Swabs to Genomes: A Comprehensive Workflow

David A Coil, Madison I Dunitz, Guillaume Jospin, Jonathan A Eisen, Aaron E Darling, Jenna M Lang

The sequencing, assembly, and basic analysis of microbial genomes, once a painstaking and expensive undertaking, has become almost trivial for research labs with access to standard molecular biology and computational tools. However, there are a wide variety of options available for DNA library preparation and sequencing, and inexperience with bioinformatics can pose a significant barrier to entry for many who may be interested in microbial genomics. The objective of the present study was to design, test, troubleshoot, and publish a simple, comprehensive workflow from the collection of an environmental sample (a swab) to a published microbial genome; empowering even a lab or classroom with limited resources and bioinformatics experience to perform it.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

Swabs to Genomes: A Comprehensive Workflow

Madison Dunitz[1,*],
Jenna M. Lang [1,*],
Guillaume Jospin[1],
Aaron E. Darling[2],
Jonathan A. Eisen[1,#],
David A. Coil[1]
1 *UC Davis, Genome Center.*
2 *ithree institute, University of Technology Sydney, Australia*
* These authors contributed equally to this work.
# Corresponding author: jaeisen@ucdavis.edu

August 7, 2014

1 Introduction

Thanks to decreases in cost and difficulty, sequencing the genome of a microorganism is becoming a relatively common activity in many research and educational institutions. However, such microbial genome sequencing is still far from routine or simple. Our objective in this project here was to design, test, troubleshoot, and publish a comprehensive workflow for microbial genome sequencing, encompassing everything from culturing new organisms to depositing sequence data; enabling even a lab with limited resources and bioinformatics experience to perform it.

In the fall of 2011 our lab began a project with the goal of having undergraduate students generate genome sequences for microorganisms isolated from the "built environment". The project focused on the built environment because it was part of the larger "microBEnet" (microbiology of the built environment network) effort. This project was initiated because it could serve many purposes including (1) engaging undergraduates in research on microbiology of the built environment (2) generating "reference genomes" for microbes that are found in the built environment (3) providing material to enhance our ability to communicate about microbes in the built environment and (4) providing a testing ground for the development of material for educational activities on microbiology of the built environment. As part of this project, undergraduate students went through the process of isolating, identifying, sequencing and assembling microbial genomes, followed by submission to databases housed by The National Center for Biotechnology Information (NCBI)and publication of each genome

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

[27][6][19][13][9][22]. Through the course of the project we found that, despite the so-called democratization of genome sequencing and the availability of diverse tools making many of the steps easier, (e.g. kits for library prep, relatively cheap sequencing, bioinformatics pipelines), there were still a significant number of stumbling blocks. Moreover, some portions of the project involve choosing between a wide variety of options (e.g. choice of assembly program) which can create a large activation energy for a lab without a bioinformatician. Each option comes with its own advantages and disadvantages in terms of complexity, expense, computing power, time, and experience required. In this workflow we have describe an approach to genome sequencing that allows a researcher to go from a swab to a published paper. We used this workflow to process a novel *Tatumella* sp. isolate and publish the genome [15]. The data from every step of the workflow, using this *Tatumella* isolate, is available on Figshare [10]

The sequencing and *de novo* assembly of genomes has already yielded enormous scientific insight revolutionizing a wide range of fields, from epidemiology to ecology. Our hope is that this workflow will help make this revolution more accessible to all scientists, as well as present educational opportunities for undergraduate researchers and classes.

There are several excellent resources that focus on smaller portions of this entire process, usually assembly and/or annotation. Examples include the Computational Genomics Pipeline [23] and a "Beginner's guide to comparative bacterial genome analysis" [18] both of which start with data on the sequences of individual small fragments from an organism's genome (each DNA sequence generated by a sequencing system is known as a "read").

2 General Notes on Bioinformatics

2.1 Command Line/Terminal Tutorial

This workflow is written assuming that the user is using a computer running Mac OS or Linux. It is also possible to carry out many of the computational parts of this workflow in a Windows environment but getting these steps to work in Windows is outside the scope of this project.

Some parts of this workflow require the user to provide text instructions for software programs by using a command line interface. While potentially intimidating to computer novices, the use of command line interfaces is sometimes necessary (e.g., some programs do not have graphical interfaces) and is also sometimes much more efficient. To access the command line on a Mac open the Terminal program in the Utilities folder under Applications.

When this application is launched a new window will appear. This is known as a "terminal" or a "terminal window." In the terminal window, you can interact with your computer without using a mouse. Many popular programs have a GUI (Graphical User Interface) but some programs used in this workflow will not. So, instead of double-clicking to make a program run, you will type a command in the terminal window. We will walk you through how to run

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

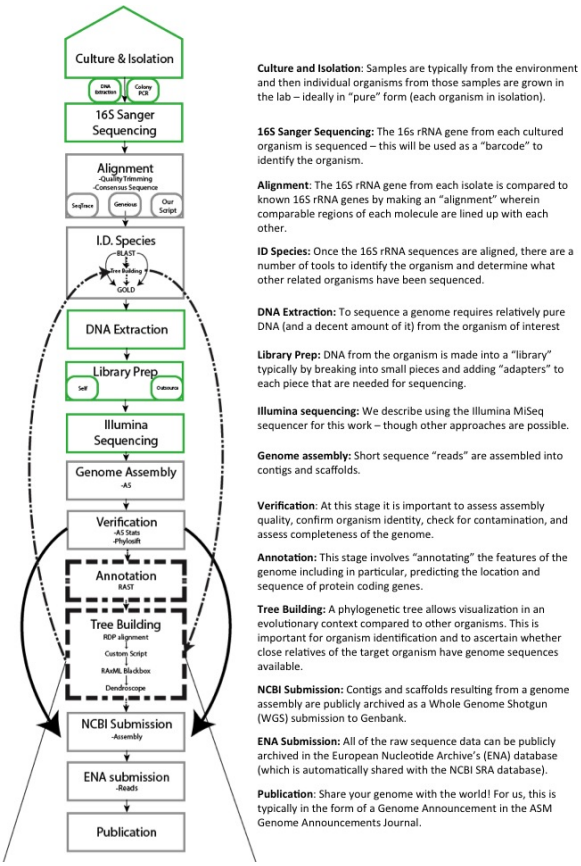

Figure 1: -Overview of the workflow

all of the programs required for this workflow, but you must first acquire a basic familiarity with how to interact with your computer through the terminal window. Below is a list of commands that will be required to use this workflow. There are many tutorials available to help you get started.

For more information on operating in the terminal, check out this informative video: `https://www.youtube.com/watch?v=zRZT4nQP3sE`

And this interactive tutorial: `http://www.ee.surrey.ac.uk/Teaching/Unix/`

2.2 Summary of commands and terms

$ **ls** lists files and directories (folders). If left as just "ls" it will list the files and directories in your current location. If a "path" is added afterwards (e.g., ls /usr) it will list the files and directories in that location.

$ **cd** use to change directories

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

$ **cd ..** use to move up one directory

$ **cd directory_name** use to move to that directory

$ **cd ~** use to move to the home directory of the current user

$ **grep "some pattern" file_name** displays lines that match the pattern you are looking for. If a line contains the same character multiple times it will only be displayed once.

$ **grep –c "what you want to count" file_name** counts the number of lines containing a specific character or sequence of characters

$ **less file_name** view a file

A few quick definitions:

command line – the command line is where you type commands in a terminal window

script – a computer program. Usually computer programs are called scripts when they perform relatively simple functions that are limited in scope. Scripts are typically only run from the command line

directory – a folder

compile - turning a human-readable file into a computer-executable program

2.3 Software updates

Software packages are updated with varying frequencies. Some software updates will render the instructions offered here obsolete. When this occurs, you should consult with the software manual for help. An internet search with a description of the problem you are having may prove helpful. Another option is to email the software developer; many are remarkably responsive. As a last resort, consult with a colleague who is more comfortable with bioinformatics. It is customary to offer a small favor or gift. Most software updates will require only minor modifications. For example, we might provide you with instructions to type:

```
./software_1.2.0/software.py
```

but a more recent release might necessitate:

```
./software_1.3.0/software.py
```

3 General notes on molecular and microbiology

This workflow assumes a basic knowledge of molecular biology and "sterile technique" (methods for carrying out lab experiments without contamination from living microorganisms). The starting point is the collection of microbes from a surface with a swab. We will cover the steps necessary to take a sample through plating, dilution streaking, overnight growth, creating a glycerol stock, 16s rDNA PCR, and preparation for Sanger sequencing to determine the identity of your bacterial or archaeal isolate.

Throughout the "Isolation" section we refer to "agar" and "culture media". The choice of media will depend on the goals of the particular project. Some

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

factors to consider when selecting media and conditions for growth include: 1. What type of organism do you want to isolate?

2. Are there types of organisms (e.g., pathogens) that you would prefer not to isolate? For example, swabbing people and growing samples on blood agar at 37 °C will often result in the isolation of pathogens.

3. How much time is available for growth and isolation?

    - growth rates differ both between organisms (e.g., species 1 versus species 2) and also in different conditions for the same organisms (e.g., species 1 at 20°C vs. 37°C)
    - for many microbes there is an "optimal growth temperature" (OGT - the temperature at which it grows best) but the OGT varies between species
    - you will be able to isolate a greater diversity of organisms if you allow a long time for slow-growing organisms to grow

4. What types of equipment are available to you?

- if an organism grows most happily at 37°C, then you will need to have an incubator and shaker available at that temperature.

For our previous work we have simply used a rich media such as lysogeny broth (LB) and growth at either room temperature or 37°C.

4 A brief introduction to phylogeny and systematics.

In order to identify to which organism a 16S rDNA sequence belongs, as well as to provide an evolutionary context for your organism of interest, we recommend inferring a phylogenetic tree comparing the new 16S sequence to other 16S sequences (see Section 11). Building such a phylogenetic tree is (relatively speaking) the easy part. Intelligent interpretation of the tree will require an investment of time, similar to the investment required to learn the basics of UNIX. Fortunately, there are a number of resources available for this purpose. We recommend this online tutorial `http://evolution.berkeley.edu/evolibrary/article/phylogenetics_02` or this paper by Baldauf [4] Here we provide a brief introduction to phylogenetic trees.

A phylogenetic tree is a diagram representing a model of evolutionary relationships. Phylogenetic trees have three main components: taxa, branches, and nodes. These are defined below:

- ***Taxon***. An individual or grouping of individuals. This could be individual sequences, species, families, phyla, etc. For phylogenetic analyses, the taxa that are drawn at the tips of branches are sometimes referred to as "leaves" on the tree.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

- ***Branch*** A representation of the evolution of a taxon over time (sometimes also known as an evolutionary lineage). There are three main types of branches in a tree. Terminal branches are those that lead to the tips or leaves in the tree. Internal branches connect branches to each other. And the root branch, also known as the root of the tree, is the branch that leads from the base of the tree to the first node in the tree.

- ***Node*** These are the points where individual branches end. In the internal parts of a phylogenetic tree, single branches can "split" producing multiple descendant branches. The point are which the branches split is known as an internal node. If a branch ends at a taxon, the end point is known as a "terminal node".

