3. Sistemas operativos

Informática

Ingeniería en Electrónica y Automática Industrial

Raúl Durán Díaz Juan Ignacio Pérez Sanz

Departamento de Automática
Escuela Politécnica Superior

Curso académico 2016–2017

Contenidos

Objetivos

- Comprender el concepto de sistema operativo y su utilidad.
- Ser capaz de describir los diferentes elementos de que consta y comprender su importancia.
- Entender el concepto de *proceso* en el contexto de los sistemas operativos.
- Entender el concepto de *sistema de ficheros*.
- Ser capaz de utilizar los comandos básicos de LINUX, manejando la entrada/salida estándar y los comandos concatenados.

¿Qué es un *sistema operativo*?

Definición

Un **sistema operativo** es un programa (o conjunto de programas) de control que tiene por objeto *facilitar* el uso del computador y conseguir que este uso sea *eficiente*.

Misiones del *sistema operativo*

El sistema operativo proporciona dos servicios fundamentales:

- extiende las instrucciones de la máquina base;
- media en los conflictos entre procesos a causa de los recursos.

Máquina extendida u operativa

- El sistema operativo esconde detalles demasiado finos como para que el programador deba ocuparse de ellos.
  - Por ejemplo: cómo se maneja la cabeza de lectura/escritura de un disco magnético.
- Presenta al usuario-programador una «máquina virtual» que se maneja de modo mucho más sencillo.
- Esa «máquina virtual» está normalmente estandarizada, de manera que un mismo programa puede valer para sistemas operativos distintos si estos ofrecen al usuario-programador la misma interfaz estándar.

Máquina extendida u operativa

- La «máquina virtual» extiende el repertorio estándar de instrucciones por medio de los llamados *system calls* o servicios del sistema.
- Esos servicios del sistema pueden verse como «instrucciones extendidas» que permiten al programador ampliar sus posibilidades a la hora de programar.
- Los servicios del sistema constituyen la "interfaz" entre el programador y el sistema operativo.
- Los servicios del sistema se dividen en dos categorías esenciales:
  - los que manejan procesos;
  - los que manejan el sistema de ficheros.

Máquina extendida u operativa

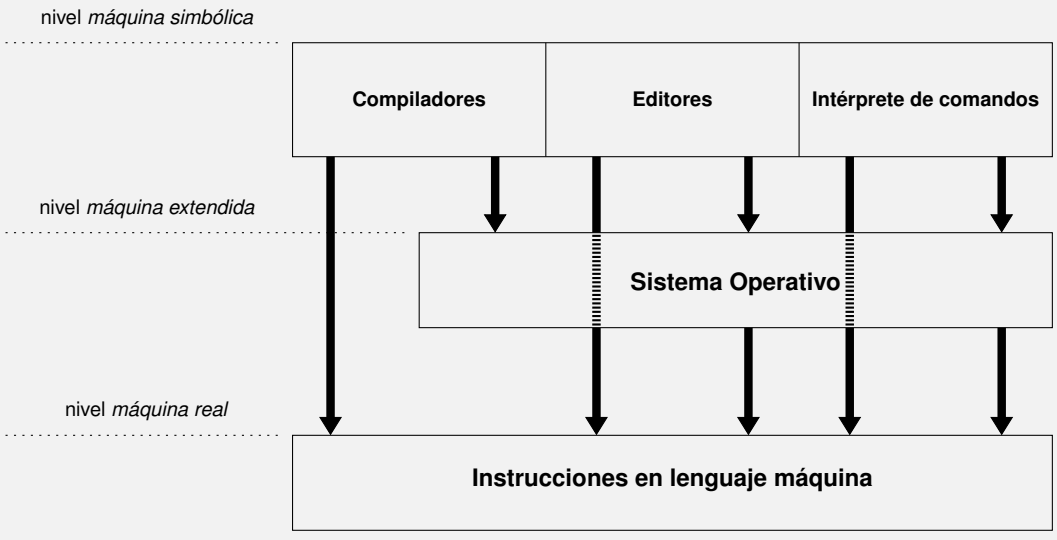


Figura: Posición relativa de las máquinas

Administrador de recursos

- La segunda misión importante del sistema operativo es actuar como árbitro entre los distintos usuarios para manejar los recursos limitados del sistema.

