



UT5: Configuración de Sistemas Operativos

Gestión de Recursos del Sistema

Sistemas Informáticos

Ciclo Formativo de Grado Superior en Desarrollo de Aplicaciones Web

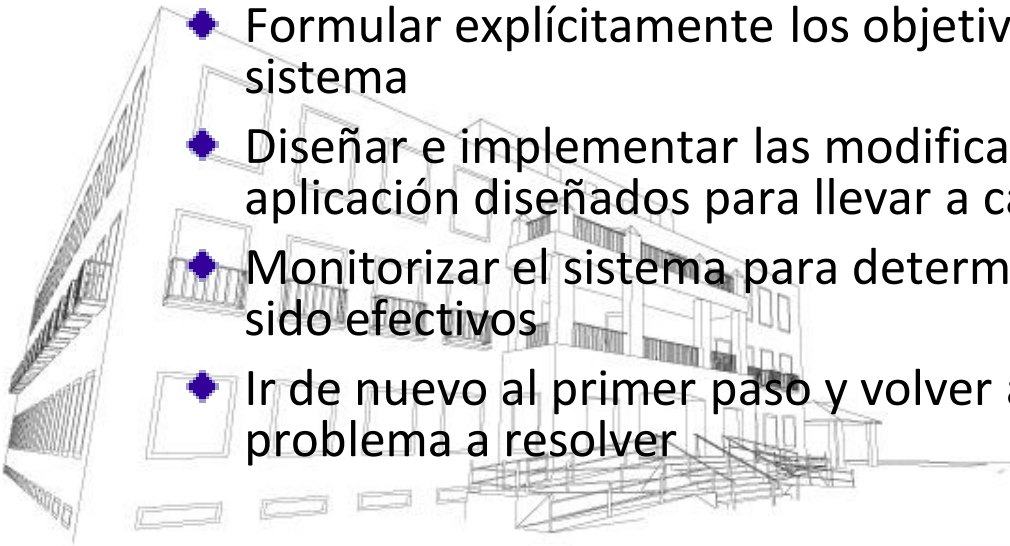
Índice

- ◆ Introducción
- ◆ Gestión de Recursos del Sistema en Linux
- ◆ Gestión de Recursos del Sistema en Windows



Introducción

- ◆ Es muy importante tener información esencial del rendimiento del sistema: procesos en ejecución, cantidad de memoria disponible, nº de particiones, etc.
- ◆ Al determinar el rendimiento del sistema se debe:
 - ◆ Definir el problema con todo el detalle que sea posible
 - ◆ Determinar la causa o causas del problema
 - ◆ Formular explícitamente los objetivos para mejorar el rendimiento del sistema
 - ◆ Diseñar e implementar las modificaciones al sistema y/o programas de aplicación diseñados para llevar a cabo esos objetivos
 - ◆ Monitorizar el sistema para determinar si los cambios realizados han sido efectivos
 - ◆ Ir de nuevo al primer paso y volver a empezar, habrá un nuevo problema a resolver

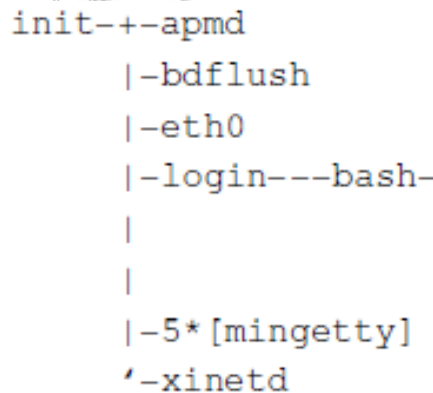


Linux: Control y gestión de la actividad de la CPU

- ◆ **uptime** → hora actual, cuánto tiempo lleva en marcha el sistema, número de usuarios conectados, y carga media del sistema en los últimos 1, 5 y 15 minutos
 - ◆ Valores altos implican que el sistema se está usando mucho
 - ◆ Valores bajos no significan que el tiempo de respuesta vaya a ser bajo

[pilar@ditec-por1 TEMA5]\$ uptime 22:58:47 up 22min, 12 users, load average: 0.24, 0.41, 0.33