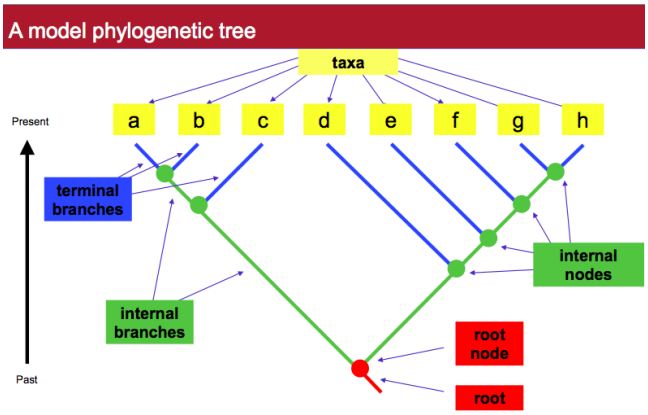

Figure 2: -A model phylogenetic tree showing nodes, branches and taxa

4.1 Some other information to know about trees.

- ***Branch rotation***. Each node in a tree can be rotated/spun around without changing the meaning of the tree. This is known as "branch rotation".
- ***Clade***. A group of organisms consisting of a single node and all the descendants of that node in a tree and nothing else.
- ***Monophyletic group***. A clade.
- ***More recent common ancestor*** of a group of taxa. A node in a tree where, going back in time, all of the branches leading up to the taxa in the group join together.
- ***Bootstrapping***. A statistical method used to measure how much a particular part of a phylogenetic tree is supported by all the data being used.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

- ***Ingroup***. The group of taxa being studied.
- ***Outgroup***. A taxon that separated in an evolutionary tree prior to the existence of the most recent common ancestor of the ingroup.

5 Isolation

This section will take you through the basics of isolating, culturing, and storing your organism.

5.1 Swab

Using a sterile cotton swab, wipe (i.e., "swab") the area you intend to sample for 10 to 15 seconds, as if you were trying to clean the area. Try to rotate the swab to ensure that all sides touch the surface.

5.2 Plate

Gently (so as not to break the agar surface) rub (aka "streak") the swab across the entire surface of an agar plate. Be sure to rotate the swab as you are doing so to ensure that all sides of the swab make contact with the plate. Incubate the plate at the desired temperature (in our case, usually 37°C or room temperature - but the desired temperature will depend on the project) for 1-3 days.

5.3 Dilution Streak (streaking for individual colonies) x2

After incubation, choose desired colonies (we typically attempt to maximize the diversity of colony morphologies) and dilution streak them onto individual plates. Dilution streaking involves a spreading out a chosen colony such that single colonies grow on a new plate (details can be easily found online).

After growth to visible colonies, repeat the dilution streaking to help ensure purity of the culture. Some organisms will only grow in tight association with others, and a mixed culture will prove difficult to work with and very difficult to sequence/assemble.

5.4 Liquid Culture

After the second dilution streaking, a liquid culture is needed both for long-term storage and for DNA extraction. Pick a single colony from each dilution streak plate into 5 mls of culture media and grow for 1-3 days until cloudy. Once the liquid culture is ready, prepare a 10% final concentration glycerol stock for long-term storage at -80°C from 2 ml of the sample.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

6 16S rDNA Sequencing and Analysis (Organism Identification)

Following the second dilution streaking, the organisms need to be identified. This is accomplished by determining and then analyzing the DNA sequence of the 16S rRNA gene. In this section we describe how the sequence of this gene is determined and readied for analysis. The general outline is as follows: DNA extraction, polymerase chain reaction (PCR) amplification of the 16S rRNA gene, and sequencing of the resulting PCR product using a method originally developed by Fred Sanger and now known as "Sanger sequencing". There are multiple approaches one can take to these steps. For example, the PCR reaction needs DNA from the organisms of interest. That DNA can come directly from a liquid culture of the organism (when this is used for PCR this is known as direct PCR). Alternatively, one can take a liquid culture and then isolate the DNA from that culture and use the "clean" DNA as material for the PCR. This adds an extra step to the process - a step known as "DNA extraction". Direct PCR significantly decreases the amount of work needed for preparation, but it can yield poorer results, both in terms of PCR success and resultant sequence quality. However, we recommend direct PCR when screening a large number of samples. DNA extraction can then be used for any recalcitrant samples. DNA extraction is significantly more work, but it often generates better Sanger sequences allowing for more accurate identification.

6.1 DNA Extraction

There are a number of different options for DNA isolation and which one should use depends on many factors including available equipment, experience, and cost. A standard approach in microbiology involves the use of phenol and chloroform extraction followed by ethanol precipitation, and any number of protocols for this approach can be found in books, articles and on the internet. A common alternative approach is to use a commercially available kit - there are many advantages to such kits - notably ease and lack of toxic chemicals. A disadvantage of kits is that they typically are more expensive per sample than other approaches (especially if one is only doing a few samples since most kits include materials for at a minimum 50 samples). For most projects, we use kits - especially the Promega-Wizard Genomic DNA Purification Kit.

Follow the protocol or kit instructions provided by the manufacturer and then proceed to "PCR reaction" below.

6.2 Direct PCR (if not extracting DNA)

Centrifuge 1 ml of the overnight culture until the cells form a pellet at the bottom of the tube (about 5 minutes at 10,000 g), pour off the liquid on top (a.k.a. the supernatant) and resuspend in 100 ul of sterile DNAase-free water. Incubate the samples at 100 deg C for 10 minutes to help lyse the cells. Use the resulting solution as the template in the PCR reaction below.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

6.3 PCR reaction

This reaction uses the 27F (AGAGTTTGATCMTGGCTCAG) and 1391R (GACGGGCGGTGTGTRCA) primers which amplify a near full-length bacterial (and many archaeal) 16S rRNA gene. Our lab uses standard PCR reagents (Qiagen or Kappa), with an annealing temperature of 54 deg C and an extension at 72 deg C of 90 seconds. Do not forget to include positive (any sample containing bacterial genomic DNA) and negative (e.g., just water) controls.

After PCR is completed, confirm the PCR reaction worked by agarose gel electrophoresis, all controls behaved as expected, and that you have DNA fragments of the correct size (˜1350bp).

6.4 Submit Samples for Sequencing

Very few single-researcher labs maintain Sanger sequencing capacity. However, there are a number of DNA sequencing facilities (commercial and academic) that provide sequencing services for researchers. They will handle as little as a single sample, or will allow you to submit an unlimited number of samples, typically arrayed in 96-well plates. You will typically provide both your PCR product as well as your PCR primers for sequencing. Don't forget to submit forward (27F) and reverse (1391R) reactions for each sample. Each facility will have its own guidelines concerning DNA and primer concentration. Our lab uses the UC DNA Sequencing Facility-UC Davis `http://dnaseq.ucdavis.edu`. If a quick internet search does not reveal the presence of a Sequencing Facility near you, most sequencing centers will allow you to ship samples to them for sequencing.

6.5 Sanger Sequence Processing

Upon receiving Sanger reads from a sequencing facility, typically via e-mail, it is necessary to do some pre-processing before they can be analyzed. These steps include quality trimming the reads, reverse complementing the reverse sequence, aligning the reads and generating a consensus sequence. There are very limited options for free software that allow the user to perform these steps. We recommend SeqTrace [35] for the user who wants to see the trace and process the sequences manually.

We have also created a script that will do all of these steps automatically, but does not allow you to adjust any of the parameters. The choice of our script (easy, little control) versus SeqTrace (more complex, more control) will depend on the user and the project.

6.6 Install and run SeqTrace

Download the program from `https://code.google.com/p/seqtrace/downloads/list`

Installation Directions `https://code.google.com/p/seqtrace/wiki/Installation`

Installing and running SeqTrace on a PC is simple, installing it on a Mac requires a few more steps than for a PC. The installation guide offers two options

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

for installing SeqTrace on a Mac, we recommend running SeqTrace with native GTK+

To install SeqTrace on a Mac you will need to download the PyGTK package from OSX. `http://sourceforge.net/projects/macpkg/files/PyGTK/2.24.0/PyGTK.pkg/download`

Confirm that you have Python version 2.x. You can do this by typing:

```
python --version
```

You should see something that looks like "Python 2.6.9" If you see Python 3.x, seek outside help to invoke an earlier version directly.

`http://www.python.org/download/releases/`

After downloading and unpacking the program, SeqTrace is ready to go. SeqTrace must be launched from a Terminal window. For a refresher or introduction to the Terminal see section 2. Move SeqTrace to your Applications folder.

Open a Terminal window and type:

```
./Applications/seqtrace-0.9.0/seqtrace.py
```

This syntax will only work if the SeqTrace folder's name is seqtrace-0.9.0, if you saved it under a different name you will need to replace seqtrace-0.9.0 with the name of that folder

This will launch SeqTrace from the terminal in a Python shell; you will need to keep the terminal window open while you are using the program.

SeqTrace provides excellent directions for using the program at `https://code.google.com/p/seqtrace/wiki/WorkingWithProjects`

6.7 Edit and Create a Consensus Sequence with SeqTrace

For this workflow we have found that the following is the simplest way to edit and create a consensus sequence from a forward and reverse read in SeqTrace.

1. Create a new project (File > New Project) Add your forward and reverse primer sequences here, we used 27F (AGAGTTTGATCMTGGCTCAG) and 1391R (GACGGGCGGTGTGTRCA) click ok
2. To add files go to Traces and click on Add trace files, then select the reads (.abi files) you want to work with.
3. The program is able to recognize forward and reverse reads from information in the file name if they are properly formatted.

- Go to Traces and click on Find and mark forward/reverse. The default setting looks for _F for forward and _R for reverse. This can be edited in the Project settings (you can pull it up by clicking on the picture of the tools at the top of the page) and changing the search strings under trace settings. For an example see Figure 3

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

- If the program is able to recognize the forward/reverse reads it will place an orange left pointing arrow in front of reverse reads and a blue right pointing arrow in front of forward reads. This step is not necessary to get a consensus sequence, it just makes organizing the reads easier.

4. Pull up the Project Settings by clicking on the picture of tools at the top of the page. Click on the Sequence Processing tab, under Sequence trimming unclick the Automatically trim sequence ends button. You should also decrease the Min. confidence score under Consensus settings. The default option is 30, which represents a 99.9% quality score, for many reads this will be too stringent and will not allow you to get enough overlap to create a consensus sequence. A minimum confidence score between 15 and 25 is normally okay but tuning may be required depending on your read quality. For an example see Figure 4.
5. Group your forward and reverse reads by highlighting both of them and clicking Group selected forward/reverse files (under Traces)
6. Under Sequences go to Generate Finished Sequences and click on for all trace files. (you will need to redo this every time you change the project settings).
7. To view your consensus sequence, click on the read pair group and then click on the magnifying glass at the top of the page. You should see something like Figure 5.
8. The Trace View shows the quality scores, the chromatogram display, and the raw base calls from both the forward and reverse reads, as well as the consensus sequence. The consensus sequence is the middle list of nucleotides. If the program is giving you a string of Ns where your forward and reverse reads do not overlap, you need to decrease the Min confidence score.
9. To export the consensus from the trace view, go to Sequence, hover on Export Sequences and select Export Sequences from Selected Trace Files. This will create a file containing the consensus sequence, which can then be used for analysis such as for searching for closely related sequences using the BLAST program [2] which can be used to identify the organism.

