

# Introduction to Linux

Inma Barrasa

Bioinformatics and Research Computing

Whitehead Institute

October 14<sup>th</sup> 2021

[http://barc.wi.mit.edu/hot\\_topics/](http://barc.wi.mit.edu/hot_topics/)



WHITEHEAD INSTITUTE



# Outline

- Linux overview
  - Logging in to tak
  - Directory structure
  - Accessing Whitehead drives/finding your lab share
- Basic commands
- Exercises
- BaRC and Whitehead resources
- LSF



# What is Linux?

- Linux is a family of open source Unix-like operating systems. The original Unix is proprietary.
- Our servers run a distribution of Linux, Ubuntu version 20.
- It is powerful, so large datasets can be analyzed.
- Many repetitive analyses or tasks can be easily automated.



# How can we access our Linux server?

- Mac computers come with Unix: You can work on the Mac or use the shell to connect to a server.
- Windows computers need an ssh client like MobaXterm to connect to a Linux server.  
<https://mobaxterm.mobatek.net>
- TAK is our Linux server. It has lots of software and databases already installed or downloaded.
- We have also access to 50 Linux servers that form our cluster.



# Logging in to tak

- Requesting a tak account

<http://bioinfo.wi.mit.edu/bio/software/unix/bioinfoaccount.php>

- Windows

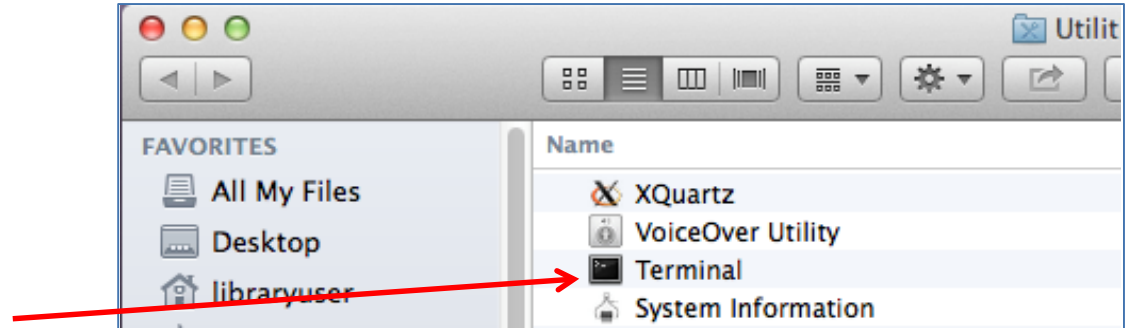
- MobaXterm

- Macs

- Access through Terminal



# Log in to tak for Mac



ssh -Y username@tak.wi.mit.edu

```
libraryuser -- byuan@tak ~ -- ssh -- 79x24
Last login: Wed Oct 1 15:45:01 on ttys000
Library-Corei5-iMac-Epson:~ libraryuser$ ssh -Y byuan@tak.wi.mit.edu
username@tak.wi.mit.edu's password:
```

```
libraryuser -- byuan@tak ~ -- ssh -- 79x24
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-58-generic x86_64)

      Tak

- System information as of Fri Sep 13 15:53:49 EDT 2019
System load: 1.31      Memory usage: 7%    Processes:   597
Usage of /:  8.0% of 438.14GB    Swap usage:  0%    Users logged in: 16

* Find genomes at /nfs/genomes
* Find bioinformatics datasets at /nfs/BaRC_datasets

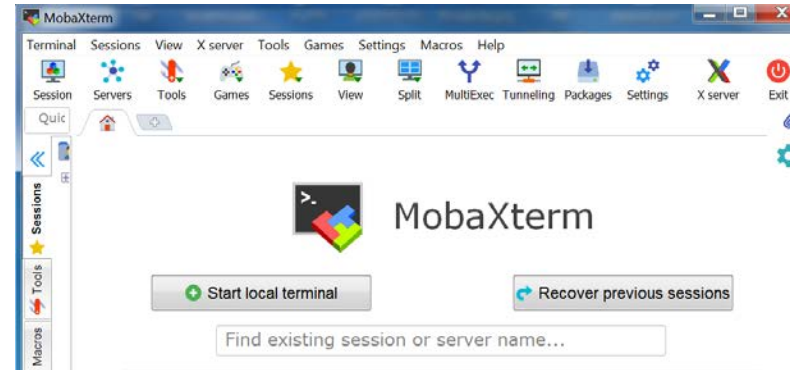
byuan@tak ~$
```



