
Sistemas operativos

Jorge Juan Chico <jjchico@dte.us.es>, Julián Viejo Cortés <julian@dte.us.es> 2011, 2014
Departamento de Tecnología Electrónica
Universidad de Sevilla

Usted es libre de copiar, distribuir y comunicar públicamente la obra y de hacer obras derivadas siempre que se cite la fuente y se respeten las condiciones de la licencia Attribution-Share alike de Creative Commons.
Puede consultar el texto completo de la licencia en <http://creativecommons.org/licenses/by-sa/3.0/>

Objetivos

- Conocer las funciones de los sistemas operativos bajo la perspectiva de los distintos tipos de usuarios
- Conocer la evolución histórica de los sistemas operativos y relacionarla con los cambios en la forma de usar el ordenador
- Conocer los tipos de tareas principales que realiza un sistema operativo moderno

Contenidos

- Funciones del sistema operativo
 - Para el usuario.
 - Para el programador.
 - Para el desarrollador del sistema operativo.
- Tareas del sistema operativo
 - Gestión de procesos.
 - Gestión de memoria.
 - Gestión de E/S
- Evolución de los sistemas operativos

Software y sistema operativo

Sin S.O. (stand-alone)

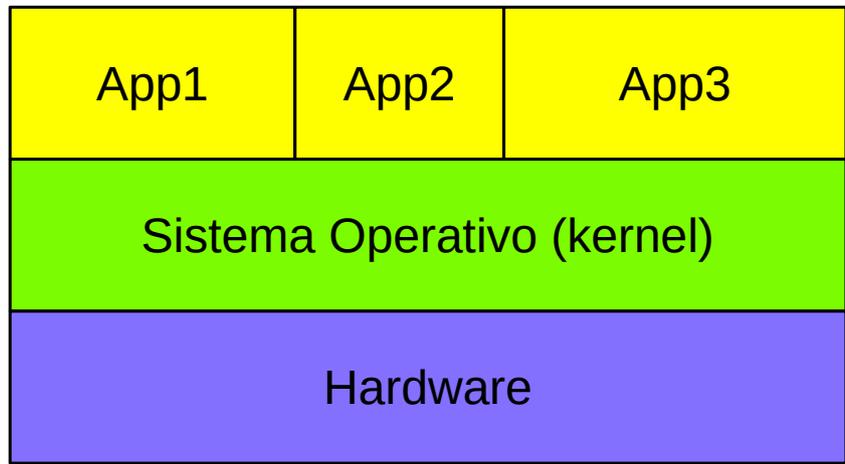
- El programa incluye todas las funciones necesarias
- Complejo: hay que programar todos los detalles
- Alto coste: programar tareas repetitivas
- Conocimiento detallado del hardware

Aplicación (programa)

