# Working with Pathogen Genomes 2022
# Introduction to Unix

Fernán Agüero fernan@iib.unsam.edu.ar
Stephen Doyle sd21@sanger.ac.uk

Instituto de Investigaciones Biotecnológicas
IIB_UNSAM

wellcome connecting science

# What is Unix?

Unix is ***a family*** of operating systems

- Unix (AT&T, 1969 – current)
- AIX (IBM; 1986 -- current) COMMERCIAL
- HP-UX (Hewlett-Packard; 1982 -- current)
- Irix (Silicon Graphics; 1988 -- 2013)
- Solaris (Sun Microsystems; 1992 -- current)
- Digital Unix (Digital)
- BSD (Berkeley; 1977 -- current)
  - FreeBSD (1993 – current) **FREE**
  - NetBSD (1993 – current) **FREE**
  - OpenBSD (1996 – current) **FREE**
  - MacOSX – OSX - macOS (2001 – current) COMMERCIAL

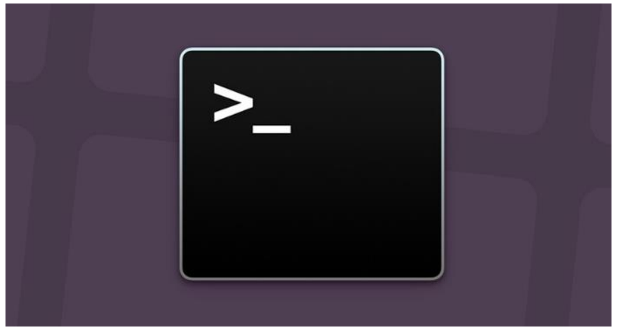
**Linux** (1991 – current)

- A *kernel*, not an operating system **FREE**
- Linux Distributions
  - Kernel + System Software + Libraries
  - RedHat (1993 – current) COMMERCIAL
  - Fedora (2003 – current) **FREE**
  - Debian (1993 – current) **FREE**
  - Ubuntu (2004 – current) **FREE**
  - SUSE (2000 – current) COMMERCIAL
  - OpenSUSE (2005 – current) **FREE**
  - And many others ...

## It seems complicated… but it’s not!

- They are all ***different***
  - Created and maintained by different groups of people
  - Different code bases
  - Different hardware support
- But they are all ***similar***
  - Unifying concept
  - Similar philosophy
    - Modular
    - Simple tools
    - Programs connect between each other
  - Multitasking
  - Multiuser

The Command Line Interface (CLI)



## Why use it?

- Output of Biological Research is often large text files
- Unix makes it easy to work with these files
- Very powerful
- Saves time
- Widely used by the Scientific Community
- Robust + Stable

## User Interfaces

**Graphical User Interface (GUI)**

- Windows, Menus
  - Cursor is controlled by mouse
- Use your mouse to select actions (commands) and run them
  - Copy, Paste, Select, Save
  - Image: Crop, Rotate, Scale
  - Web: Paste Input, Click Button (Run)

