



# Programación de Sistemas

## CCPG1008

---

Federico Domínguez, PhD.

Unidad 1 – Sesión 2: El Shell de Linux

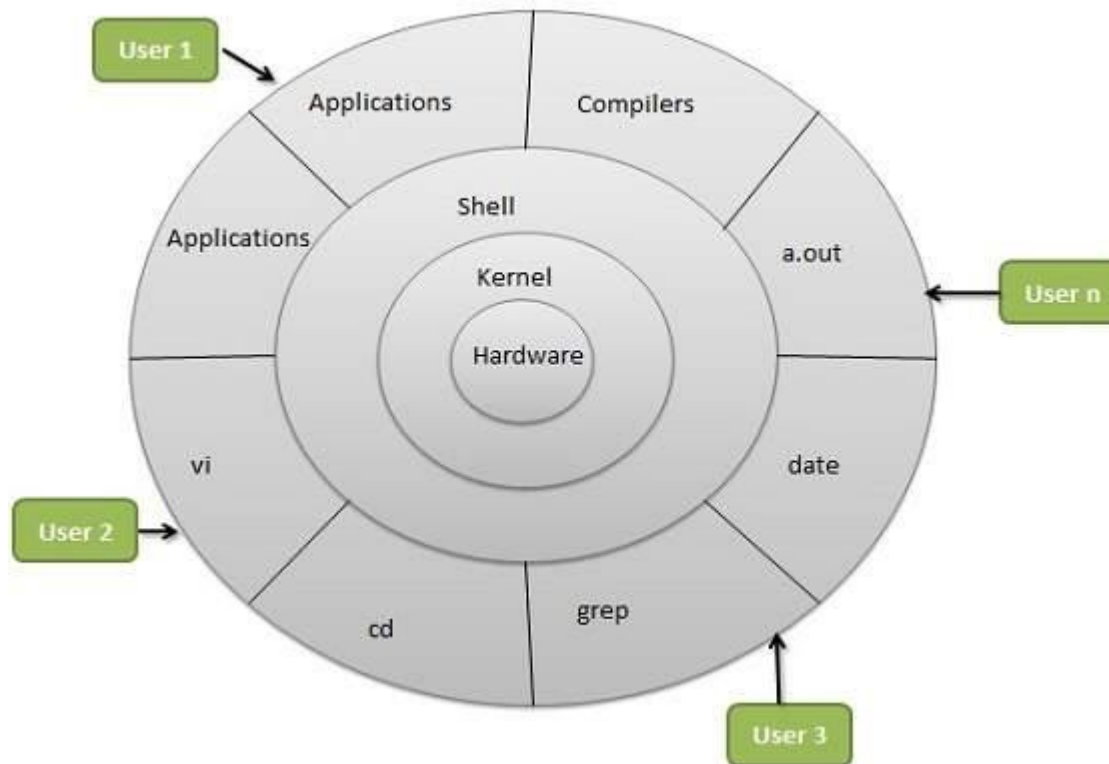
# Contenidos

---

1. Repaso
2. ...más sobre el sistema de archivos de Linux
3. ¿Qué es el Shell de Linux?
4. Comandos básicos: *ls*, *cd*, *pwd*, *date*, *rm*, *mkdir*, *mv*...
5. Práctica de laboratorio: Uso básico del shell