Hardware

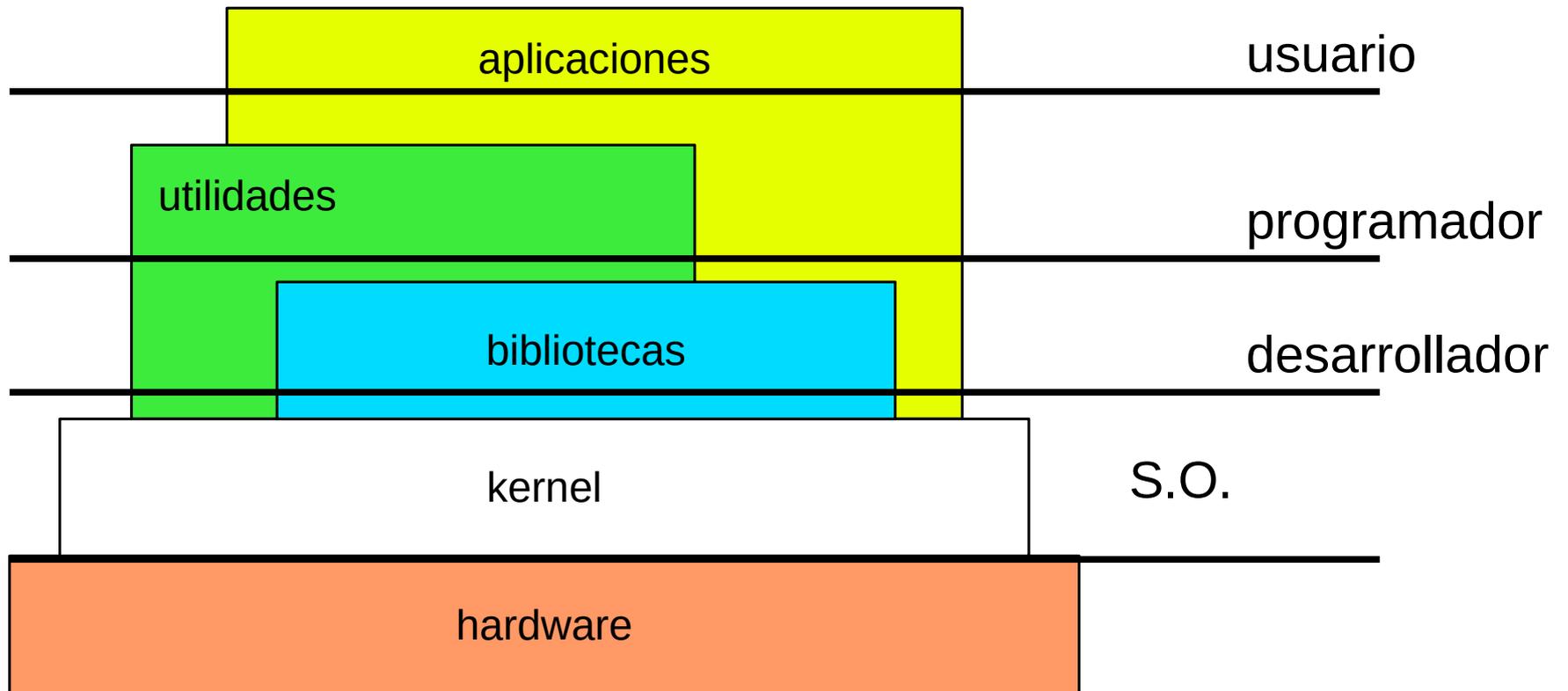
Software y sistema operativo

Con S.O.



- El sistema operativo gestiona el hardware y organiza la ejecución de los programas
- Operaciones comunes programadas en el S.O.
 - Llamadas al sistema
- Programación más fácil
- Ejecución de varios programas
- No se necesita conocimiento detallado del hardware

Jerarquía del software



Funciones del S.O. El Usuario: interfaz

- Software genérico que “viene” instalado con el ordenador
 - Interfaz hombre/máquina: interfaz de usuario
 - Utilidades varias: editor de textos, calculadora, etc.
 - Aplicaciones de uso general: reproductor multimedia, navegador web, correo electrónico, etc.

Funciones del S.O. El programador: interfaz de programación

- Funciones y utilidades básicas que permiten la instalación y desarrollo de otras aplicaciones
 - Programas básicos: editor de textos, comandos básicos
 - Biblioteca de funciones básicas
 - Herramientas de desarrollo: compilador, enlazador, ensamblador, etc.