**Command Line Interface (CLI)**

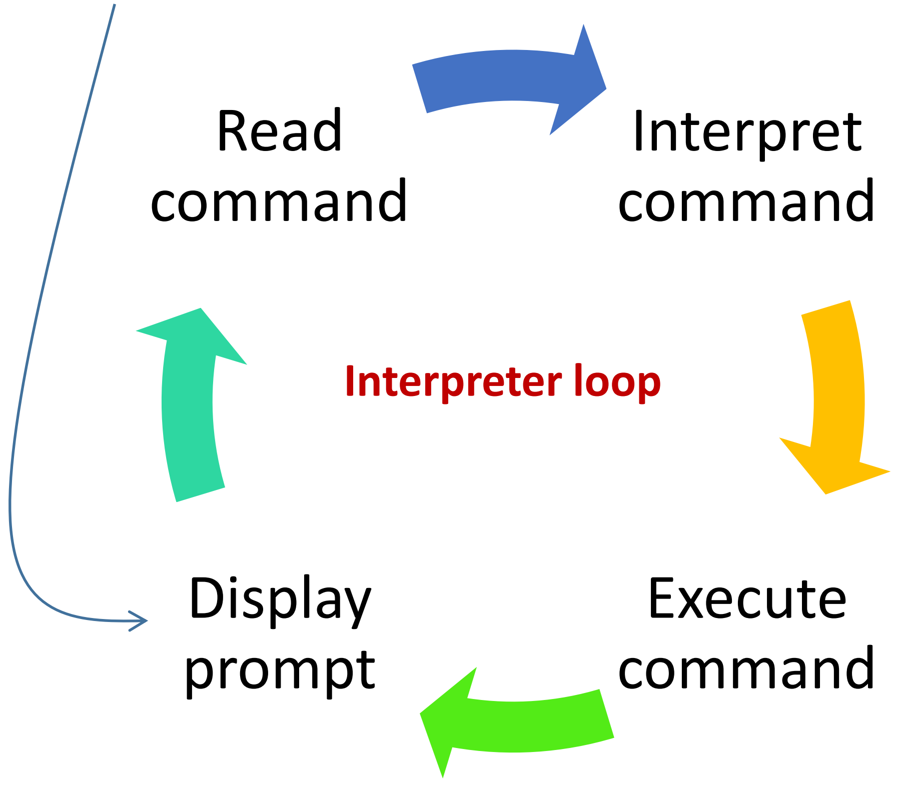
- Prompt (expects input)
  - Cursor blinks
- Use your keyboard to write commands (actions) and run them
  - cat, sed, awk, grep, find, mkdir …
  - blastp, hmmalign, velvet …

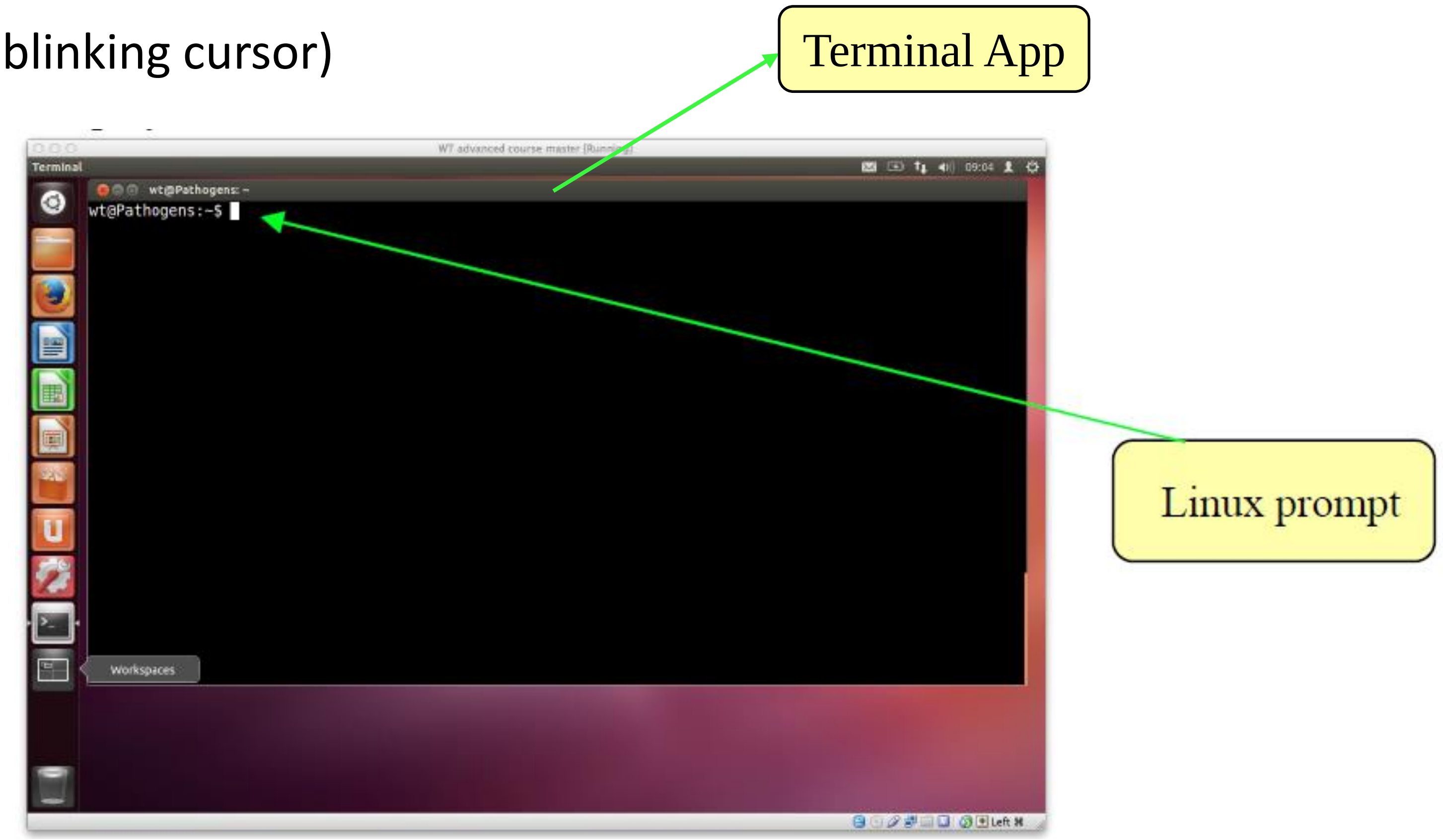
Instituto de Investigaciones Biotecnológicas
IIB_UNSAM