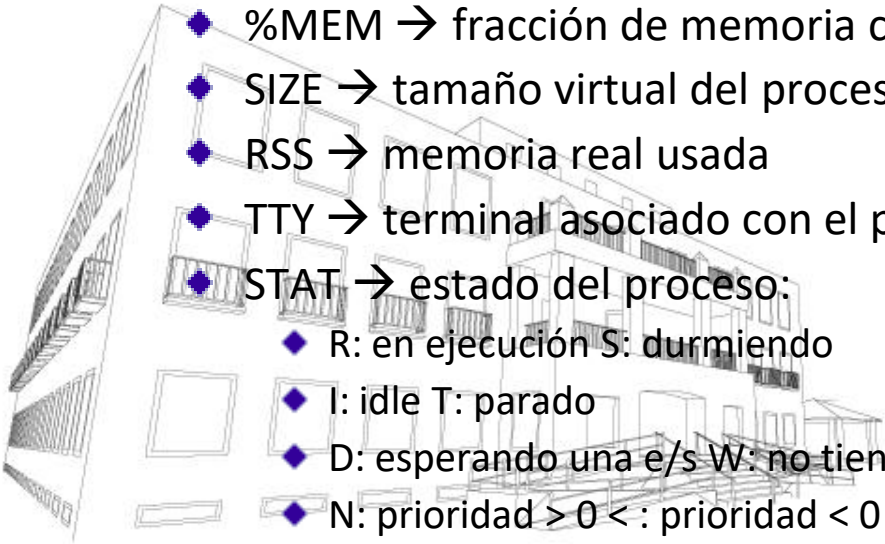
- ◆ **ps tree** → visualiza un árbol de los procesos en ejecución



```
init--+-apmd
      |-bdflush
      |-eth0
      |-login---bash---startx---xinit--+-X
      |                                   '-startkde--+-kwrapper
      |                                   '-ssh-agent
      |-5*[mingetty]
      '-xinetd
```

Linux: Control y gestión de la actividad de la CPU (ii)

- ◆ **ps** → información sobre los procesos en ejecución
 - ◆ USER → usuario que lanzó el programa
 - ◆ PID → identificador del proceso
 - ◆ PPID → identificador del proceso padre
 - ◆ %CPU → porcentaje entre el tiempo usado realmente y el tiempo que lleva en ejecución
 - ◆ %MEM → fracción de memoria consumida (es una estimación)
 - ◆ SIZE → tamaño virtual del proceso: código+datos+pila
 - ◆ RSS → memoria real usada
 - ◆ TTY → terminal asociado con el proceso
 - ◆ STAT → estado del proceso:
 - ◆ R: en ejecución S: durmiendo
 - ◆ I: idle T: parado
 - ◆ D: esperando una e/s W: no tiene páginas residentes
 - ◆ N: prioridad > 0 < : prioridad < 0



Linux: Control y gestión de la actividad de la CPU (iii)

◆ ps aux

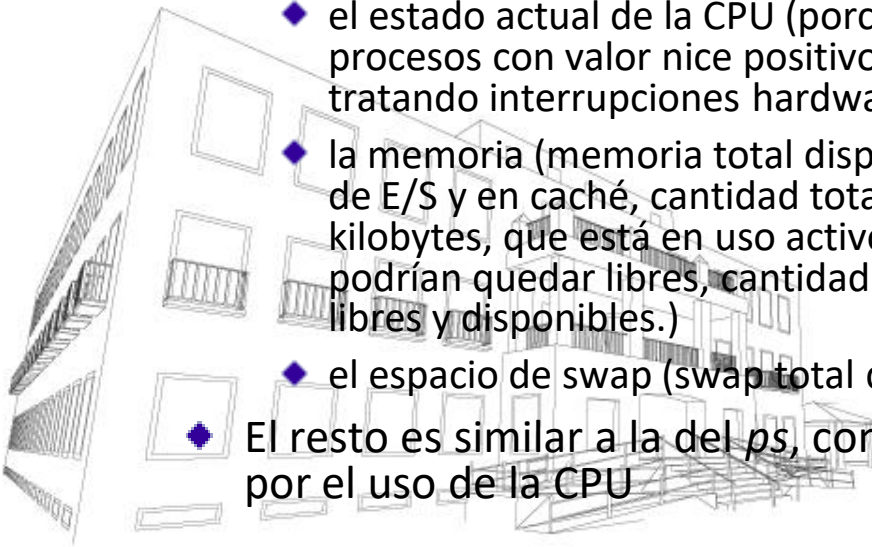
USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.1	0.0	1368	60	?	S	22:36	0:04	init
root	9	0.0	0.0	0	0	?	SW	22:36	0:00	[bdf flush]
root	10	0.0	0.0	0	0	?	SW	22:36	0:00	[kupdated]
root	15	0.0	0.0	0	0	?	SW	22:36	0:00	[kjournald]
root	429	0.0	0.0	1440	76	?	S	22:37	0:00	syslogd -m 0
lp	603	0.0	0.0	6264	32	?	S	22:37	0:00	[lpd]
root	965	0.0	0.0	1352	4	tty4	S	22:37	0:00	/sbin/mingett
pilar	972	0.0	0.0	4580	4	tty1	S	22:37	0:00	-bash
pilar	1200	0.0	0.2	4580	252	pts/5	S	22:38	0:00	/bin/bash
pilar	1726	0.0	0.7	4072	832	pts/5	S	23:26	0:00	ps aux