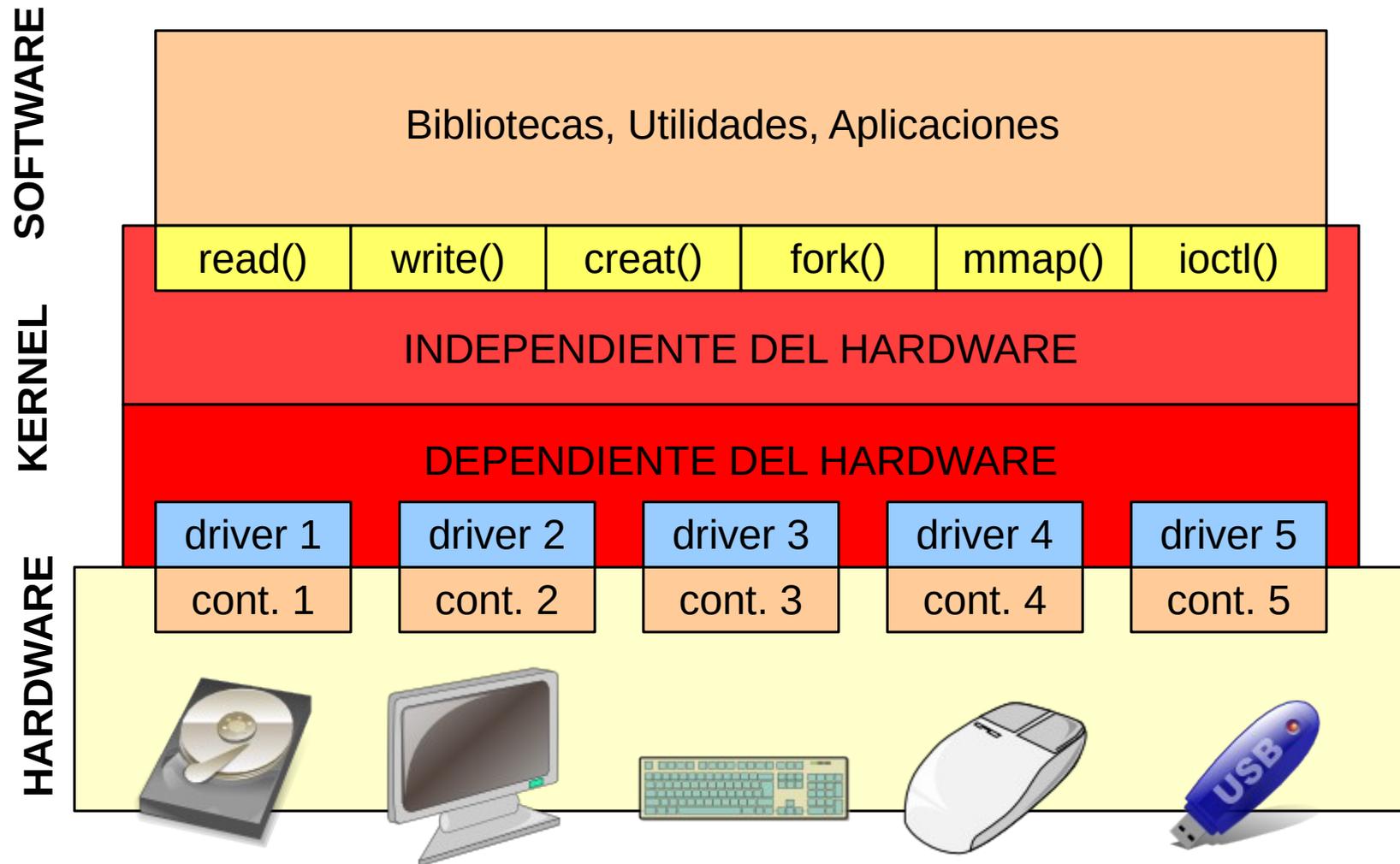
Funciones del S.O.

El desarrollador: máquina virtual

- Como gestor de recursos y controlador de la ejecución de procesos
 - Gestión de archivos
 - Gestión de memoria
 - Gestión de tareas
 - Gestión de entrada/salida
- Objetivo: suministrar una “máquina virtual” independiente del hardware

Funciones del S.O.

