

13.09.2018

Sistemas Operativos

Alejandro Alonso

Departamento de Ingeniería de Sistemas Telemáticos

<http://moodle.dit.upm.es>



Algunos derechos reservados. Este documento se distribuye bajo licencia
[Creative Commons Reconocimiento-NoComercial-CompartirIgual 4.0 Internacional.](https://creativecommons.org/licenses/by-nc-sa/4.0/)

Objetivos del tema

- **Describir las características principales de los SO**
- **Describir los principios básicos de su operación**
- **Identificar los componentes principales de los SO**

1. Sistemas Operativos

- ▶ Introducción
- ▶ Funcionamiento del Sistema Operativo
- ▶ Interfaz de programación: llamadas al sistema
- ▶ Componentes de un sistema operativo

2. Gestión de procesos

3. Gestión de memoria

4. Gestión de almacenamiento

5. Gestión de E/S

1.1 Introducción a los sistemas operativos

- **¿Qué es un sistema operativo?**

- ▶ Difícil de definir. Una posible definición es:

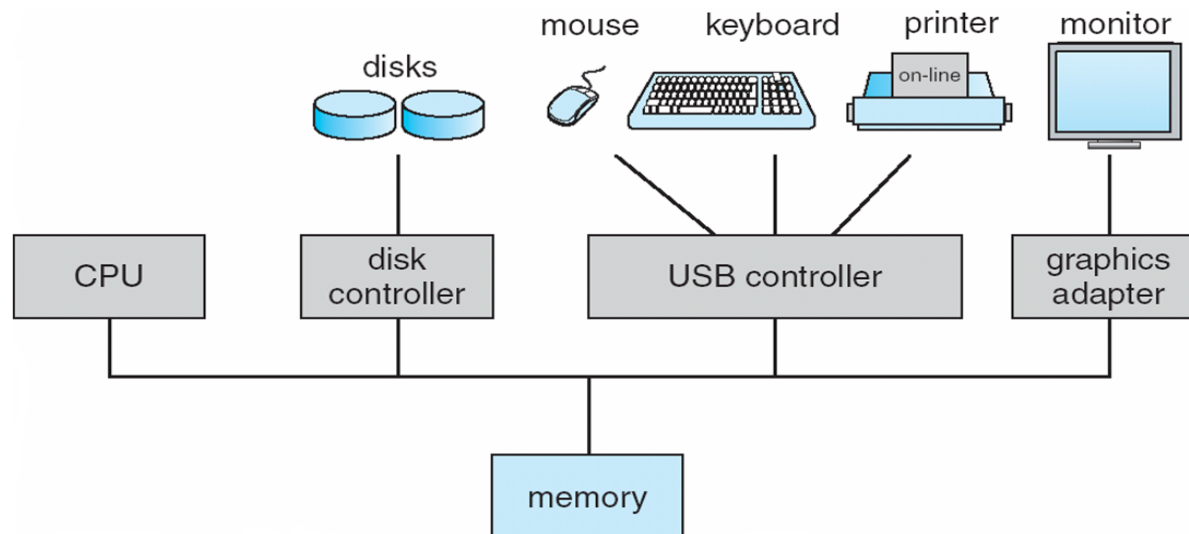
Programa que actúa como intermediario entre el hardware y los usuarios de un computador

- **Objetivos del sistema operativo**

- ▶ Gestión de recursos:
 - Oculta complejidad del hardware
 - Uso eficiente de recursos y control ante accesos simultáneos
- ▶ Facilitar el uso del ordenador: abstracciones adecuadas
 - Mayor nivel de abstracción que el hardware: máquina virtual
 - Abstracciones de alto nivel: procesos, ficheros,
 - Orientado para usuarios y desarrolladores de aplicaciones