6.8 Custom Script to Create a Consensus Sequence (merge_sanger_16s.pl)

6.8.1 Download/Install

1. Create a new folder called Sanger_seq on your desktop
2. Download the zip file, containing three scripts (merge_sanger_16s.pl, cleanup.pl and subsample_reads.pl) from [24]
3. Open the zip file and move the merge_sanger_16s.pl file to the new Sanger_seq folder

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

6.8.2 MUSCLE

In order to run this script you will need to download MUSCLE [17] from here: `http://www.drive5.com/muscle/downloads.htm`. Use the Archive Utility to open the file, change the name of the executable file from something like "muscle3.8.31-_i86darwin64" to "muscle," and move it into your bin directory via the terminal with the following syntax (you will need to know your admin password to do this):

```
sudo cp /Downloads/muscle /usr/bin
```

6.8.3 Convert Files from .abi to . fastq

To run the merge_sanger_16s.pl you will first need to convert your read files from .abi to .fastq

This can be done at `http://sequenceconversion.bugaco.com/converter/biology/sequences/`

Use the drop down menus to set it to convert .abi files to .fastq. Upload a file and convert it. The converted file will save to your downloads folder under the name sample.fastq. If you are working with a lot of reads we recommend immediately renaming the files to match the original abi file name to avoid confusion.

6.8.4 Edit and Create a Consensus Sequence

Once all of your files are in fastq format, move all of them to the Sanger_seq folder in which you saved the merge_sanger_16s.pl script. Use the terminal to navigate to within this folder by typing:

```
cd Desktop/Sanger_seq
```

Then, to run the script, type:

```
perl merge_sanger_16s.pl file1.fastq file2.fastq
```

The script will return one of 2 messages:

1. "Found N conflicting case(s) during merging of X residues"
2. "Not enough data to overlap confidently."

In the first case the merging happened, however there may be some conflicting bases. The fewer the better. It can be an indication of how confident the user should be with the results. Since this is a very crude method it should be noted that there is no fancy algorithm behind the merge. There is a crude comparison for which we keep the base that had the highest quality score.

In the second outcome, the sequences were trimmed too much when doing the QC. The length of both sequences end to end was smaller than the fragment length that we are looking for. This is an indication of poor quality sequence

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

and most users should not proceed (others can lower the quality threshold set by the script).

The newly merged file will be saved as file1_merged.fasta and can be uploaded to BLAST for identification.

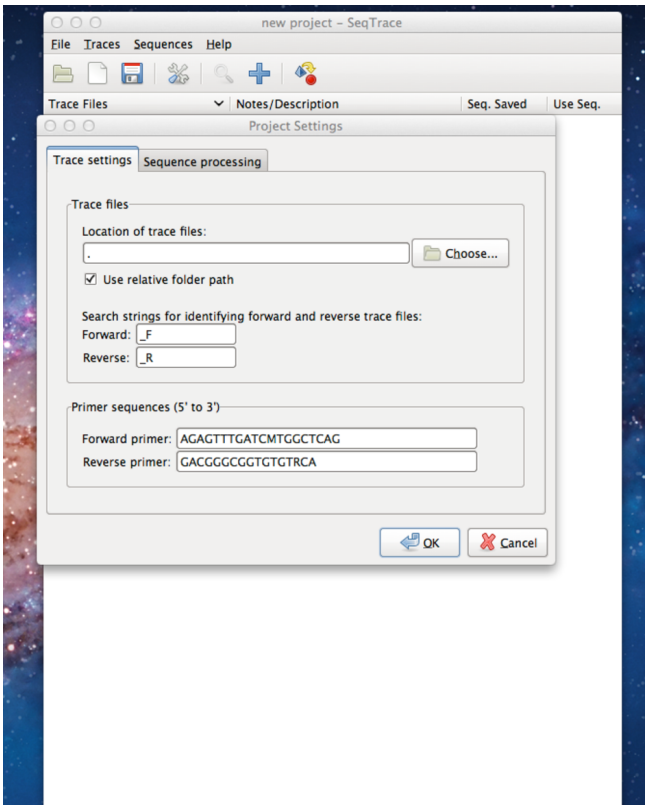

Figure 3: -The above figure shows the appropriate Trace Settings for SeqTrace for step 3 of the "Creating and Editing a Consensus Sequence" for Sanger Sequence Processing.

7 Organism Identification using 16s rRNA gene sequence

In a classroom or undergraduate research setting the project may not have a particular bacterial species in mind. In this case it is necessary to screen the 16S Sanger sequencing results for possible genome sequencing candidates. We recommend starting with BLAST results, then continuing onto the Genomes Online Database (GOLD), and simply Google searching. In many cases it will

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

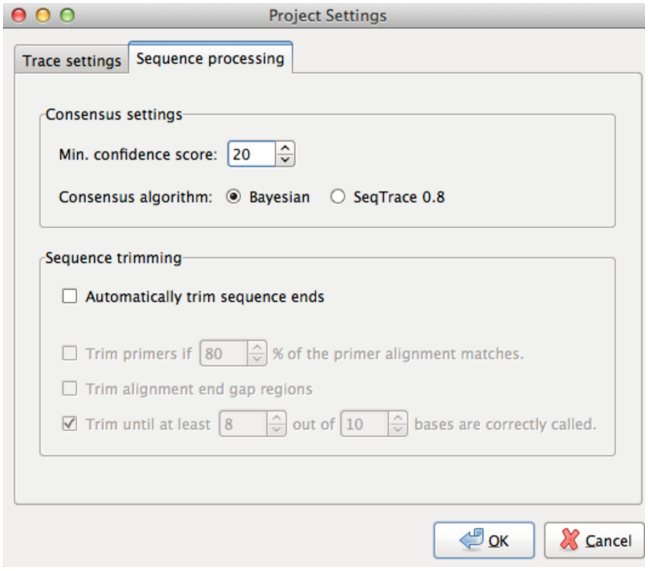

Figure 4: -The above screenshot shows the appropriate Sequence Processing Settings for SeqTrace for step 4 of the "Creating and Editing a Consensus Sequence" for Sanger Sequence Processing.

Figure 5: -The above screen shot shows a chromatogram of an .abi trace file in SeqTrace. This is what you should see in step 7 of "8.7 Edit and Create a Consensus Sequence with SeqTrace".

be handy to also build a phylogenetic tree to aid in identification since the BLAST results may not be sufficiently informative.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

7.1 BLAST 16S rDNA sequence

Begin by navigating to the Standard Nucleotide BLAST at NCBI:

`http://blast.ncbi.nlm.nih.gov/Blast.cgi?\discretionary{-}{}{}PROGRAM=blastn&PAGE_TYPE=BlastSearch&LINK_LOC=blasthome`

Paste in your Sanger consensus sequence. We recommend checking the box to exclude Uncultured/environmental sample sequences since these will not be informative for identification. Be sure the nucleotide collection (nr/nt) is selected under database and click the BLAST button.

7.2 Interpreting the results

Depending on the quality of the Sanger sequencing and the particular bacteria sequenced, the BLAST results can range from definitive to relatively uninformative. Examples of both are discussed below.

1. In some cases it is not necessary to build a phylogenetic tree for further identification. If all of the top hits are the same species (or end in sp.), have e-values of 0.0, good query coverage, and 99% to 100% identity you can proceed to "Using GOLD".
2. In other cases the results are much more ambiguous. The results may show more than 99% identity to multiple species within multiple genera. In this case, proceed to section 11 "Building a 16S rDNA Phylogenetic Tree", before using GOLD.

7.3 Using GOLD (the Genomes Online Database)

Go to: `http://genomesonline.org/cgi-bin/GOLD/index.cgi`

Click the search button on the left side of the page and you should be taken to a page that looks like the screen shot displayed in Figure 9.

Fill out the blue Organism Information (Organism Name) section, with information about your microbe from BLAST and click submit search. We usually search for only the genus to get a sense for how well that genus is represented in the database and which species are present. Figure 10 shows an example screen shot of the results for "*Brachybacterium*." Clicking on a project ID will take you to a more detailed description of the project including its project status (complete, permanent draft, incomplete, targeted).

If you have relatively ambiguous identification results (e.g. you think you have some sort of *Brachybacterium* but aren't sure which species) it could be worthwhile to perform an alignment of your 16S sequence with those from genomes already in Genbank.

7.4 Align 16S Sequences using Align Sequences Nucleotide BLAST

First locate the 16S sequences of the genome you'd like to compare to, by searching the NCBI Nucleotide database for "Species 16s gene".

`http://www.ncbi.nlm.nih.gov/nuccore/`

Click on the sequence of interest, then click on the "FASTA" link to get the sequence in FASTA format. Now navigate to the "Align Sequences Nucleotide BLAST" page:

`http://blast.ncbi.nlm.nih.gov/Blast.cgi?PAGE_TYPE=BlastSearch&BLAST_SPEC=blast2seq&LINK_LOC=align2seq`

Paste in the two 16S rDNA sequences and click on the "BLAST" button. Unless both your sequence and the sequence you are comparing to were amplified with the same primers, the query coverage will not be 100%. A low identity can be the result of poor sequence quality or taxonomic distance.

A choice of whether to sequence an organism based on these results depends on the project goal. For example an identity of 100% suggests that at least at the 16S level, the candidate organism is very similar to what is already in the database. However, many organisms vary greatly in gene content between strains and an additional genome may still be informative. There is also significant debate over what level of relatedness at the 16S level should be used to determine the difference between species, or if this is even a relevant question [7][14][21][34].

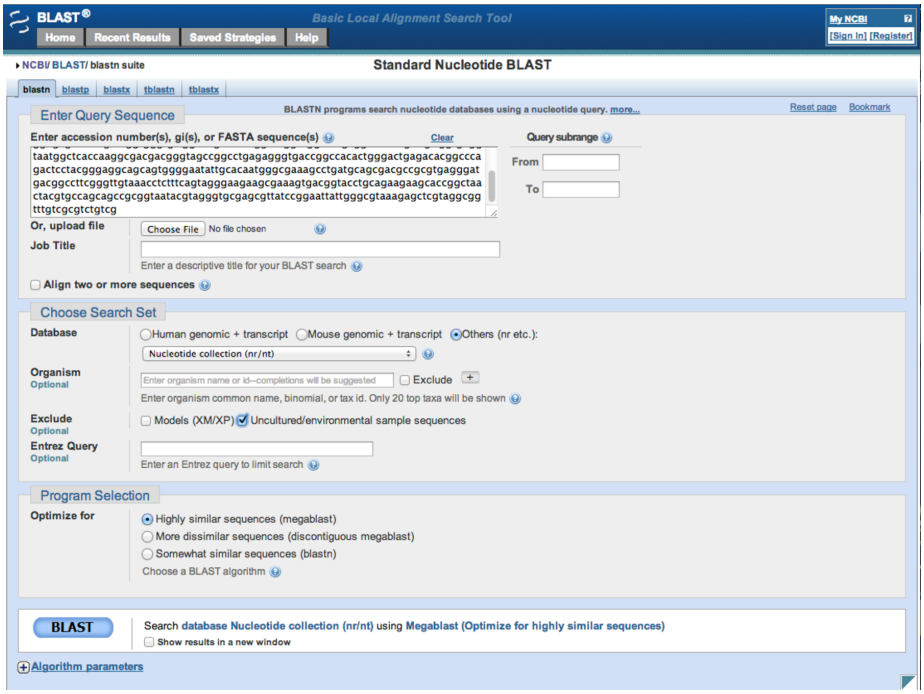

Figure 6: -This is what the BLAST submission page looks like. BLAST stands for Basic Local Alignment Search Tool and allows you to search the database for a known sequence that matches, or is similar to the sequence you provide.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