El desarrollador: máquina virtual



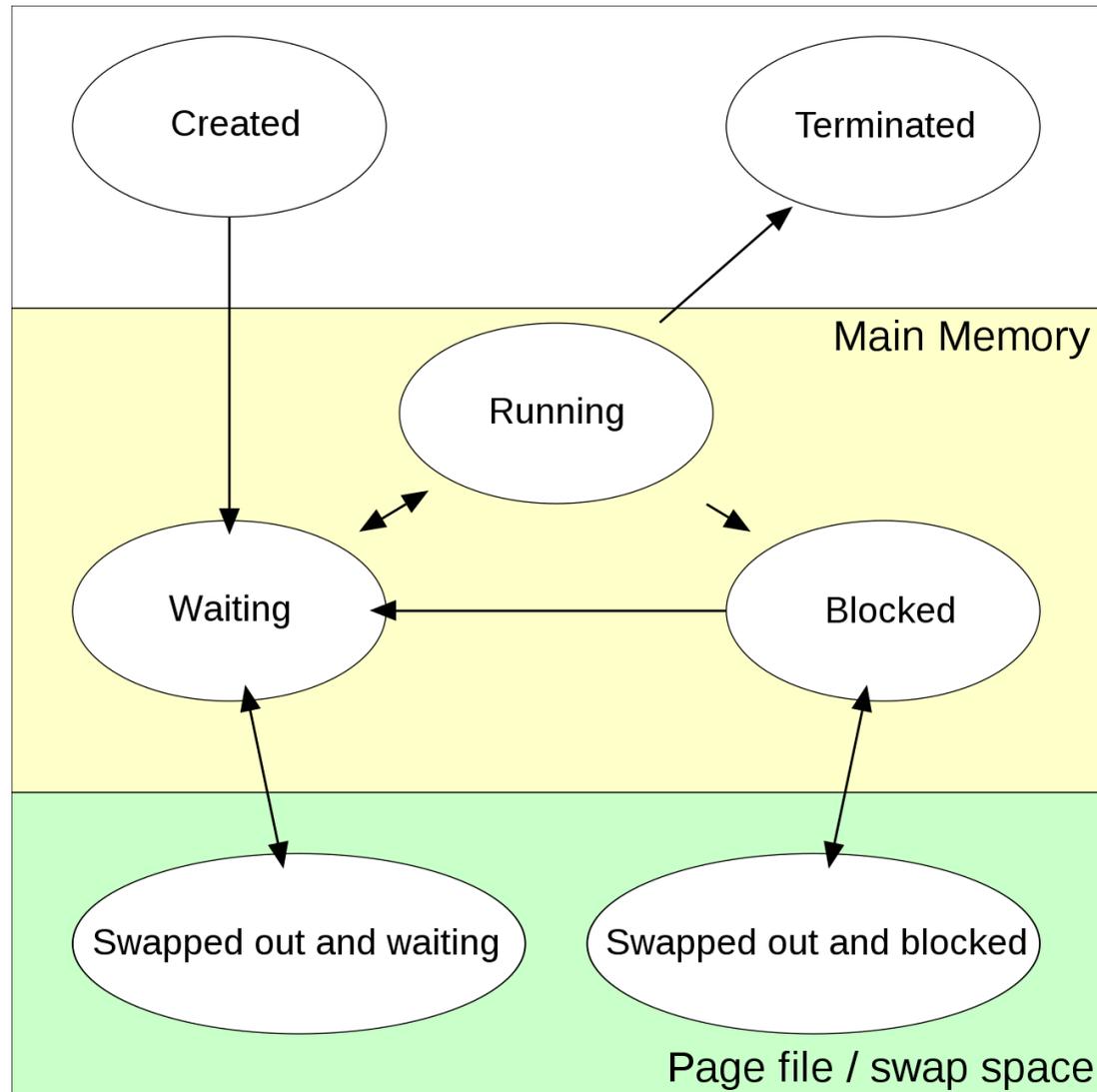
Contenidos

- Funciones del sistema operativo
 - Para el usuario.
 - Para el programador.
 - Para el desarrollador del sistema operativo.
- **Tareas del sistema operativo**
 - **Gestión de procesos.**
 - **Gestión de memoria.**
 - **Gestión de E/S**
- Evolución de los sistemas operativos

Gestión de procesos

- Proceso
 - Programa cargado en la memoria para su ejecución
- Problema
 - Los ordenadores modernos ejecutan varios procesos a la vez (multitarea)
 - El tiempo de CPU debe repartirse entre todos los procesos en estado de ejecución
- Tareas del S.O.
 - Implementación de la multitarea: control del estado de los procesos
 - Protección entre procesos
 - Comunicación entre procesos
- El S.O. es un proceso “privilegiado”
 - Algunas funciones del ordenador sólo están disponibles para el S.O. (instrucciones privilegiadas)
 - Se necesita soporte del hardware

Gestión de procesos. Estados.



