

# Introducción

# 1

## Sistemas Operativos y Distribuidos


Mg. Javier Echaiz

D.C.I.C. – U.N.S.

<http://cs.uns.edu.ar/~jechaiz>

[je@cs.uns.edu.ar](mailto:je@cs.uns.edu.ar)

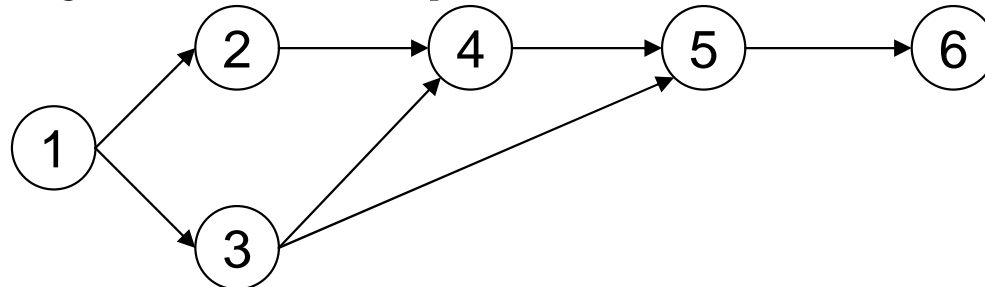


A vertical red bar with a textured, slightly grainy appearance, featuring a bright light flare at the top left corner.

***Los Sistemas Operativos  
son programas para la  
administración eficiente  
de los recursos de la  
computadora.***

# Problemas que resuelve un S.O. (1)

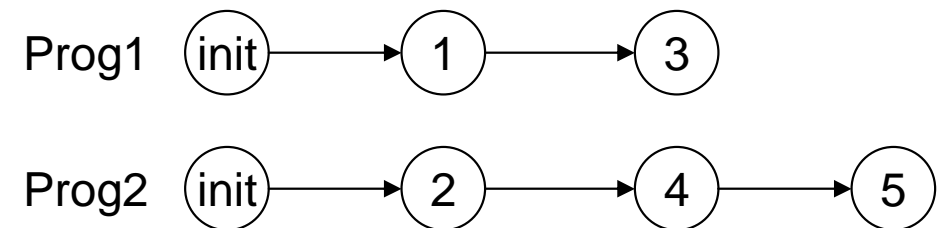
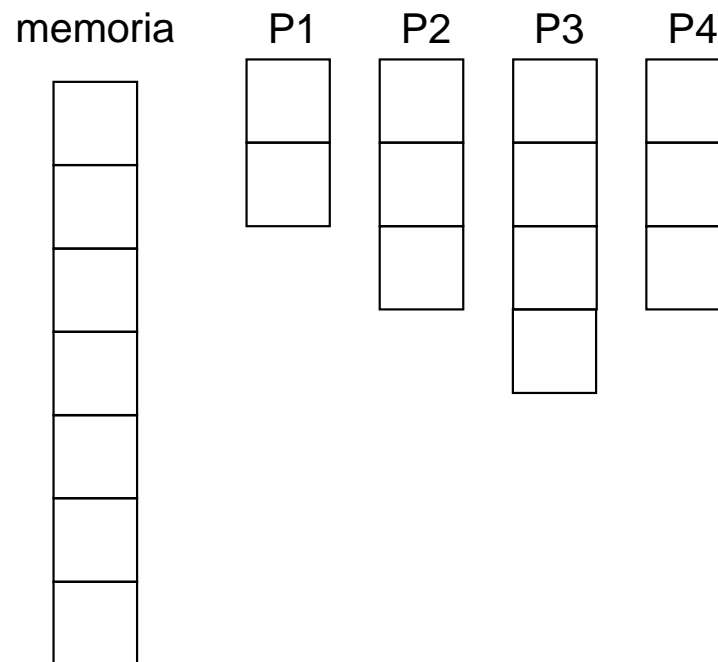
- Una aplicación en ejecución es un conjunto de procesos en ejecución.



- Cómo sabe el proceso 2 que el proceso 1 ya terminó.
- Dado que 2 y 3 se pueden ejecutar en paralelo, ¿quién se ejecuta primero?
- Si 2 y 3 ya terminaron, ¿cómo se enteró 4 de este suceso?