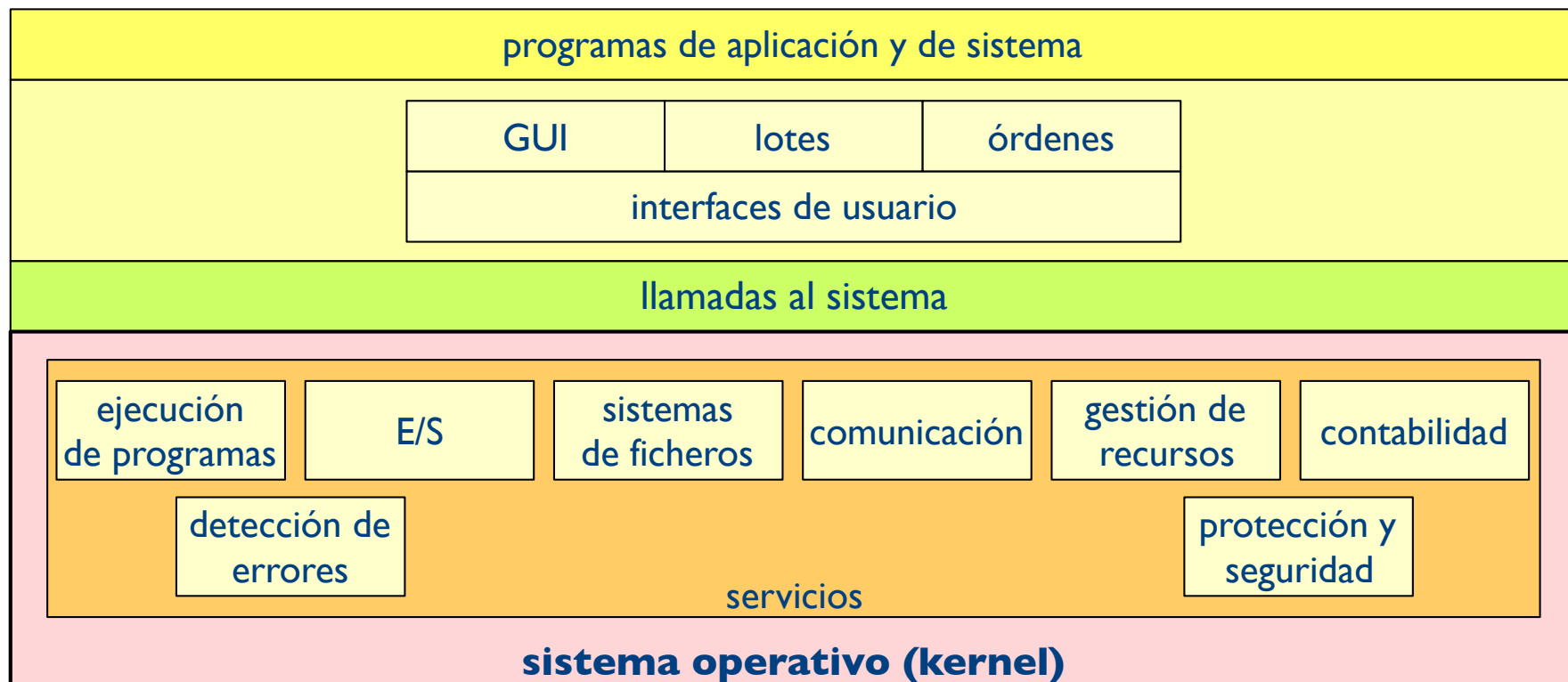
Gestión de los recursos de un Computador *dit* UPM

- **Controladores de dispositivos variados y complejos**
 - ▶ Interfaz de alto nivel para ocultar peculiaridades del hardware
 - ▶ Ficheros: Sucesión de registros lógicos. En Unix: son bytes
- **Contención en el uso de recursos**
 - ▶ Varios usuarios tratan de usar simultáneamente recursos hardware, como procesador, memoria o dispositivos de E/S



Servicios del sistema operativo

- **SO proporciona un entorno para ejecutar programas**
 - ▶ se realiza mediante un conjunto de llamadas al sistemas
- **Los programas o bibliotecas para usuarios les invocan**



- **Los servicios más importantes para los usuarios son:**

- ▶ ejecución de programas
 - carga en memoria y arranque de ejecución
- ▶ operaciones de E/S
 - lectura y escritura en programas
- ▶ gestión de ficheros
 - organización y acceso a la información
- ▶ comunicaciones
 - entre procesos de la misma o distintas máquinas
- ▶ detección de errores
 - errores de hardware o de software
 - acciones de corrección para preservar la integridad del sistema

- **Otros servicios están dirigidos a facilitar el funcionamiento del sistema**
 - ▶ gestión de recursos
 - tiempo de CPU, memoria, espacio en disco, etc.
 - ▶ contabilidad
 - datos sobre uso de recursos
 - ▶ protección y seguridad
 - los procesos no deben interferir con otros procesos
 - los usuarios no deben poder acceder a los recursos sin autorización