Ejemplo

Si varios programas quieren escribir en una impresora, hay que arbitrar los medios para que lo hagan en serie, uno detrás de otro, y que no se obstaculicen mutuamente.

Ejemplo

Mientras un proceso espera un dato del disco, otro puede usar la capacidad de cómputo del procesador.

Procesos

- El elemento ejecutivo fundamental de un sistema computador es el proceso.
- Un sistema computador en funcionamiento no es más que un conjunto de uno o más procesos en ejecución según diversas posibles planificaciones temporales.

¿Qué es un proceso?

Definición

Un *proceso* es, en esencia, un programa en ejecución.

Observación

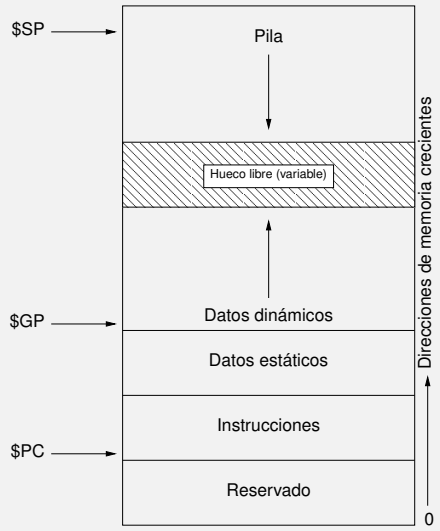
Podemos ver un proceso como la instanciación dinámica de un programa. Puede, pues, ocurrir que varios procesos ejecuten (es decir, instancien) el mismo programa.

Elementos de un proceso

- Un proceso consta de
  - un espacio de memoria, estructurado en secciones;
  - los registros del procesador.

Estructura de un proceso

Vista de un proceso en relación a la memoria virtual y registros:



- Un proceso necesita manejar registros de la CPU:
  - contador de programa `$PC`.
  - puntero de pila `$SP`.
  - puntero a datos dinámicos `$GP`.
  - registro de estado `$PSW`.

Características un proceso

- Un proceso solo puede ser creado por otro proceso.
- Cualquier proceso puede crear a su vez procesos.
- Se utilizan dos servicios del sistema para crear nuevos procesos:
  - `fork`: crea una réplica del proceso llamante;
  - `exec(p1)`: pone en ejecución el programa en el archivo ejecutable `p1`.

El servicio del sistema `exec` se ocupa de trasladar el archivo ejecutable a memoria, dándole la adecuada estructura a cada sección, para que pueda comenzar la ejecución en el contexto del nuevo proceso.