Gestión de entrada/salida

- Problema
 - Varios procesos necesitan acceder a los mismos dispositivos de entrada/salida (teclado, ratón, pantalla, red, etc.)
 - ¿Qué proceso puede usar un dispositivo en un momento dado?
 - ¿Qué proceso recibe los datos que llegan a un dispositivo?
- Solución
 - Sólo el S.O. puede acceder directamente a los dispositivos de E/S
 - El resto de procesos acceden a E/S usando funciones del S.O.
- Tareas del S.O.
 - Organizar las peticiones de acceso de E/S de los procesos
 - Gestionar permisos de ciertas funciones de E/S
 - Crear “abstracciones” fáciles de usar por los procesos
- Ejemplos
 - Sistema de archivos: archivos y carpetas
 - Funciones de red: paquetes, puertos y conexiones

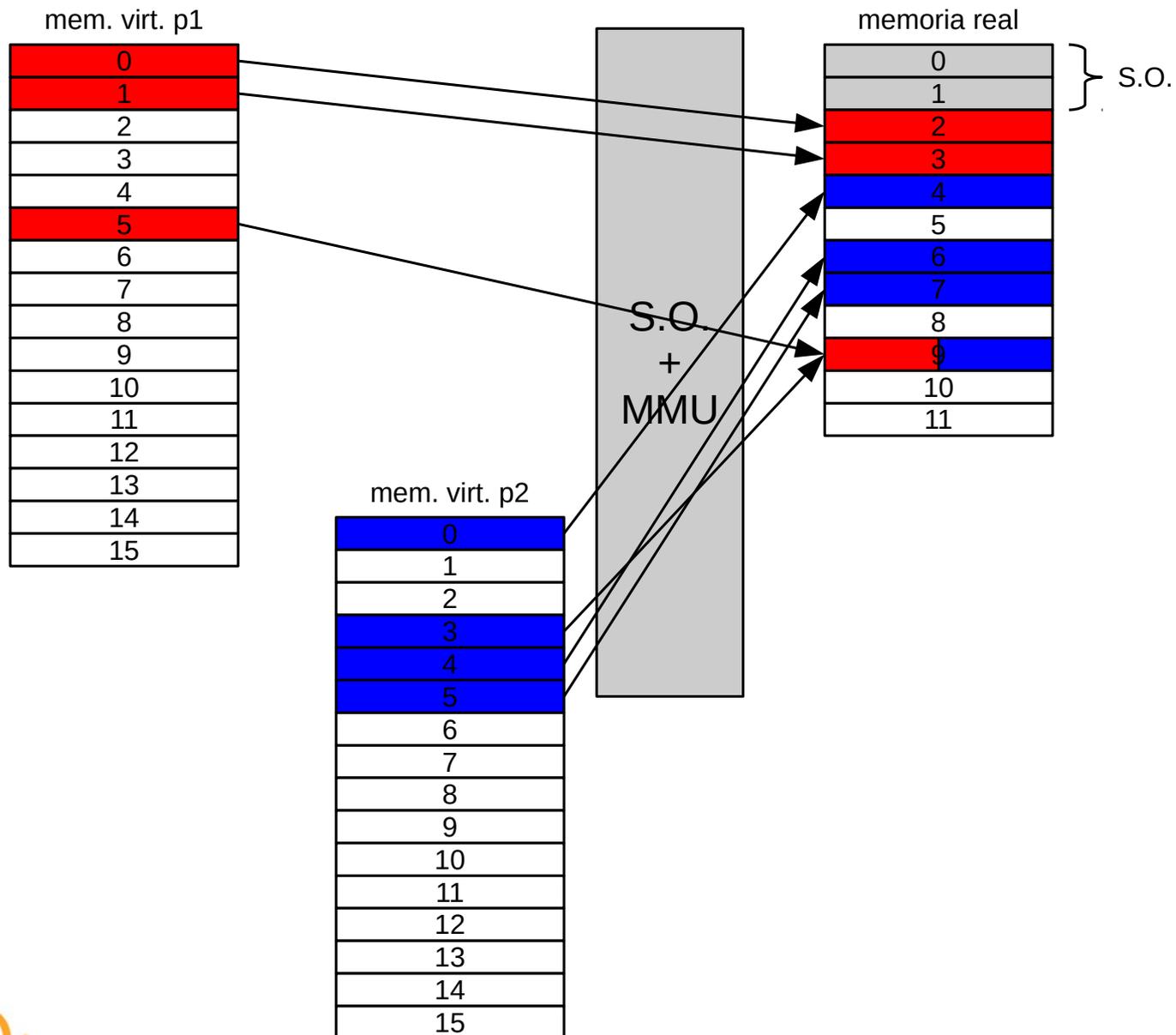
Gestión de memoria