1.2. Funcionamiento del Sistema Operativo

- **Generación de las interrupciones:**

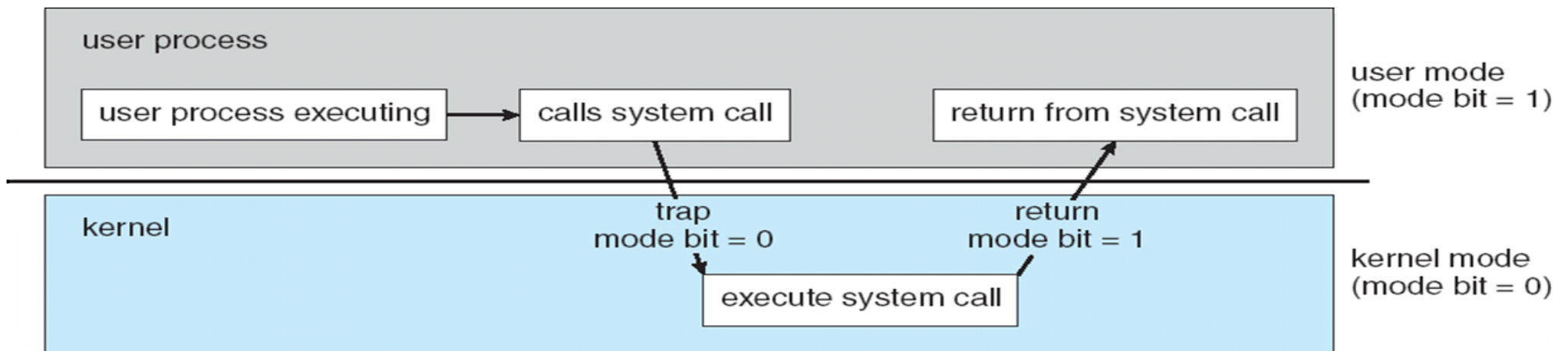
- ▶ Dispositivos (HW) de E/S para indicar fin/inicio de una operac.
- ▶ Software para que el SO realice alguna operación (trap):
 - servicios al usuario (llamadas al sistema)
 - errores de ejecución: División por cero, instrucciones ilegales

- **SO se basa en el modelo dual del procesador:**

- ▶ Dos modos: usuario y núcleo o privilegiado
 - El hardware proporciona un bit de modo
- ▶ Hay instrucciones que se deben ejecutar en modo privilegiado
- ▶ El núcleo se ejecuta en modo privilegiado
- ▶ Fundamental para proteger al sistema de programas erróneos o accesos indebidos

Interrupciones o Excepciones

- **Cuando se produce una interrupción:**
 - ▶ Se salva el contexto de la ejecución en curso
 - ▶ Se pasa a modo supervisor / modo núcleo
 - ▶ Se inhabilitan las interrupciones para que no se pierdan
 - ▶ Transfiere control a la rutina de tratamiento correspondiente
 - Se suele emplear vectores de interrupciones