# Repaso

## Arquitectura de LINUX



**Hardware:** CPU y periféricos

**Kernel:** Núcleo del sistema operativo, controla todo el funcionamiento del sistema.

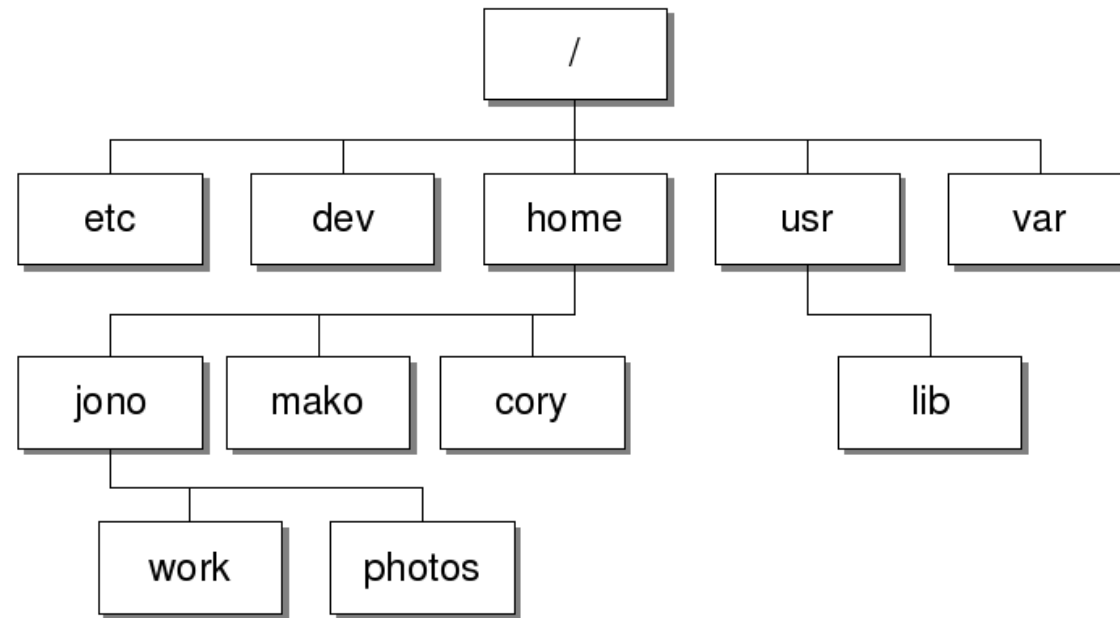
**Shell:** Interface entre el Kernel y las aplicaciones de usuario. Esconde la complejidad del Kernel, ejecuta comandos de usuarios y aplicaciones.

# Repaso

---

## Sistema de archivos

- Es monolítico, empieza desde la raíz o “root” representada por “/”



# ...más sobre el sistema de archivos de Linux

---

La distribución y uso del sistema de archivos de Linux se mantiene relativamente similar entre distribuciones.

- `/etc`: Archivos de configuración globales
  - `/etc/passwd` – Usuarios del sistema
  - `/etc/fstab` – dispositivos de almacenamiento y sus puntos de montaje
  - `/etc/network` – Configuración de red del equipo
- `/home`: Carpetas de usuarios
- `/lib`: Librerías del sistema
- `/media`: Puntos de montaje de dispositivos de almacenamiento
- `/mnt`: Igual que `/media` pero para versiones viejas de Linux
- `/usr`: Contiene todos los programas y archivos de soporte
  - `/usr/bin`: Contiene los ejecutables del sistema.
  - `/usr/lib`: Contiene librerías compartidas
- `/var`: Datos que pueden cambiar son grabados aquí (bases de datos, logs, correos del sistema, etc..)
  - `/var/log`: Aquí se guardan los logs
- `/dev`: Representación de periféricos y dispositivos de hardware del sistema.