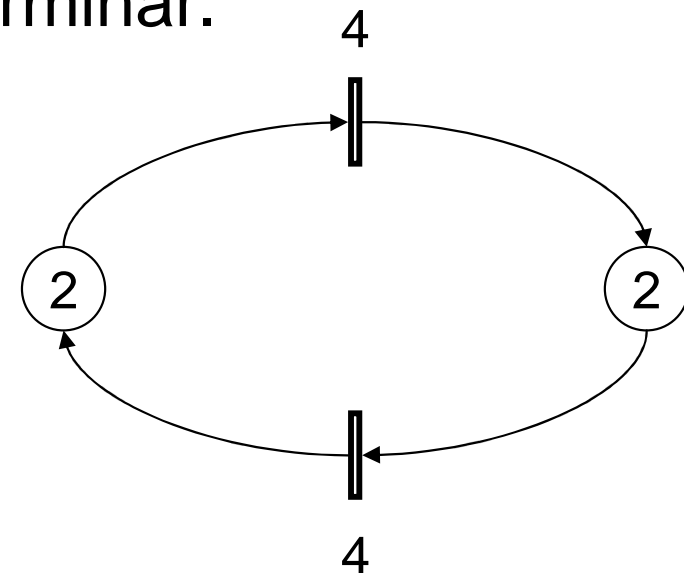
# Problemas que resuelve un S.O. (2)

- Los procesos requieren de recursos para ser ejecutados, cada uno de los siguiente procesos requiere la cantidad indicada de recursos. Ambos programas se ejecutan en paralelo. ¿en qué orden se debe de asignar los recursos a los proceso?. Considere que mientras un programa no termina, los procesos asociados no pueden soltar los recursos. Considere que una propiedad deseable de un SO es la multiprogramación.



# Problemas que resuelve un S.O. (3)

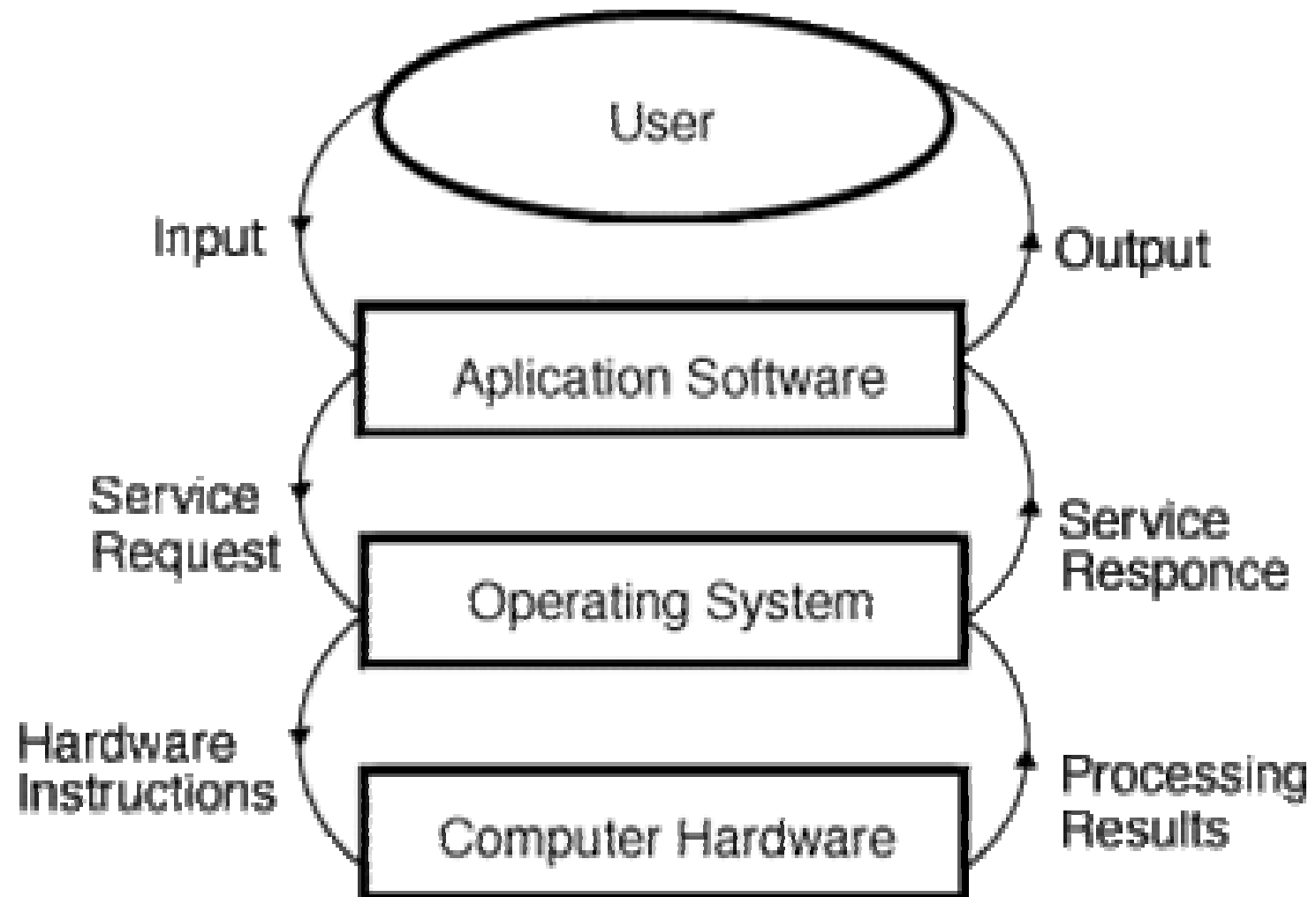
- El siguiente problema se denomina “*deadlock*” y se presenta cuando el CPU no puede asignar un recurso compartido a un conjunto de procesos pendientes de terminar. Los recursos que podría liberar uno de los procesos cuando termine serían suficientes como para continuar los otros procesos y por lo tanto terminar.