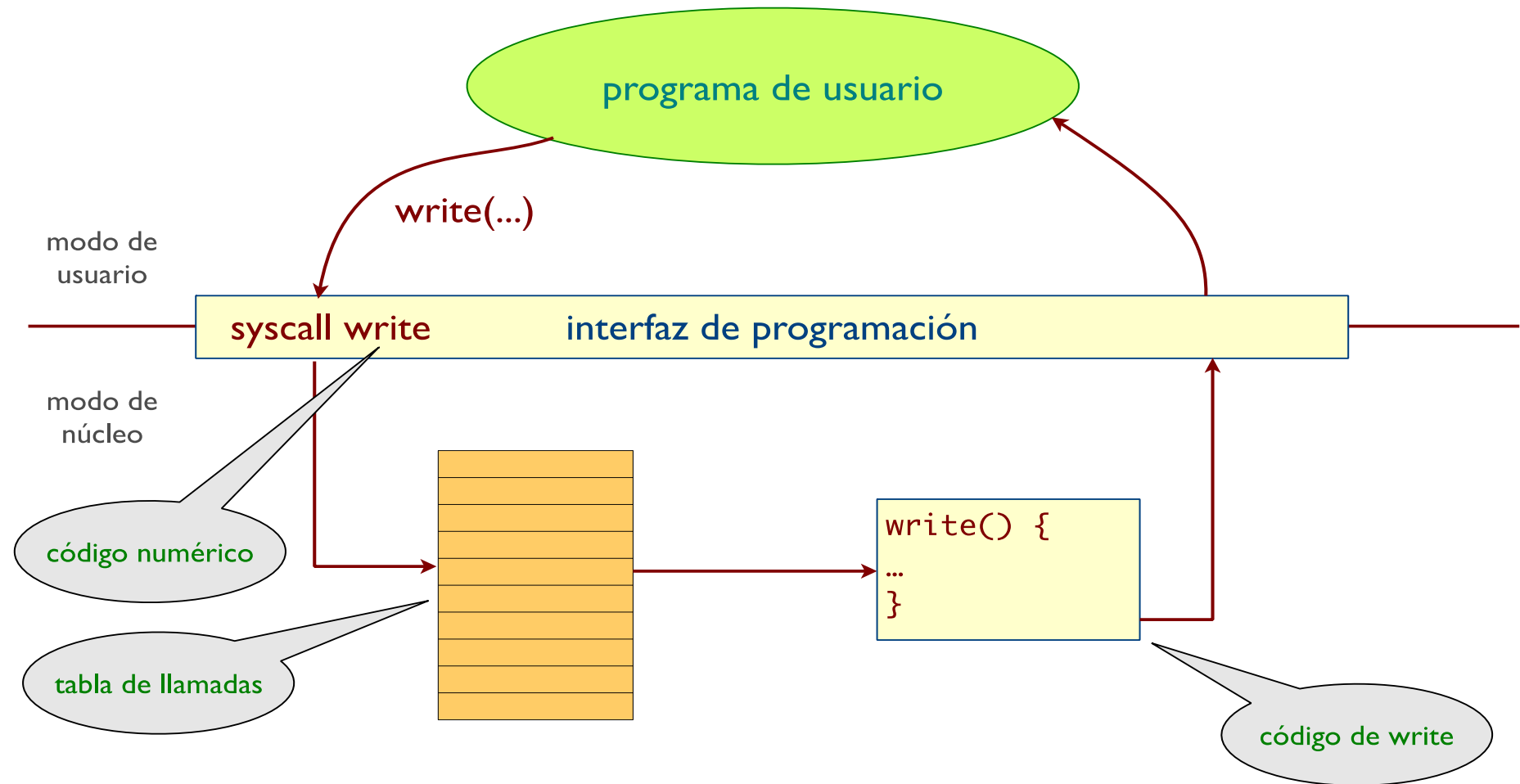
1.3. Interfaz de programación

- **Los programas pueden acceder a los servicios del sistema invocando llamadas al sistema**
 - ▶ operaciones efectuadas por el SO
- **Ejemplo: copiar un fichero en otro**
 - ▶ Obtener los nombres de los ficheros de entrada y salida
 - entrada de texto o selección de un menú
 - ▶ Abrir el fichero de entrada
 - ▶ Crear el fichero de salida
 - avisar si ya existe
 - ▶ Repetir mientras haya datos en el fichero de entrada:
 - leer datos del fichero de entrada
 - escribir datos en fichero de salida
 - ▶ Cerrar los ficheros de entrada y salida

Interfaz de programación (API)

- **Especificación de operaciones del sistema en términos de un lenguaje de programación**
 - ▶ funciones, subprogramas o métodos con parámetros
 - ▶ utiliza la sintaxis de un lenguaje de programación
 - normalmente C, también Java u otros
- **Ejemplo (Java): `java.io.OutputStream.write`**
 - ▶ `public abstract void write (byte[] b,int off,int len)) throws IOException`
- **Ejemplo (C):**
 - ▶ `ssize_t write(int fd, const void *buf, size_t count);`
- **Algunas APIs comunes:**
 - Win32 (sistemas Windows), POSIX (UNIX, Linux, Mac OS), Java API

