Bioinformatics

Computational Methods I:
Genomic Resources and Unix

George Bell
WIBR Biocomputing Group

Human genome databases

- Human Genome Sequencing Consortium
- Major annotators:
  - NCBI
  - Ensembl (EMBL-EBI and Sanger Institute)
  - UCSC "Golden Path"
- Which annotation(s) and assembly best address your needs?
- Levels of use:
  1. Query remote database using web interface
  2. Write scripts to query remote database
  3. Install database locally and create queries however you want

---

NCBI

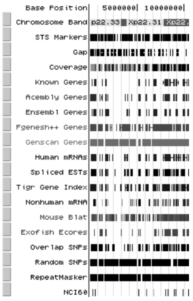
- http://www.ncbi.nlm.nih.gov/entrez/query.fcgi?db=Genome: > 800 genomes
- http://www.ncbi.nlm.nih.gov/genome/guide/human : most recent build (27) from October 28, 2001
- Use Map Viewer and Evidence Viewer to browse data
- Lots of other genome-centered resources!

Ensembl

- Human: http://www.ensembl.org/
- Version 3.26.1 (1/23/02) based on NCBI 26 assembly
- Use different "views" to browse data
  - ex: MapView, DiseaseView, GeneView
- BioPerl code created for accessing data
- Mouse: http://mouse.ensembl.org/

UC Santa Cruz

- The "Golden Path"
- http://genome.ucsc.edu/
- Differs from NCBI assembly
- Current version: August 6, 2001
- UCSC Genome Browser
- Home of BLAT search



Introduction to Unix

- Why Unix?
- The Unix operating system
- Files and directories
- Ten required commands
- Input/output and command pipelines
- Shell scripts

Did you know...
You can avoid neck injury and eyestrain at work by just quitting your job and finding one that doesn't involve computers in any way.

Did you know...
You can type the word "dog" simply by pressing the keys marked "d," "o," and "g," in that order. If it is the first word of a sentence, you will need to press "Shift" + "D."

Did you know...
When creating funeral annou
is generally viewed as bad tas
such gimmick fonts as "Ghou

all *Mad* images © DC Publications

Objectives for this week

- get around on a Unix computer
- run bioinformatics programs
  "from the command line"
- design potential ways to streamline data manipulation and analysis with scripts

---

Why Unix (for me)?

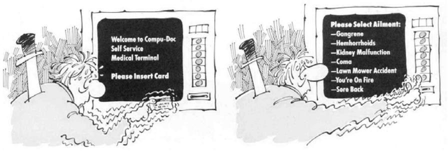
- **GEISHA**, the *Gallus gallus* (chicken) EST and in situ hybridization (ISH) database
- Analysis of lots of promoters for binding sites
- Searching for polyY/R in human promoters
- Prediction of novel non-coding human genes

Why Unix (in general)?

- Features: multiuser, multitasking, network-ready, robust
- Others use it – and you can benefit from them (open source projects, etc.)
- Good programming and I/O tools
- Types: Solaris, Linux, etc.

Why Unix for Bioinformatics?

- Good for manipulating lots of data
- Many key tools written for Unix
- Don't need to re-invent the wheel
- Unix-only packages: GCG, EMBOSS, etc.
- Unix tools with other OSs: Mac (OS X) & PC (Cygwin)



Unix O.S.

- kernel
  - managing work, memory, data, permissions
- shell:
  - working environment and command interpreter
  - link between kernel and user
  - choices: tcsh, etc.
  - History, filename completion [tab], wildcard (*)
  - Shell scripts to combine commands
- filesystem
  - ordinary files, directories, special files, pipes

It was fun for a while, but soon, it wasn't enough to satisfy me.

Logging in

- ssh (secure shell; for encrypted data flow)

```
ssh -l user_name
fladda.wi.mit.edu
```

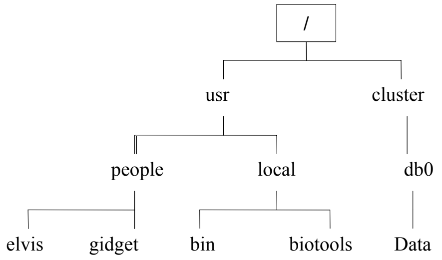
- passwd: to change your passwd
- logging out

```
logout
```

Every on-line account features several security checks, so you know that your stored information is absolutely inaccessible –

– including to you, after you forget your invisible password.