# ¿Qué es un sistema operativo?

- Un programa que actúa como un intermediario entre un usuario de una computadora y el *hardware* de la misma.
- Objetivos del Sistema Operativo:
  - Ejecutar los programas de usuario y permitir la solución de problemas del usuario más fácilmente.
  - Conveniencia de uso del sistema de la computadora.
- Uso del hardware de la computadora de manera eficiente.

# Componentes del Sistema de Cómputo (1)





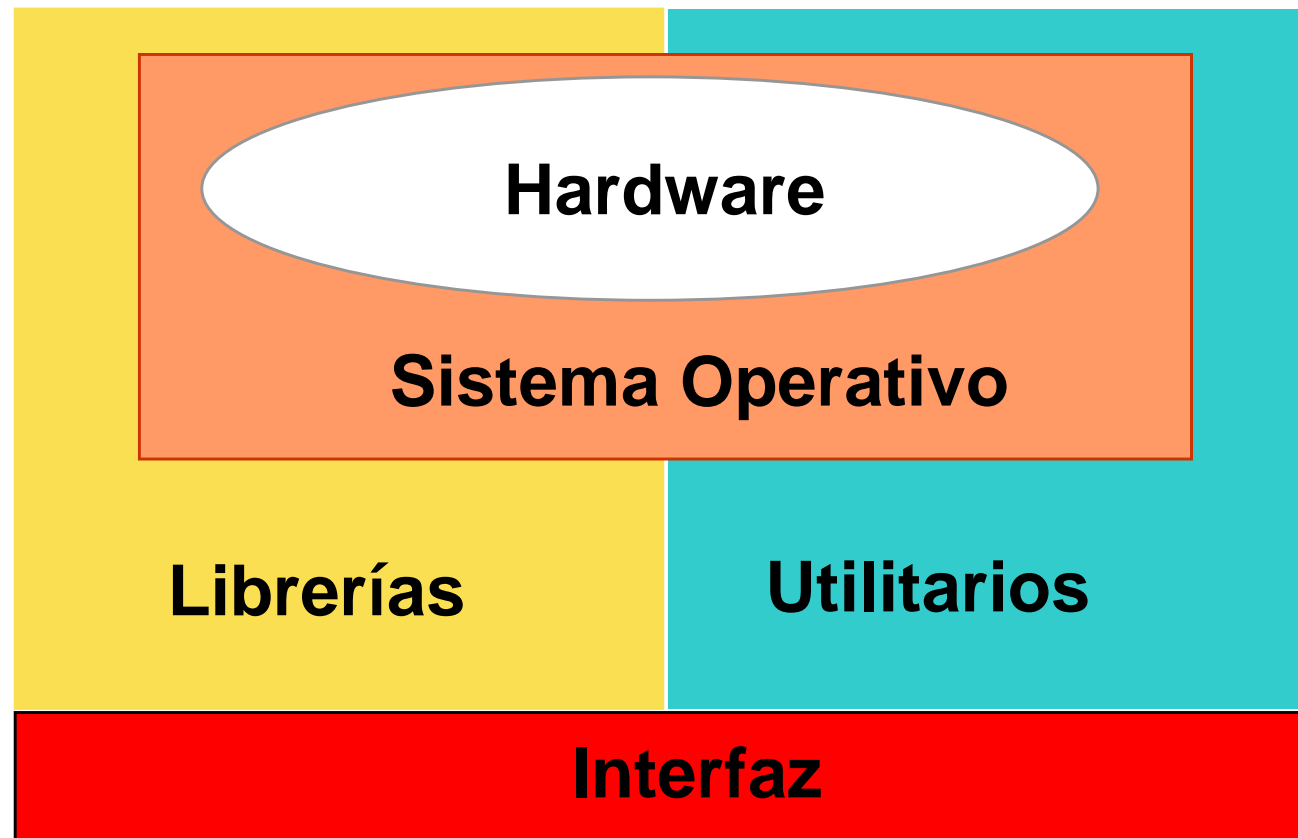
# Componentes del Sistema de Cómputo (2)

