQT
  - Version: 0.3.0
  - License: GPLv3

9.11 RNA-Seq Analysis

- PyPiReT: Pipeline for Reference based Transcriptomics.
  - Citation:
  - Site: https://github.com/mshakya/PyPiReT
  - Version: 0.3.2
  - License: GPLv3

CHAPTER 10

FAQs and Troubleshooting

10.1 FAQs

- Can I speed up the process?

  You may increase the number of CPUs to be used from the "additional options" of the input section. The default and minimum value is one-eighth of total number of server CPUs.

- There is no enough disk space for storing projects data. How do I do?

  There is an archive project action which will move the whole project directory to the directory path configured in the $EDGE_HOME/sys.properties. We also recommend a symbolic link for the $EDGE_HOME/edge_ui/EDGE_input/public/ directory which points to the location where the users' (or sequencing centers') raw data are stored, obviating unnecessary data transfer via web protocol and saving local storage.

- How to decide various QC parameters?

  The default parameters should be sufficient for most cases. However, if you have very depth coverage of the sequencing data, you may increase the trim quality level and average quality cutoff to only use high quality data.

- How to set K-mer size for IDBA_UD assembly?

  By default, it starts from kmer=31 and iterative step by adding 20 to maximum kmer=121. Larger K-mers would have higher rate of uniqueness in the genome and would make the graph simpler, but it requires deep sequencing depth and longer read length to guarantee the overlap at any genomic location and it is much more sensitive to sequencing errors and heterozygosity. Professor Titus Brown has a good blog on general k-mer size discussion.

- How many reference genomes for Reference-Based Analysis and Phylogenetic Analysis can be used from the EDGE GUI?

  The default maximum is 20 and there is a minimum 3 genomes criteria for the Phylogenetic Analysis. But it can be configured when installing EDGE.

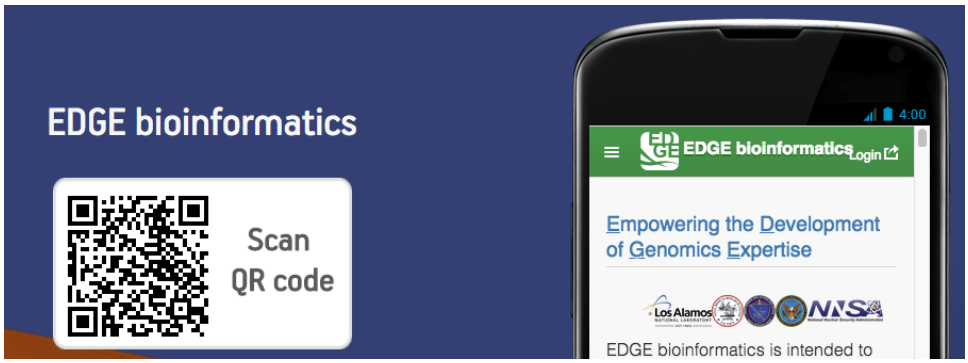
- Which aligner should I choose?

We use default setting of the aligner. Bowtie2 default is for global alignment and BWA mem algorithm will do local alignment. If users would like to overwrite the setting, users can use "Aligner Options" to do so. For example, use "–local" to run bowtie2 with local alignment mode. Or, use "-x ont2d" to run BWA mem with Nanopore reads.

- How to make an app icon on the mobile device?

  Launch the Safari browser on Apple's iOS and navigate to the https://bioedge.lanl.gov/edge_ui/ or your EDGE instance website. (Please refresh the page few times to update the cache) Tap the Share button on the browser's toolbar — that's the rectangle with an arrow pointing upward. It's on the bar at the top of the screen on an iPad, and on the bar at the bottom of the screen on an iPhone or iPod Touch. Tap the Add to Home Screen icon in the Share menu.

  Launch Chrome for Android and open the https://bioedge.lanl.gov/edge_ui/ or your EDGE instance website. (Please refresh the page few times to update the cache) Tap the menu button and tap Add to homescreen. You'll be able to enter a name for the shortcut and then Chrome will add it to your home screen. Alternatively, we have bioedge Web App as APK file to download and install in your android device too. You can download by scan the QR code below.