wellcome connecting science

## Interacting with a Unix system (CLI)

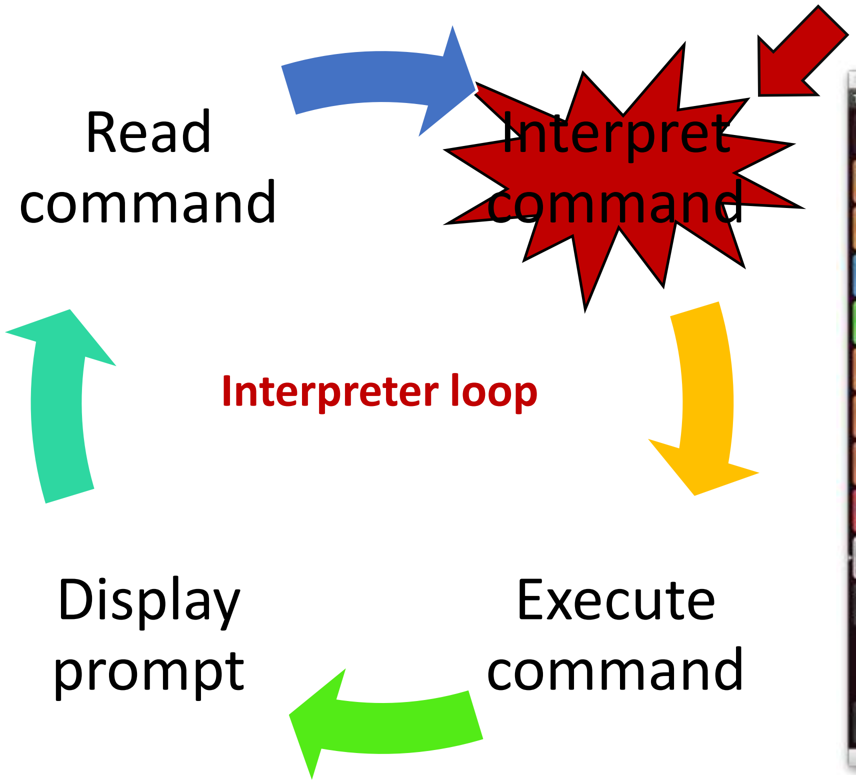
- Use a Terminal to enter commands
  - All terminals run a shell / interpreter
  - The shell waits for input (prompt + blinking cursor)