Ejecución de llamadas al sistema



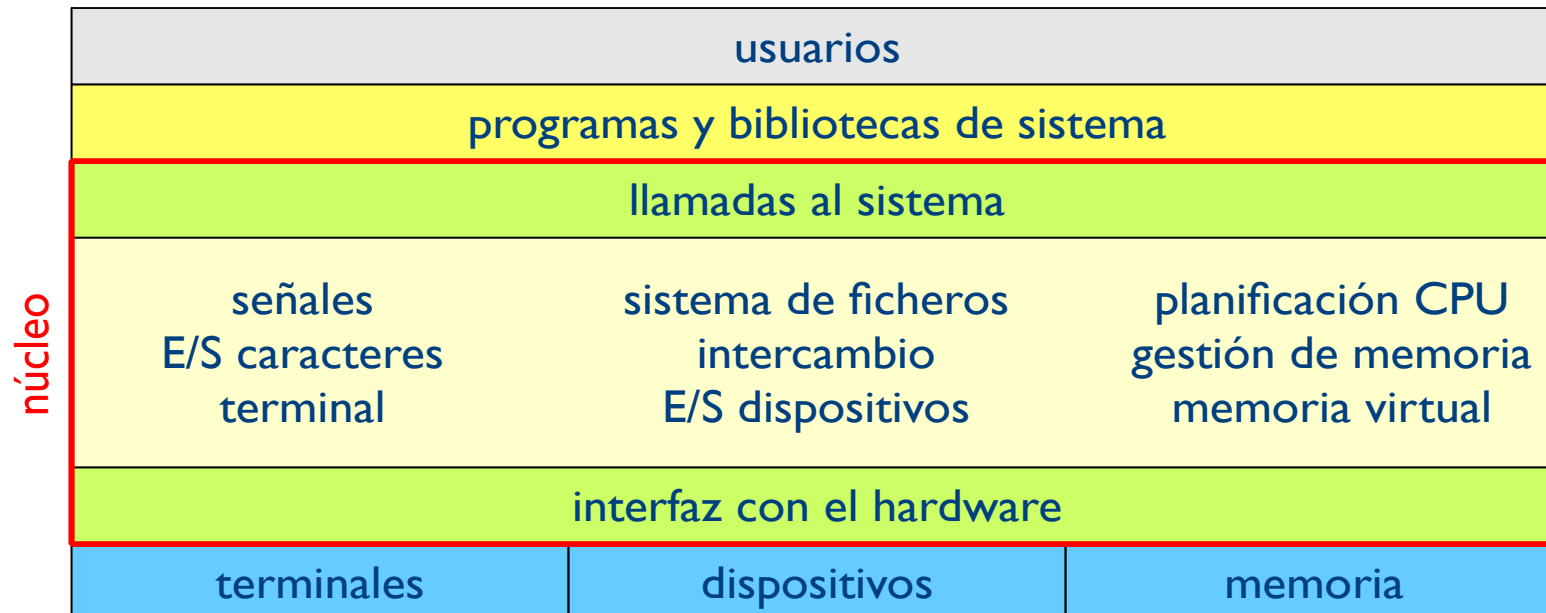
Tipos de llamadas al sistema

	Windows	Unix
Gestión de procesos	CreateProces() ExitProcess() WaitForSingleObject()	fork() exit() wait()
Gestión de ficheros	CreateFile() ReadFile() WriteFile() CloseHandle()	open() read() write() close()
Gestión de dispositivos	SetConsoleMode() ReadConsole() WriteConsole()	ioctl() read() write()
Mantenimiento de información	GetCurrentProcessId() SetTimer() Sleep()	getpid() alarm() sleep()
Comunicaciones	CreatePipe() CreateFileMapping() MapViewOfFile()	pipe() shmget() mmap()
Protección	SetFileSecurity() InitializeSecurirty Descriptor() SetSecurityDescriptorGroup()	chmod() umask() chown()