Estructura de un árbol de procesos

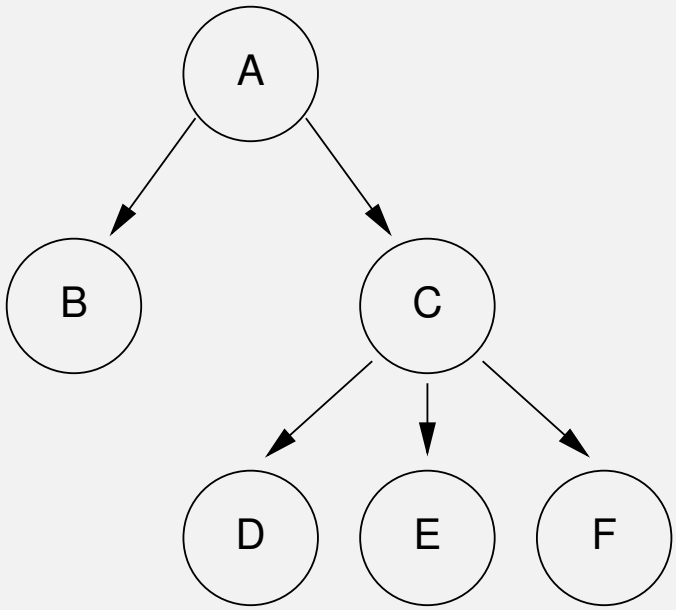


Figura: Vista de un árbol de procesos

Arranque (*boot*) del sistema computador

- En el momento de arranque de la máquina, hay un solo proceso en ejecución, responsable de ir creando todos los demás. Este proceso está grabado en la imagen de arranque (*boot image*).
- El arranque del sistema se ejecuta en varios pasos:
  - Se ejecuta un pequeño código almacenado en memoria no volátil que carga un cargador más potente y lo pone en ejecución.
  - Este segundo cargador carga la imagen del sistema operativo (*boot image*) y pone en ejecución el primer proceso de esa imagen.

Estructura de un fichero de imagen ejecutable

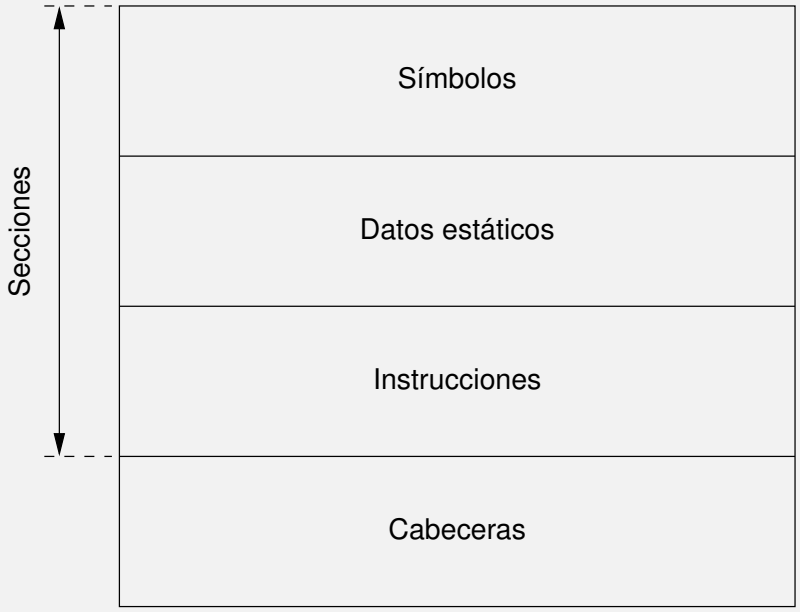


Figura: Estructura de un fichero ejecutable

Turno en la ejecución de los procesos

- El recurso más valioso es el procesador.
- El procesador solo puede ejecutar un proceso cada vez.
- Cuando el proceso en ejecución se bloquea (porque espera un dato del disco, por ejemplo) el proceso *distribuidor* suspende el proceso en curso y selecciona otro que tenga posibilidad de ejecutarse en ese momento de acuerdo con cierta *planificación*.
- Cada proceso se describe por el espacio de memoria que usa y por el valor de los registros del procesador en el momento de la suspensión.