- Problema: varios procesos comparten la memoria
 - 1) (Re)ubicación de procesos en la memoria
 - 2) Fragmentación de la memoria
 - 3) ¿Qué pasa si un proceso accede a la memoria de otro?
 - 4) ¿Cómo se comunican los procesos? ¿Compartir memoria?
 - 5) ¿Qué pasa si no hay suficiente memoria para todos los procesos?
- Solución: Memoria Virtual + paginación de memoria
 - 1) Cada proceso tiene un espacio de direcciones continuo
 - 2) El S.O. divide la memoria en páginas y las asigna a la memoria real
 - 3) La misma dirección en dos procesos corresponde a páginas reales distintas
 - 4) Los procesos pueden compartir páginas
 - 5) Algunas páginas pueden almacenarse fuera de la memoria principal

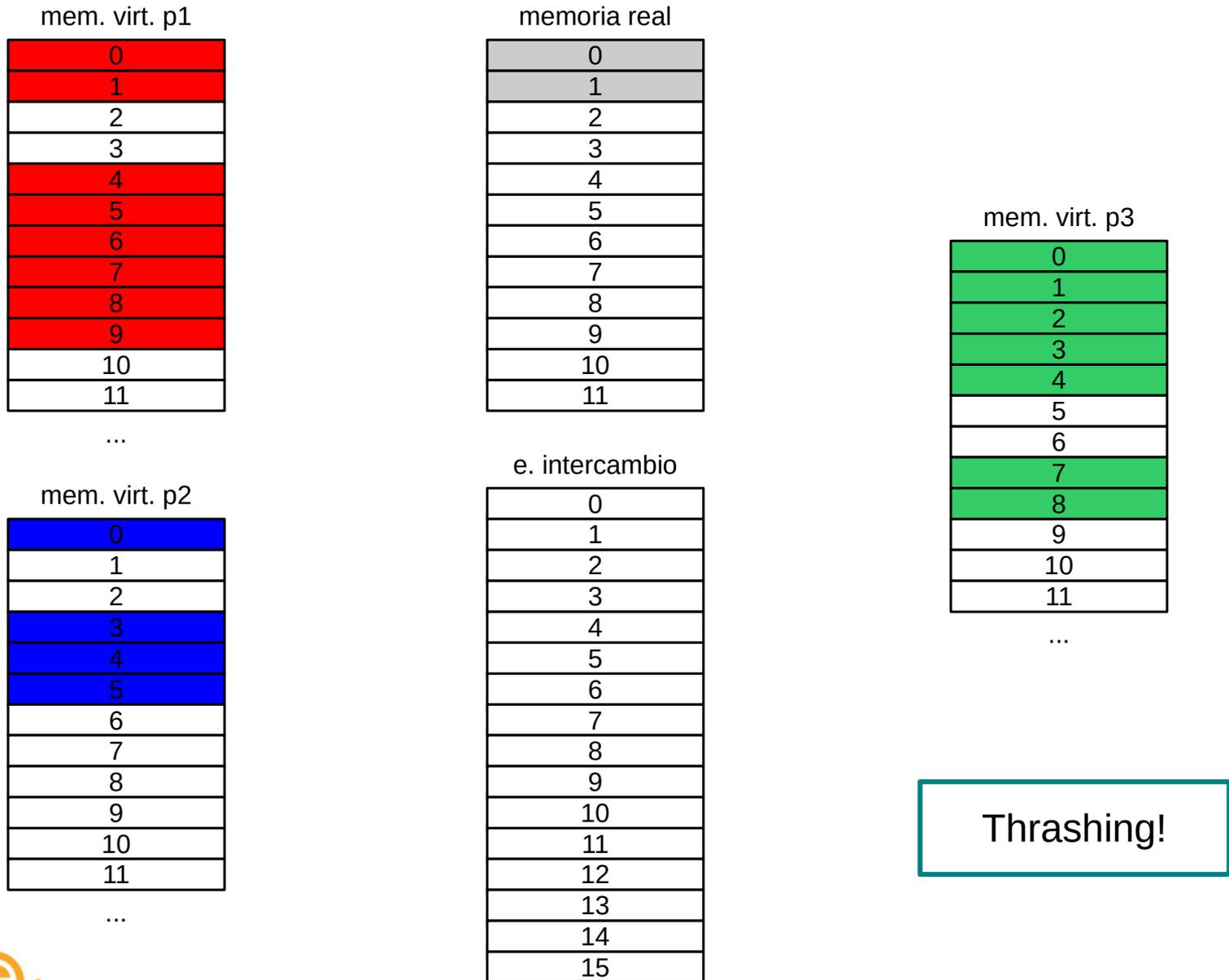
Virtual Memory 1–3
(Youtube)