◆ ps al

F	UID	PID	PPID	PRI	NI	VSZ	RSS	WCHAN	STAT	TTY	TIME	COMMAND
0	500	1065	1013	15	0	4528	1068	wait4	S	pts/6	0:00	/bin/bash
0	500	2800	1065	21	0	3268	1404	-	R	pts/6	0:00	ps la

Linux: Control y gestión de la actividad de la CPU (iv)

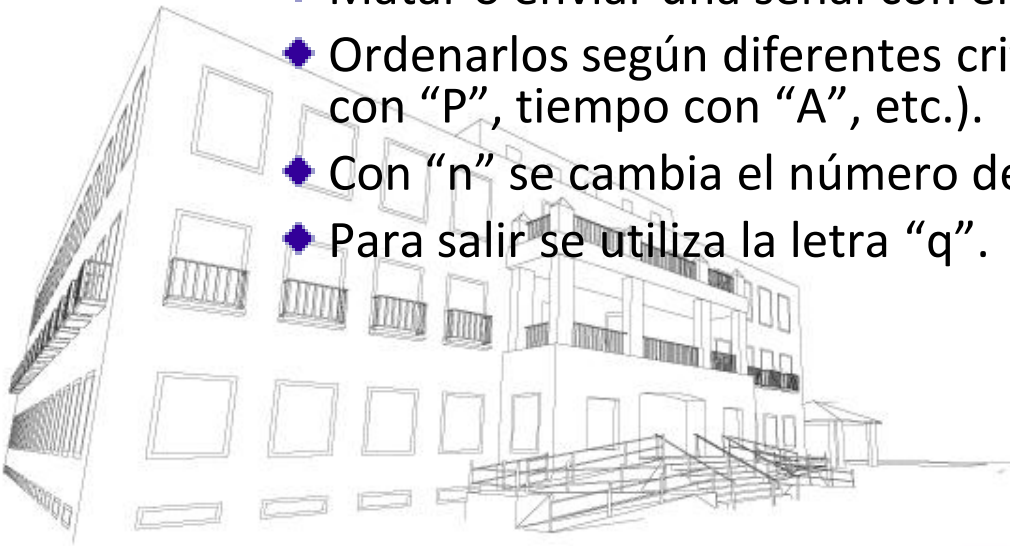
- ◆ **top** → proporciona una visión continuada de la actividad del procesador en tiempo real, muestra las tareas que más uso hacen de la CPU, y tiene una interfaz interactiva para manipular procesos
 - ◆ Las cinco primeras líneas muestran información general del sistema:
 - ◆ las estadísticas del comando *uptime*,
 - ◆ estadísticas sobre los procesos del sistema (número de procesos, procesos durmiendo, procesos ejecutándose, procesos zombies y procesos parados),
 - ◆ el estado actual de la CPU (porcentaje en uso por usuarios, por el sistema, por procesos con valor nice positivo, por procesos esperando E/S, desocupada, y tratando interrupciones hardware o software),
 - ◆ la memoria (memoria total disponible, usada, libre, compartida, usada como *buffer* de E/S y en caché, cantidad total de *buffer* o memoria caché de página, en kilobytes, que está en uso activo, cantidad total de *buffer* y páginas de la caché que podrían quedar libres, cantidad total de *buffer* o páginas de la caché que están libres y disponibles.)
 - ◆ el espacio de swap (swap total disponible, usada y libre).
 - ◆ El resto es similar a la del *ps*, con los procesos ordenados decrecientemente por el uso de la CPU



Linux: Control y gestión de la actividad de la CPU (v)

◆ **top**, continúa. . .