1. **Hardware** – provee los recursos básicos de computación (CPU, memoria, dispositivos de E/S).
2. **Sistema Operativo** – controla y coordina el uso del hardware entre varios programas de aplicación para varios usuarios.
3. **Programas de Aplicación** – define el modo en el cual los recursos del sistema son usados para resolver los problemas de computación de los usuarios (compiladores, motores de bases de datos, juegos, programas administrativos, etc.).
4. **Usuarios** (persona, máquina, otras computadoras).

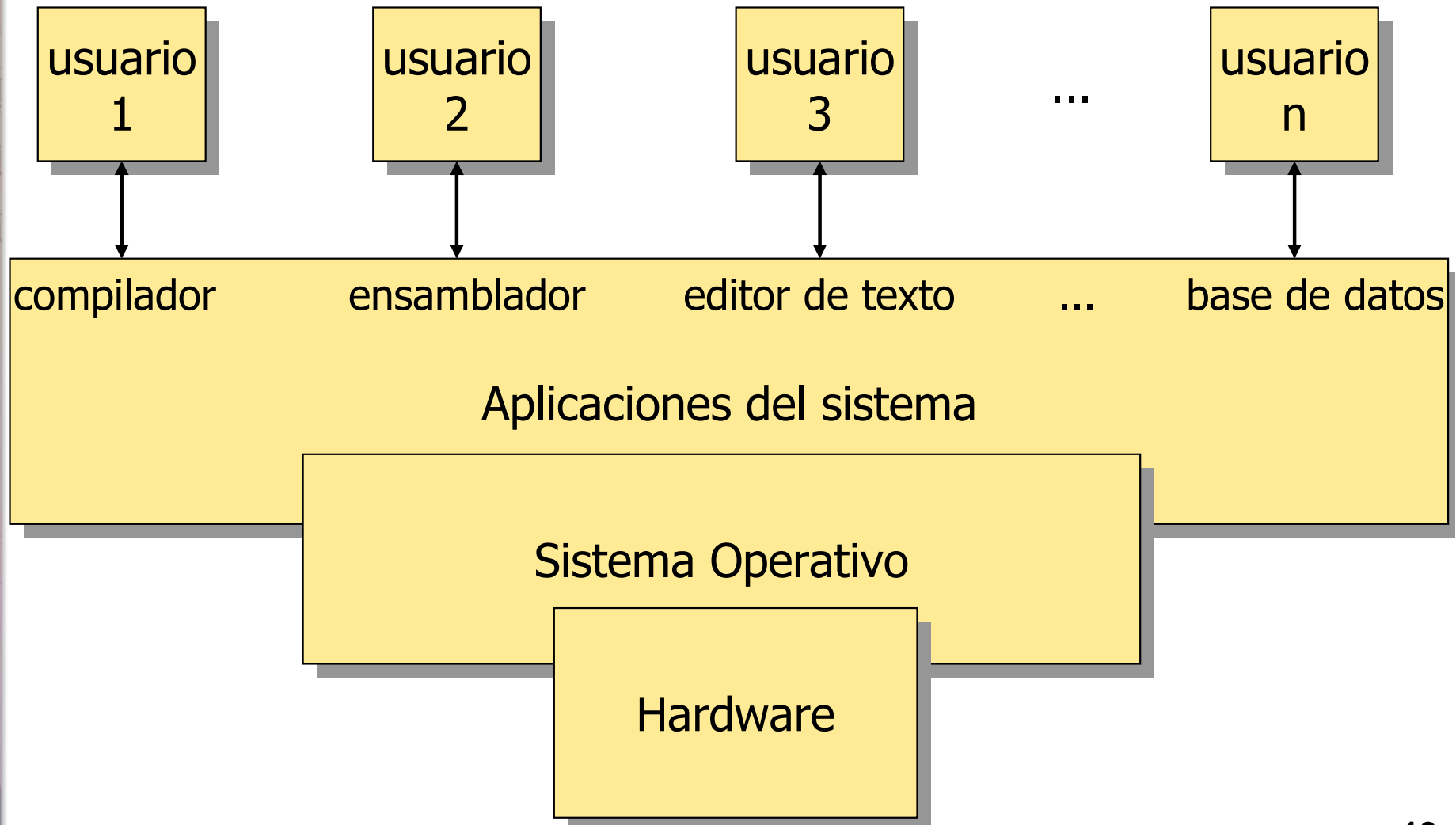


# Introducción

## Vista Abstracta de los Componentes del Sistema



# Visión abstracta del sistema



# Definiciones de Sistema Operativo

- **Administrador de recursos** – gestiona y asigna los recursos de la computadora.
- **Programa de Control** – controla la ejecución de los programas del usuario y las operaciones de los dispositivos de E/S.
- **Kernel o núcleo** – es el programa que reside en memoria y corre en todo momento.
- **Máquina virtual** – extiende y enmascara la funcionalidad del hardware.

# Historia de los Sistemas Operativos (1)

Se asocia al avance tecnológico.

## **Primera generación (1945-1955):**

Tubos de Vacío y Tableros Programables.

Sistemas operativos y lenguajes de programación eran desconocidos. En general resolvía problemas de cálculo directo (tablas de senos, cosenos y logaritmos) y se programaban por medio de tableros.

En los primeros años de los 50 la rutina fue una mejora con la introducción de la tarjeta perforada.

# Historia de los Sistemas Operativos (2)

## Segunda Generación (1955-1965):

Transistores y Sistemas Batch.

Con la introducción del transistor las computadoras fueron mas confiables. Por primera vez hay una clara separación entre diseñadores, constructores, programadores y personal de mantenimiento.

