

Pablo Vinuesa (vinuesa@ccg.unam.mx)
Centro de Ciencias Genómicas-UNAM, México
<http://www.ccg.unam.mx/~vinuesa/>

Tema 1: Introducción al biocómputo en sistemas Unix/Linux
<https://github.com/vinuesa/TIB-filoinfo>

- ¿Qué son la bioinformática y el biocómputo, y cómo pueden ayudarme para mi trabajo en biología?
- ¿Qué es UNIX y Linux?
- Tengo una PC que corre windows, ¿cómo puedo correr Linux en mi máquina?
- ¿Cómo hago trabajar a UNIX/Linux? – el shell: comandos y conceptos básicos
- Uso de tuberías de comandos UNIX/Linux para procesar archivos de secuencias

Introducción al biocómputo en sistemas UNIX/Linux

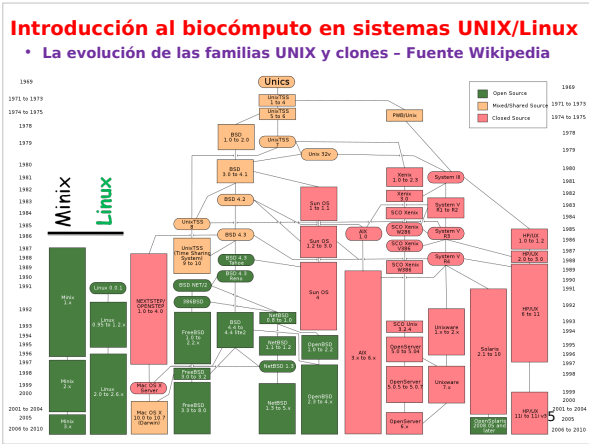
• ¿Qué es UNIX? - fuente: Wikipedia

- Unix (registrado oficialmente como UNIX®) es un sistema operativo portable, multitarea y multiusuario
- su desarrollo inicia en 1969 por un grupo de empleados de los laboratorios Bell de AT&T, entre los que figuran Ken Thompson, Dennis Ritchie y Douglas McIlroy
- UNIX es un Sistema Operativo no libre muy popular, porque está basado en una arquitectura que ha demostrado ser técnicamente estable.
- MacOS X es un derivado de UNIX BSD!










Ken Thompson y Dennis Ritchie
Fuente: Wikipedia

3



¿Dónde estoy y cómo desarrollo habilidades en biocómputo?

Biología	Biología computacional	Bioinformática	C. de la computación
Yo uso Windows y software pirata viejo, además de free-ware de código fuente cerrado: Virus, spyware, malware me acosan... SOCORRO!	Yo programo en el ambiente Linux y uso software de código fuente abierto		
	Bash, Perl, Python, R, MySQL, PHP, HTML5, Apache ...		
	*****	C, C#, C++, Java ...	
	Sé un poco de R	*****	Ensamblador FTW
	Sé un poco de Perl o Python	*****	
	Sé un poco de Linux y Shell	*****	
Hago filogenias con MEGA y gráficos con Excel			
Corro BLAST en el portal de NCBI con secuencias que almaceno en documentos Word			

Introducción al biocómputo en sistemas UNIX/Linux

• ¿Qué es Linux? - Evolución de sistemas UNIX y similares a UNIX (fuente: Wikipedia)

GNU: En 1983, Richard Stallman anunció el Proyecto GNU, un ambicioso esfuerzo para crear un sistema similar a Unix, que pudiese ser distribuido libremente. El software desarrollado por este proyecto -por ejemplo, GNU Emacs y GCC - también han sido parte fundamental de otros sistemas UNIX. (vean conferencias de R. Stallman en youtube ...)



Richard Stallman

Linus Torvalds

Linux: En 1991, cuando Linus Torvalds empezó a proponer el núcleo Linux y a reunir colaboradores. Las herramientas GNU eran la elección perfecta. Al combinarse ambos elementos, conformaron la base del sistema operativo (basado en POSIX) que hoy se conoce como GNU/Linux.

Las distribuciones basadas en el núcleo, el software GNU y otros agregados como Red Hat Linux y Debian GNU/Linux, se han hecho populares tanto entre los aficionados a la computación como en el mundo empresarial y científico. Linux tiene un origen independiente, por lo que se considera un 'clon' de UNIX y no un UNIX en el sentido histórico.