Intro to files and directories



ex: Full path to lots of applications is /usr/local/bin

Intro to files and directories

- Arranged in a branching tree
- Root of tree at "/" directory
- User elvis lives at /usr/people/elvis (fladda)
- Full vs. relative pathnames
  - At his home, Elvis' home dir is "."
  - To get to /usr/people/gidget, go up and back down: (../gidget relative to /usr/people/elvis)
- Anywhere, your home directory is "~".

Intro to Unix commands

- Basic form is

```
command_name options argument(s)
```

examples:

```
mv new_data old_data
blastall -p blastn -i myFile.seq -e 0.05
  -d nt -T T -o myFile.out
```

- Use history (↑, ↓, !*num*) to re-use commands
- Cursor commands: ^A(beginning) and ^E (end)
- To get a blank screen: `clear`
- For info about a command: `man` *`command`*

Key commands p. 1

- Where am I?

```
elvis@fladda[1]% pwd
/usr/people/elvis
```

- What's here?

```
elvis@fladda[2]% ls
A01.tfa
```

```
elvis@fladda[4]% ls -a
.     .cshrc      A01.tfa
..    .twmrc
```

```
elvis@fladda[3]% ls -l
-rw-r--r--  1 elvis musicians  1102 Dec 19 10:45 A01.tfa
```

7 1/2 MONTHS AGO
A NOSTALGIC LOOK BACK AT THE OLD DAYS

Hard to believe,
but CyberNerds way back in August were forced to make do with a 950 MEGABYTE Hard Drive!!!

Key commands p. 2

- Change directories:

```
cd ../gidget
/usr/people/gidget
```

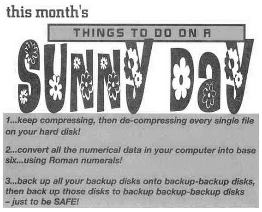
- Make a new directory:

```
mkdir spleen
```

- Remove a directory:

```
rmdir spleen
```

(needs to be empty first)



this month's THINGS TO DO ON A SUNNY DAY

1...keep compressing, then de-compressing every single file on your hard disk!

2...convert all the numerical data in your computer into base six...using Roman numerals!

3...back up all your backup disks onto backup-backup disks, then back up those disks to backup backup-backup disks – just to be SAFE!

File permissions

- Who should be reading, writing, and executing files?
- Three types of people: user (u), group (g), others (o)
- 9 choices (rwx or each type of person; default = 644)

0 = no permission	4 = read only
1 = execute only	5 = r + x
2 = write only	6 = r + w
3 = x + w	7 = r + w + x

- Setting permissions with chmod:

`chmod 744 myFile` or `chmod u+x myFile`

```
-rwxr--r--  1 elvis musicians  110 Dec 19 10:45 myFile
```

```
chmod 600 myFile
-rw------  1 elvis musicians  110 Dec 19 10:45 myFile
```

Key commands p.3

- Copying a file:

```
cp [OPTION]... SOURCE DEST
Ex: cp mySeq seqs/mySeq
```

- Moving or renaming a file:

```
mv [OPTION]... SOURCE DEST
Ex: mv mySeq seqs/mySeq
```

- Looking at a file (one screenful) with ‘more’

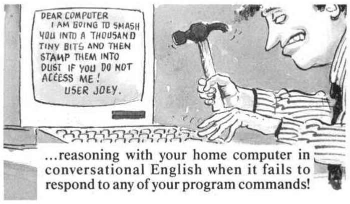
```
Ex: more mySeq
```

(Spacebar a screenful forward,
<enter> a line forward; ^B a screenful back; q to exit)

Key commands (summary)

ssh
pwd
ls
cd
mkdir
mvdir
chmod
cp
mv
more

It never works but you gotta try. . .



