

# Procesos

The screenshot displays a Linux desktop environment. On the left, the 'Actividad del sistema' window is open, showing a list of running processes. A context menu is visible over the 'plasma-desktop' process, with options like 'Terminar proceso...', 'Búsqueda rápida', and 'Todos los procesos, Árbol'. The process list includes columns for PID, Name, User, CPU %, Memory, Shared Memory, and Title. The 'plasma-desktop' process is highlighted with 1% CPU usage and 100.5 M of memory.

No	Termina el proceso seleccionado. Tenga cuidado, pues perderá el trabajo que no haya guardado.	sys:	cpu %	Memoria	Memoria compartida	Título de la vent
		desktop	1%	100,5 M	78,8 M	
	Haga clic derecho sobre un proceso para enviarle otras señales.	sergio	1%	14,2 M	16,4 M	
	Consulte «Qué es esto» para obtener información técnica.	sergio		12,7 M	25,1 M	Actividad del ...
	Para señalar una ventana concreta a matar, presione Ctrl+Alt+Esc en cualquier momento.	sergio		9,1 M	15,9 M	
		sergio		7,9 M	21,3 M	
		sergio		7,3 M	15,5 M	
		sergio		5,3 M	14,6 M	
		sergio		5,2 M	13,5 M	
		sergio		4,8 M	2,8 M	
		sergio		4,0 M	9,5 M	
		root		3,4 M	1,0 M	
		sergio		3,2 M	9,9 M	
		sergio		3,2 M	9,1 M	
		sergio		3,1 M	11,8 M	
		sergio		3,0 M	9,6 M	
		sergio		3,0 M	9,1 M	
		root		3,0 M	3,7 M	
		sergio		3,0 M	11,2 M	
		sergio		2,8 M	11,8 M	
		sergio		2,7 M	11,1 M	
		sergio		2,4 M	8,2 M	
		root		2,0 M	2,7 M	
		sergio		1,9 M	888 K	
		root		1,6 M	616 K	
		sergio		1,5 M	4,8 M	
		sergio	2%	1,5 M	3,0 M	
		sergio		1,5 M	4,3 M	
		dbus-daemon		1,4 M	1,3 M	
		root		1,3 M	2,7 M	
		smmsp		1,3 M	616 K	
		root		1,3 M	5,2 M	
		sergio		1,0 M	2,9 M	
		root		0,9 M	2,3 M	
		sergio		872 K	2,1 M	
		root		836 K	2,8 M	
		root		820 K	2,7 M	

On the right, a system monitoring dashboard provides real-time data for various components:

- CPU:** Intel(R) Core(TM) i3-2310M CPU @ 2.10GHz. CPU temperature: 48°C. MB temperature: 48°C. Core 0: 0.11%, Core 1: 9%, Core 2: 8%, Core 3: 6%.
- GPU:** Intel Corporation Device 0116 (rev 09). GPU @ 300 MHz, GPU Temp: 48°C. RAM @ 1000 MHz, MB. OpenGL Mesa DRI Intel(R) Sandybridge.
- Memory:** 1189 of 3906 Mb (Ram). 0 of 8000 Mb (Swap).
- Disk usage:** 30444 of 98429 Mb (Root). 37420 of 282221 Mb (home). 0 of 0 Mb (storage). 0 of 0 Mb (/usr). 0 of 0 Mb (/var).
- Networking:** External IP: 192.168.1.100. Hostname: sebelkjenovo. Total upload: 0.0 KB/s. Total download: 0.0 KB/s.
- Hardware Sensors:** CPU Fan: 1200 RPM, System Fan: 1200 RPM, Case Fan: 1200 RPM, Aux. Fan: 1200 RPM. Voltage: VCore: +12.0V, +5.0V; Aux: +3.3V.
- Tasks:** 209. COMM USER %CPU ALL. Running: 2. Load: 12%. System: 4%. User: 7%.

A 'Temperatura' widget is also visible in the bottom right corner, showing temperature gauges for various components.