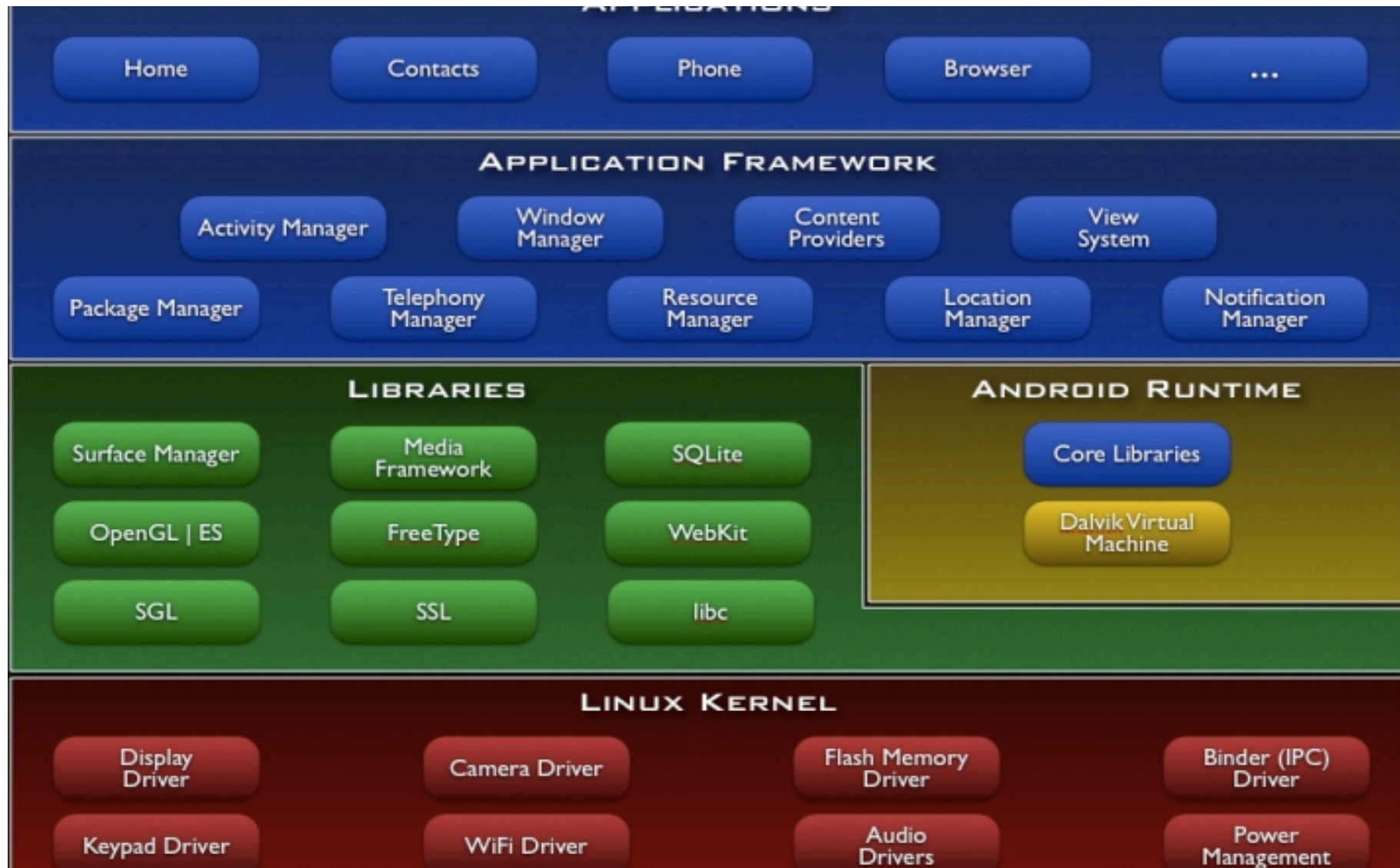
Est

Estructura de sistemas operativos

- Se divide en componentes jerárquicamente
 - de mayor a menor nivel de abstracción
- Se pueden añadir componentes durante la ejecución mediante módulos que se cargan dinámicamente
- Ejemplo: estructura original de Unix



Arquitectura de Android



- **Configuración del SO e instalación en una máquina específica a partir del código de distribución**
 - ▶ CPUs, memoria, discos, particiones, sistemas de ficheros, comunicaciones, otros dispositivos
- **Carga del núcleo (boot)**
 - ▶ cargador inicial, arranca por hardware: *bootstrap loader*
 - en ROM/EPROM (firmware): comprobación de dispositivos básicos
 - Cargador general desde un bloque de disco predeterminado (GRUB)
 - ▶ a continuación se carga, el cargador completo
 - y éste carga el núcleo del SO en memoria
- **Hay cargadores permiten elegir entre varios SO**

1.4. Componentes de un Sistema Operativo

- **El SO se puede dividir desde un punto de vista funcional:**
 - ▶ Gestor de procesos
 - ▶ Gestor de Memoria
 - ▶ Gestor de Almacenamiento Secundario
 - Sistemas de ficheros
 - ▶ Gestor de Entrada Salida
 - ▶ Protección y Seguridad

- **La mayoría de los SO son multiproceso**

- ▶ Se ejecutan varias actividades a la vez
- ▶ Una sola actividad no es suficiente para mantener la CPU y los dispositivos ocupados
- ▶ La multiprogramación organiza las actividades (código y datos) para intentar que la CPU tenga algo que ejecutar
- ▶ Parte de estas actividades se mantienen en memoria
 - Tienen que estar en memoria para poder ejecutarlas
- ▶ Un planificador decide qué actividad se ejecuta. La actividad en ejecución se cambia cuando
 - se bloquea en una operación de E/S
 - se ejecuta durante un periodo dado de tiempo

- **Las instrucciones y los datos de un proceso tienen que estar en memoria para ejecutarse**
- **El gestor decide cuáles son los contenidos de la memoria:**
 - ▶ Mapa de memoria, qué (partes) procesos, ...
 - ▶ Se trata de optimizar el uso de la CPU y la respuesta a los usuarios
- **Actividades del gestor de memoria:**
 - ▶ Memoria libre ocupada y quién la usa
 - ▶ Qué procesos (o partes) se mantienen en memoria