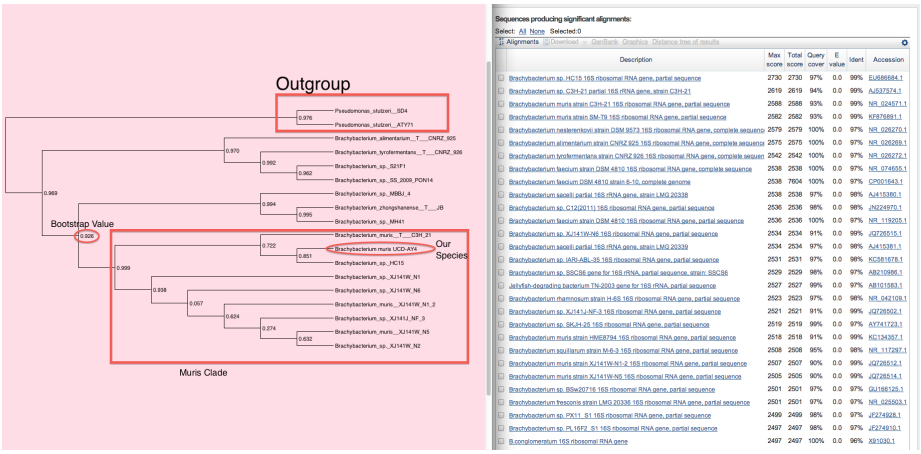

Figure 7: -The left side of the figure shows an example of an informative phylogenetic tree. Our species of interest (*Brachybacterium muris* UCD-AY4) falls within a clade where every named member has the same name "*Brachybacterium muris*", and this species name does not occur elsewhere on the tree. The bootstrap value of the Muris clade is high (0.926) indicating that this clade is well supported. *Pseudomonas stutzeri* was chosen as the outgroup because it is phylogenetically distinct to *Brachybacterium*. On the right side of the figure are the BLAST results for the 16s SSU rRNA of our species. As shown in the tree, the most similar species (highest identity scores and lowest E values) are *Brachybacterium muris* or unspecified *Brachybacterium* species.

8 Library Preparation and Sequencing

8.1 Library Preparation

The first choice in library preparation is whether to do the library prep yourself or to have the library made by your sequencing provider. The economics of this decision are usually dependent on the number of samples involved. For example an Illumina TruSeq library prep kit costs around $2600 for 48 samples. That's far cheaper than the $150 to $300 that a typical sequencing provider would charge per sample. However, if you're only preparing a couple of samples there's no reason to buy an entire kit. The requisite time and ancillary consumables and equipment must also be taken into account (see Figure 14). Most sequencing facilities offer library preparation services.

8.2 Kit Options

Whether you chose to make libraries yourself, or use a provider, the next major choice is of the type of kit. The two major different choices with Illumina kits are

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

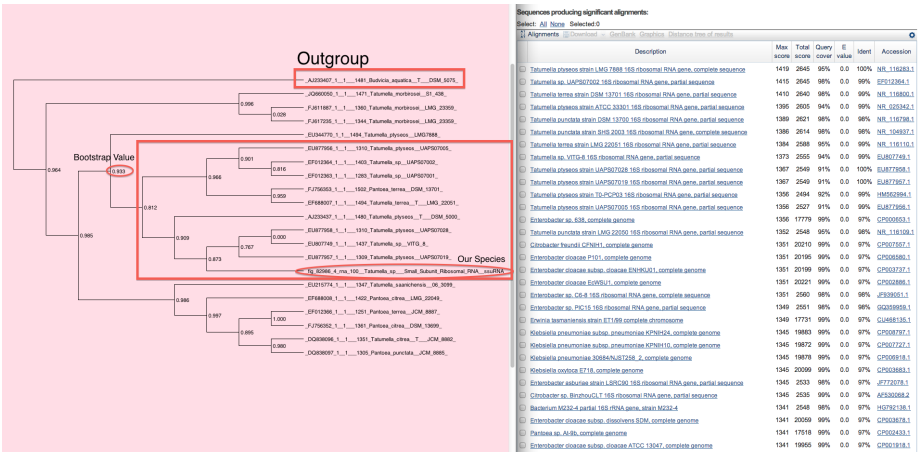

Figure 8: -The left half of the figure shows an uninformative tree. The species in question (*Tatumella* sp) is located within a poorly defined clade, which contains multiple species some of which are represented elsewhere in the tree. The bootstrap value of the selected clade is high (0.933) indicating it is well supported by the data, however multiple species are placed inside and outside of the clade indicating some phylogenetic uncertainty. *Budvicia aquatica* was chosen as the outgroup because it is phylogenetically distinct to *Tatumella*.The right half of the figure shows the BLAST results for the 16s SSU rRNA of our species. As in the tree, there is significant phylogenetic uncertainty for the target species.

the Nextera transposase-based kits or the TruSeq kits (with or without PCR). These kits are available from Illumina, but there are also comparable options from other vendors (e.g. New England Biolabs and Kapa Bioscience). The pros and cons of each type of kit are listed below:

- Nextera: *Pro* – It allows for very low amounts of input DNA, down to 1ng in the case of the Nextera XT kit. *Con* – the transposase has an insertion bias and the extensive PCR required for low input samples will also impact the final assembly[1].
- TruSeq (our recommendation): *Pro* – The PCR-free protocol minimizes library bias by using mechanical instead of enzymatic DNA fragmentation, and by eliminating PCR, resulting in better assemblies. *Con* – requires a large amount of DNA (at least 1 ug for PCR-free). There is also now a TruSeq LT kit which only requires 100ng of DNA but does entail some PCR so may provide a middle option between PCR-free TruSeq and Nextera.

When growing bacteria in culture, as described in this workflow, it should almost always be possible to get enough DNA to use PCR-free TruSeq and therefore minimize library preparation biases in the genome assembly.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

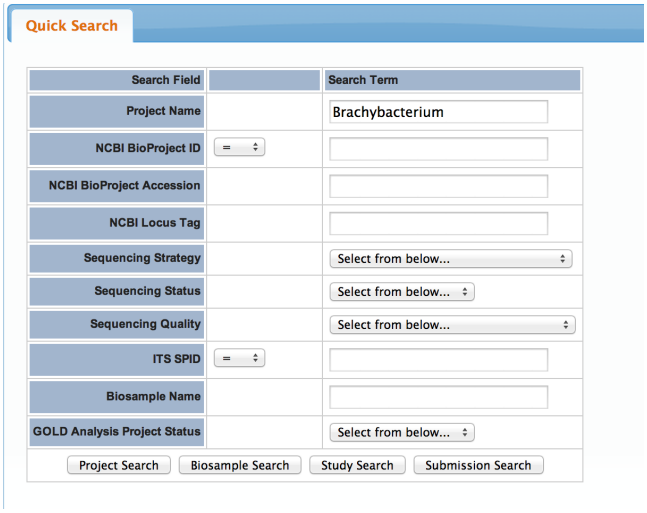

Figure 9: -This is GOLD's (Genome OnLine Database) search page. Enter the name of the organism you are interested in under project name.

GOLD Project ID +	Project Name Brachybacterium [remove]
Gp0001925	Brachybacterium faecium 6-10, DSM 4810
Gp0004437	Brachybacterium muris
Gp0011776	Brachybacterium paraconglomeratum LC44
Gp0012502	Brachybacterium squillarum M-6-3
Gp0028874	Brachybacterium alimentarium CNRZ 925
Gp0028876	Brachybacterium nesterenkovii CNRZ 926
Gp0033260	Brachybacterium muris UCD-AY4
Gp0035956	Brachybacterium tyrofermentans CNRZ 926
Gp0086679	Brachybacterium phenoliresistens W13A50
Gp0089909	Brachybacterium phenoliresistens W13A50
Gp0093608	Brachybacterium zhongshanense JCM 15471

RESET [1 - 11] of 11 Show 25 results.

Figure 10: -This is GOLD's results page, 11 *Brachybacterium* projects are listed, some of which may be complete and others in a more ambiguous state.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

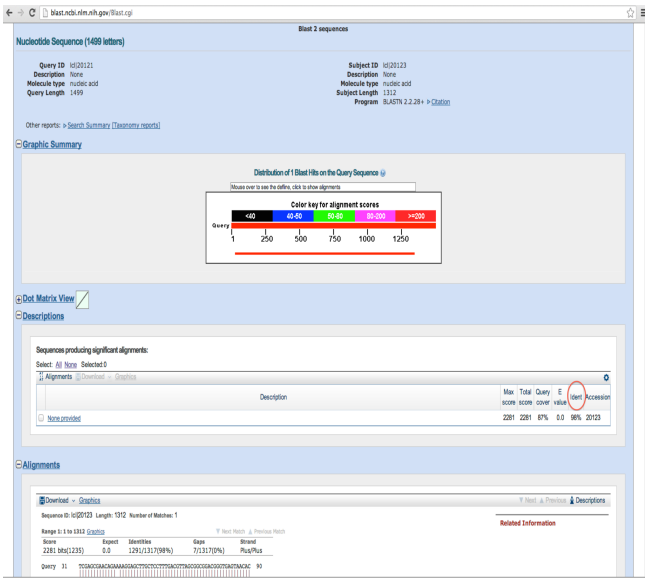

Figure 11: -This is the align2seqs results page showing the alignment between two sequences. The identity metric is circled above.

Figure 12: - This figure shows the Dendroscope editing options. The expansion tools are circled while the arrow points to the phylogram option.

8.3 Considerations in Library Preparation

Insert size: The tradeoff with insert size is between utility for assembly (larger is better) and ability of those fragments to amplify on the Illumina flowcell for sequencing (smaller is better). The optimal fragment size also depends on the length of reads used (with longer read-lengths, longer insert sizes are useful for scaffolding). The final consideration is the amount of DNA available for sequencing. While having all inserts be exactly 750 base pairs (bp) might be ideal, such a stringent size-selection could result in the recovery of only a very small amount of DNA. In our lab, with paired end 300bp (PE300) reads on the Illumina MiSeq, we shoot for a fragment size (including adapters) of 600-900bp. Different sequencing facilities have different opinions on this topic and it is worth having a discussion with your sequencing facility's point of contact before making any libraries. It is very important that all samples have similar library sizes if multiplexing as described below.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

8.4 Multiplexing

The capacity of an Illumina MiSeq with PE300 reads is around 15 Gigabases (Gb), which would result in a coverage of 4300X for a typical bacterium with a 3.5Mb genome. On the HiSeq with PE125bp reads, this would be over 14,000X coverage. Currently, the recommended coverage for a bacterial genome assembly is 20-200X depending on the choice of assembler. Therefore, sequencing a single bacterial genome on a full MiSeq or HiSeq run is a significant waste of money and reagents. Furthermore, some current genome assembly algorithms do not perform well given an excess of data, and require down-sampling (i.e., throwing away data) to achieve the recommended coverage for assembly. We typically multiplex 10-48 genomes on a PE300 MiSeq run and many more on a HiSeq run. If using a kit for library prep, multiplexing is quite straightforward since there are a number of barcoded adaptors that come with the kit. Demultiplexing can be performed by the sequencing facility.

8.5 Collaborate

Current Illumina sequencing systems have much greater capacity than is needed for sequencing a single genome. This means it can be generally beneficial to combine many samples into a single run of a machine. Unfortunately, our experience has been that sequencing facilities will typically not help in the coordination of such pooling of samples (we assume because they do not want to oversee the pooling or deal with the associated accounting hassles). Therefore this means it is up to the users to carry out such coordination. Though this can sometimes be complicated it is generally worth it since one can pool together many genomes or metagenomes into single runs of a system and still get enough data for each project, thus making the sequencing cost per project significantly lower. For this to work well, what one needs to do is to coordinate the use of barcodes to tag each sample, coordination of the pooling, and some informatics work at the end to "demultiplex" samples from each other.