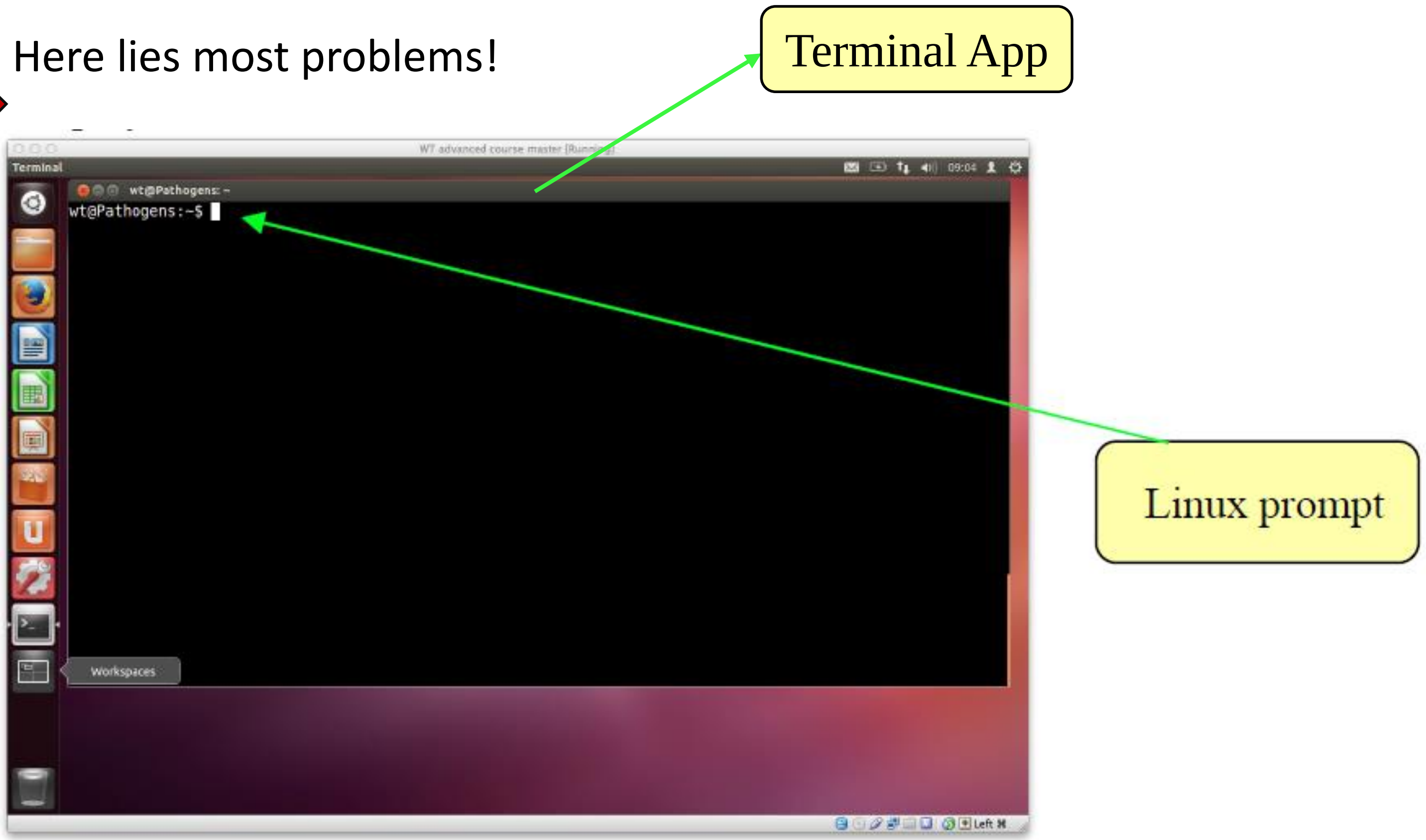


```
wt@Pathogens:~$ 
```

# Interacting with a Unix system (CLI)



Here lies most problems!



```
wt@Pathogens:~$ 
```

Instituto de Investigaciones Biotecnológicas
IIB_UNSAM

wellcome connecting science

## Hints for interacting with the shell

- Unix is ***CASE-SENSITIVE***
  - LS is not the same as ls
- The shell separates what you type into ***words***
  - Spaces must go between a command and the values
  - *mkdir newdir* will work
  - *mkdirnewdir* will not
- **Spelling** must be correct. The shell does not guess, nor autocorrects.
  - *blastn –query this_sequence –db GenBank –out blastn_results*
  - *blasyn –query this_sequence –db GenBank –out blastn_results*

## Hints for interacting with the shell

- There are some special characters that have ***particular meanings*** for the shell, these are
  - ~
  - #
  - $
  - &
  - *
  - ()
  - \
  - /
  - |
  - <
  - >
  - ?
  - !
  - {}
  - []
  - ;
  - ‘
  - “

*And the space character, of course!*