- **Proporciona una visión uniforme y virtual del almacenamiento de información**
 - ▶ Fichero: abstracción de almacenamiento de información que oculta los detalles de los dispositivos físicos
 - ▶ Cada dispositivo tiene características diferentes
- **Sistema de ficheros**
 - ▶ Los ficheros se suelen organizar en directorios
 - ▶ Control de acceso a la información
 - ▶ Operaciones del SO:
 - Creación y borrado de ficheros y directorios
 - Operaciones para manipular ficheros y directorios
 - Almacenar los ficheros en los dispositivos
 - Copias de seguridad en dispositivos no volátiles

- **Se suelen usar discos para:**
 - ▶ Almacenar lo que no cabe en memoria
 - ▶ Información que se quiere guardar
- **El rendimiento de los subsistemas de gestión de disco tienen gran influencia en las prestaciones del sistema**
- **Actividades del SO:**
 - ▶ Gestión del espacio libre
 - ▶ Asignación de espacio de almacenamiento
 - ▶ Planificación del acceso a disco
- **Almacenamiento de copias de seguridad**
 - ▶ Dispositivos ópticos y cinta magnética
 - ▶ Menos importante las prestaciones

- **Un objetivo del SO es ocultar los detalles y peculiaridades de los dispositivos de E/S**
- **Responsabilidades del SO:**
 - ▶ Gestión de la memoria de E/S:
 - Almacenamiento temporal de datos, mientras se transfieren
 - Caché de disco
 - Spooling, etc
 - ▶ Interfaz con los gestores de dispositivos
 - ▶ Gestores para los diferentes dispositivos hardware

- **Protección: mecanismos para controlar el acceso de procesos o usuarios a recursos del sistema**
- **Seguridad: defensa del sistema contra ataques internos y externos**
 - ▶ Denegación de servicio, virus, gusanos, robo de identidad, robo de servicios, etc.
- **Se suele usar la identidad del usuario para determinar qué puede hacer**
 - ▶ Cada usuario tiene un identificador
 - ▶ Se asigna a cada fichero y proceso del usuario para determinar las operaciones permitidas

5. Tipos de sistemas

- **Procesador: CPU**

- ▶ Microprocesador: procesador integrado en un único chip.
 - Intel4004 (1971) con ancho de palabra de 4 bits y 2000-3000 transis.
- ▶ Multiprocesador: computador con más de un procesador que comparten la memoria principal y ejecutan instrucciones en paralelo.
 - Multicore: multiprocesador integrado en único chip.
 - Manycore: multicore con muchos procesadores (¿> 50?)

- **SoC (System on Chip)**

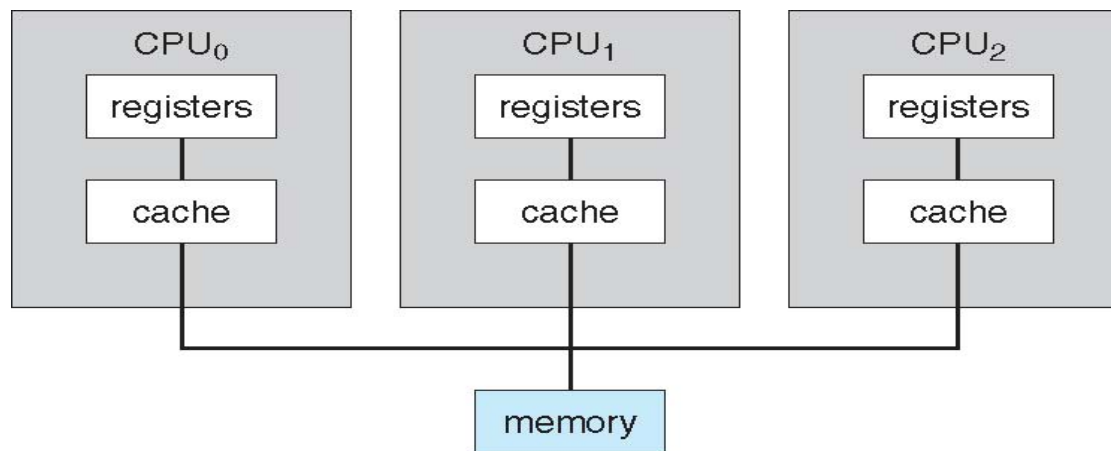
- ▶ Computador completo (CPU(s), sistema de entrada-salida, dispositivos periféricos y memorias) en un único chip.
- ▶ Término de uso habitual en hardware configurable que permite construir computadores a medida.