Estas computadoras, *mainframes*, se ubican en lugares especiales con aire acondicionado con staff de operadores profesional. Los costos eran multimillonarios.

Sistemas Batch con tarjetas.

Lenguajes: Assembler y FORTRAN.

Sistemas Operativos: FMS (FORTRAN Monitor Sys.).<sup>13</sup>

# Historia de los Sistemas Operativos (3)

## Tercera Generación (1965-1980):

Circuitos Integrados y Multiprogramación.

Aparece un complejo sistema operativo OS/360.

Se desarrollan técnicas como:

- Multiprogramación.
- Spooling (de *Simultaneous Peripheral Operations On Line*).
- Tiempo Compartido.

Surgen sistemas operativos como CTSS (*Compatible Time Sharing System*), MULTICS, UNIX (BSD y System V).

# Historia de los Sistemas Operativos (4)

## Cuarta Generación (1980-Presente):

Computadoras Personales (PC).

Aparecen las computadoras personales (**microcomputadoras**) por el desarrollo de circuitos LSI (*Large Scale Integration*).

Aparece el primer sistema operativo (monitor?) sobre estas microcomputadoras: **CP/M** (*Control Program for Microcomputer*). Luego para la PC IBM desarrolla MS-DOS.

Aparecen las pantallas gráficas - **GUI** (*Graphical User Interface*).

M\$ Windows (Windows 2k, XP, Vista, 7).



# Historia de los Sistemas Operativos (5)

## Cuarta Generación (1980-Presente) (Cont.):

Computadoras Personales (PC).

UNIX soporta **X Window System** (*display, teclado, pointing devices*), sobre el cual corren los *desktop environments* (KDE, gnome, xfce, etc).

\*BSD, GNU/Linux.

Mac OS.

A mediados de los '80 aparecen sistemas operativos de red y sistemas operativos distribuidos.

# Sistemas *Batch* Simples (1)

- Requiere un operador.
- Usuario  $\neq$  operador.
- Agrega un lector de tarjetas.
- Reduce el tiempo inicial loteando jobs similares.