8.6 Downsampling

Coverage (read depth) is the average number of reads representing a given nucleotide and is a function of the number and size of genomes pooled onto a run. The optimal amount of coverage depends on the read length, the assembler being used, and other factors. For Illumina data assembled using this workflow we recommend that this number be between 20x and 200x. See our more detailed discussion in section 9.1.3 "Interpretation of A5-miseq stats". If you have coverage significantly higher than 200x and wish to downsample your data we have written a script (sub_sample_reads) for this purpose. You will first need to calculate how many reads you want the script to sample. We recommend determining how many reads would be equivalent to 100x coverage (divide the genome size by the average read length and multiply by 100). You can download the script from the figshare zipped script file [25]. Create a new directory

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

containing the script (sub_sample_reads) and the reads you wish to downsample.

To downsample the data navigate to the directory you just created (in the terminal) and use the following command

```
/sub_sample_reads file1 file2 #_reads_to_keep output_file_name
```

for example

```
/Users/Madison/Desktop/sub_sample/sub_sample_reads.pl
test_1.fq test_2.fq 250 my_reads.fastq
```

For further directions and documentation you can view the script on github
`https://github.com/gjospin/scripts/blob/master/subsample_reads.pl`

9 Genome Assembly and Annotation

9.1 Assembly

Genome assembly consists of

1. data pre-processing (quality filtering and adaptor removal)
2. error correction
3. contig assembly
4. scaffolding
5. verification of scaffolds/contigs

There is a plethora of programs that can perform some, or most of these steps. These programs include commercial and open-source options, some are very user friendly and some are extremely difficult to use/install. Common assemblers for bacterial genomes include SPAdes [5], MIRA [8], SGA [33], Velvet [37] CLC (CLC Bio), and A5 [36]. Good sources for overviews of genome assemblers and the assembly process include the GAGE project [31], the GAGE-B project [28], and the Assemblathon Project [16].

For this workflow we recommend use of the open source A5 assembly pipeline which automates all of the steps described above with a single command [36]. A5 is designed to work with raw, demultiplexed Illumina data and a recent version, A5-miseq, has been optimized for longer reads from the MiSeq (Coil et al submitted). Input reads must be paired, and the files can be separate (forward reads in one file, reverse reads in another) or interleaved. These files should have the .fastq extension. See (`http://en.wikipedia.org/wiki/FASTQ_format`) for a description of the fastq format. You may need assistance from your sequencing center in locating and accessing these files. You will need one of the two following (per genome): 1) a single .fastq file that contains both forward and reverse reads, or 2) two .fastq files, one with forward reads and one with the corresponding reverse reads. These FastQ files can optionally be gzip compressed (as indicated by the .gz file name extension).

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

Download/Install A5 from `http://sourceforge.net/projects/ngopt/`

Follow the (expert) instructions located `http://sourceforge.net/projects/ngopt/files/?source=navbar`

or

Follow a video made by David Coil `https://www.youtube.com/watch?v=Ad6HJevC5U8`

or

Follow these instructions:

After downloading and unzipping the program, change the name of the folder to a5_pipeline and move it from your downloads folder to your Applications folder. Then, create a new folder which will contain the files generated by the pipeline on your Desktop. By the way, there's nothing special about having your file on the Desktop, it's just there to simplify our instructions. We will refer to this folder as "a5_output", but you should use a more informative name.

9.1.1 Running A5-miseq

Open a Terminal window and navigate to a5_output. A5-miseq will write all of the assembly output files to the same folder from which you run the program. In this example the newly created folder is on the Desktop and named a5_output so the syntax for navigating to the folder in a Terminal window is

```
cd Desktop/a5_output/
```

Now that you are in the folder where you want your genome assembly to appear, you are ready to run the program. First, type (don't hit return yet!):

```
/Applications/a5_pipeline/bin/a5_pipeline.pl
```

Then, drag and drop in the input file(s) into the same Terminal window (or type the path to them if you know it). Finally, type a name that will be used as part of all of your output files. So, your command line should look like this:

```
/Applications/a5_pipeline/bin/a5_pipeline.pl SequenceFile1.fastq
SequenceFile2.fastq MyGenome
```

The program may take a few hours to run. Once it is completed the terminal will display Final assembly in MyGenome.final.scaffolds.fasta. The complete assembly will be located in the a5_output folder.

Among the numerous files generated by A5, two of particular importance are the "MyGenome.contigs.fasta" and "MyGenome.final.scaffolds.fasta" which contain the contigs and scaffolds, respectively.

In addition, A5-miseq generates a file containing information about the quality of the assembly called "MyGenome.assembly_stats.csv"

To view this file use the "less" command:

```
less MyGenome.assembly_stats.csv
```

For more on interpreting these numbers proceed to "Assembly Validation".

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

9.1.2 Assembly Validation

There are three components to genome assembly validation. The first is the overall "quality" of the assembly, assessed by examining the stats provided by A5-miseq (discussed below). The second is verification that the organism sequenced is the organism of interest, simply by checking the assembled 16S sequence with BLAST. The third is "completeness" which is difficult to measure except in cases where a close reference is available. Nevertheless, we can get an idea of how complete the genome is by looking for highly conserved "housekeeping" genes that are found in almost every bacterial genome. To do this, we use a program called PhyloSift [11] to assess the presence or absence of 37 housekeeping genes in the assembly to infer completeness.

9.1.3 Interpretation of A5-miseq stats

To open A5-miseq stats, import it into excel as a tab delimited CSV file. The first two numbers, shown in columns 2 and 3, are the number of contigs and scaffolds. Defining a "good" or "bad" assembly starts here. A finished assembly would consist of a single contig with no unresolved nucleotides but that is extremely unlikely to result from short read data. At the other extreme, we would consider a bacterial assembly in 1000 contigs to be very fragmented. In our experience, acceptable bacterial assemblies using Illumina PE300 data, assembled with A5, tend to range from 10-200 contigs. It is also worth noting that unless studying genomic organization, the number of contigs is less important than the gene content recovered by the assembly which is typically >99% using A5-miseq (Coil et al, submitted).

"Genome Size" and "Longest Scaffold" are simply represented as base pairs. While genome size can vary within taxa, this can be a second useful sanity check for the assembly. When expecting a 5MB genome based on other sequenced isolates from the same genus, if the assembled genome size is 2MB or 10MB, a red flag should be raised. "N50" represents the contig size at which at least 50% of the assembly is contained in contigs of that size or larger. This metric, combined with the number of contigs is the most common measure of assembly quality... larger is better. An N50 of 5,000 bp would be quite poor... meaning that half of the entire assembly is in contigs smaller than 5,000 bp. On the other hand an N50 of 1,000,000 bp is considered very good for bacterial genomes sequenced with Illumina technology.

The number of raw reads/raw nucleotides "Raw reads"/"Raw nt" and error-corrected reads/nucleotides "EC Reads"/"Raw nt" counts are useful for seeing what percentage of the data has been discarded. A very large difference between these numbers ("% reads passing EC"/"% nt passing EC") would indicate either poor quality sequence data or significant adapter contamination. Adapter contamination rates can be high when the insert size is too small or if there were problems during library preparation. Poor quality sequence data can result from loading the libraries at a molar concentration that was too high for the instrument, from mechanical issues preventing focus of the sequencing instrument's

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

cameras, or from use of a compromised batch of sequencing reagents. Resolution of these issues would entail a discussion with your sequencing provider.

A5-miseq reports three depth of coverage statistics which can be used to assess whether sufficient data has been collected for genome assembly. First is the "Raw cov" which is simply the total number of base pairs of sequence data, divided by the assembly size. This gives an estimate of the average number of reads covering each base in the assembly. The actual number of reads at each site can and will vary substantially from the average. The second statistic is the "Median cov" which gives the median depth of coverage among all sites in the assembly. That is, 50% of sites will have greater coverage and 50% will have less than this value. "10th percentile cov" indicates a coverage level below which only 10% of sites in the assembly fall. For Illumina data, the ideal median coverage will lie between ˜20X and 100X. Much less than 20X median coverage and the quality of individual base calls may be compromised. Ideally, the 10th percentile coverage will be higher than 10, for similar reasons.

A separate metric of the base call quality is also reported by A5-miseq as "bases >= Q40". Following assembly, A5-miseq realigns the reads to the assembled sequence and estimates the accuracy of the nucleotide called at each site in the assembly. These accuracies are provided as PHRED quality scores [20], which represent log-scaled probabilities of accuracy. For example a PHRED score of 20 indicates a 99% chance of the correct base, while Q30 and Q40 indicate 99.9% and 99.99% probabilities of the correct base being called. A5-miseq reports the number of assembly bases called with at least Q40.

9.1.4 Verification of 16S Sequence

Follow the steps described in Section 11, "Making a Phylogenetic Tree" for obtaining and performing a BLAST search of the full length 16s sequence.

PhyloSift: Navigate to `http://phylosift.wordpress.com`

Download and unzip the latest version of Phylosift

In the terminal, navigate to the directory containing the unzipped Phylosift

Run

```
./phylosift search contig_file_name
```

For example:

```
./phylosift search /Users/microBEnet/Desktop/Data-Genomes/Pantoea_Tatumella/
tatumella/tatumella.contigs.fasta
```

Note: The first time you run PhyloSift it has to download a marker gene database so it may take a few minutes.

From the PhyloSift directory Move to the "PS_temp" directory

Within this directory, Phylosift has created a directory with the same name as the input file. Move to this new directory, and then move to "blastDir".

Open the marker_summary.txt file in the blastDir

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

```
less marker_summary.txt
```

The DNGNGWU0001-00040 markers represent 37 highly conserved bacterial genes, if one is missing it won't show up as a zero, it is necessary to manually verify the list. Most of the genes should only appear once. An occasional 2 is fine, but if all/a majority of the genes appear twice or even three times you have most likely sequenced multiple bacteria together. Additionally check to make sure there is no 18S RNA (at the top of the list) to ensure your sample has not been contaminated with a eukaryote (e.g. yeast).

Important Note: Markers 4, 8 and 38 are no longer included in the Phylosift analysis so do not be concerned if they are not listed.

9.2 Annotation

9.2.1 Options

Genome annotation is the process of predicting genes (open reading frames) within a genome sequence and attempting to assign function to those genes based on homology to known sequences. Note that we are not describing a genome "analysis" here. While genome annotation marks the final step in our data wrangling workflow, it is just the beginning of a thorough genome analysis. We recommend performing this step as the bare-minimum analysis required to include a very basic description of the genomic content for a genome announcement publication.

There are a number of different pipelines available for annotation of bacterial genomes. These include Prokka [32], IMG [29], RAST [30], GLIMMER [12], PGAP [3] and others.

Each of these pipelines has advantages and disadvantages, and each will give slightly different results. Here we recommend RAST since it is web-based, easy to use, returns results within hours, and provides a convenient toolbox for analyzing the results. However, RAST annotations are very difficult to submit to NCBI so we recommend allowing NCBI to re-annotate the genome with PGAP upon submission. Also, we recommend reporting the annotation results from the PGAP annotations in the genome announcement (for consistency.) Why do we also run a RAST annotation? Because we are impatient and we like to see results right away. We do not like having to wait for the NCBI submission process to be completed before we start exploring our data.