Multiprocesador

- **La mayoría de los sistemas usan un procesador de propósito general**
 - ▶ También procesadores de propósito específico (Gráficos, FPU)
- **Sistemas multiprocesador**
 - ▶ También se conocen como sistemas paralelos o fuertemente acoplados
 - Mejores prestaciones: motivo más importante
 - Mayor fiabilidad: se pueden tolerar fallos, pero el software se complica
 - Economía de escala: varios procesadores comparten periféricos, especialmente en grandes sistemas.
 - ▶ Dos tipos: simétricos y asimétricos

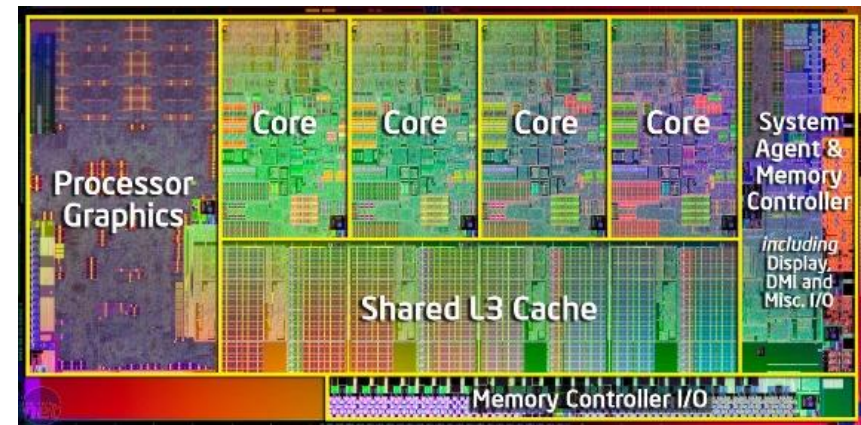
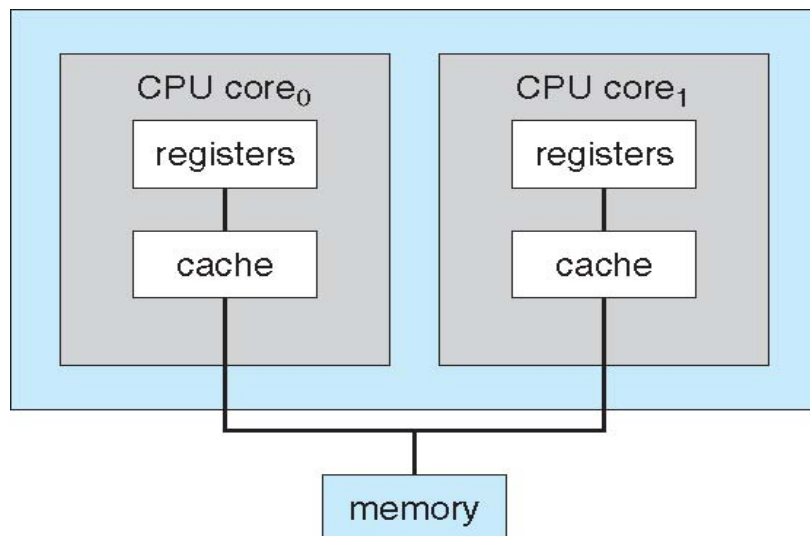
Multiprocesador

- **Varios procesadores, conectados por el mismo bus:**
 - ▶ Comparten memoria y dispositivos
 - ▶ Registros propios y memoria oculta
- **Se complica el SO:**
 - ▶ Pueden ejecutar varias actividades en paralelo
 - ▶ Los datos de E/S deben llegar al procesador adecuado



Sistema con cores

- **Multiprocesador con varios CPUs en el mismo chip**
- **Aumentar la potencia de los procesadores:**
 - ▶ Aumentar la frecuencia es mucho más complicado actualmente
- **El SO se complica para optimizar el uso de los núcleos**
- **Comunicaciones entre núcleos más rápidas y menor consumo**



- **Objetivos SO:**

- ▶ Gestión eficiente de recursos
- ▶ Interfaz adecuado con usuario

- **Sistema Operativo**

- ▶ Fundamental para adecuado funcionamiento de computadores
- ▶ Dirigido por interrupciones
- ▶ Núcleo: se ejecuta en modo protegido

- **Componentes fundamentales**

- ▶ Gestión de procesos
- ▶ Gestión de memoria
- ▶ Gestión de almacenamiento secundario
- ▶ Gestión de E/S

- A. Silberschatz, P. Galvin y G. Gagne: **Operating System Concepts with Java**. 8ª edición. Addison Wesley, 2011.
- A. Silberschatz, P. Galvin, y G. Gagne: **Fundamentos de Sistemas Operativos**, 7a edición, 2005, McGraw-Hill.
- G. Fernández, **Conceptos básicos de Arquitectura y Sistemas Operativos**, 5ª Edición. Publicaciones ETSIT, 2005.