Planificación: modo monoprograma

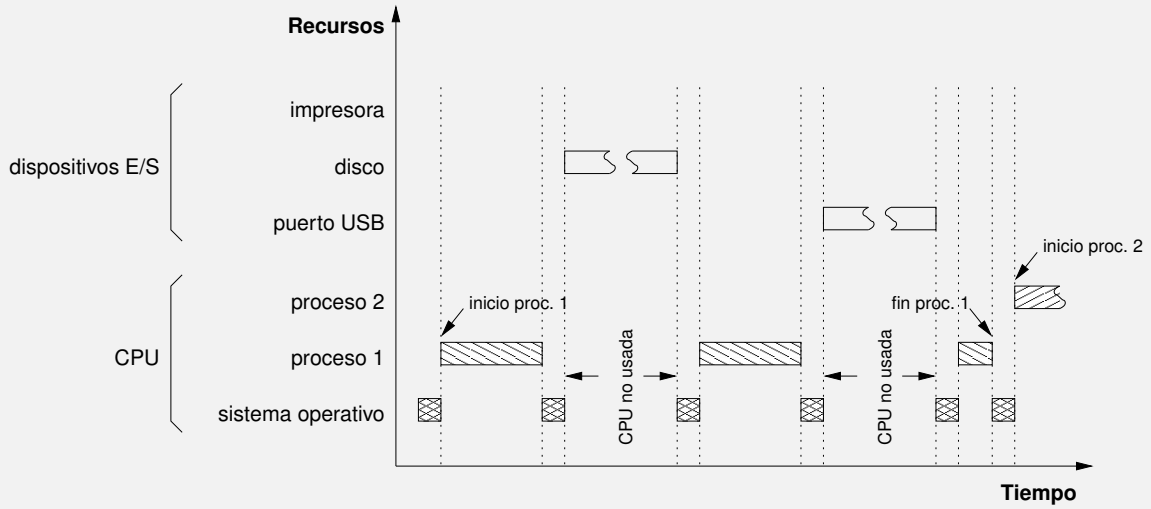


Figura: Planificación en modo monoprogramación

Planificación: modo multiprograma

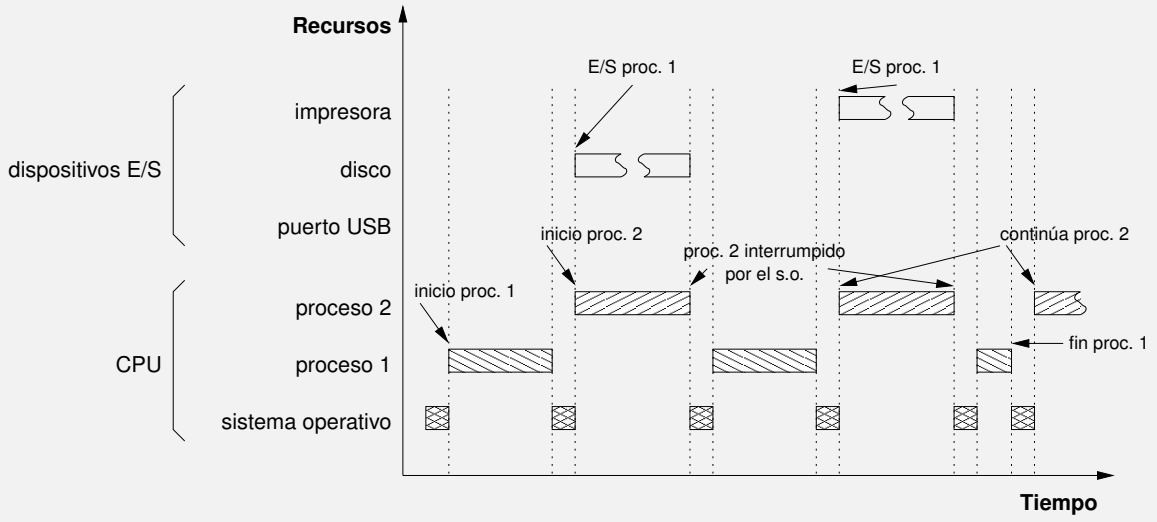


Figura: Planificación en modo multiprogramación

Planificación: modo turno rotatorio

- El modo multiprograma tiene el inconveniente de que si un proceso tiene poca E/S, monopoliza demasiado tiempo el procesador.

Solución: multiprogramación apropiativa

El *distribuidor* del sistema operativo puede interrumpir un proceso en ejecución, *apropiándose* del procesador y dando turno a otro proceso que esté listo para ejecutarse.