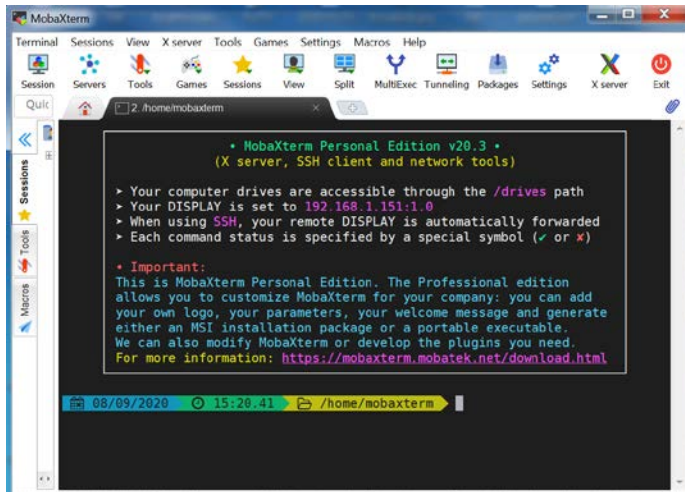
# Connecting to tak from Windows



1) Open MobaXterm

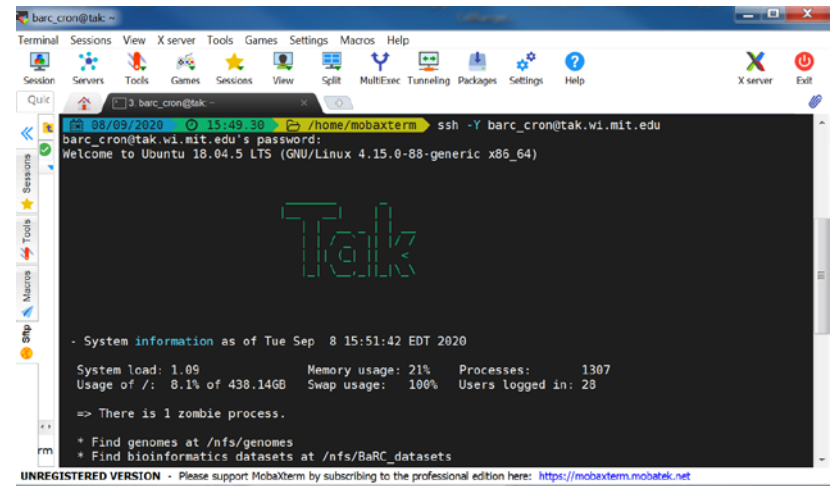


2) Click on the "Start local terminal" button



3) Type:

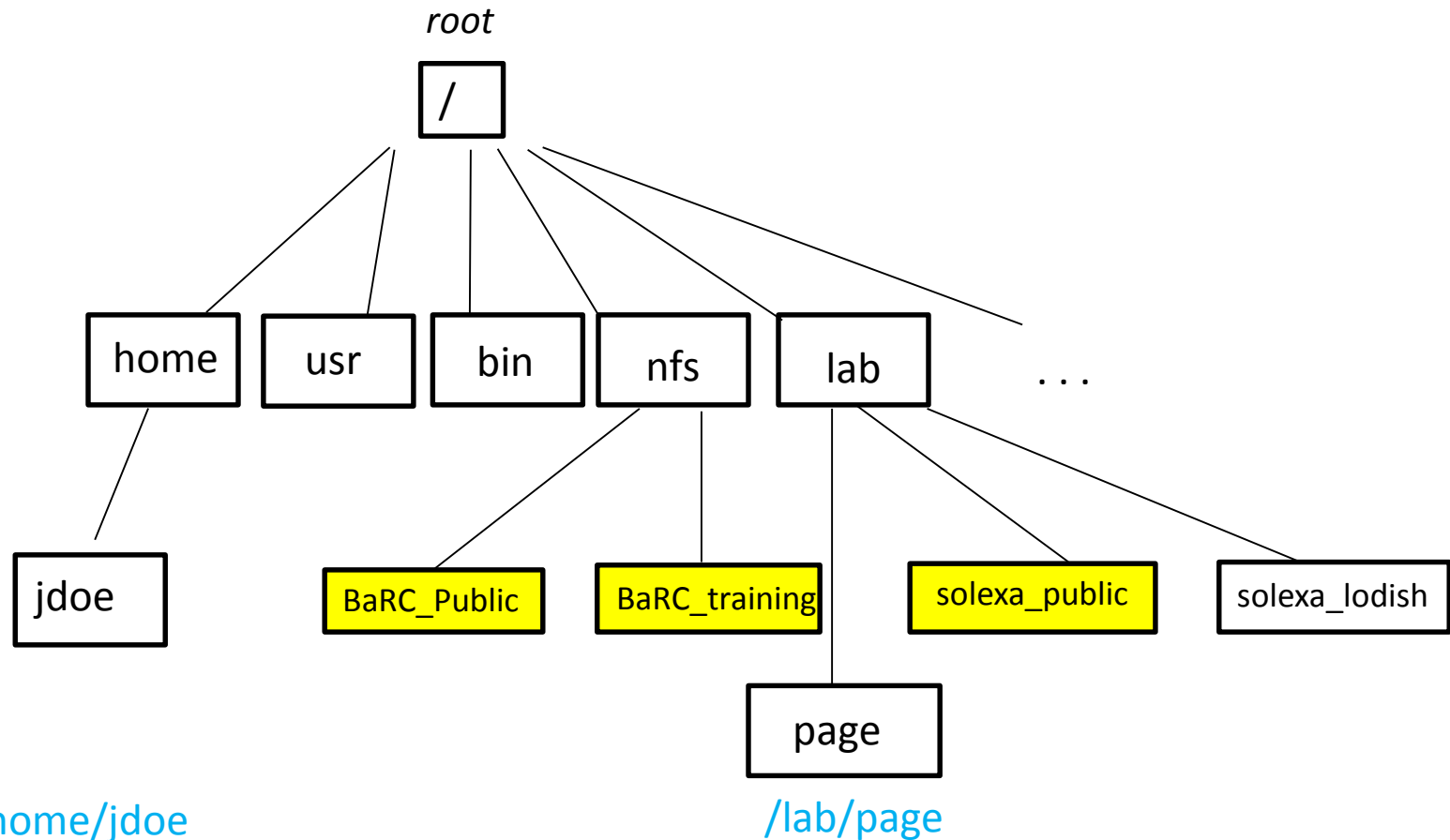
`ssh -Y username@tak.wi.mit.edu`



*Note:*

When you write the password you won't see any characters being typed.

# Unix Directory Structure



See <http://www.thegeekstuff.com/2010/09/linux-file-system-structure/> for more detailed info.  
nfs and lab are directories specific to Whitehead



# Accessing Shared Resources at Whitehead

- Unix
  - /nfs/BaRC\_Public
  - /nfs/BaRC\_training
  - /lab/solexa\_public
- Windows (access using *Start Menu* → *Search*)
  - \\wi-files1\BaRC\_Public
  - \\wi-files1\BaRC\_training
  - \\wi-bigdata\solexa\_public
- Macs (access using *Go* → *Connect to Server...*)
  - smb://wi-files1/BaRC\_Public
  - smb://wi-files1/BaRC\_training
  - smb://wi-bigdata/solexa\_public

Where's my lab's share?

- <http://it.wi.mit.edu/systems/file-storage/lab-share-paths>



# Get ready for the exercises

- Use this link to copy paste the commands for the exercises  
[http://barc.wi.mit.edu/education/hot\\_topics/IntroToLinux\\_2021/IntroToLinux\\_HandsOn.txt](http://barc.wi.mit.edu/education/hot_topics/IntroToLinux_2021/IntroToLinux_HandsOn.txt)
- These folders contain today's slides and the material for the exercises

\\wi-files1\BaRC\_Public\Hot\_Topics\IntroToLinux\_2021

cifs://wi-files1/BaRC\_Public/Hot\_Topics/IntroToLinux\_2021



# Unix/Linux Tips

- Use **↑ ↓** to reuse previous commands
- **Ctrl-c**: stop a process that is running
- **Tab-completion**:
  - Complete commands/file names
- **Unix is case-sensitive**