# Sacar fotos de procesos

- Mostrar procesos que corren solamente en la terminal  
*ps*
- Mostrar procesos que corren en otras terminales  
*ps a*
- Mostrar procesos que corren en terminales y aquellos que no están conectados a ninguna terminal  
*ps ax*
- Id. anterior, mostrando más información (notar que se muestra el usuario)  
*ps aux*

# Estilos de uso de ps

- BSD: Las opciones no tienen guiones  
*ps ax o comm,ni,pid*
- GNU: Las opciones son largas expresadas con doble guión  
*ps --user sergio --headers*
- UNIX: Las opciones son cortas con precedidas por un guión  
*ps -ax -o comm,ni,pid*
- En muchos casos estos estilos se pueden combinar  
*ps -F --tty /dev/tty3*
- Notar que 'ps a' y 'ps -a' no es lo mismo. El estilo BSD muestra además, los procesos que crean sesiones

# Mostrar en tiempo real

- *top*
- El comando `top` muestra los procesos que usan más el procesador
- Durante la ejecución de `top` se puede
  - Matar un proceso (letra `k`)
  - Cambiar la prioridad de un proceso (letra `r`)

# Estado de los procesos

- S: Sleeping; el proceso duerme esperando un evento
- R: Running, el proceso se ejecuta o es ejecutable
- T: Detenido
- D: el proceso está durmiendo y no se puede interrumpir (por ejemplo, escribiendo en disco)
- Z: Zombie; el proceso terminó (por lo tanto no tiene sentido matar un proceso zombie), pero el proceso padre no esperó a que finalizara

# Estado de los procesos BSD

- <: El proceso corre con nice negativo
- s: Es un generador de sesiones
- +: El proceso corre en primer plano
- N: El proceso corre con un nice positivo
- L: El proceso tiene páginas de memoria bloqueadas
- l: El proceso es multihilo

# Jobs

- Ejecutar un comando en segundo plano  
*comando &*
- Listar jobs  
*jobs*
- Llamar al job 1 en segundo plano  
*bg 1*
- Llamar al job 1 en primer plano  
*fg 1*

# Señales

- Terminar el proceso con PID 1234  
*kill -TERM 1234*
- Matar el proceso con PID 1234  
*kill -KILL 1234*
- Interrumpir el proceso con PID 1234  
*kill -INT 1234*
- Detener el proceso con PID 1234  
*kill -STOP 1234*

# Señal HangUp

- Generalmente se usa para aplicar un cambio en la configuración de un servicio sin detenerlo
- *kill -HUP 1234*
- El comando de arriba es en general equivalente a:  
*/etc/init.d/servicio reload*
- Hacer que el proceso de un comando siga corriendo aun cuando se cierre la terminal  
*nohup comando &*

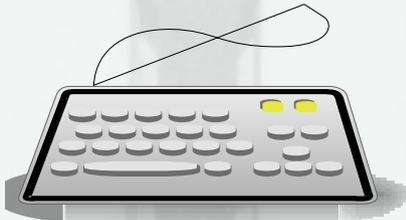
# Prioridad de procesos

- nice: Escala de “amabilidad” del proceso. Los procesos más amables consumen menos el procesador
- El valor máximo de nice es 19
- El valor mínimo de nice es -20
- Solamente root puede correr procesos con nice negativas
- Solamente root puede disminuir el nivel de “amabilidad” de un proceso
- Un proceso con nice negativa significa que tendrá mayor prioridad y viceversa

# nice y renice

- Si no se especifica el comando nice asignará nice = 10
- Correr comando con nice -5
  - *nice -n -5 comando*
- Correr comando con nice 19
  - *nice -n 19 comando*
- Cambiar el nice a -5 del proceso con pid 1234
  - *renice -5 1234*
- Cambiar el nice a 19 del proceso con pid 1234
  - *renice 19 1234*