# ¿Qué es el Shell de Linux?

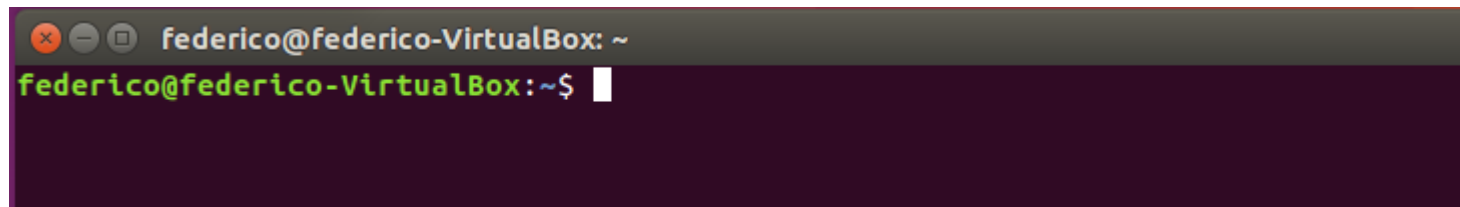
---

Conocido también como la línea de comandos, es el programa que interpreta los comandos del usuario y los pasa al sistema operativo (kernel) para su respectiva ejecución.

En Linux, el Shell más popular es **Bash** (*Bourne-again Shell*), creado para el proyecto GNU por Brian Fox. **Bash** está basado en **sh**, el Shell usado en UNIX y creado por Steve Bourne.

**bash** y **sh** no son los únicos shells, por ejemplo:

- ash: Shell de bajos recursos computacionales usado en FreeBSD y similares.
- dash: Shell usado en Debian.
- csh: Shell que usa C como lenguaje de scripting.

A screenshot of a terminal window. The title bar at the top shows window control buttons and the text 'federico@federico-VirtualBox: ~'. The terminal content shows a green prompt 'federico@federico-VirtualBox:~\$' followed by a white cursor block.

# Comandos básicos del shell

---

¿Dónde estoy? → **pwd**: Print Working Directory

¿Qué hay aquí? → **ls**: List directory

Me voy a ... → **cd**: Change directory

```
federico@federico-VirtualBox: /etc
federico@federico-VirtualBox:~$ pwd
/home/federico
federico@federico-VirtualBox:~$ ls
Desktop    Downloads    foo    Pictures    Templates
Documents  examples.desktop  Music  Public    Videos
federico@federico-VirtualBox:~$ cd /etc
federico@federico-VirtualBox:/etc$ pwd
/etc
federico@federico-VirtualBox:/etc$
```

# Pathnames

---

Pathnames (caminos) es una dirección en el sistema de archivos. Tiene dos formas:

- Absoluta: Empieza desde la raíz
  - `/etc/passwd`
  - `/home/pedro`
  - `/usr/lib/ssl/libcrypt.so.5`
- Relativa: Empieza desde el directorio actual usando la notación especial `./`
  - `./bin` o `bin` (si no empieza con `./` se asume que es relativo)
  - Casos especiales:
    - `../bin`, `..` significa el directorio padre
    - `~/Pictures`, `~` significa el home del usuario



# Opciones y argumentos

---

Comandos de Shell pueden recibir opciones y argumentos:

`comando -opciones argumentos`

Opciones cortas, son una letra, se especifican con –

- `ls -l` : Listar con formato largo
- `ls -a`: Listar mostrando archivos ocultos

Las opciones cortas se acumulan:

- `ls -la`: Listar con formato largo mostrando archivos ocultos

Opcionaes largas, son una palabra, se especifica con --

- `ls --all` : igual a `ls -a`

# Formato largo de `ls -l`

```
% ls -al
```

total 94							
drwxr-xr-x	2	john	doc	512	Jul 10 22:25	.	
drwxr-xr-x	4	bin	bin	1024	Jul 8 11:48	..	
-rw-r--r--	1	john	doc	136	Jul 8 14:46	.exrc	
-rw-r--r--	1	john	doc	833	Jul 8 14:51	.profile	
-rw-rw-rw-	1	john	doc	31273	Jul 10 22:25	ch1	
-rw-rw-rw-	1	john	doc	0	Jul 10 21:57	ch2	

type

access modes

# of links

owner

group

size (in bytes)

modification date and time

name

# Explorar el contenido de un archivo de texto

---

Muchos archivos en Linux son de texto (en contraste a los archivos binarios). Un uso común de los archivos de texto son los archivos de configuración.

