# Command Line Unix
# BLAST, RM, Herne

Christopher Shaffer
Genomics Education Workshop
Washington University, Summer '06

## Command line

- Why use BLAST command line?
  - Avoid slow responses at NCBI
  - Use a static database, so answers does not change
  - Search unpublished data (search all fosmids)
- Finishing
- Level 3 Annotation

# Building a BLAST command

| | |
|---|---|
| >blastall -p blastx | Search Type |
| -i 99M21.seq | Query |
| -d ../../../Databases/protein/melanogaster | Database |
| -o dmel_blastx.html | Output |
| -T -e 1e-2 -F F | Settings |

# Goals

- Unix directories & command line
- BLAST (via command line)
- RepeatMasker (via command line)
- herne (blast visualization)

## General Unix Tips

- To use the command line start X11 and type commands into the “xterm” window
- A few things about unix commands:
  - UNIX is case sensitive
  - Little or no feedback if successful, “no news is good news”

## Unix Help on the Web

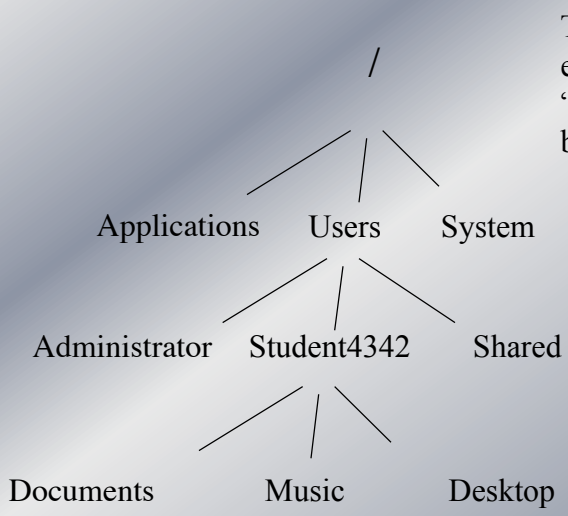
Here is a list of a few online Unix tutorials:

- Unix for Beginners
  http://www.ee.surrey.ac.uk/Teaching/Unix/
- Unix Guru Universe
  http://www.ugu.com/sui/ugu/show?help.beginners
- Getting Started With The Unix Operating System
  http://www.leeds.ac.uk/iss/documentation/beg/beg8/beg8.html

# Unix directories

- Unix systems can have many thousands of files
- Directories are a means of organizing your files on a Unix computer.
  - They are equivalent to folders in Windows and Macintosh computers

# Directory Structure




The directory that holds everything is called "root"and is symbolized by "/"

## Unix directories (cont)

- When logged into a unix computer there is always one directory called the "current working directory"
- When you type a command with a filename unix will look for that file in the "current working directory"
- You can change the "current working directory" at any time with the cd command.
- You can see the "current working directory" with the pwd command
- If the file is not in the "current working directory" you must tell unix where the file is. You do this by giving the full "path" to the file.

## Unix directories (cont)

- The "path" is a list of every directory from the "current working directory" to the file in which the name of each directory is separated by a "/"
- For example, when doing a BLAST search by command line, a path to the database to search might look like this:

  Databases/protein/invert/melanogaster
- Thus you are telling the computer to look inside the current working directory for the directory Database; inside is a directory called "protein"; inside that is a directory called "invert"; and inside that is the "melanogaster" database

## Your Home Directory

- When you login to any Unix server, the "current working directory" always start in your **Home** directory.
  - On Mac my home is /Users/chris
- To change which directory is the current working directory type cd and then the path to the new directory

  cd data/virilis/fosmids/14J19

## Paths special characters

- You can also start a path at two other locations besides the current working directory
- / can be used at the very beginning of the path to tell the computer to start at the very top of the tree (hard drive)

  /Users/Chris/data/virilis/fosmids/14J19
- ~ can be used at the beginning to tell the computer to start in your home dir

  ~/data/virilis/fosmids/14J19

# Other Path Shortcuts

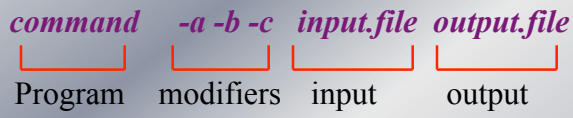
- There are two other symbols that are useful in paths
  - . (dot) means "the current working directory"
  - .. means "the parent directory" - the directory one level above the current directory, you can string many ../ together to keep going up and up and up

E.g. ../../../Databases/protein/invert/melanogaster

This would go up three directories to find the "Databases" directory then go down into protein and invert to eventually find the melanogaster database

# Unix Commands

Most commands have this basic structure:




•Program: the name of the program you want to run (blastall, consed, etc.)

•Modifiers (optional, usually in any order): a CASE SENSITIVE letter which controls exactly how the program will run

•Some will be plain "-l" some will have values "-l guest"

•Input (optional): the file(s) you wish the program to act upon.