Recorrido de un proceso



Entrada Estándar  
Descriptor de archivo: **0**



Salida Estándar  
Descriptor de archivo: **1**



Error Estándar  
Descriptor de archivo: **2**

Redireccionar la entrada



Entrada Estándar  
Descriptor de archivo: **0**



Salida Estándar  
Descriptor de archivo: **1**



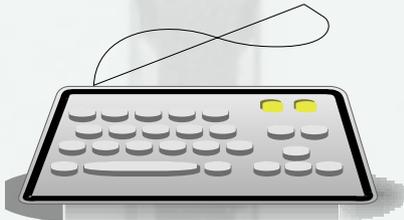
Error Estándar  
Descriptor de archivo: **2**

```
case busco in [a-z]*)echo "Letras";; [0-9])echo "Numeros";; *) \  
echo otro;; esac < /sys/devices/pci0000\:00/0000\:00\:00.0/enable
```

Redireccionar la salida

file

Salida Estándar  
Descriptor de archivo: **1**



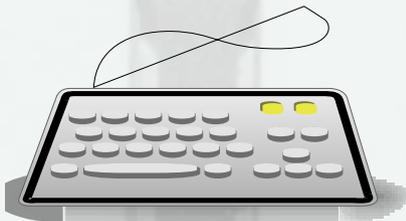
Entrada Estándar  
Descriptor de archivo: **0**



Error Estándar  
Descriptor de archivo: **2**

```
find / -name "k*" 1> /tmp/salida
```

Redireccionar el error estándar



Entrada Estándar  
Descriptor de archivo: **0**



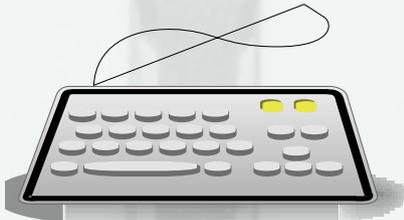
Salida Estándar  
Descriptor de archivo: **1**



Salida Estándar  
Descriptor de archivo: **2**

```
find / -name "k*" 2> /tmp/error
```

Redireccionar la salida estándar y el error estándar por separado



Entrada Estándar  
Descriptor de archivo: **0**

file

Salida Estándar  
Descriptor de archivo: **1**

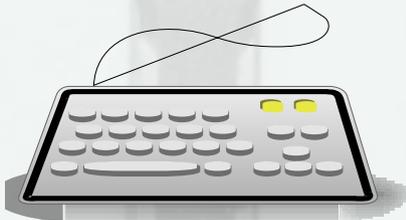
file

Salida Estándar  
Descriptor de archivo: **2**

```
find / -name "k*" 1> /tmp/salida 2> /tmp/error
```

Redireccionar la salida estándar y el error estándar al mismo archivo

file



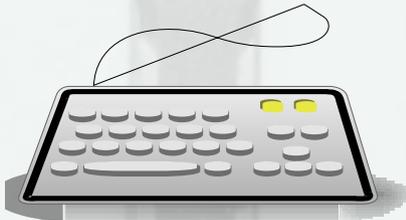
Salida Estándar  
Descriptor de archivo: **1**

Entrada Estándar  
Descriptor de archivo: **0**

Salida Estándar  
Descriptor de archivo: **2**

```
find / -name "k*" 1> /tmp/salida_y_error 2>&1
```

**IMPORTANTE:** el orden en que se escribe el comando es importante: el ejemplo siguiente, es parecido pero el efecto es totalmente distinto



Entrada Estándar  
Descriptor de archivo: **0**



Salida Estándar  
Descriptor de archivo: **1**



Los errores se muestran en pantalla y la salida termina en el archivo

Error Estándar  
Descriptor de archivo: **2**

```
find / -name "k*" 2>&1 1> /tmp/salida_y_error
```