**Tip:** Try to avoid these characters in the names of files and folders

**Tip:** You can use the back slash character (\) to quote (escape) the next character.

Instituto de Investigaciones Biotecnológicas
IIB_UNSAM

wellcome connecting science

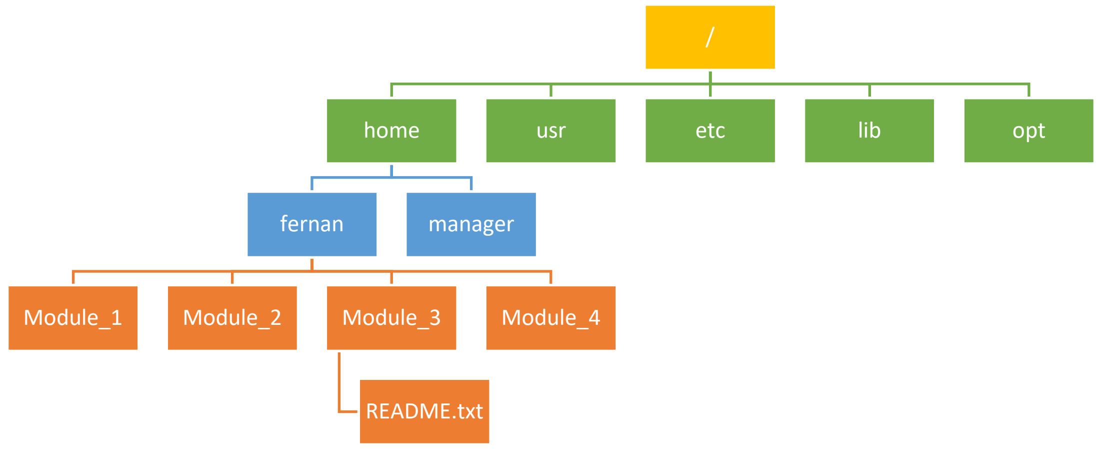
# Structure of Unix commands

- **Command**
  - Always the first word!
- **Parameters**
  - Usually begin with dash(es)
  - Mandatory/Optional?
  - Provide additional information for the program
    - **Input**
    - **Output**
  - Or act as ***modifiers***
    - Change how the command runs or how it does it

```
mkdir  directory_1 directory_2 directory_3

mkdir –p directory_4/sub_directory_a/sub_sub_directory_z

blastp –query prot.fasta –db swissprot –gapopen 10 –word_size 5 –out results.txt

rsync –a –v –z fernan@lab.unsam.edu.ar:/home/fernan/work MyBackups/
```

**Tip:** depending on the command, parameters can be grouped, e.g. for rsync –avz is the same as –a –v –z (but do check the documentation for each command!)