9.2.2 RAST Annotation

Navigate to `http://rast.nmpdr.org/` and register a new account. Once you have created an account, log in. Hover over the "Your Jobs" tab at the top of the page and click on "Upload New Job." In order to proceed you must specify a domain, a genus, a species, and the genetic code (usually "11".) Click "Finish the Upload."

The annotation will take some time, ranging from 2 hours to a few days, depending on server load. RAST will email you when it is complete. Once the

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

annotation is complete, use their SEED Viewer to explore the annotation and metabolic pathways of the organism. From the RAST results, you can obtain information such as the presence or absence of a particular gene/pathway and you can compare the annotation to other genomes in their database.

10 Obtain the Full-Length 16S Sequence from the Assembly

(Skip this step if you are building the tree using the 16S sequence from Sanger sequencing)

1. Go to RAST and sign in
2. On the "Jobs Overview" page, click on view details (under annotation progress) for the microbe you are working with.
3. Click on Browse annotated genome in SEED viewer (At the top of the page)
4. Click on Browse through the features of [organism name]
5. Under Function search for "ssurna" or "SSU rRNA" (if it doesn't work at first then refresh the page)
6. Find the ssuRNA that is 1400-1800 bp in length (often Illumina assemblies also have fragments of 16S sequence that are only a few hundred bp long)
7. Click on the Feature ID for that sequence
8. Click on the Sequences tab (around the middle of the page)
9. Click on Show Fasta
10. Click on Download Sequences and save as a fasta file. Rename the file to something useful.
11. Double check the identity of the sequence at BLAST: `http://blast.ncbi.nlm.nih.gov/Blast.cgi?PROGRAM=blastn&PAGE_TYPE=BlastSearch&LINK_LOC=blasthome`

11 Building a 16S rDNA Phylogenetic Tree

What *is* this thing? At this point, you have an organism in pure culture, but you do not know what it is. If you found a creature crawling on the ground and wanted to identify (or classify) it, you might look at it's morphology and ask what it most *looks* like. If it has six legs, you might hypothesize it is some kind of insect. If it has hard outer wings folded over its back, you might hypothesize that it is some kind of beetle. If it also had antler-style horns on its head, you might hypothesize that it is some kind of stag beetle. If you do not have enough information available to hypothesize what kind of stag beetle you have, then you have reached the limit of *taxonomic resolution* for your creature.

With an unknown microbial species, the best way to identify it is to sequence one of its genes (most people use the 16S rRNA gene) and ask what *it* most looks like. With animal classification, commonly used key features are

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

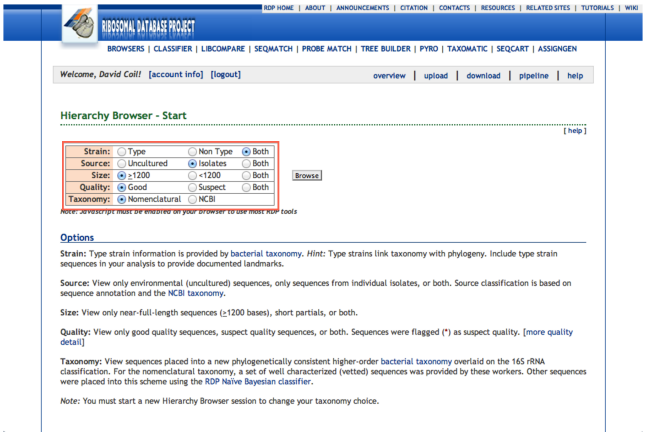

Figure 13: -This figure shows the RDP Hierarchy Browser with the recommended parameters selected (red box).

things like legs and wings and horns; with microbial classification, the key features to examine are the nucleotides in different positions in a DNA sequence. Fortunately, we have computer programs to help us make sense of the DNA sequence information. Our preferred approach to classifying microbial species is to place an unknown sequence in the context of a phylogenetic tree of known sequences. Building a phylogenetic tree from a 16S rRNA sequence is fairly straightforward, but the interpretation of the tree can be a bit complex. Here, we attempt to guide you through both. However, some complicated cases will require consultation with an expert in the field of phylogenetics or systematics.

The outline of the workflow is to use the Ribosomal Database Project (RDP) to generate an alignment of the sequence with close relatives and an outgroup, following by cleanup of the RDP headers, tree-building with FastTree and viewing/interpretation of the tree using Dendroscope.

11.1 Obtain an RDP alignment

The goal of this section is to obtain a 16S alignment from RDP that can be used to build a tree. This procedure has the added benefit of providing an independent verification of the taxonomic assignment of your sequence based on the BLAST results.

1. Go to `http://rdp.cme.msu.edu`
2. Create an account
3. Click on my RDP/login
4. Upload the fasta file containing your 16S sequence
5. Assign it a group name (this is what the program will label your sequence/organism). Choose this carefully since that will be the name on

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

the final tree.

6. Click the "+" next to the sequence, to add it to your cart
7. Click on CLASSIFIER at the top of the page
8. Click on "Do Classification With Selected Sequences" button. This will show you a hierarchical view of the classification of your sequence (from Phylum to Genus.) You will use this information to navigate to other sequences that you want to include in your alignment that you will use to build your phylogenetic tree. For example, Figure 13 shows the hierarchy for the *Tatumella* 16S sequence.
9. Click on BROWSERS. We recommend opening BROWSERS in a new tab so that you can keep the hierarchy information handy.
10. Click on "Isolates" to select only isolates for further analysis. Then click "Browse"
11. Click on the + sign next to "Archaea outgroup." This will add an Archaeal sequence to your cart, which will be used to root your phylogenetic tree.
12. If using the example sequence provided, click on "Proteobacteria", then Gammaproteobacteria, then "Enterobacteriales", then Enterobacteriaceae. This will take you to the Genus Tatumella, which currently has over 69 entries in it. If the genus you are working with has too many sequences to analyze easily (for example, *Bacillus* currently has >26000,) one way to reduce this number is to exclude the uncultured taxa in the database. To do this, scroll down to the Data Set Options and click on the "Isolates" button. Click "Refresh" and you will see that there are fewer sequences in the Genus. To reduce this number further, click on the "Type" Strain button (though if you do this you'll have to build a tree later for species identification since each species will only be represented once in the tree). As a worst-case scenario, you will need to manually select a subset of organisms to include in your alignment.
13. Click on the + sign next to **genus** *Tatumella* to add all of those sequences to your cart.
14. Click on "Sequence Cart" and confirm that your uploaded sequence, the outgroup sequence, and all of the other sequences you'd like to include in your tree are displayed.
15. Click on "download," leave the download options as the defaults (fasta, aligned, uncorrected,) and then click on the appropriate download button. Save the file and then rename it to something informative.

11.2 Clean up the RDP taxon names

The RDP alignment will have taxon names that most of the downstream software tools will not tolerate because they consist of special text characters. So, we have written a little Perl script (cleanup.pl) that will remove those special characters and replace them with underscores. This script is included in the zip file of scripts on figshare [26]. To run cleanup.pl, first move it to your Applications folder. Then, in a Terminal window, navigate to the directory that

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

contains the RDP alignment that you've just downloaded. Then, type:

```
perl /Applications/cleanup.pl -i RDP_alignment.fa -o RDP_alignment_clean.fa
```

11.3 Building the Tree with FastTree

There are two ways to get FastTree, which will be required for building the tree from your alignment. The first is to use Phylosift (installed in 9.1.4) which contains a working version of FastTree. In this case, you will simply call the program from the Phylosift directory with the following command (be sure the path to Phylosift calls the correct version):

```
/phylosift/osx/FastTree -nt RDP_alignment_clean.fa > tree_file.tre
```

The other option is to install FastTree directly, which is a bit more involved.

Go to `http://www.microbesonline.org/fasttree/#Install` and download the FastTree.c program by right clicking on it and saving the link to your Applications folder. To compile the software, navigate to your Applications folder in a Terminal window:

```
cd /Applications
```

Then, type:

```
gcc -O3 -finline-functions -funroll-loops -Wall -o FastTree FastTree.c -lm
```

This compiling of FastTree requires a software tool called gcc (the Gnu Compiler Collection, if you want to know - see `http://gcc.gnu.org` for more detail). If your attempt to compile FastTree with the instructions above fails, the most likely reason is that you do not have gcc. You can download and install gcc from Xcode here https://developer.apple.com/downloads/index.action?q=xcode

In order to download Xcode you will need to register as a developer with Apple which takes only a couple of minutes. After you register, click on the apple next to "Developer" at the top of the page. Then, click on the Xcode download link, which will ultimately take you to the Mac App Store, where you can follow the instructions to install Xcode. Once it is installed, open the program and open preferences (under the Xcode tab). Click on the downloads option and install the command line tools.

Once you have successfully downloaded and installed Xcode and the command line tools, return to your Applications folder in a Terminal window and type again:

```
gcc -O3 -finline-functions -funroll-loops -Wall -o FastTree FastTree.c -lm
```

Now, you should have a working version of FastTree. To build your tree, using the cleaned up RDP alignment, type the following (be sure the output name ends in ".tre" to ensure it will be recognized by Dendroscope):

```
/Applications/FastTree -nt RDP_alignment_clean.fa > tree_file.tre
```


PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

11.4 Viewing the Tree in Dendroscope

Download and install Dendroscope. `http://ab.inf.uni-tuebingen.de/software/dendroscope/`

You will need to obtain a license here `http://www-ab2.informatik.uni-tuebingen.de/software/dendroscope/register/`

Enter the license number into Dendrosope and then you can open your phylogenetic tree from the File menu to view it.

Once the tree is visible, the first step is to re-root the tree to the outgroup. Expand the tree by clicking the expansion button (labeled in Figure 12), then scroll through the tree to locate the outgroup. Click on the beginning of the taxa name, to select it, and reroot the tree by going to edit and selecting re-root.

We recommend viewing the tree as a phylogram which can be accomplished by clicking on the phylogram button (labeled in Figure 12). From this tree it should be possible to determine the phylogenetic placement of the candidate sequence, and in some cases to give it a name with more certainty than a simple BLAST search. Below are examples of a relatively informative tree and a relatively uninformative tree:

In tree shown in Figure 7 (genus *Brachybacterium*), our sample of interest from an assembly is "Brachybacterium muris UCD-AY4" [27]. It falls within a clade where every named member has the same name "Brachybacterium muris", and this name does not occur elsewhere on the tree. Hence, we were confident enough to name our sample as that species. In other words, this sequence falls within a well-supported monophyletic clade of *Brachybacterium muris*.

In the tree shown in Figure 8 (genus *Tatumella*) our species of interest is *Tatumella* sp. [6]. In contrast to the Brachybacterium example, here our species falls within a poorly defined clade containing multiple species. In this case we did not assign a species name to this isolate.

12 Data Submission

This section describes how to submit contigs and scaffolds (if applicable) as a Whole Genome Shotgun (WGS) submission to Genbank. We also recommend allowing NCBI to annotate the genome themselves, since submitting RAST annotations to Genbank can be prohibitively complicated. The genomes are automatically shared with the DNA Data Bank of Japan (DDBJ) and the European Molecular Biology Laboratory (EBML). In addition, genomes from Genbank are automatically pulled into the Integrated Microbial Genomes (IMG) database hosted at the Joint Genome Institute (JGI), and are annotated there as well. This section also describes how to submit the raw reads, in this case we use the European Nucleotide Archive (ENA) for ease of use but the reads will be automatically incorporated into the Short Read Archive (SRA) at NCBI as well.