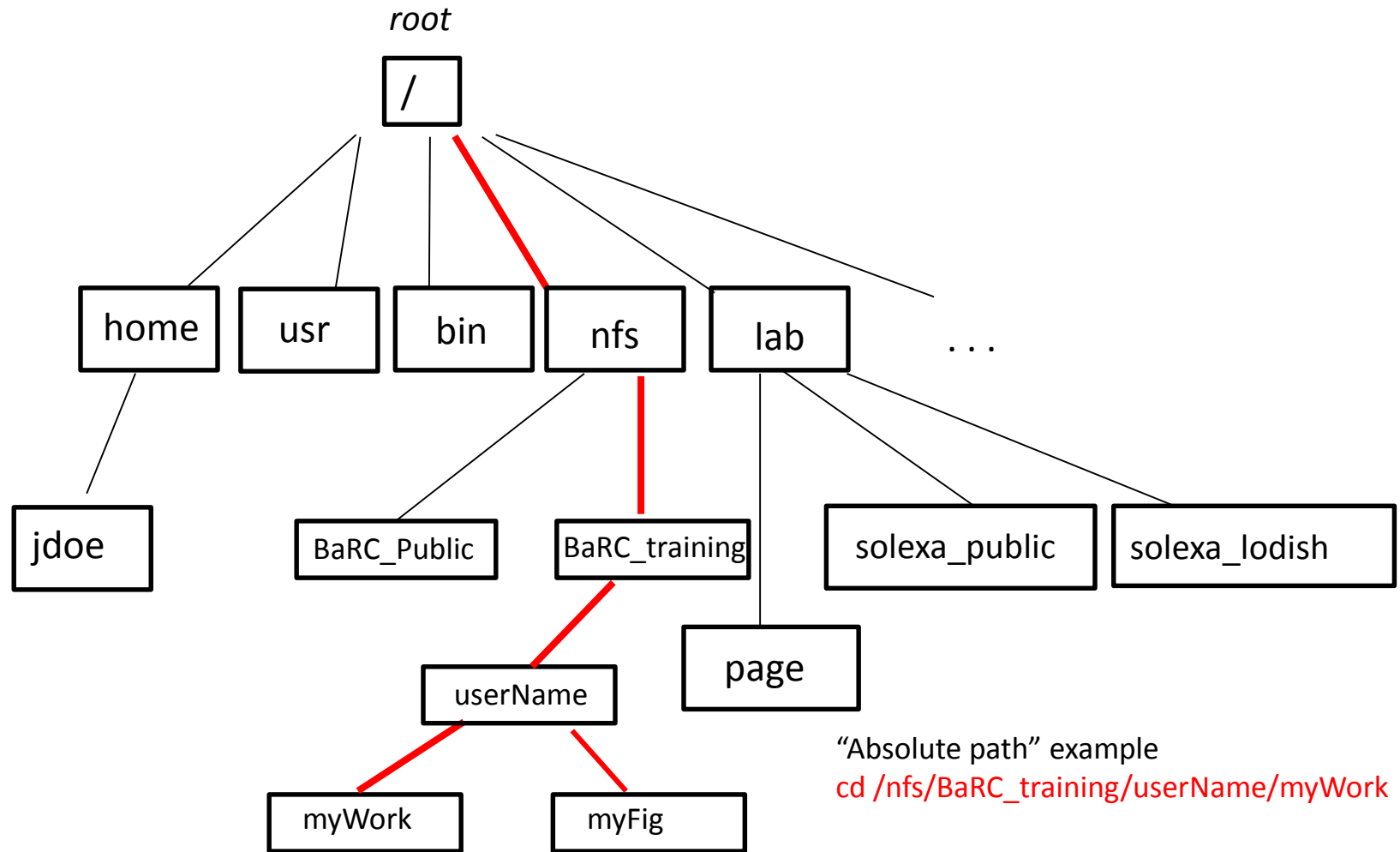
# Basic commands

## listing and organizing files/folders

- list the contents of a directory:
  - `ls` [only show names]
  - `ls -l` [long listing: show other information too]
  - `ls -h` [human readable]
- make a directory: `mkdir dirname`
- change directory: `cd dirname`
- The directories `.` and `..`
  - The current directory (`.`) `ls ./subfolder`
  - The parent directory (`..`)
    - `cd ..` (go to the parent directory) `ls ..` (list to the parent directory)
- print working directory: `pwd`
- Go to your home directory: `cd` or `cd ~`
- Go to the previous directory: `cd -`