To get more info (syntax, options, etc.):

```
man command
```

Input/output redirection

- Defaults: stdin = keyboard; stdout = screen
- To modify,

```
command < inputFile > outputFile
```

- input examples

```
sort < my_gene_list
```

- output examples

```
ls > file_name  (make new file)
ls >> file_name  (append to file)
ls foo >& file_name (stderr only)
```

Pipes (command pipelines)

- In a pipeline of commands, the output of one command is used as input for the next

- Link commands with the “pipe” symbol: |

```
ex1: ls *.fasta | wc -l
ex2: head -1 *.fasta | grep '^>' | sort
```

Managing jobs and processes

- Run a process in the foreground (fg):
  command
- Run a process in the background (bg):
  command `&`
- Change a process (fg to bg):
  1. suspend the process: `^Z`
  2. change to background: `bg`

Managing jobs and processes (cont.)

- See what’s running (ps)

```
elvis@fladda[1]% ps
   PID TTY       TIME CMD
22541 pts/22   0:00 perl
22060 pts/22   0:00 tcsh
```

- Stop a process:

```
kill PID
ex: kill 22541
```

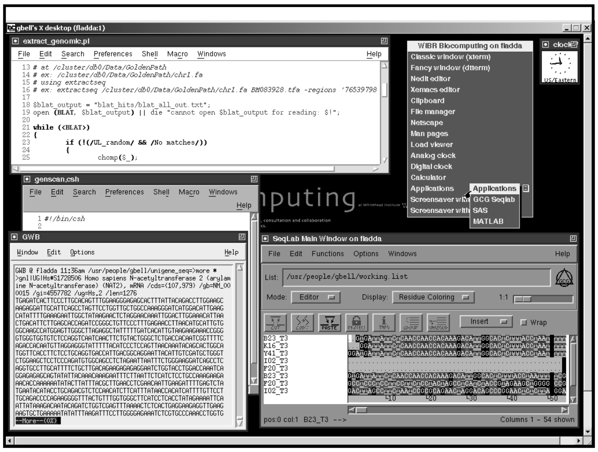
Icons For The Little Things Your Computer ACTUALLY Does

Text editors

- emacs, vi (powerful but unfriendly at first); pico
- xemacs, nedit (easier; X windows only)
- desktop text editors (BBEdit; TextPad) + ftp

X Windows

- method for running Unix graphical applications
- still allows for command-line operation
- See help pages on fladda for getting started
- Some powerful graphical applications on fladda:
  - GCG
  - Matlab
  - SAS
- Requires a fast network/internet connection
- Info at http://fladda.wi.mit.edu/bioc/help/x.html



```
# at /cluster/db0/Data/GoldenPath
# ex: /cluster/db0/Data/GoldenPath/chr1.fa
# using extractseq
# ex: extractseq /cluster/db0/Data/GoldenPath/chr1.fa BM083928.tfa -regions '76539798

$blat_output = "blat_hits/blat_all_out.txt";
open (BLAT, $blat_output) || die "cannot open $blat_output for reading: $!";

while (<BLAT>)
{
        if (!(/UL_random/ && /No matches/))
        {
                chomp($_);
```

```
#!/bin/csh

```

```
GWB @ fladda 11:36am /usr/people/gbell/unigene_seq=>more *
>gnl|UG|Hs#S1728506 Homo sapiens N-acetyltransferase 2 (arylam
ine N-acetyltransferase) (NAT2), mRNA /cds=(107,979) /gb=NM_00
0015 /gi=4557782 /ug=Hs.2 /len=1276
TGAGATCACTTCCCTTGCACAGTTTGGAAGGGAGAGCACTTTATTACAGACCTTGGAAGC
AAGAGGATTGCATTCAGCCTAGTTCCTGGTTGCTGGCCAAAGGGATCATGGACATTGAAG
CATATTTTGAAAGAATTGGCTATAAGAACTCTAGGAACAAATTGGACTTGGAAACATTAA
CTGACATTCTTGAGCACCAGATCCGGGCTGTTCCCTTTGAGAACCTTAACATGCATTGTG
GGCAAGCCATGGAGTTGGGCTTAGAGGCTATTTTTGATCACATTGTAAGAAGAAACCGGG
GTGGGTGGTGTCTCCAGGTCAATCAACTTCTGTACTGGGCTCTGACCACAATCGGTTTTC
AGACCACAATGTTAGGAGGGTATTTTTACATCCCTCCAGTTAACAAATACAGCACTGGCA
TGGTTCACCTTCTCCTGCAGGTGACCATTGACGGCAGGAATTACATTGTCGATGCTGGGT
CTGGAAGCTCCTCCCAGATGTGGCAGCCTCTAGAATTAATTTCTGGGAAGGATCAGCCTC
AGGTGCCTTGCATTTTCTGCTTGACAGAAGAGAGAGGAATCTGGTACCTGGACCAAATCA
GGAGAGAGCAGTATATTACAAACAAAGAATTTCTTAATTCTCATCTCCTGCCAAAGAAGA
AACACCAAAAAATATACTTATTTACGCTTGAACCTCGAACAATTGAAGATTTTGAGTCTA
TGAATACATACCTGCAGACGTCTCCAACATCTTCATTTATAACCACATCATTTTGTTCCT
TGCAGACCCCAGAAGGGGTTTACTGTTTGGTGGGCTTCATCCTCACCTATAGAAAATTCA
ATTATAAAGACAATACAGATCTGGTCGAGTTTAAAACTCTCACTGAGGAAGAGGTTGAAG
AAGTGCTGAAAAAATATATTTAAGATTTCCTTGGGGAGAAATCTCGTGCCCAAACCTGGTG
--More--(0%)
```