Planificación: ciclo de vida de un proceso

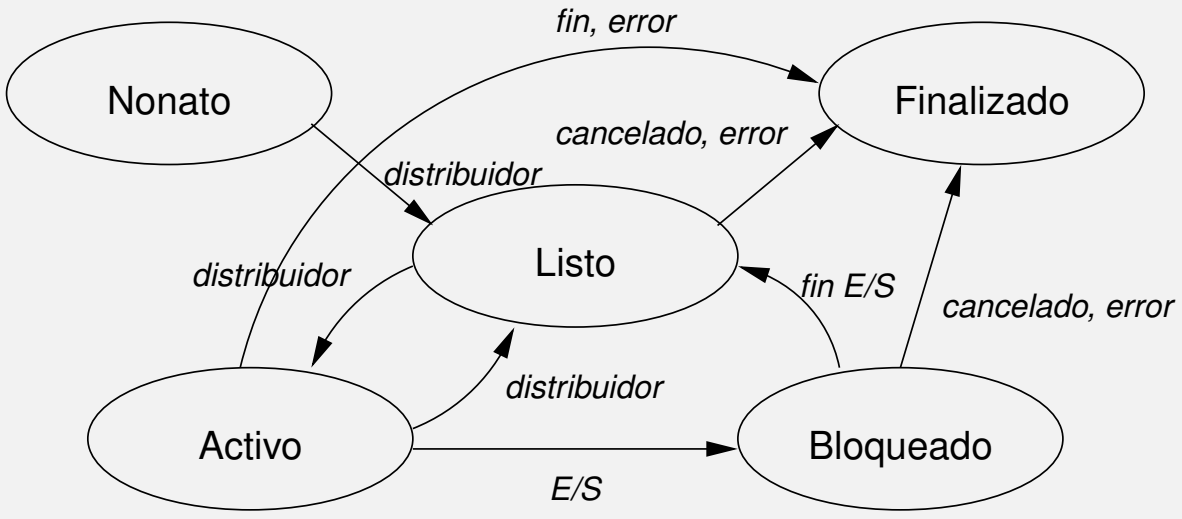


Figura: Diagrama de estados para un proceso

Memoria virtual

- El sistema operativo ofrece al proceso la ilusión de que puede utilizar toda la memoria disponible. Esto se denomina *memoria virtual*.
- En realidad, la memoria física que ocupa un proceso puede ser mucho más pequeña.
- El sistema operativo se encarga de mantener una tablas específicas para cada proceso mediante las cuales realiza la traducción de la dirección de memoria virtual a la dirección física.
- Así pueden convivir simultáneamente en un sistema más procesos de los que realmente caben en la memoria física de que ese sistema dispone.

Sistema de ficheros

- Concepto de fichero y directorio.
- Directorio de trabajo.
- Rutas absolutas y relativas.

Sistema de ficheros

Definición

Abstracción limpia para manejar ficheros con independencia del hardware real que exista.

Ejemplo

Existen muchos tipos de sistemas de ficheros. Cada sistema operativo soporta unos u otros. Como ejemplo tenemos: `ext4`, `NTFS`, `FAT32`, `ReiserFS`, `ISO9660`, etc. Pueden convivir varios simultáneamente en un mismo sistema.

Servicios del sistema para manejar ficheros

- El sistema nos da servicios para manejar ficheros:
  - `create/open`: para crear un fichero nuevo o abrir uno existente;
  - `read/write`: para leer o escribir en un fichero;
  - `close`: cerrar el canal de comunicación con un fichero;
  - `unlink`: borra un fichero de un directorio.

Nota

Los servicios de manejo de ficheros son independientes del sistema de ficheros usado.