# Relative and absolute paths



“Absolute path” example

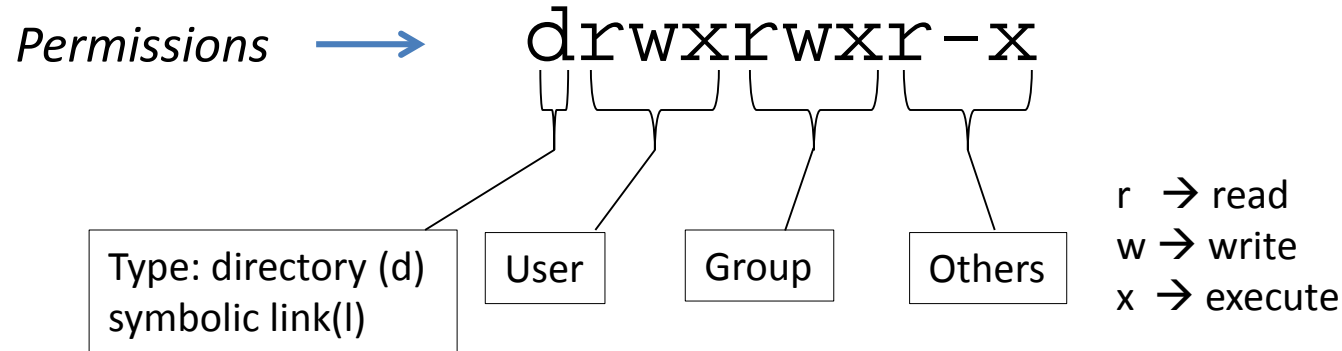
`cd /nfs/BaRC_training/userName/myWork`

“Relative path” example

`cd ../myFig`



# Who do you want to be able to read and/or modify your files?



- Use **chmod** to change permissions
  - user(u), group(g), others(o), all(a)
  - `chmod u+x foo.pl` (user can execute)
  - `chmod g-w foo.pl` (group can't write)

```
ls -l myFile.txt
-rw-r--r-- 1 thiruvil barc 0 2012-10-10 13:32 myFile.txt
```

```
chmod g+w myFile.txt
ls myFile.txt
-rw-rw-r-- 1 thiruvil barc 0 2012-10-10 13:32 myFile.txt
```



# Basic commands

## copying, moving files, getting help

- copy: **cp**
- move: **mv**
- remove: **rm**
- remove directory: **rmdir**
- get help on a command: **man {command}**
- Run a program in the background: **&**

*Do Exercises 1, 2 and 3*



# Displaying the contents of a file on the screen

- **cat** filename: Dump a file to the screen
- **more** filename: Progressively dump a file to the screen:  
ENTER = one line down SPACEBAR = page down q=quit
- **less** filename: like more but with extended capabilities
- **head** filename: Show the first few lines of a file
- **head -n** filename: Show the first n lines of a file
- **tail** filename: Show the last few lines of a file
- **tail -n** filename: Show the last n lines of a file
- **clear**: clear screen





# Editing a File

- Command-line editors
  - pico
  - nano
  - emacs (`emacs -nw`)
  - vi
- Graphical editors (Windows users need an X-windows emulator)

Note: may not be part of standard installation

- nedit
  - gedit
  - xemacs
- Put an `&` at the end of command line to run it in the background when using a graphical editor so that you can continue to use the terminal window

e.g. `gedit myFile.txt&`



# Output Redirection and Piping

- Write output of a command to file
  - Write to output file
    - `sort myFile.txt > myFile_sorted.txt`
  - Over-write output file (if it exists)
    - `sort myFile.txt >| myFile_sorted.txt`
  - Append to output file
    - `sort myFile.txt >> myFile_sorted.txt`
- Piping “|”: use output of one command as input for another command
  - `sort myFile.txt | more`



# Parsing a File: cut

## Word count: wc

- Select columns of interest

```
cut -f 9,12-15 myGeneValues.txt > col_9.12to15.txt
```

### Options:

-f output only these fields

-d field delimiter, default is “\t”

- Count number of lines/words/characters in file

```
wc myFile
```

wc -w (count words only)

wc -l (count lines only)

*Do Exercises 4, 5, 6 and 7*



# Sorting and removing redundancy

## sort and uniq

- Sort on column(s)

```
sort -k 3,3 myGeneExpression.txt | more
```

**Options:**

- n, --numeric-sort compare according to string numerical value
- g, --general-numeric-sort compare according to general numerical value
- r reverse
- k pos1,pos2 start sorting at pos1, end it at pos2
- t, --field-separator=SEP use SEP instead of non-blank to blank transition

- Get only unique entries

```
uniq mySortedGenes.txt > myUniqGenes.txt
```

**Options:**

- c count entries
- d duplicate counts

make sure that the file is sorted before running uniq



Do Exercises 8 and 9



# Searching the contents of a file

- **grep** (global regular expression print)

Find words, or patterns, occurring in lines of a file

```
grep TMEM geneList.txt
TMEM131
TMEM9B
TMEM14C
TMEM66
TMEM49
```