- ◆ La lista es actualizada de forma interactiva, y además se permite realizar una serie de tareas sobre los procesos, como por ejemplo:
 - ◆ Cambiar la prioridad de alguno utilizando el comando “r”.
 - ◆ Matar o enviar una señal con el comando “k”.
 - ◆ Ordenarlos según diferentes criterios (por PID con “N”, uso de CPU con “P”, tiempo con “A”, etc.).
 - ◆ Con “n” se cambia el número de procesos que se muestran.
 - ◆ Para salir se utiliza la letra “q”.



top

```
16:41:50 up 5:46, 18 users, load average: 0,14, 0,14, 0,10
90 processes: 88 sleeping, 2 running, 0 zombie, 0 stopped
CPU states: 1,4% user 0,9% system 0,0% nice 0,0% iowait 97,5% id
Mem: 643136k av, 629060k used, 14076k free, 0k shrd, 31068k
      452436k actv, 0k in_d, 11520k in_c
Swap: 265064k av, 14316k used, 250748k free 481472k cached
```

PID	USER	PRI	NI	SIZE	RSS	SHARE	STAT	%CPU	%MEM	TIME	CPU	COMMAND
891	root	15	0	35896	6508	836	S	1,4	1,0	4:36	0	X
1013	pilar	15	0	6640	5976	3136	R	0,4	0,9	0:23	0	kdeinit
2839	pilar	15	0	1024	1024	800	R	0,4	0,1	0:00	0	top
1	root	15	0	108	76	56	S	0,0	0,0	0:04	0	init
2	root	15	0	0	0	0	SW	0,0	0,0	0:00	0	keventd
3	root	15	0	0	0	0	SW	0,0	0,0	0:00	0	kapmd
9	root	25	0	0	0	0	SW	0,0	0,0	0:00	0	bdf flush
5	root	15	0	0	0	0	SW	0,0	0,0	0:00	0	kswapd
10	root	15	0	0	0	0	SW	0,0	0,0	0:00	0	kupdate
562	root	15	0	196	156	112	S	0,0	0,0	0:00	0	syslogd
686	root	15	0	620	520	380	S	0,0	0,0	0:00	0	sshd

Linux: Consumo de Memoria

- ◆ **vmstat** → Información sobre la memoria virtual (tb de procesos)

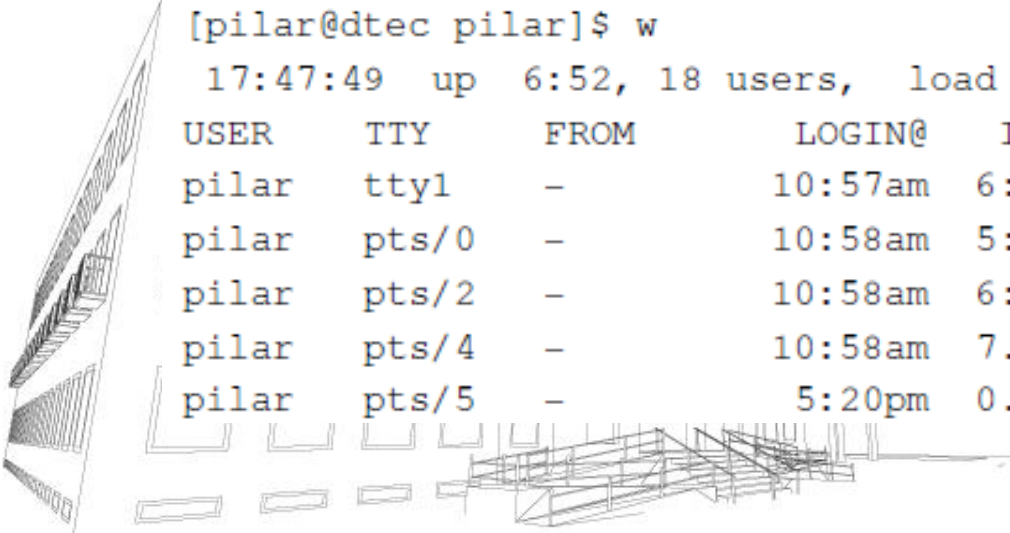
- ◆ r: número de procesos esperando su tiempo de ejecución
- ◆ b: número de procesos en espera ininterrumpible
- ◆ w: número de procesos en espacio de intercambio
- ◆ us: tiempo de usuario como porcentaje de tiempo total
- ◆ sy: tiempo de sistema como porcentaje de tiempo total
- ◆ id: tiempo de inactividad como porcentaje de tiempo total