# Basic commands

- ***pwd*** (**present working directory**) shows the name of the current working directory:

```
> pwd
/Users/Chris
```

- ***ls*** (**list**) gives you a list of the files in the current directory:

```
> ls
Desktop          Library          Public
Documents        Movies           Sites
```

- ***cd*** (**change directory**) set a new current working dir

```
>cd Documents
> pwd
/Users/Chris/Documents
```

# Basic commands

- There are many other commands that can be used to copy, move, view, and delete files
- You can read about these and other commands in the unix help pages mentioned above or use the mac or windows operating systems to do these things

## Unix tries to help you:

1. Command History
2. Command line completion

## Command history

- The computer remembers you last 100 or so commands you have entered.
- Use the “up arrow” to scroll back through these commands.
- You can use right and left arrow to move around inside the command to edit it
- Very nice for long commands (e.g. blast searches) or to look back at a command that did not work looking for typos

## Files name completion

- Since the computer will only look in the current working dir for files when you want to type in a file name all you have to type is enough characters at the beginning to uniquely define the file
- When you hit <TAB> the computer will finish typing the name for you

```
> ls
af151074.gb_pr5  test.seq apt.txt
> rm af
<hit tab>
> rm af151074.gb_pr5
```

- Have long names but keep the differences at the beginning to facilitate using this feature

## Bioinformatics commands

- There are three commands that we will talk about that you might want to run on a command line:
  - BLAST
  - RepeatMasker
  - Herne (our BLAST results viewer)

# Command line BLAST

- There are two versions of BLAST available for the command line:
  - NCBI BLAST (free from NCBI)
  - WU BLAST (blast.wustl.edu)
- Why run BLAST command line?
  - Avoid slow responses at NCBI
  - Use a static database, so answers does not change
  - Search unpublished data (search all fosmids)

# NCBI BLAST

- For Bio4342 we use NCBI BLAST
  - Mostly historical
  - Command line program is called blastall
  - Web page with help for NCBI BLAST:

    http://goose.wustl.edu/~chris/blastall.html

## NCBI BLAST

- NCBI blastall has 4 required entries which can be given in any order:
  - Which program (blastn, blastx  etc) -p *name*
  - Query -i *seq_file.seq*
  - Database -d *database_name*
  - Name of file for results -o *output_filename*

## NCBI BLAST

- NCBI blastall has many optional settings the most common include:
  - Format the output as a web page: -T
  - Change the E cutoff: -e *number*
  - Explicitly turn filter on: -F T
  - Explicitly turn filter off: -F F

# Building a BLAST command

- Usually you will set the current working directory to the one with the query sequence using the cd command. For example

  > cd data/virilis/fomids/99M21/

  Then enter the the blastall command with all relevant settings:

```
>blastall   -p blastx   -i 99M21.seq
        -d ../../../Databases/protein/melanogaster
          -o dmel_blastx.html   -T     -e 1e-2     -F F
```

# RepeatMasker

- This program is used to screen any DNA sequence for the presence of know repetitive sequences.
- It will take a sequence file as input and create a new file in which any sequences identified repetitive will be changed to “N”. Can improve subsequent analysis (e.g. gene finders).
- It will also create other files which summarize the repetitive content (.tbl) and complete list of every hit (.out)

## RepeatMasker

- Web pages that run RepeatMasker limit the size of the input sequence
- Computationally intensive, 1-2 hours for class to get results for 12 fosmids on moderately powerful unix workstation
- Once installed you can get a mini help page by typing “RepeatMasker” on the command line with nothing else

## RepeatMasker command line

- The command is RepeatMasker
- There are two required fields

  -species *species_name*

  *sequence_file_to_mask*

- RepeatMasker command might look like this:

  >RepeatMasker -species drosophila 99M21.seq

## Herne the BLAST output viewer

- We have available a viewer that can be run on any computer that supports Tcl/Tk. (Most linux, mac OS 10.4, can be installed)
- The command has two required settings and two optional settings
- This is a nice way to get an overall view of where in a large sequence various highly similar sequences align

## Herne the BLAST output viewer

- The command in herne you must supply:
  - the output of a blastall search (-b)
  - the sequence file used in the search (-s)

> herne -b dmel_blastx.html -s 99M21.seq

## Herne the BLAST output viewer

- There are also two optional settings
  - You can have herne show you the location of all the repeats identified by RepeatMasker by adding

    -m *repeatmasker.out*

    e.g. -m 99M21.seq.out

  - If you are red/green colorblind add the setting -r

## Interpretation of Herne window

- The green bar represents the query sequence
- If you added the repeatmasker.out file, the repeats are shown in grey
- Each single hit is shown in its proper position, black, green, yellow, red show increasing quality
- All hits from the same subject are boxed
- Clicking on any hit will open a new window showing the alignment