## Options:

- v select non-matching lines
- i ignore case
- n print line number

Example: get TMEM but exclude TMEM14C

```
grep TMEM geneList.txt | grep -v "TMEM14C" | more
```

*Do Exercise 10*



# Getting Files

- Getting files or directories

- Files

- `wget` [http://data.broadinstitute.org/igv/projects/downloads/2.4/IGVSource\\_2.4.2.zip](http://data.broadinstitute.org/igv/projects/downloads/2.4/IGVSource_2.4.2.zip)

- Directories from (outside) servers

- `scp -r origin destination`

- `scp -r userName@serverToCopyFrom:/pathToFolderToCopyFrom .`

- `scp -r jdoe@copper.broadinstitute.org:/broad/lab/works .`



# (Un)Compressing Files

- **.gz file**

Compress: **gzip** file.txt (file.txt.gz will be created)

Uncompress: **gunzip** file.txt.gz (file.txt will be created)

- **.tar.gz file**

Compress: **tar -czvf** myFiles.tar.gz myFiles

Uncompress: **tar -xzvf** myFiles.tar.gz

### Options

-c create an archive

-x extract an archive

-f FILE name of archive

-v be verbose, list all files being archived/extracted

-z create/extract archive with gzip/gunzip

- **View compressed files using:**

**zmore** , **zgrep**



# Other useful commands

- Remove the end of line character introduced by programs like Excel

```
fromdos file.txt
```

(works for files modified on a Mac or PC)

- See all your commands

```
history
```

- See all the hidden characters on the file

```
cat -A file.txt
```





# BaRC Resources

<http://barc.wi.mit.edu/>

The screenshot shows a web browser window displaying the BaRC website. The browser's address bar shows the URL <http://barc.wi.mit.edu/>. The website header features the title "Bioinformatics & Research Computing" in orange, with the subtitle "at Whitehead Institute" and the WI logo. Below the header, a message states: "Consultation and collaboration, training and education, and software in the areas of Bioinformatics and Graphics." A secondary message reads: "Even though we're not physically at Whitehead these days, we continue our mission to help Whitehead scientists! If you'd like to contact us, email, and we can follow up that way or by phone or videoconference." The navigation menu includes "Site Map", "Bioinfo Basics", "Bioinfo Tools", "Graphics", and "Search". The "Bioinfo Tools" dropdown menu is open, listing several resources, with "How-To's (Best Practices)" highlighted by a red arrow. The "Group members:" section lists George Bell, Inma Barrasa, Prathapan Thiru, Troy Whitfield, and Bingbing Yuan. The "contact:" section provides the email address [barc@wi.mit.edu](mailto:barc@wi.mit.edu). The footer contains links for "Whitehead Home", "Inside WI", "Education", and "Publications".

BaRC: Bioinformatics and Research Computing

at Whitehead Institute

Consultation and collaboration, training and education, and software in the areas of Bioinformatics and Graphics.

Even though we're not physically at Whitehead these days, we continue our mission to help Whitehead scientists! If you'd like to contact us, email, and we can follow up that way or by phone or videoconference.

Site Map Bioinfo Basics Bioinfo Tools Graphics Search

Education and training Short course materials  
Linux for bioinfo at WI Hot Topics notes  
Bioinfo resources at WI FAQs  
Bioinfo software at WI How-To's (Best Practices) ← How-To's  
Unix cheat sheet  
List of Unix utilities  
Unix commands for bioinfo

Group members:

George Bell  
Inma Barrasa  
Prathapan Thiru  
Troy Whitfield  
Bingbing Yuan

contact:  
[barc@wi.mit.edu](mailto:barc@wi.mit.edu)

Whitehead Home Inside WI Education Publications



# BaRC Best practices

SOPs – BaRC Wiki - Mozilla Firefox

File Edit View History Bookmarks Tools Help

SOPs – BaRC Wiki

barcwiki.wi.mit.edu 80%

Search

**BaRC**  
Bioinformatics and Research Computing

[Login](#) | [Preferences](#) | [Help/Guide](#) | [About Trac](#)

Wiki Search

wiki: SOPs

## BaRC Best practices

These are "how-to's" detailing the methods that BaRC uses and finds to work effectively. Email BaRC if you have any questions about how or why to perform what is described on these pages.

Frequently Asked Questions (of particular interest to Whitehead scientists) are on our [FAQs site](#).