List: /usr/people/gbell/working.list

```
B23_T3   ..GTGATTATTCTCAACCAACCACAAAGACATTGGCACTCTTTACCTAATTTTT
K16_T3   ...GGATTATTCTCAACCAACCACAAAGACATTGGCACTCTTTACCTAATTTTT
Y41_T3   GGATGATTATTCTCAACCAACCACAAAGACATTGGCACTCTTTACCTAATTTTT
I02_T3   ......................................................
F20_T3   ......................................................
B23_T3   GTGATTATTCTCAACCAACCACAAAGACATTGGCACTCTTTACCTAATTTTTCG
F20_T3   GCCTCCCTCCTTCCTTCTCCTACTAGCCTCATCTACCGTAGAAGCTGGGG CCG
I02_T3   GACTTAGCCTTCTAATCCGCGCAGAACTAGGACAGCCCCGGAACTCTCTTA GGA
                   10        20        30        40        50
pos:0 col:1  B23_T3  -->                           Columns 1 - 54 shown
```

Unix packages for sequence analysis

- EMBOSS programs at http://www.uk.embnet.org/Software/EMBOSS/Apps
- ex: Smith-Waterman local alignment (`water`)
- Programs have two formats: interactive and one-line
- Conducive to embedding in scripts for batch analysis

- GCG programs at http://fladda.wi.mit.edu/gcg/gcgmanual.html
- Both command line and X Windows versions

EMBOSS examples

- dreg: regular expression search of a nucleotide sequence

```
dreg –sequence mySeq.tfa –pattern
  GGAT[TC]TAA –outfile mySeq_dreg.txt
```

- fuzznuc: nucleic acid pattern search

```
fuzznuc –sequence mySeq.tfa –pattern
  GGATTTAA –mismatch 1 –outf
  mySeq_fuzznuc.txt
```

Some other helpful commands

- cat: concatenate files

  ex: `cat *.seq > all_seq.tfa`
- alias: create your own command shortcuts

  ex: `alias myblastx blastall –p blastx –d nr`
- find: find a lost file

  ex: `find . –name \*.pl`
- tar: combine files together for storage or transfer
- ftp: transfer files between machines
- gzip & gunzip: compress or uncompress a file

  (see examples in demo)

Shell script example: runGenscan.csh

```
#!/bin/csh
# runGenscan.csh: to run genscan on a folder of sequences

# Go to the directory
cd /usr/people/elvis/seqs

# Do for each seq file (ending in .tfa)
foreach seqFile (*.tfa)

    # Put output in genscan folder with filename replacing .tfa with .txt
    set outFile = genscan/`basename $seqFile .tfa`.txt
    echo Processing $seqFile into $outFile

    # Run my application on sequence file (Genscan in this example)
    genvert $seqFile > $outFile

end
                    # Don’t forget to make ‘runGenscan.csh’ executable
```

Demo

- compress, move, and uncompress lots of single sequence files
- make a multiple sequence file
- create a BLAST database
- run BLAST on your database
- extract a sequence from the database
- view results using X Windows

Next week

Sequence analysis with Perl

When a certain DNA string is played backwards on a gene-sequencing machine, it translates into "We can't believe you morons fell for that 'Paul is dead' gag we pulled back in '68! Signed, The Universe"!