4

Introducción al biocómputo en sistemas UNIX/Linux

• Tengo una PC que corre windows, ¿cómo puedo correr Linux en mi máquina?

1. Puedes instalar Linux en una nueva partición (Lo más recomendable) .

descargas gratuitas de distribuciones desde:

=> 1. Ubuntu 18.04 LTS - <http://www.ubuntu.com/getubuntu/download>

Ver instrucciones de instalación aquí:

<http://www.ubuntu.com/download/desktop/install-desktop-latest>

Centos - <https://www.centos.org/download/>

Fedora - <http://fedoraproject.org/es/get-fedora>

Biolinux - <http://environmentalomics.org/bio-linux/>

2. Puedes instalar MobaXterm, que proporciona una terminal para Windows con un servidor de ambiente gráfico X11, un cliente SSH para establecer sesiones remotas seguras con un servidor, diversas herramientas de red y más.

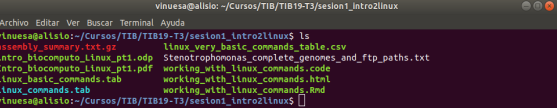
<https://mobaxterm.mobatek.net/download.html>

6

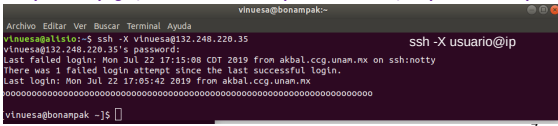
Introducción al biocómputo en sistemas UNIX/Linux

Tipos de sesiones shell

1) local (acceder a tu computadora)



2) remota (login; conexión a otra máquina en Internet, vía protocolo ssh)

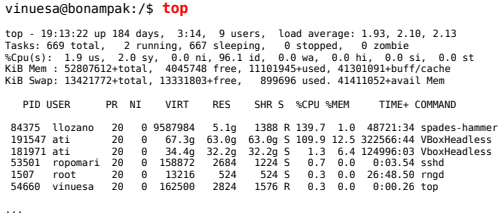


Introducción al biocómputo en sistemas UNIX/Linux

explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

1. - ¿Qué procesos se están corriendo en el sistema?

top - monitor system load

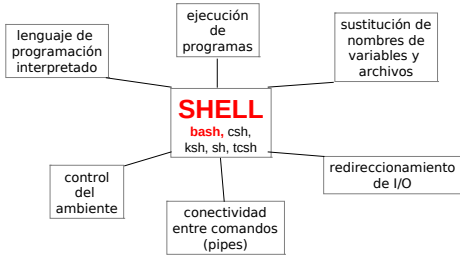


9

Introducción al biocómputo en sistemas UNIX/Linux

explorando el nuevo ambiente ... El shell y la interpretación de comandos

- Las responsabilidades del shell -



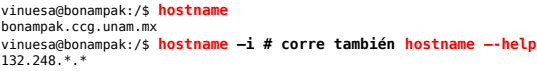
11

Introducción al biocómputo en sistemas UNIX/Linux

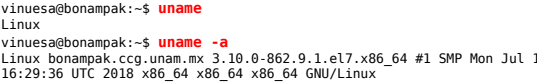
explorando el nuevo ambiente ... El shell y la interpretación de comandos, unos ejemplos

1. - ¿Qué máquina es ésta a la que estoy conectando?

hostname - read or set the hostname or the NIS domain name



uname - Print certain system information

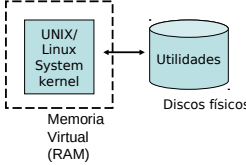


8

Introducción al biocómputo en sistemas UNIX/Linux

explorando el nuevo ambiente ... El shell (consola) y la interpretación de comandos

- ¿Qué es el shell?



Los sistemas UNIX/Linux se dividen lógicamente en dos piezas: el **kernel** y las **utilidades**.

El kernel es el corazón del sistema y reside en la memoria de la computadora desde el momento que se arranca y hasta que se apaga.

Las utilidades (comandos), residen en el disco físico y se cargan en memoria sólo cuando son llamadas.

El shell también es un programa. Se carga automáticamente en memoria desde que uno hace el login a una máquina para que el usuario pueda interactuar con ella.

10

Introducción al biocómputo en sistemas UNIX/Linux

explorando el nuevo ambiente ... El shell y la interpretación de comandos

- ¿Dónde encuentro una lista y descripción básica de los comandos disponibles ?