To submit a genome, you must first create a "BioProject" at NCBI. When that is complete, a separate process is required to submit the genome sequence. Before submitting your genome, you will need to have available 4-5 files which are listed below.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

File types used in data submission:

- AGP file (.agp). This is a file required by NCBI to describe scaffolding (if applicable)
- FASTA file (.fasta). This is the standard file type for sequence data, produced in this case by A5-miseq
- FSA file (.fsa). Same as a FASTA file but with a different extension
- SQN file (.sqn). The file type for sequence data required by NCBI
- SBT file (.sbt). This is a template file type used by NCBI

12.1 FASTA2AGP

First, create the .agp file In the terminal, navigate to the directory containing your scaffolds file

Run the fasta2agp.pl script included with A5 on the scaffold file output by the A5 assembly "my_scaffolds.fasta". Syntax is:

```
perl fasta2agp.pl my_scaffolds.fasta > my_scaffolds.agp
```

eg:

```
perl /Users/Madison/Desktop/a5_miseq_macOS_20140113/bin/fasta2agp.pl
/Users/Madison/Desktop/a5_miseq_macOS_20140113/example/
phiX.a5.final.scaffolds.fasta > phiX.a5.scaffolds.agp
```

If this runs successfully then you should see a both the FSA and AGP files in your current directory.

Important Note: NCBI considers a gap of less than 10 nucleotides to be "missing information" in a contig, not a gap between contigs (whereas A5 has no minimum gap size). Therefore NCBI requires that contigs separated by less than 10 nucleotides be merged. This script performs that merging, meaning that the number of contigs in the FSA file may be less than in your input file. Therefore we recommend counting the contigs in the FSA file:

To count them in the terminal use the syntax

```
grep -c ">" name_of_your_.fsa_file
```

Important Note: If after running the fasta2agp.pl script and counting the contigs you have the same number of contigs as starting scaffolds, then you should only submit the SQN file to Genbank and specify that scaffolding did not take place (otherwise NCBI will reject the AGP file).

Now, navigate to http://www.ncbi.nlm.nih.gov. Create an account and/or login. Then, create a BioProject at NCBI by navigating to `https://submit.ncbi.nlm.nih.gov/subs/bioproject/` and clicking on "New submission." Fill in the personal information for the submitter.

Below, in italics are the responses that we typically give for a genome sequencing project.

Project type

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

- Project data type-*genome sequencing*
- Sample scope-*monoisolate*
- Material-*genome*
- Capture-*whole*
- Methodology-*sequencing*
- Objective-*assembly*

Target

- Organism Name
- If you have other information feel free to add it

General info

- We recommend choosing "*Release immediately following curation*"
- Project Title
- Public Description
- Relevance-*Environmental*
- Biosample-*blank*
- Publications-*blank*

Once the project is submitted, refresh the page and copy down the Bioproject ID (it starts with "PRJNA")

12.2 Create a SBT template

Create a SBT template file at NCBI http://www.ncbi.nlm.nih.gov/WebSub/template.cgi The BioProject # is the Bioproject ID starting with "PRJNA" which you received in the previous step, BioSample can be left blank

When you click create the template, it will automatically download to your computer as template.sbt. We recommend immediately renaming the file to the appropriate project.

12.3 Tbl2asn

Download the tbl2asn program from `ftp://ftp.ncbi.nih.gov/toolbox/ncbi_tools/converters/by_program/tbl2asn/`

If you are using Safari, a window will pop up asking for login information, just choose guest and unzip the version of the program that is compatible with your operating system. Other browsers will take you to a page with a lot of tbl2asn programs, download the one compatible with your operating system.

After downloading the desired command-line program, uncompress the archive and rename the resulting file to tbl2asn

Now change the file permissions of the file (in the terminal) since transfer by FTP resets the permissions.

Syntax is:

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

```
chmod 755 tbl2asn
```

Once you have changed the permissions, create a new directory and place tbl2asn along with the SBT file and FSA files into the folder.

Run the tbl2asn program using the following syntax. You will need to fill out the organism name, strain, location, collection date, isolation source specific to your own project.

```
path_to_program/tbl2asn -p path_to_files -t template_file_name -M n -Z
discrep -j "[organism=X] [strain=X] [country=X: city, state abbreviation]
[collection_date=X] [isolation-source=X] [gcode=11]"
```

Following the -p is the path to the directory containing the FSA file, following the -t is the path to and name of the SBT template file

Sample syntax

```
Desktop/ncbi/tbl2asn -p ~/Desktop/ncbi -t ~/Desktop/ncbi/template-1.sbt
-M n -Z discrep {j "[organism=Ruthia magnifica str. UCD-CM][strain=UCD-CM]
[country=USA: Davis, CA][collection_date=2002]
[isolation-source=Calyptogena magnifica tissue][gcode=11]"
```

The program will output the necessary files into the directory you created earlier

(ensure no errors were generated by opening the errorsummary.val file and making sure it is blank, or listing the directory contents ($ ls –lh) to ensure it has zero bytes)

12.4 Create a Whole Genome Shotgun (WGS) Submission

Navigate to https://submit.ncbi.nlm.nih.gov/subs/wgs/ Click on the New Submission button at the top Submitter -fill in your own information

General Info

- BioProject- *Yes*, add the BioProject identification sequence (from the BioProject submission, starts with PRJNA)
- Biosample-*No*
- Release date-Optional but we recommend "*Release immediately following curation*"

Don't check the box stating, "Genome assembly structured comment is in the contig .sq file"

- Assembly Method-Choose *other*, fill in the blank with A5 Assembly Pipeline (version can be found in the asssembly_stats.csv file)
- Version or date program was run – *a5-miseq-macOS-20140521*
- Assembly name – give your assembly an appropriate name

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

- Genome coverage- this is provided in the output from A5
- Sequencing technology – *Illumina* (Miseq or HiSeq)
- Is this the full representation of the genome? *Yes*
- Is this the final version? *Yes*
- Do you intend to annotate this version? *No*
- Is it a part of a multiisolate project? *No*
- Is it a de novo assembly? *Yes*
- Is it an update of existing submission? For most projects the answer to this will be *no*
- BioSample Type: *Microbe*

BioSample attributes

- Sample Name
- Organism
- Strain
- Collection date
- Geographic location
- Isolation source
- Files
- Select *We have files for traditional split contigs OR gapped sequences*
- Select *AS.1 (.sqn)*__ and upload your .sqn file +"Do you have AGP files that assemble the split contigs into scaffolds and/or chromosomes, OR assemble the gapped sequences into chromosomes?" If you have scaffolds that are not identical to your contigs select yes, if not select no and continue onto the next section

If you do have scaffolding:

- "Do you have an AGP file for unplaced scaffolds built from the split contigs (these are scaffolds without chromosome or plasmid information)?" *Yes* -upload the AGP file
- "Are there also AGP files that assemble chromosomes, plasmids and/or unlocalized scaffolds?" *No*
- "Did you annotate the scaffolds or chromosomes that are assembled in the AGP files (not gapped submissions)?" *No*
- "Bacteria is available from" *If the bacteria is available in a culture collection, feel free to indicate where. We recommend submission of sequenced strains to a culture collection if possible.*
- Source DNA is available from-*See above*

-Check the box below to annotate this prokaryotic genome in the NCBI prokaryotic annotation pipeline before being released. This will allow NCBI to use their PGAAP pipeline to annotate the genome, and this annotation will be automatically attached to the project.

Files

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

- Click on "We have files for contigs"
- Did you assemble the contigs or other components into scaffolds and/or chromosomes? *Yes*
- Do you have unplaced scaffolds (scaffolds without chromosome or plasmid information)? *Yes*-upload AGP file
- Did you assemble chromosomes, plasmids and/or unlocalized scaffolds? *No*
- Do you have sequence files for scaffolds and/or chromosomes and/or plasmids? *No*

Click "Submit" and you're done! You will receive a series of e-mails from NCBI confirming your submission and notifying you of any problems. Once the submission is pre-processed you'll get an Accession Number. Note however that the data will not be released until final processing. The Accession Number is not acceptable for publication until after the final release of the data.

Submitting Raw Reads to ENA/SRA

We recommend using Safari or Firefox for this step, in our hands Chrome can have issues with the Java requirements for uploading files.

Go to:

https://www.ebi.ac.uk/ena/about/sra_submissions

And create an account

Successful creation of an account should take you to the "Welcome to ENA's Sequence Read Archive (SRA) Webin submission system." screen

Click on New Submission tab

Select Submit sequence reads and experiments

Click on Data Upload Instructions towards bottom of page

This takes you to a variety of options for uploading files depending on your preference and operating system. We use the Webin Data Uploader. Click on the link which will download a .jlnp file. Open and run this file. Depending on your system you may have to download and install a new version of java. On some systems you may have to right-click the .jlnp file and Open with "Java Web Start".

Login using your e-mail address and password

In the WebinDataUploader, in the blank area to the right of the Local Upload directory, navigate to the directory on your computer containing the reads (using the path as you would in the terminal)

Select the file(s) containing the reads and click Upload.

(Note that paired-end data is required to be in two separate fastq files. If your data came as one interleaved file, then the separated fastq files can be found in the directory where the A5 assembly was performed as [project name].raw1_p1.fastq.gz and [project name].raw1_p2.fastq.gz)

Note that the only acceptable file types for submission are gzip (.gz) and bzip (.bz2). To gzip files in the Terminal use the following syntax:

```
gzip [filename]
```


PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

After completion, return to EMBL (the new submission tab of the SRA Webin submission system) and select the Next button. During this process, refreshing the page or navigating away from the page will reset the form and the information will be lost.

Click Create a New Study. Fill in descriptions of the project and proceed to next tab. Select the appropriate metadata format, or in most cases the ENL default sample checklist at the bottom. Note that the default release date is three months from the current date, change this if the data should be released sooner.

You should now be at the Sample page. Required fields are listed on the right and optional additional fields can be selected from the options on the right. Fill out the appropriate fields and click on Next.

Note: If you are submitting data for an organism that doesn't have a Taxon ID ("Tax ID") then you need to e-mail ENA to receive one (datasubs@ebi.ac.uk). If you have already submitted the genome to NCBI then you can retrieve the Taxon ID from your BioProject page there. On the ENA page, you will be able to search for the Taxon ID and find your organism under the Organism Details tab but you won't be able to find it using the name of the organism.

On the Sample page Click the + Add button under sample group details Fill in the unique name under basic details, add the Tax ID if it wasn't added previously and click next

On the Run page Select the appropriate data type

Fill in the required fields (they change with data type)

Note: "Insert size" cannot be a range, only a number.

Click Submit and confirm submission. You will immediately receive a confirmation e-mail but it takes some time before the information is actually live at the ENL links.

13 Discussion

In an effort to democratize the process of microbial sequencing and de novo assembly, we have designed a workflow that would allow a small lab, one operating without a specialized technician or bioinformatician, to take a sample and carry it through from swab to publication. There are many options for sequencing, assembling, and annotating microbial genomes. This workflow is only one path through the numerous choices that could be made in a genome sequencing project.