Estructura de un árbol de directorios

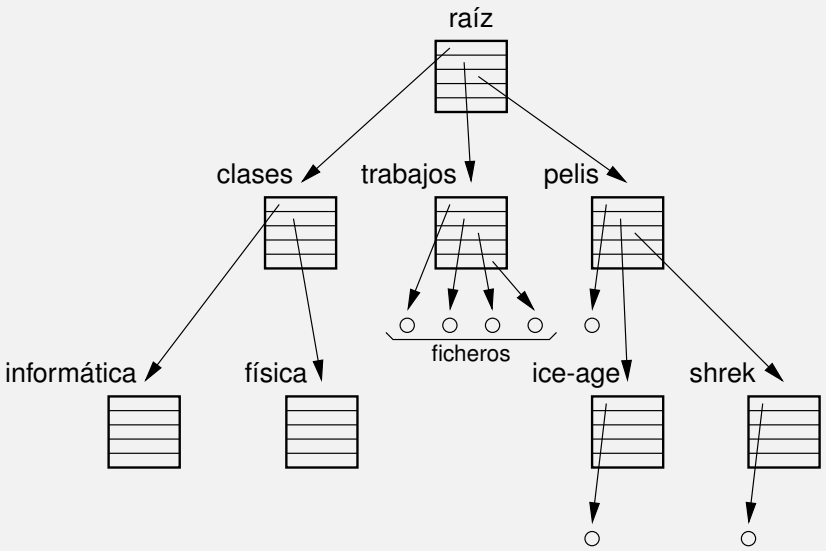


Figura: Vista de la estructura de ficheros en Unix/Linux

Aplicación a Unix/Linux

- jerarquía de ficheros;
- dispositivos como ficheros (virtuales);
- puntos de montaje.

Dispositivos y ficheros

- Para proporcionar una interfaz coherente, en Unix/Linux se accede a los dispositivos también a través de ficheros.
- Se trata de fichero especiales, normalmente situados bajo el directorio `/dev`.
- Los servicios del sistema que se usan son los mismos que para los ficheros normales.

Puntos de montaje en el sistema de ficheros

- Para aumentar el espacio disponible se pueden agregar más discos al sistema de ficheros existente. El nuevo disco ha de contener a su vez un árbol de directorios.
- Para hacer conocido al sistema el nuevo disco, se utiliza el concepto de *punto de montaje*.
- El punto de montaje no es más que un directorio del sistema original bajo el cual se *monta* el nuevo disco.
- El usuario ve que aparecen nuevos directorios y archivos bajo un directorio originalmente vacío. No tiene conciencia de en qué disco está escribiendo.

Interfaz de usuario

- En todo sistema computador siempre existe al menos un proceso para cada usuario que permite a este la interacción con el sistema.
- En su forma más sencilla, la interfaz consiste en una consola textual donde el usuario puede teclear órdenes. Esas órdenes mandan en su mayor parte poner en ejecución distintos programas que el usuario quiere utilizar.
- El proceso interfaz recoge esas órdenes, busca los programas correspondientes, los pone en ejecución y dirige hacia la consola la salida que estos programas producen.

Tipos de interfaz de usuario

Existen varios tipos de interfaces de usuario, cada uno con sus ventajas y sus inconvenientes.

- Interfaz de línea de órdenes: las órdenes se teclean sobre una terminal obedeciendo a cierta sintaxis. En la misma terminal se muestra también la salida producida.
- Interfaz de menús: el usuario selecciona las órdenes a partir de un elenco pre-establecido.
- Interfaz con iconos: las posibles órdenes están simbolizadas con iconos o figuras que se ejecutan al pulsar con el ratón sobre ellas.

Programa shell

Se trata de un programa muy completo que funciona en modo interfaz de línea de órdenes. El usuario teclea sus órdenes y el `shell` las manda ejecutar. Presenta varias características típicas:

- entrada y salida estándar;
- concatenación de entrada/salida, «tuberías»;
- variables de entorno;
- estructura típica de un comando de línea: llamada al comando, argumentos, variable de entorno PATH.
- algunos comandos muy básicos:

  `ls -l`, `cd`, `pwd`, `mkdir`, `rmdir`, `ps -ef`, `pstree`.