1. Mira estas entradas en Wikipedia:

http://en.wikibooks.org/wiki/Linux_Guide/Linux_commands

http://en.wikipedia.org/wiki/List_of_Unix_programs

2. Y corre estos comandos para ver parte de los programas del sistema instalados en el servidor o en tu máquina:

```
ls /bin
```

```
ls /usr/bin
```

3. Un sencillo tutorial que todos deberían haber revisado ya (caps 1-5):

<http://www.ee.surrey.ac.uk/Teaching/Unix/>

12

© Pablo Vinuesa, 2019. vinuesa[at]ccg.unam.mx;
<http://www.ccg.unam.mx/~vinuesa/>

2

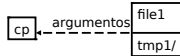
Introducción al biocómputo en sistemas UNIX/Linux

* explorando el nuevo ambiente ... El shell y la interpretación de comandos

Ejecución de programas por el shell – ejemplos

formato básico de un comando

comando [argumento1 arg2 arg3 ...]



```
-bash-3.1$  
-bash-3.1$ cp file1 tmp1/  
-bash-3.1$
```

¡los espacios separan argumentos!

necesito al menos 1 espacio entre comandos y argumentos!
Nombres de archivos en UNIX/Linux de preferencia no deben contener espacios

* Además de **argumentos**, los comandos pueden tener **opciones**, las cuales preceden a los argumentos y llevan un guión sencillo delante de una o más opciones.

```
-bash-3.1$ ls -l /home # prueba también solo ls /home  
drwxr-xr-x. 2 aagarcia students 4096 Oct 30 2014 aagarcia  
drwxr-xr-x. 2 aanaya students 4096 Oct 30 2014 aanaya  
drwxr-xr-x. 2 acamos students 4096 Oct 30 2014 acamos  
drwxr-xr-x. 2 acarmona students 4096 Oct 30 2014 acarmona  
drwxr-xr-x. 4 agodinez students 4096 Jul 1 2016 agodinez  
drwxr-xr-x. 2 aguzman students 4096 Oct 30 2014 aguzman
```

13

Comandos Básicos

- ls (list)
 - \$ ls -l
 - \$ ls -a
 - \$ ls -la
 - \$ ls -l --sort=time # ls -ltr
 - \$ ls -l --sort=size -r #ls -lsr
 - \$ ls *.txt
- cd (change directory)
 - \$ cd dir
 - \$ cd /dir
 - \$ cd ./
 - \$ cd ../
 - \$ cd /export/space2/tib/filo
- pwd (print working directory)
 - \$ pwd
- ~
- \$ cd ~
- ~user
- \$ cd ~vinuesa
- ¿Qué hará "cd ~vinuesa"?
- ¿y cd?

- which
 - \$ which blastn
- locate
 - \$ locate get_homologues.pl
 - \$ locate mi_archivo
- find
 - \$ find / | grep stdio.h
 - \$ find /usr/include | grep stdio.h
 - \$ find . -type d
 - \$ find /home/vinuesa -name "TIB"
- man (manual pages for command)
 - \$ man ls
 - \$ man find
 - \$ man man

Comandos Básicos (cont.)

- tar & gzip
 - \$ tar cvzf dir1.tar.gz dir1
 - \$ ls -l /home | less
 - \$ tar cvzf dir1.tar.gz dir1
 - \$ tar cvpf dir1.tar dir1
- gzip (gunzip, compress)
 - \$ gzip -9 dir1.tar
 - # genera dir1.tar.gz
- untar & unzip
 - \$ tar xvfz dir1.tar.gz
- touch
 - \$ touch foo
- head
 - \$ head archivo.txt
- tail
 - \$ tail archivo.txt
 - \$ tail -f archivo2.txt
- Pipe, >, grep, sort, cut, uniq
 - \$ ls -l /home | less
 - \$ cat /etc/passwd | cut -d: -f1 | sort
 - \$ ls -l /home | grep stud | wc -l > stud.txt
 - \$ ls -l /home | grep -v stud | sort | uniq -c
- backticks
 - \$ echo "The date is `date`"
 - \$ echo `seq 1 10`
- Hard, soft (symbolic) link
 - ln -s /path/to/gbk_files/*.gbk .
- scp (secure copy)
 - scp arch1 vinuesa@buluc.lcg.unam.mx: \$HOME/tmp

Introducción al biocómputo en sistemas UNIX/Linux

* explorando el nuevo ambiente ... El shell y la interpretación de comandos

Tecleando comandos para el shell – el ciclo de comandos



El shell encuentra y ejecuta los programas (binarios o ejecutables) que se encuentran en los

directorios guardados en la **variable de ambiente \$PATH**

Ejecuta en la terminal el siguiente comando:

(echo imprime el valor de una variable y/o una lista de argumentos)

```
-bash-3.1$  
vinuesa@bonampak:/ $ echo $PATH
```

14

Comandos Básicos (cont.)