Para explorar el contenido de archivos de texto:

- `less`
- `more`
- `cat`
- `tail`
- Editores de texto como `vi`, `vim`, `nano`, `emacs`, etc.
- Ejemplo, ver las últimas 5 líneas del archivo de log del sistema:
  - `tail -n 5 /var/log/syslog`
- Recomendado: **`less`**, bastante completo y **`cat`**, bastante simple

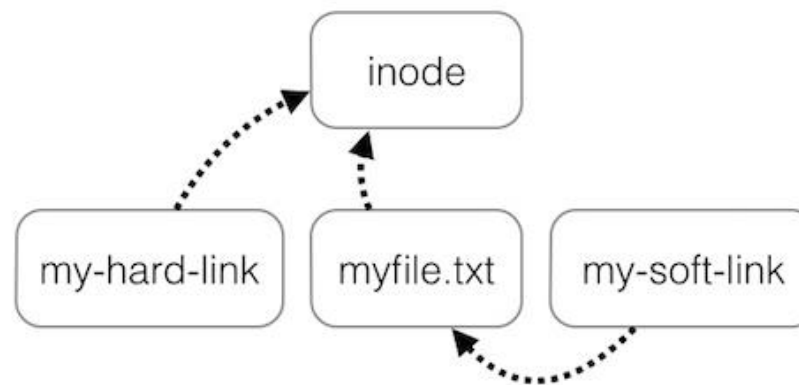
# Enlaces simbólicos

---

Comando `ln` crea enlaces o “alias” de archivos.

Dos tipos de enlaces simbólicos:

- `ln`: Enlace hard, es igual al archivo original, es un enlace a bajo nivel
- `ln -s`: Enlace soft, similar a un acceso directo.
- Ejemplo:
  - `ln -s /etc/passwd usuarios` (crea un enlace simbólico suave llamado usuarios al archivo /etc/passwd)



# Wildcards

---

Una característica del Shell que nos permite agrupar nombres de archivos y directorios:

Wildcard	Meaning
*	Matches any characters
?	Matches any single character
[ <i>characters</i> ]	Matches any character that is a member of the set <i>characters</i>
[! <i>characters</i> ]	Matches any character that is not a member of the set <i>characters</i>
[[: <i>class</i> :]]	Matches any character that is a member of the specified <i>class</i>

# Wildcards

---

Pueden ser usadas con cualquier comando del shell, ejemplo:

- Listar todos los archivos que empiezan con b: `ls b*`
- Mostrar el contenido de todos los archivos que terminan con un número: `cat *[0-9]`

<code>Data???</code>	Any file beginning with “Data” followed by exactly three characters
<code>[abc]*</code>	Any file beginning with either an “a”, a “b”, or a “c”
<code>BACKUP.[0-9][0-9][0-9]</code>	Any file beginning with “BACKUP.” followed by exactly three numerals
<code>[[:upper:]]*</code>	Any file beginning with an uppercase letter
<code>[![:digit:]]*</code>	Any file not beginning with a numeral
<code>*[[:lower:]]123</code>	Any file ending with a lowercase letter or the numerals “1”, “2”, or “3”

# Comandos de manipulación de archivos y directorios

---

`cp` : copiar archivos

`mv`: mover archivos, cambiar de nombre a archivos

`rm`: borrar archivos

`ln`: crear enlaces simbólicos

`mkdir`: Crear un directorio

Ejemplos:

- Cambiar de nombre al archivo `hola.txt` a `hello.txt`: `mv hola.txt hello.txt`
- Mover todos los archivos que terminan en `.txt` al directorio `texto`: `mv *.txt ./texto`

# Tarea: Uso básico del shell

---

Tarea se encuentra en la sección Programa Analítico

Referencia: Capítulos 1 al 4 del libro The Linux Command Line

Para la próxima clase:

- Lectura Capítulos 5,6 y 9 del libro The Linux Command Line