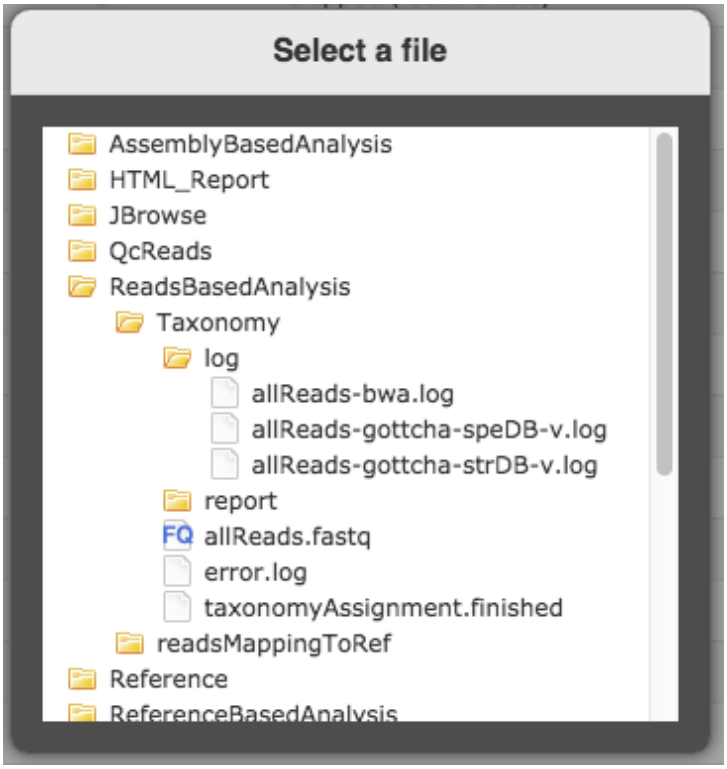
- Why a job is queued and never autostarted?

  The queued job had too much CPUs request. The autorun feature will start running queued job when there is available CPU resource. The queued job CPUs usage plus running jobs CPUs usage should be less than (<) **edgeui_tol_cpu** configured in the $EDGE_HOME/edge_ui/sys.properties.

- Why some of the taxonomy profiling result are N.A.?

Tool	#Reads	%Reads	Rank	Top1	Top2	Top3
gottcha-speDB-b	189,392	3.2	species	Enterococcus faecium	Parabacteroides distasonis	Bacteroides vulgatus
gottcha-speDB-v	539	0.0	species	Pepper mild mottle virus	Streptococcus phage 20617	Paprika mild mottle virus
bwa	**0**	**0.0**	**species**	**N/A**	**N/A**	N/A

  Please check the log file to give us more information. For above example on BWA result, at web UI, you can open log file by Clicking the link next to "Output Directory" at "General" section -> ReadsBasedAnalysis -> Taxonomy -> log -> allReads-bwa.log.



In this case, it is out of memory. EDGE requires at least 16G memory. see System requirements

For machine with < 32Gb memory, we suggest to use the smaller BWA index (~13Gb) and contains the databases for bwa taxonomic identification pipeline

```
wget -c https://edge-dl.lanl.gov/EDGE/dev/edge_dev_bwa_mini_index.tgz
```

10.2 Troubleshooting

- Process.log and error.log files may help on the troubleshooting.

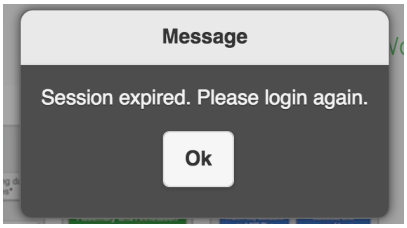
10.2.1 EDGE WEB GUI

- In the GUI, if you are trying to enter information into a specific field and it is grayed out or won't let you, try refreshing the page by clicking the icon in the right top of the browser window.
- After installation, I can login but cannot select any files for input. the selection pop-up is empty.



This could be the permission issue on the EDGE_input/EDGE_output directory for Apache user. Please see Apache Web Server Configuration

- I can not login to EDGE, it keeps saying Session expired.



The login session will expire in 12 hours. If you keeps get session expired message. It may indicate the '/' (root) space is full. Please try to clean up log files or others you/admins can delete. For example, /var/log/message-2016xxxx is the archived log rotations which can be deleted for the space.

10.2.2 Coverage Issues

- Average Fold Coverage reported in the HTML output and by the output tables generated in {output directory}/AssemblyBasedAnalysis/ReadsMappingToContigs/ are calculated with mpileup using the default options for metagenomes. These settings discount reads that are unpaired within a contig or with an insert size out of the expected bounds. This will result in an underreporting of the average fold coverage based on the generated BAM file, but one that the team feels is more accurate given the intended use of this environment.