- echo (print to STDOUT)
 - \$ echo "Hello World"
 - \$ echo -n "Hello World"
- cat (concatenate)
 - \$ cat /proc/cpuinfo
 - \$ cat arch1 arch2
- cp (copy)
 - \$ cp arch1 dir1
 - \$ cp -r dir1 ~vinuesa/tmp
- mv (move or rename)
 - \$ mv arch1 archivo1
 - \$ mv arch1 ~vinuesa/tmp
- mkdir (make directory)
 - \$ mkdir dir2
 - \$ mkdir -p dir2/practica1
- rm (remove)
 - \$ rm arch1
 - \$ rm -r dir2
- less (paginador)
 - \$ less archivo.txt # q para salir
- head (ver cabecera del archivo)
 - \$ head -5 archivo1.txt
- tail (ver cola del archivo)
 - \$ tail -1 archivo.txt
 - \$ tail -f logfile.txt
- sed (stream editor)
 - \$ sed 's/esto/aquello/' archivo.txt
- vim (vi improved; a powerful command line text editor in Linux)
 - \$ gedit (editor de texto con interfaz gráfica en gnome)
 - \$ nedit (otro editor de texto con interfaz gráfica)

Comandos Básicos (cont)

- Uso del disco duro
 - \$ df -h /
- Uso disco de archivos
 - \$ du -sxh ~/
- Uso avanzado: programación del Shell ☺
 - Asignación de variables y bucles for
 - var1=123; echo \$var1; var2=/home/pepin.&& echo \$var2
 - for file in *faa; do muscle < \$file > \${file%.*}_aln.faa; done
 - for file in \$(ls *faa | grep rpoB); do echo -n \$file; grep -c '>' \$file; done

Linux text editors

- con entorno gráfico
 - **gedit**
 - nedit
- Sin entorno gráfico
 - **Vim**
 - emacs
 - nano
 - pico

Introducción al biocómputo en sistemas UNIX/Linux

- explorando el nuevo ambiente ... Trabajando eficientemente con el shell

¿Cómo me muevo en la línea de comandos?

- Usa **ctrl-e** para ir al final de la línea
- Usa **ctrl-a** para ir al principio de la línea

¿Cómo edito la línea de comandos?

- Usa la tecla **backspace** para eliminar uno a uno caracteres (del final hacia el principio)
- Usa **ctrl-w** para eliminar una palabra completa (del final hacia el principio)
- Usa **ctrl-u** para eliminar la línea completa (del final hacia el principio)

¿Cómo aborto o suspendo la ejecución de un comando?

- Usa **ctrl-c** para abortar la ejecución del último comando
- Usa **ctrl-z** para suspender la ejecución del último comando
- Usa **bg** para poner este último comando a correr en el fondo (background)

Repetición de la ejecución de un comando y completado de nombres de comandos/archivos

- Unix recuerda los comandos ejecutados: usa flecha arriba o abajo para moverte por el historial de comandos "history" file o escribe **history | grep comando**
- Usa **TAB** para completar automáticamente el nombre de comandos, archivos o directorios

21

Vim – the Linux power text editor

- 2 modes
 - Input mode
 - ESC to back to cmd mode
 - Command mode
 - Cursor movement
 - h (left), j (down), k (up), l (right)
 - ^f (page down)
 - ^b (page up)
 - ^ (first char.)
 - \$ (last char.)
 - G (bottom page)
 - :1 (goto first line)
 - Switch to input mode
 - a (append)
 - i (insert)
 - o (insert line after)
 - O (insert line before)
- Delete
 - dd (delete a line)
 - d10d (delete 10 lines)
 - d\$ (delete till end of line)
 - dG (delete till end of file)
 - x (current char.)
- Paste
 - p (paste after)
 - P (paste before)
- Undo
 - u
- Search
 - /
- Save/Quit
 - :w (write)
 - :q (quit)
 - :wq (write and quit)
 - :q! (give up changes)

Hagamos unas prácticas -
Navegando el sistema de archivos y herramientas de filtrado