[pilar@ditec pilar]\$ vmstat 2

procs			memory			swap		io		system		cpu			
r	b	w	swpd	free	buff	cache	si	so	bi	bo	in	cs	us	sy	id
2	0	0	18596	6784	36408	467096	0	1	67	15	161	258	4	1	94
1	0	0	18596	7808	36420	465836	0	0	0	604	269	1113	96	4	0
1	0	0	18604	7748	36416	466476	0	4	0	4	197	971	94	6	0
2	0	0	18604	6876	36432	467072	0	0	0	634	190	657	95	2	3
2	0	0	18612	7968	36420	466260	0	4	0	4	178	1053	96	4	0
2	0	1	18612	7108	36424	466864	0	0	0	530	232	955	97	3	0
4	0	0	18612	9728	36436	466252	0	0	0	130	211	991	91	9	0
3	0	0	18612	8160	36440	465960	0	0	0	0	222	1497	94	6	0

Linux: Control de Usuarios

- ◆ **w** → quién está conectado y qué está haciendo
 - ◆ JCPU: tiempo usado por todos los procesos asociados a ese terminal (incluye los procesos en background actuales, pero no los pasados).
 - ◆ PCPU: tiempo de CPU usado por el proceso actual



```
[pilar@dtec pilar]$ w
```

```
17:47:49 up 6:52, 18 users, load average: 0.02, 0.09, 0.11
```

USER	TTY	FROM	LOGIN@	IDLE	JCPU	PCPU	WHAT
pilar	tty1	-	10:57am	6:50m	0.12s	0.04s	-bash
pilar	pts/0	-	10:58am	5:36	0.55s	0.01s	/bin/bash
pilar	pts/2	-	10:58am	6:49m	0.05s	0.05s	/bin/bash
pilar	pts/4	-	10:58am	7.00s	9.57s	3.68s	vim tema5.te
pilar	pts/5	-	5:20pm	0.00s	0.08s	0.02s	w

Linux: Recursos disponibles

- ◆ **ulimit** → proporciona un control sobre los recursos disponibles para una shell y los procesos lanzados por esa shell

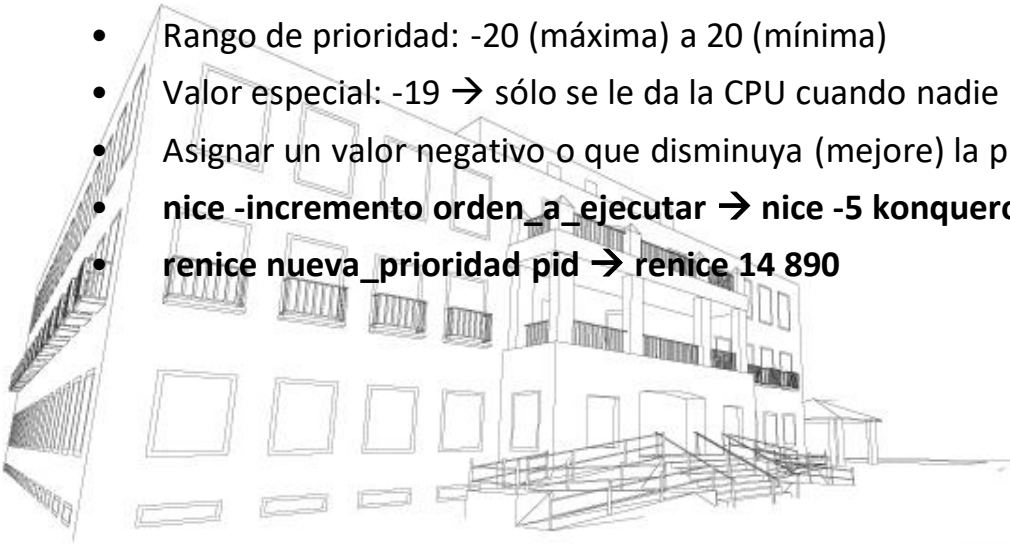
```
[pilar@momo TEMA5]$ ulimit -a
core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
file size               (blocks, -f) unlimited
max locked memory       (kbytes, -l) unlimited
max memory size         (kbytes, -m) unlimited
open files              (-n) 1024
pipe size               (512 bytes, -p) 8
stack size              (kbytes, -s) unlimited
cpu time                (seconds, -t) unlimited
max user processes      (-u) 5120
virtual memory          (kbytes, -v) unlimited
```