Tuberías

Proceso 1

Proceso 2

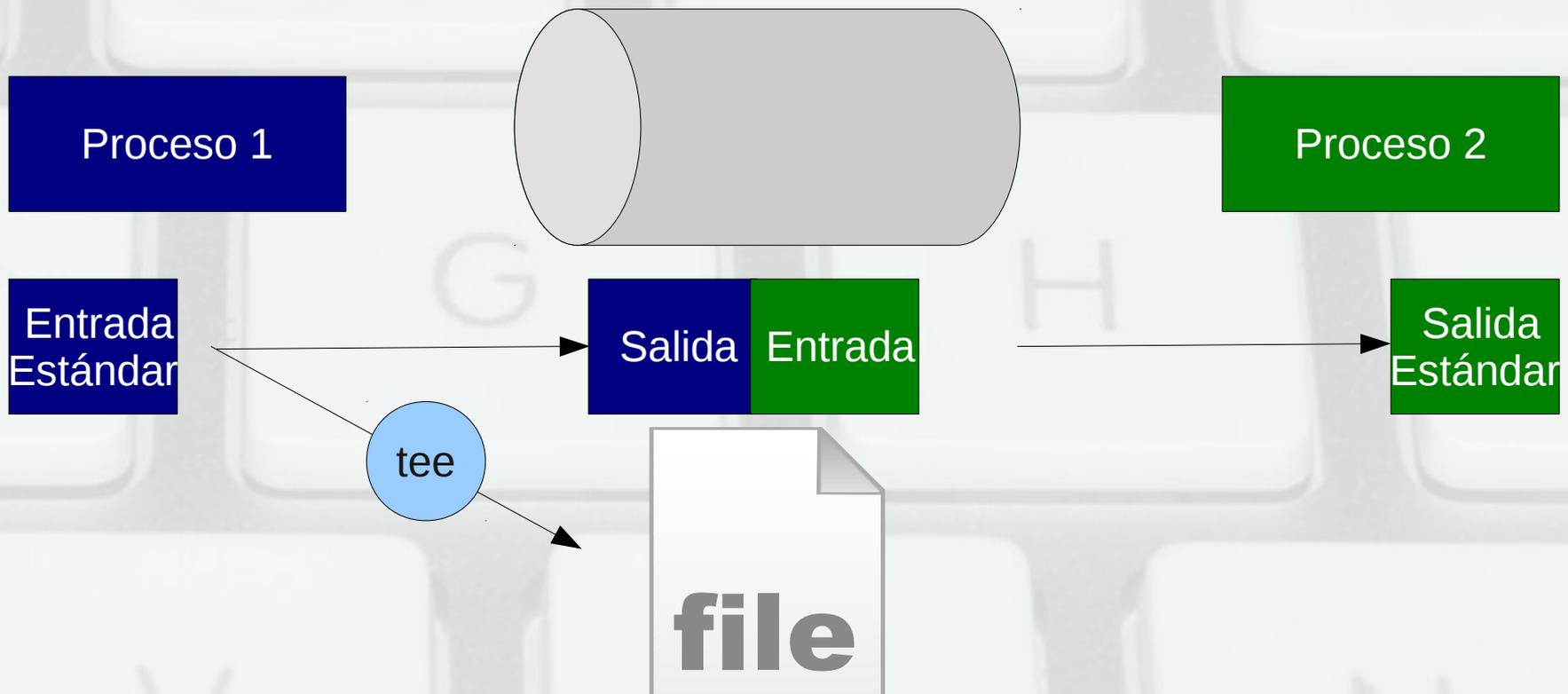
Entrada  
Estándar

Salida    Entrada

Salida  
Estándar

```
grep 'sh$' /etc/bash | wc -l
```

Comando tee: Bifurcación



```
grep 'sh$' /etc/passwd | tee /tmp/tienenshell | wc -l
```

Comando xargs: Convertir la salida estándar en argumentos para un comando



```
find / -name *.iso | xargs md5sum
```

# Recordar

- Para la shell bash un proceso termina un error cuando su estado de salida es distinto de 0.
- '*comando > archivo*' es equivalente a '*comando 1> archivo*'
- “>” sobrescribe, “>>” agrega al final
- Se pueden acumular sucesivas tuberías
- Las 'pipes' entuban solamente la salida de un proceso. (Existen trucos para entubar el error)

# Bibliografía y recursos

- Páginas del manual de todos los comandos vistos
- Página del manual de wait
- Página del manual de proc
- [Illustrated Redirection Tutorial](#)