# Sistemas *Batch* Simples (2)

- Secuenciamiento automático de *jobs* – transfiere automáticamente el control de un job a otro. Primer sistema operativo rudimentario.
- Monitor residente
  - control inicial en el monitor.
  - transfiere el control a un job.
  - cuando el job se completa el control se transfiere al monitor.

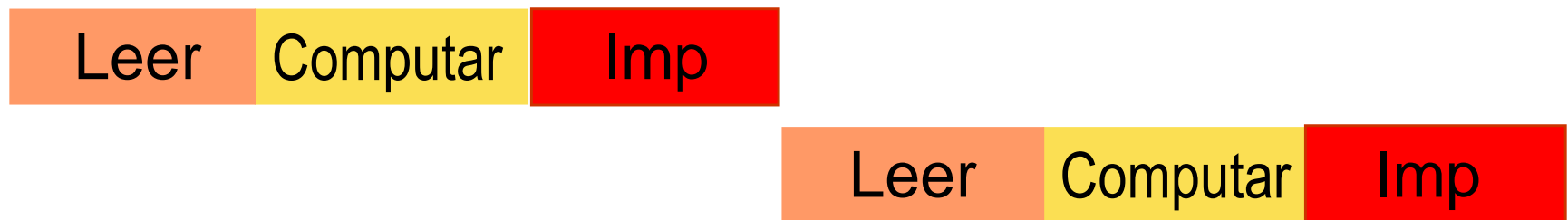
# Distribución de memoria

## Distribución de la Memoria para un Sistema Batch Simple

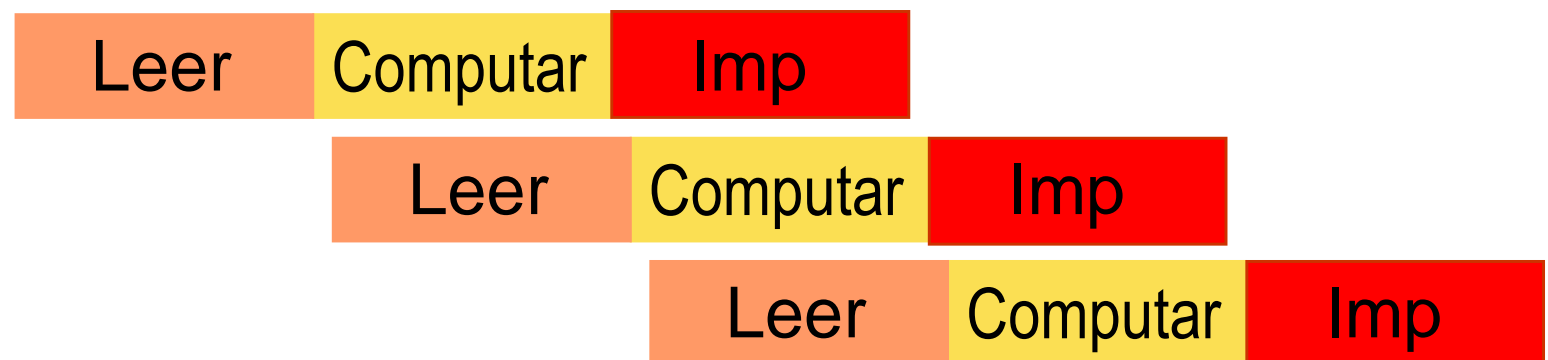


# Sistemas Batch

## ◆ Sistemas Batch Simples

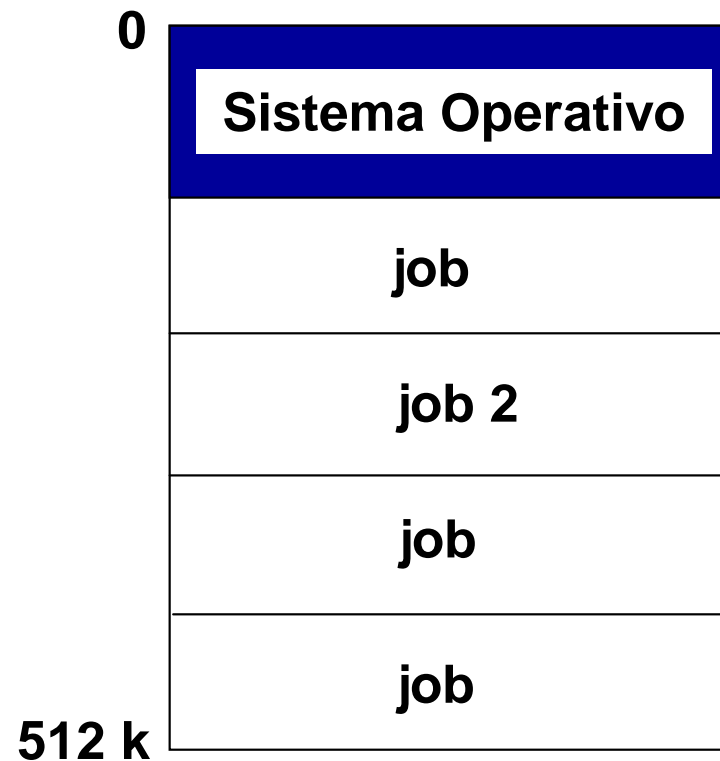


## ◆ Sistemas Batch Simples con *spooling*



# Sistemas Batch Multiprogramados

Varios jobs se mantienen en memoria principal al mismo tiempo y el CPU se alterna entre ellos.