### Analysis of short read sequencing

- [Downloading and processing NCBI SRA files](#)
- [Quality control and preprocessing of short reads \(fastq files\)](#)
- [Mapping short reads](#)
- [SAM/BAM summarizing and processing](#)
- [SAM/BAM quality control](#)
- [Using RNA-Seq to quantify gene levels and assay for differential expression](#)
- [Using RNA-Seq to assemble or annotate transcripts](#)
- [Using ChIP-Seq to identify and/or quantify bound regions \(peaks\)](#)
- [Using ATAC-Seq to identify open chromatin](#)
- [ngsplot to make stacked heatmaps and profiles of genes or genomic regions \(like ChIP-seq peaks\)](#)



# Unix Commands and BaRC Scripts

**Bioinformatics & Research Computing**  
at Whitehead Institute

Consultation and collaboration, training and education, and software in the areas of Bioinformatics and Graphics.

enter site:

**Site Map**

- Bioinfo Basics
- Bioinfo Tools
- Graphics

Group members:

- George Bell
- Inma Barrasa
- Prathapan Thiru
- Troy Whitfield
- Bingbing Yuan

contact:  
barc@wi.mit.edu

Whitehead Home   Inside WI   Education   Publications

**Navigation Menu:**

- Education and training
  - Unix for bioinfo at WI
  - Bioinfo resources at WI
  - Bioinfo software at WI
- Bioinfo Tools
  - Short course materials
  - Hot Topics notes
  - FAQs
  - How-To's (SOPs)
  - Unix cheat sheet
  - List of Unix utilities
  - Unix commands for bioinfo

A red arrow labeled "UNIX" points to the "Unix for bioinfo at WI" link.

**Bioinformatics & Research Computing**  
at Whitehead Institute

Consultation and collaboration, training and education, and software in the areas of Bioinformatics and Graphics.

enter site:

**Site Map**

- Bioinfo Basics
- Bioinfo Tools
- Graphics
- Search

Group members:

- George Bell
- Inma Barrasa
- Prathapan Thiru
- Troy Whitfield
- Bingbing Yuan

contact:  
barc@wi.mit.edu

Whitehead Home   Inside WI   Education   Publications

**Navigation Menu:**

- Education and training
  - Unix for bioinfo at WI
  - Bioinfo resources at WI
  - Bioinfo software at WI
- Bioinfo Tools
  - Perl scripts for bioinformatics (sample)
  - Perl script repository
  - R scripts (sample)
  - R script repository
  - Local BLAST databases
  - Local genomes
  - BaRC datasets

Red arrows labeled "Perl" and "R" point to the "Perl scripts for bioinformatics (sample)" and "R scripts (sample)" links respectively.



# Running scripts on Unix

- Perl, R or Python scripts

```
bed2gff.pl or ./bed2gff.pl or path/bed2gff.pl  
run_rma_customCDF.R or ./run_rma_customCDF.R  
myScript.py or ./myScript.py
```

- Java Archive (JAR)

```
java -Xmx1000m -jar /usr/local/share/IGVTools/igv.jar
```

- The script has to be executable to be run

```
-rwxrwxr-x 1 byuan  barc  2331 Mar  5  2013 bed2gff.pl
```

- The script that will be executed depends on the path written in front of the script name

"bed2gff.pl" : Unix looks for that script in your path

"./bed2gff.pl" : Unix looks for that script in the current folder path



# BaRC code

Script type	Location
Perl	/nfs/BaRC_Public/BaRC_code/Perl
R	/nfs/BaRC_Public/BaRC_code/R
Python	/nfs/BaRC_Public/BaRC_code/Python



# Commonly used data stored locally at Whitehead

Location	Description
<code>/nfs/genomes</code> <code>\\wi-files1\genomes</code> <code>cifs://wi-files1/genomes</code>	Genome data: gff, gtf, fasta, bowtie indexed files, blat indexed file, etc. for several organisms <code>nfs_genomes_info.html</code> has summary table
<code>/nfs/seq/Data</code>	Sequence data, including blast databases, for several organisms
<code>/nfs/BaRC_datasets</code> <code>\\wi-files1\BaRC_datasets</code> <code>cifs://wi-files1/BaRC_datasets</code>	<code>BaRC_datasets.html</code> in that folder has a summary table



# Submitting jobs to the cluster

## LSF Commands

- **bsub** to submit jobs

```
bsub wc -l reads.fq
```

```
bsub "sort foo.txt > sorted.txt"
```

**Options:**

-e error\_file

-o standard\_out\_file

-m machine (send the job to that machine)