## The Unix filesystem

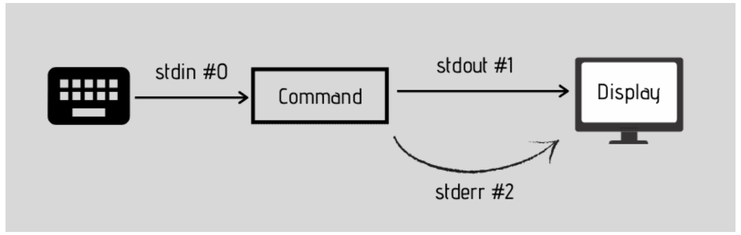


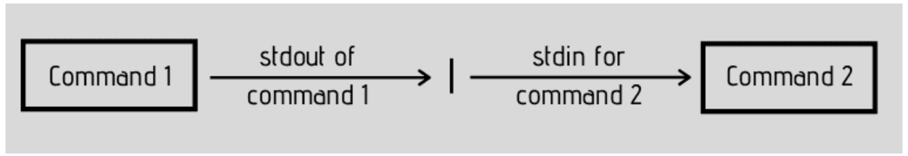
**Tip:** absolute paths always start from the root. Relative paths can start anywhere. Use the special characters "." (a dot), or ".." (two dots) to refer to the current directory (.) or to the parent directory (..)

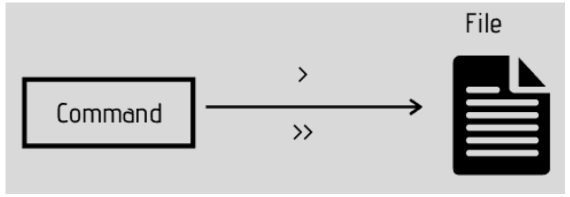
- /
  - The ***root*** directory
  - Yes, literally a single character!
- Because all other directories or files are descended from root, the ***absolute path*** of any file is traversed through root.
  - The path to a *README.txt* file in the Module_3 folder is /home/fernan/Module_3/README.txt

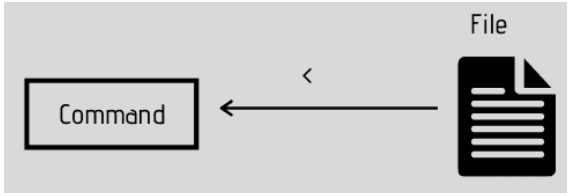
# Connecting programs in Unix

- All commands have access to 3 streams of data: INPUT, OUTPUT, ERROR
- The ***standard*** streams of data are:
  - STANDARD INPUT (**STDIN**) = the keyboard
  - STANDARD OUTPUT (**STDOUT**) = the screen
  - STANDARD ERROR (**STDERR**) = the screen (for error messages)
  - But these can be **redirected!**
- Using ***pipes*** the output of a command can be redirected as the input of the next command. The pipe operator is **|**
- The output of a command can be redirected to a file. The output redirection operator is **>**
- The input to a command can come from a file (instead of the keyboard). The input redirection operator is **<**









Instituto de Investigaciones Biotecnológicas IIB_UNSAM

More about this here: [https://linuxhandbook.com/redirection-linux/](https://linuxhandbook.com/redirection-linux/)

wellcome connecting science

# Programming with the Unix Shell

- You can store values into ***variables***
  - files="genome1.fasta genome2.fasta genome3.fasta"
- A ***series of shell commands*** can be executed as a "program"
  - for file in ${files}; do blastn –query ${file} –db refseq –out ${file}.out.blastn; done
- Write and save these commands in a ***text file***
- Make the file executable using the ***chmod*** command (change mode)
  - chmod +x my_shell_script.sh
- Execute it using a shell interpreter (bash, csh, tcsh, zsh, sh)!
  - bash my_shell_script.sh

**Tip:** yes, there are many shells out there. Check your sysadmin to see which one is installed for you. In Ubuntu it is usually bash.

Instituto de Investigaciones Biotecnológicas
IIB_UNSAM

wellcome connecting science