Linux: Planificación de procesos

Número nice y prioridad de procesos

- Linux realiza una planificación por prioridades dinámicas
- Al lanzar un proceso se le asigna un valor de prioridad **número nice**, por defecto hereda la del proceso padre
- La **prioridad dinámica** del proceso se calcula en función del número nice, junto con el consumo de CPU realizado
- Valores bajos (negativos) → más prioridad
- Valores altos (positivos) → menos prioridad
- Rango de prioridad: -20 (máxima) a 20 (mínima)
- Valor especial: -19 → sólo se le da la CPU cuando nadie más la quiera
- Asignar un valor negativo o que disminuya (mejore) la prioridad del proceso sólo puede hacerlo el **root**
- **nice -incremento orden_a_ejecutar → nice -5 konqueror**
- **renice nueva_prioridad pid → renice 14 890**



Linux: Planificación de procesos (ii)

Señales

- ◆ En ocasiones es necesario enviar señales a los procesos: pararlos (SIGSTOP-19), eliminarlos, que continúen (SIGCONT-18), etc.
- ◆ **kill [-señal] pid's** → enviar una señal
- ◆ **kill pid's** → se le dice al proceso que termine, de forma correcta, puede ser capturada, envía la señal SIGTERM (15)
- ◆ La señal SIGKILL (9), no puede ser capturada y es seguro que el proceso finaliza
- ◆ **killall [-señal] orden** → enviar una señal a todos los procesos «orden», p.e.: killall -15 bash
- ◆ Hay procesos que no mueren a pesar de recibir la señal KILL:
 - ◆ Procesos zombies
 - ◆ Procesos que esperan un recurso vía NFS que no está disponible
 - ◆ Procesos que esperan una petición de E/S realizada a un dispositivo

La Configuración del SO en Windows

ADMINISTRANDO EL EQUIPO

Para realizar distintas tareas de administración se dispone de la utilidad **Administración de equipos** con la que se pueden realizar, entre otras, las siguientes operaciones:

- Monitorizar sucesos del sistema como la hora de inicio de sesión y los errores de programa (**Visor de sucesos o Visor de eventos**).
- Crear y administrar los recursos compartidos (**Carpetas compartidas**).
- Ver una lista de los usuarios conectados a un equipo local o remoto (**Carpetas compartidas**).
- Administrar usuarios y grupos del equipo (**Usuarios locales y grupos o Usuarios y grupos locales**).
- Configurar los contadores de rendimiento y las alarmas (**Registros y alertas de rendimiento o Rendimiento**).
- Ver la configuración de los dispositivos y agregar controladores de dispositivo nuevos (**Administrador de dispositivos**).
- Administrar los discos del equipo (**Administración de discos**).
- Iniciar y detener los servicios del sistema (**Servicios y aplicaciones**).

La Configuración del SO en Windows

EL VISOR DE SUCESOS

El **Visor de eventos** (**Visor de sucesos** en Windows XP) es la herramienta que permite examinar y administrar los eventos ocurridos en el equipo.

Un **evento** o **suceso** es un acontecimiento significativo del sistema o de una aplicación que requiere una notificación al usuario).