10.2.3 Data Migration

- The preferred method of transferring data to the EDGE appliance is via SFTP. Using an SFTP client such as FileZilla, connect to port 22 using your system's username and password.
- In the case of very large transfers, you may wish to use a USB hard drive or thumb drive.
- If the data is being transferred from another LINUX machine, the server will recognize partitions that use the FAT, ext2, ext3, or ext4 filesystems.
- **If the data is being transferred from a Windows machine, the partition may use the NTFS filesystem. If this is the case, th**
  - Open the command line interface by clicking the Applications menu in the top left corner (or use SSH to connect to the system).
  - Enter the command: ''sudo yum install ntfs-3g ntfs-3g-devel -y"

  - Enter your password if required.

- After a reboot, you should be able to connect your Windows hard drive to the system, and it will mount like a normal disk.

10.2.4 Known Issues

- Installations on CentOS 6.4 with Apache 2.2 are known to have difficulty executing jobs that have ".real" anywhere in the name. This is due to a CGI execution issue. The recommended resolution is to use an underscore in place of the period, or simply name your job something else.

10.3 Discussions / Bugs Reporting

- We welcome questions, feedback and bug reports that may help us improve the EDGE platform. Ideally, any bug report should include the process.log, the error.log and the failed module log files. If it is a system error, the tomcat, apache and mysql logs are helpful. If possible, please share a subset of the input dataset/files for us to re-create the bug. Here is a bug report template for your reference.:

```
**Describe the bug**
A clear and concise description of what the bug is.

**To Reproduce**
Steps to reproduce the behavior:

Go to '...'
Click on '....'
Scroll down to '....'
See error
Expected behavior
A clear and concise description of what you expected to happen.

**Screenshots**
If applicable, add screenshots to help explain your problem.

**Desktop (please complete the following information if applicable):**

OS: [e.g. iOS]
Browser [e.g. chrome, safari]
Browser Version [e.g. 22]
EDGE Version [e.g. 2.3.1]

**Additional context**
If you ran a project, can you provide process.log, error.log and the failed␣
↪module log files. If it is system error, the tomcat, apache and mysql logs are␣
↪helpful.
```

- We have created a mailing list for EDGE users. If you would like to recieve notifications about the updates and join the discussion, please join the mailing list by becoming the member of edge-users groups.

  EDGE user's google group

- We appreciate any feedback or concerns you may have about EDGE. If you encounter any bugs, you can report them to our GitHub issue tracker.

  Github issue tracker

- Any other questions? You are welcome to *Contact Us and Citation* (page 97)

CHAPTER 11

Copyright

Copyright (2018). Triad National Security, LLC. All rights reserved.

This program was produced under U.S. Government contract 89233218CNA000001 for Los Alamos National Laboratory (LANL), which is operated by Triad National Security, LLC for the U.S. Department of Energy/National Nuclear Security Administration.

All rights in the program are reserved by Triad National Security, LLC, and the U.S. Department of Energy/National Nuclear Security Administration. The Government is granted for itself and others acting on its behalf a nonexclusive, paid-up, irrevocable worldwide license in this material to reproduce, prepare derivative works, distribute copies to the public, perform publicly and display publicly, and to permit others to do so.

This is open source software; you can redistribute it and/or modify it under the terms of the GPLv3 License. If software is modified to produce derivative works, such modified software should be clearly marked, so as not to confuse it with the version available from LANL. Full text of the GPLv3 License can be found in the License file in the main development branch of the repository.

CHAPTER 12

Contact Us and Citation

Questions? Concerns? Please feel free to email our google group at edge-users@googlegroups.com or contact a dev team member listed below.

Name	Email
Patrick Chain	pchain@lanl.gov
Chien-Chi Lo	chienchi@lanl.gov
Paul Li	po-e@lanl.gov
Karen Davenport	kwdavenport@lanl.gov
Logan Voegtly	logan.j.voegtly.ctr@mail.mil
Kim Bishop-Lilly	kimberly.a.bishop-lilly.ctr@mail.mil

12.1 Citation

Enabling the democratization of the genomics revolution with a fully integrated web-based bioinformatics platform

Po-E Li; Chien-Chi Lo; Joseph J. Anderson; Karen W. Davenport; Kimberly A. Bishop-Lilly; Yan Xu; Sanaa Ahmed; Shihai Feng; Vishwesh P. Mokashi; Patrick S.G. Chain

Nucleic Acids Research 2016;

doi: 10.1093/nar/gkw1027