Virtual Memory
(Wikipedia)

Memoria Virtual



Memoria Virtual



Contenidos

- Funciones del sistema operativo
 - Para el usuario.
 - Para el programador.
 - Para el desarrollador del sistema operativo.
- Tareas del sistema operativo
 - Gestión de procesos.
 - Gestión de memoria.
 - Gestión de E/S
- **Evolución de los sistemas operativos**

Evolución de los S.O.

- Sistemas primitivos
- Sistemas de colas manuales
- Sistemas de colas sencillos (monitor)
- Sistemas de colas sofisticados
- Sistemas de tiempo compartido

History of Operating Systems. Background
(Wikipedia)

Evolución de los S.O. Sistemas primitivos

- Programación directa en código máquina
- Los programas (cada instrucción) se cargan en memoria directamente (ej: mediante tarjetas perforadas)
- Los errores se indican con indicadores luminosos
- Ordenador operado por los propios usuarios

Problema:
uso del ordenador complejo y poco eficiente

Evolución de los S.O. Colas manuales

- Existe software pre-programado de uso general
 - ensambladores, compiladores, bibliotecas, ...
- Se escriben programas en lenguajes de alto nivel
- Entrada/salida:
 - Tarjetas/cinta magnética
 - Impresora
- Planificación de tareas manual: operador del ordenador

Problema:
tiempo de preparación y gestión de tareas excesivo

Evolución de los S.O. Colas sencillas (monitor residente)

- Trabajos ordenados en una cola de entrada
- Monitor residente: programa que carga y ejecuta trabajos automáticamente
 - Preparación automática: lenguaje de control de trabajos
- Software de soporte
 - Compiladores, funciones de E/S, etc.
- Soporte del hardware:
 - interrupciones
 - protección de memoria
 - temporización
 - instrucciones privilegiadas: E/S, etc.

Problema:
tiempo de CPU desperdiciado durante E/S

Evolución de los S.O. Colas sofisticadas

- Multiprogramación:
 - Varios programas cargados en memoria a la vez
 - Un programa se ejecuta mientras otro espera E/S
 - Necesaria planificación de tareas
- Soporte del hardware
 - Múltiples interrupciones
 - Gestión avanzada de la memoria
 - Acceso directo a memoria (DMA)

Problema:
tiempo de espera entre la modificación del programa
y la obtención de los resultados

Evolución de los S.O. Sistemas de tiempo compartido

- Multitarea preeventiva
 - Ejecución de procesos de forma aparentemente simultánea
 - División del tiempo de CPU entre los diferentes procesos
- Uso interactivo del sistema mediante “terminales”
 - (Desaparecen) las colas de programas
 - Los programadores preparan los programas en el mismo ordenador
 - Aparecen programas de usuario:
 - Editores de texto, hojas de cálculo, etc.
- Multiprocesado
 - Se desarrollan ordenadores con varias CPU
 - Algunas tareas realmente simultáneas

Tipos de S.O.

- Solían haber distintos tipos de sistemas operativos para distintos tipos de ordenadores
 - Sistemas empujados (específicos)
 - Ordenadores personales
 - Estaciones de trabajo
 - Mainframes/superordenadores
- Actualmente:
 - Todos los tipos de ordenadores se parecen mucho entre si
 - Los diferentes S.O. se parecen mucho entre si
 - Hay una tendencia a la uniformidad: un sistema operativo para todo
- El Software Libre y GNU/Linux han sido favorecidos
 - Posibilidad de adaptación
 - Modularidad
 - Herramientas de desarrollo muy maduras

Tipos de S.O. Ejemplo de evolución