-n number (use that many processors in the cluster)

bsub -n 4 -R "span[hosts=1]"(use 4 processors all in the same machine in the cluster)

- **bjobs** to view your jobs

```
bjobs
```

- **bkill** to kill a job

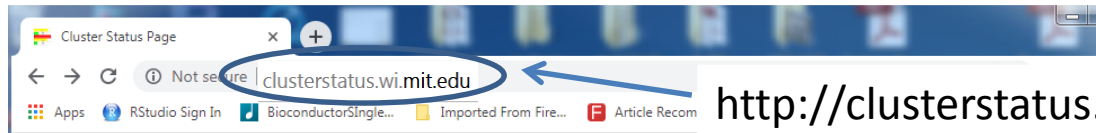
```
bkill 237878
```

"bkill 0" will kill all your jobs

By default the user will receive an email with the job status when the job finishes or stops



# LSF Cluster Activity



<http://clusterstatus.wi.mit.edu/>

Welcome to Tak 4.0

### Software installation requests or issues:

- [Create a new ticket](#)
- [Get an Account](#)
- [Account Help](#)

### Installed software and modules:

- [Installed Packages](#)
- [Installed Perl Modules](#)
- [Installed Python Modules](#)
- [Installed R Modules](#)

### Available Data Sets

- [Find genomes at /nfs/genomes](#)
- [Find bioinformatics datasets at /nfs/BaRC\\_datasets](#)

### Additional Computational Resources

- [R Studio](#) Language and environment for statistical computing and graphics.
- [Jupyter Notebook](#) Web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text.
- [WI UCSC Browser](#) An on-line, and downloadable, genome browser hosted by the University of California, Santa Cruz, but this copy is local!

[Light Tak](#)

[Dark Tak](#)

LSF Cluster Activity		
User	Running	Pending
anhoang	1	0
abraham	1	0
schwarz	48	0
genya	1	0
<b>Totals:</b>	<b>51</b>	<b>0</b>

Last Updated: Tue, 17 Sep 2019 09:11:21

Status of the Hosts				
Host Available	Host Full	Admin Closed	Exclusive Use	Unavailable

Job Slot Utilization of Running Jobs			
Idle	Low	Medium	High

### LSF Cluster Activity

Host and Status	Max Job Slots	Running	Load Average	Cores	CPU (GHz)	Memory (GB)	Uptime	System Info	GPU(s)	Note
c2b10	47	0	0	24	2.60	256	23 days	BL460c G9		New!
c2b11	47	0	0	24	2.60	256	23 days	BL460c G9		New!
c2b12	47	0	0	24	2.60	256	24 days	BL460c G9		New!
c1b2	39	0	0	20	2.80	256	11 days	BL460c G8		
c1b3	39	0	0	20	2.80	256	11 days	BL460c G8		
c1b4	39	0	0	20	2.80	256	11 days	BL460c G8		
c1b5	47	0	0	24	2.70	256	11 days	BL460c G8		
c1b9	47	0	0	24	2.70	256	19 days	BL460c G8		
c1b10	47	0	0	24	2.70	256	19 days	BL460c G8		
c1b11	39	0	0	20	2.80	256	19 days	BL460c G8		
c1b12	39	0	0	20	2.80	256	19 days	BL460c G8		
c1b13	47	0	0	24	2.70	256	23 days	BL460c G8		
c2b7	39	24	23	20	2.80	256	23 days	BL460c G8		
c2b8	47	0	0	24	2.70	256	6 days	BL460c G8		





# Further Reading

- **BaRC: Unix Info**

[http://bioinfo.wi.mit.edu/bio/education/unix\\_intro.php](http://bioinfo.wi.mit.edu/bio/education/unix_intro.php)

<http://bioinfo.wi.mit.edu/bio/bioinfo/scripts/#unix>

- **LSF Cluster (incl. examples)**

[http://bioinfo.wi.mit.edu/bio/bioinfo/docs/LSF\\_help.php](http://bioinfo.wi.mit.edu/bio/bioinfo/docs/LSF_help.php)

- **Whitehead IT Computing Tutorials**

<http://it.wi.mit.edu/training-tutorials/computing-tutorials>



# Upcoming Hot Topics

- Unix: Beyond the Basics (Oct 19th)
- Python: An Introduction - November
- Python: Advanced Topics – November

[http://barc.wi.mit.edu/education/hot\\_topics/upcoming/](http://barc.wi.mit.edu/education/hot_topics/upcoming/)