# Multiprogramación

## Características Necesarias para la Multiprogramación

- **Mecanismos de E/S.**
- **Manejo de Memoria** – el sistema debe alojar la memoria a varios jobs.
- **Planificación de CPU** – el sistema debe seleccionar un job entre los que están listos para ejecutar.
- **Asignación de Dispositivos.**



# Tiempo Compartido (1)

## Sistemas de Tiempo Compartido Computación Interactiva (1)

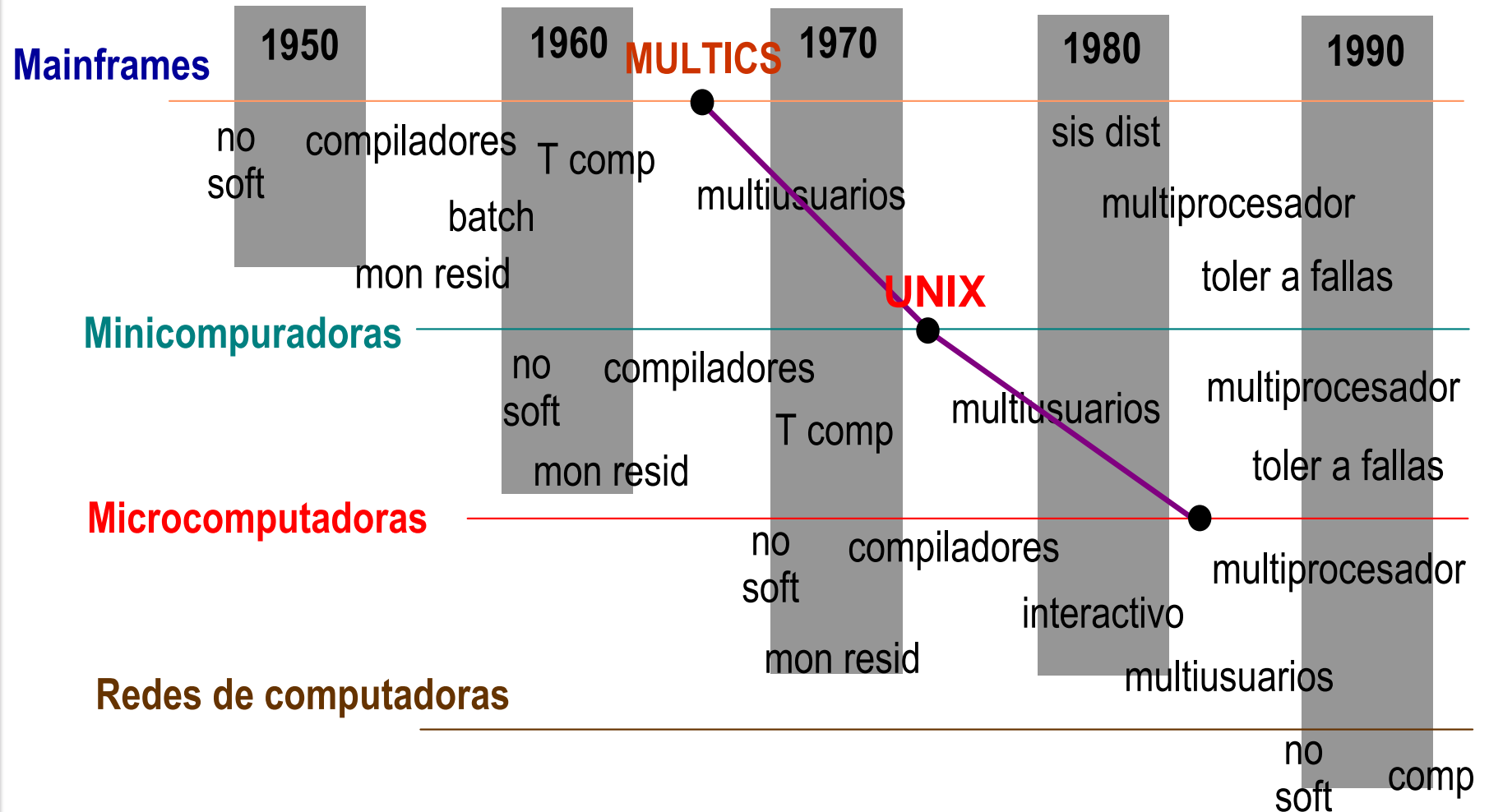
- El CPU se comparte entre varios jobs que se mantienen en memoria (RAM) y en disco (el CPU se asigna a un job sólo si el job está en memoria).
- Un job se intercambia entre la memoria y el disco.