All of the scripts and programs for this workflow are open-source and available online for free to ensure that individual researchers and small groups are able to access and utilize the tools necessary to complete the workflow. In general, many available bioinformatic tools are free and open source, but the installation, operation and maintenance of these programs is often complex, requiring specific technical expertise or extensive detailed instructions and best practices in the field remain undefined.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

Sequencing, sharing, and publishing a genome sequence can certainly be considered as an important process in its own right. Once a genome is shared other people can use that genome for various and diverse purposes. However, just because one can stop after publishing and releasing a genome sequence that does not mean one should ignore what one can do with the data. A genome sequence is also a starting point for many computational and laboratory analyses that can provide insight into evolution, ecology, physiology, biochemistry, metabolism, and more. Such analyses are beyond the scope of this workflow and paper but that should not be taken as implying they are not interesting, useful or important.

Projected Cost

Item	Best Case (Per Sample)	Worst Case (Per Sample)
DNA Extraction[1]	$1.66	$166
PCR[2]	$0.60	$150
PCR Cleanup[3]	$2.00	$100
Sanger[4]	$14.00	$14
Library Prep[5]	$58.33	$2,800
Illuminia Sequencing[6]	$35.42	$1,700
Total	$112.01	$4,930

Figure 14: -This figure shows the estimated materials (i.e. without labor) cost of performing a genome sequencing project with this workflow. The "Best Case" shows the marginal cost of sequencing one genome in a case where you are multiplexing 48 samples, and have the appropriate kits and reagents on hand. The "Worst Case" shows the cost of doing a single genome, with no multiplexing, in a lab where every reagent needed to be purchased new and was not used for anything else. Specific assumptions are as follows; 1) This assumes the purchase of a standard DNA extraction kit, good for 100 samples. 2) This assumes purchase of a standard 200U PCR reagent kit. 3) PCR cleanup can be performed in a number of ways; gel extraction, beads, or columns for example. Here we assume purchase of a standard column-based kit. 4) Sanger sequencing cost is given as the price per reaction ($7 at our sequencing facility), times the forward and reverse reactions. 5) This assumes the purchase of a 48-sample Nextera or TrueSeq kit from Illumina, however kits from other manufacturers can be cheaper. 6) Our sequencing cost estimate assumes purchase of an Illumina MiSeq run from a sequencing facility.

References

[1] Daniel Aird, Michael G Ross, Wei-Sheng Chen, Maxwell Danielsson, Timothy Fennell, Carsten Russ, David B Jaffe, Chad Nusbaum, and Andreas Gnirke. Analyzing and minimizing PCR amplification bias in Illumina sequencing libraries. *Genome Biology*, 12(2):R18, 2011.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

[2] S Altschul. Basic Local Alignment Search Tool. *Journal of Molecular Biology*, 215(3):403–410, Oct 1990.

[3] Samuel V. Angiuoli, Aaron Gussman, William Klimke, Guy Cochrane, Dawn Field, George M. Garrity, Chinnappa D. Kodira, Nikos Kyrpides, Ramana Madupu, Victor Markowitz, and et al. Toward an Online Repository of Standard Operating Procedures (SOPs) for (Meta)genomic Annotation. *OMICS: A Journal of Integrative Biology*, 12(2):137–141, Jun 2008.

[4] Sandra L. Baldauf. Phylogeny for the faint of heart: a tutorial. *Trends in Genetics*, 19(6):345–351, Jun 2003.

[5] Anton Bankevich, Sergey Nurk, Dmitry Antipov, Alexey A. Gurevich, Mikhail Dvorkin, Alexander S. Kulikov, Valery M. Lesin, Sergey I. Nikolenko, Son Pham, Andrey D. Prjibelski, and et al. SPAdes: A New Genome Assembly Algorithm and Its Applications to Single-Cell Sequencing. *Journal of Computational Biology*, 19(5):455–477, May 2012.

[6] Z. A. Bendiks, J. M. Lang, A. E. Darling, J. A. Eisen, and D. A. Coil. Draft Genome Sequence of Microbacterium sp. Strain UCD-TDU (Phylum Actinobacteria). *Genome Announcements*, 1(2):e00120–13–e00120–13, Mar 2013.

[7] Jacqueline Z-M Chan, Mihail R Halachev, Nicholas J Loman, Chrystala Constantinidou, and Mark J Pallen. Defining bacterial species in the genomic era: insights from the genus Acinetobacter. *BMC Microbiology*, 12(1):302, 2012.

[8] B. Chevreux. Using the miraEST Assembler for Reliable and Automated mRNA Transcript Assembly and SNP Detection in Sequenced ESTs. *Genome Research*, 14(6):1147–1159, May 2004.

[9] D. A. Coil, J. I. Doctor, J. M. Lang, A. E. Darling, and J. A. Eisen. Draft Genome Sequence of Kocuria sp. Strain UCD-OTCP (Phylum Actinobacteria). *Genome Announcements*, 1(3):e00172–13–e00172–13, May 2013.

[10] David Coil;. From Swab to Publication Sample Data (Tatumella), 2014.

[11] Aaron E. Darling, Guillaume Jospin, Eric Lowe, Frederick A. Matsen, Holly M. Bik, and Jonathan A. Eisen. PhyloSift: phylogenetic analysis of genomes and metagenomes. *PeerJ*, 2:e243, Jan 2014.

[12] A. L. Delcher, K. A. Bratke, E. C. Powers, and S. L. Salzberg. Identifying bacterial genes and endosymbiont DNA with Glimmer. *Bioinformatics*, 23(6):673–679, Mar 2007.

[13] A. L. Diep, J. M. Lang, A. E. Darling, J. A. Eisen, and D. A. Coil. Draft Genome Sequence of Dietzia sp. Strain UCD-THP (Phylum Actinobacteria). *Genome Announcements*, 1(3):e00197–13–e00197–13, May 2013.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

[14] M. Drancourt and D. Raoult. Sequence-Based Identification of New Bacteria: a Proposition for Creation of an Orphan Bacterium Repository. *Journal of Clinical Microbiology*, 43(9):4311–4315, Sep 2005.

[15] M. I. Dunitz, P. M. James, G. Jospin, J. A. Eisen, D. A. Coil, and J. A. Chandler. Draft Genome Sequence of Tatumella sp. Strain UCD-D suzukii (Phylum Proteobacteria) Isolated from Drosophila suzukii Larvae. *Genome Announcements*, 2(2):e00349–14–e00349–14, Apr 2014.

[16] D. Earl, K. Bradnam, J. St. John, A. Darling, D. Lin, J. Fass, H. O. K. Yu, V. Buffalo, D. R. Zerbino, M. Diekhans, and et al. Assemblathon 1: A competitive assessment of de novo short read assembly methods. *Genome Research*, 21(12):2224–2241, Dec 2011.

[17] R. C. Edgar. MUSCLE: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Research*, 32(5):1792–1797, Mar 2004.

[18] David J Edwards and Kathryn E Holt. Beginner's guide to comparative bacterial genome analysis using next-generation sequence data. *Microb Inform Exp*, 3(1):2, 2013.

[19] J. C. Flanagan, J. M. Lang, A. E. Darling, J. A. Eisen, and D. A. Coil. Draft Genome Sequence of Curtobacterium flaccumfaciens Strain UCD-AKU (Phylum Actinobacteria). *Genome Announcements*, 1(3):e00244–13–e00244–13, May 2013.

[20] P. Green. Phrap. *version 1*, page 090518., 2009.

[21] W. P. Hanage, C. Fraser, and B. G. Spratt. Sequences, sequence clusters and bacterial species. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 361(1475):1917–1927, Nov 2006.

[22] H. E. Holland-Moritz, D. R. Bevans, J. M. Lang, A. E. Darling, J. A. Eisen, and D. A. Coil. Draft Genome Sequence of Leucobacter sp. Strain UCD-THU (Phylum Actinobacteria). *Genome Announcements*, 1(3):e00325–13–e00325–13, Jun 2013.

[23] A. O. Kislyuk, L. S. Katz, S. Agrawal, M. S. Hagen, A. B. Conley, P. Jayaraman, V. Nelakuditi, J. C. Humphrey, S. A. Sammons, D. Govil, and et al. A computational genomics pipeline for prokaryotic sequencing projects. *Bioinformatics*, 26(15):1819–1826, Aug 2010.

[24] David Coil; Guillaume Jospin; Jenna Lang;. Miscellaneous Scripts for Workflow, 2014.

[25] David Coil; Guillaume Jospin; Jenna Lang;. Miscellaneous Scripts for Workflow, 2014.

[26] David Coil; Guillaume Jospin; Jenna Lang;. Miscellaneous Scripts for Workflow, 2014.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug

[27] J. R. Lo, J. M. Lang, A. E. Darling, J. A. Eisen, and D. A. Coil. Draft Genome Sequence of an Actinobacterium, Brachybacterium muris Strain UCD-AY4. *Genome Announcements*, 1(2):e00086–13–e00086–13, Mar 2013.

[28] T. Magoc, S. Pabinger, S. Canzar, X. Liu, Q. Su, D. Puiu, L. J. Tallon, and S. L. Salzberg. GAGE-B: an evaluation of genome assemblers for bacterial organisms. *Bioinformatics*, 29(14):1718–1725, Jul 2013.

[29] V. M. Markowitz, I.-M. A. Chen, K. Palaniappan, K. Chu, E. Szeto, M. Pillay, A. Ratner, J. Huang, T. Woyke, M. Huntemann, and et al. IMG 4 version of the integrated microbial genomes comparative analysis system. *Nucleic Acids Research*, 42(D1):D560–D567, Jan 2014.

[30] R. Overbeek, R. Olson, G. D. Pusch, G. J. Olsen, J. J. Davis, T. Disz, R. A. Edwards, S. Gerdes, B. Parrello, M. Shukla, and et al. The SEED and the Rapid Annotation of microbial genomes using Subsystems Technology (RAST). *Nucleic Acids Research*, 42(D1):D206–D214, Jan 2014.

[31] S. L. Salzberg, A. M. Phillippy, A. Zimin, D. Puiu, T. Magoc, S. Koren, T. J. Treangen, M. C. Schatz, A. L. Delcher, M. Roberts, and et al. GAGE: A critical evaluation of genome assemblies and assembly algorithms. *Genome Research*, 22(3):557–567, Mar 2012.

[32] T. Seemann. Prokka: rapid prokaryotic genome annotation. *Bioinformatics*, Mar 2014.

[33] J. T. Simpson and R. Durbin. Efficient construction of an assembly string graph using the FM-index. *Bioinformatics*, 26(12):i367–i373, Jun 2010.

[34] E. Stackebrandt. Report of the ad hoc committee for the re-evaluation of the species definition in bacteriology. *INTERNATIONAL JOURNAL OF SYSTEMATIC AND EVOLUTIONARY MICROBIOLOGY*, 52(3):1043–1047, May 2002.

[35] B. J. Stucky. SeqTrace: A Graphical Tool for Rapidly Processing DNA Sequencing Chromatograms. *Journal of Biomolecular Techniques*, 23:90–93, 2012.

[36] Andrew Tritt, Jonathan A. Eisen, Marc T. Facciotti, and Aaron E. Darling. An Integrated Pipeline for de Novo Assembly of Microbial Genomes. *PLoS ONE*, 7(9):e42304, Sep 2012.

[37] D. R. Zerbino and E. Birney. Velvet: Algorithms for de novo short read assembly using de Bruijn graphs. *Genome Research*, 18(5):821–829, Feb 2008.

PeerJ PrePrints | http://dx.doi.org/10.7287/peerj.preprints.453v1 | CC-BY 4.0 Open Access | received: 8 Aug 2014, published: 8 Aug