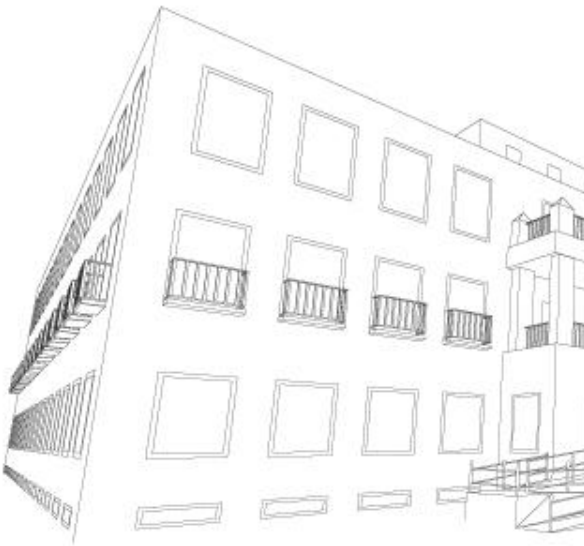
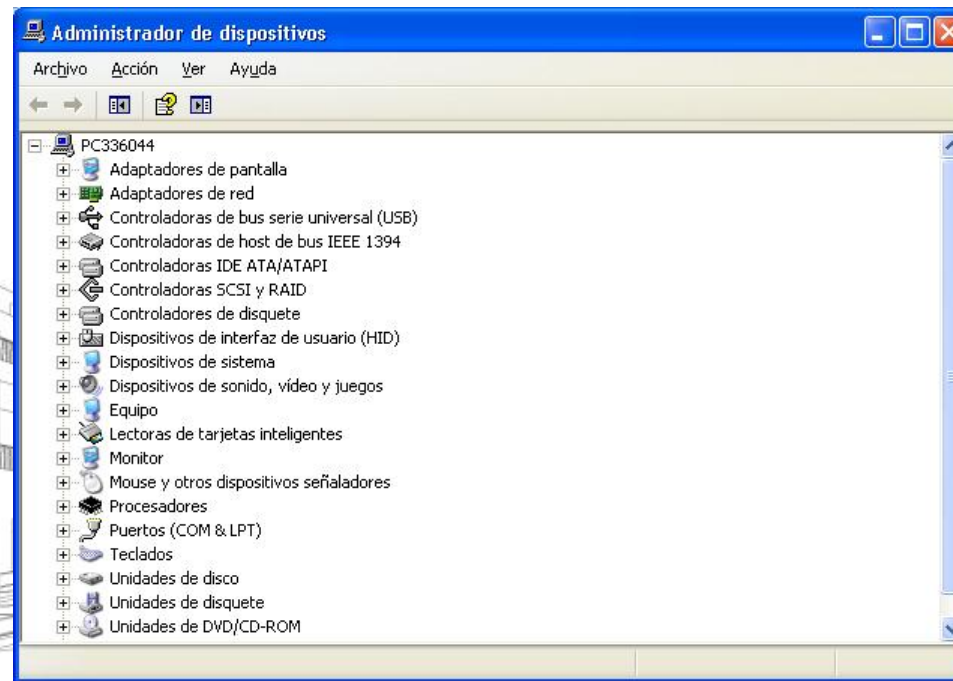
Puede mostrar los siguientes tipos de sucesos:

- **Crítico** (únicamente en Windows 7): corresponde a un error del que no puede recuperarse automáticamente la aplicación o el componente que desencadenó el evento.
- **Error**: corresponde a un problema importante que puede afectar a la funcionalidad externa a la aplicación o al componente que desencadenó el evento.
- **Advertencia**: corresponde a un evento que no es importante necesariamente pero que indica la posibilidad de problemas en el futuro.
- **Información**: corresponde a un evento que describe el funcionamiento correcto de una aplicación, un controlador o un servicio.
- **Auditoría correcta**: indica que se ha realizado correctamente el ejercicio de los derechos de un usuario.
- **Error de auditoría**: indica que se ha producido un error en el ejercicio de los derechos de un usuario.

La Configuración del SO en Windows

EL ADMINISTRADOR DE DISPOSITIVOS

Windows dispone de una utilidad que permite ver la configuración de los dispositivos instalados en el equipo y añadir o actualizar sus controladores.



La Configuración del SO en Windows

EL MONITOR DE RENDIMIENTO

El **monitor de rendimiento** es una herramienta gráfica que sirve para visualizar datos sobre el rendimiento, en tiempo real y desde archivos de registro. Entre sus posibilidades se encuentran:

- Reunir datos de rendimiento en tiempo real tanto del equipo local como de cualquier otro equipo de la red.
- Ver los datos reunidos (tanto los actuales como los anteriores) en un registro de contadores de rendimiento.
- Presentar los datos en un gráfico, en un histograma o en un informe.
- Exportar los datos a Word u otras aplicaciones de Microsoft Office.
- Crear páginas HTML a partir de las vistas de rendimiento.

La Configuración del SO en Windows

EL MONITOR DE RENDIMIENTO