# Tiempo Compartido (2)

## Sistemas de Tiempo Compartido Computación Interactiva (2)

- Se provee una comunicación entre el usuario y el sistema; cuando el SO finaliza la ejecución de un comando, busca la próxima *sentencia de control*, no de la lectora de tarjetas sino de teclado del usuario.
- El sistema está “*en línea*” disponible para que los usuarios accedan a

# Evolución de los Conceptos de los SO



# Sistemas de Computadoras Personales

- **Computadoras Personales** – sistema de cómputo dedicado a un único usuario.
- **Dispositivos de E/S** – teclados, mouses, monitores, impresoras, etc.
- Conveniencia para el usuario y respuesta.
- Pueden adoptarse tecnologías desarrolladas para SO grandes.
- No necesitan características especiales de protección dado que son utilizadas por usuarios únicos. Esto hoy no es tan así...

# Sistemas de Tiempo Real

- Son usados frecuentemente como dispositivos de control en una aplicación dedicada como control de experimentos científicos, sistemas de imágenes médicas, sistemas de control industrial, etc.
- Restricciones de tiempo bien definidas.
- *Sistemas de tiempo real duros.*
  - Almacenaje secundario limitado o ausente, datos almacenados en memoria de corto término/ROM.
  - Conflictos con sistemas de tiempo compartido, no soportado por SO de propósito general.
- *Sistemas de tiempo real blandos*
  - Limitada utilidad en control industrial o robótica
  - Útil en aplicaciones (multimedia, realidad virtual) que requieren características avanzadas de SO.

# Sistemas Paralelos (1)

- Sistemas Multiprocesador (varios CPUs intercomunicados).
- *Sistemas fuertemente acoplados* – los procesadores comparten memoria y reloj; se comunican por medio de la memoria compartida.
- Ventajas de un sistema paralelo:
  - Incrementa el *procesamiento total*.
  - Es mas económico.
  - Incrementa la confiabilidad.
    - degradación suave.
    - software de sistemas a prueba de fallos.



# Sistemas Paralelos (2)

- *Multiprocesamiento Simétrico*
  - Cada procesador ejecuta una copia idéntica del SO.
  - Muchos procesos pueden ejecutarse a la vez sin deterioro del rendimiento.
- *Multiprocesamiento Asimétrico*
  - Cada procesador es asignado a una tarea específica; el procesador maestro planifica y asigna trabajo a procesadores esclavos.
  - Muy común en sistemas grandes.



# Sistemas Distribuidos

- Distribuye la computación entre varios procesadores físicos.
- ***Sistema débilmente o flojamente acoplado***  
Cada procesador tiene su propia memoria local; los procesadores se comunican unos con otro por medio de variadas líneas de comunicación, como canales de alta velocidad o incluso líneas telefónicas.
- Ventajas de los sistemas distribuidos:
  - Comparten recursos.
  - Acelera la computación por carga compartida.
  - Confiabilidad.
  - Comunicaciones.

**Coming  
Next**

**Estructura de  
los Sistemas  
de Cómputo**

