

# Sistemas Operativos

Algo de Sistemas Operativos,  
Arquitecturas y Paralelismo

# ¿Qué es un Sistema Operativo?

---

- **Administra recursos**
- Arma de una máquina virtual
- Comunicación con el usuario
- Facilita el uso (herramientas, bibliotecas, etc.)

# Funciones de un Sistema Operativo

---

Secuenciar las tareas

Interpretar un lenguaje de control y comandos

Administrar errores

Administrar las interrupciones

Scheduling

Proteger (administrar recursos)

Comunicación con el usuario

# Características de un Sistema Operativo

---

- Concurrencia
- Uso compartido de recursos
- Almacenamiento a largo plazo
- Determinismo VS Indeterminismo
- Eficiente – Fiable
- Facilidad de corrección
- Tamaño Pequeño

# Repaso Vocabulario

- Monoprocesador

Monoprogramación o Monotarea  
(DOS – [trick TSR])

Multiprogramación o Multitarea

Batch - Time-Sharing

Interactivo

(Terminales Bobas o Inteligentes)

# Repaso Vocabulario

Multiprogramación o Multitarea

UNIX (en todos sus sabores)

Linux – IRIX – AIX – BSD – Solaris – etc.

Windows (en casi todos sus sabores)

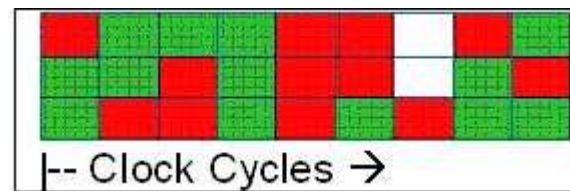
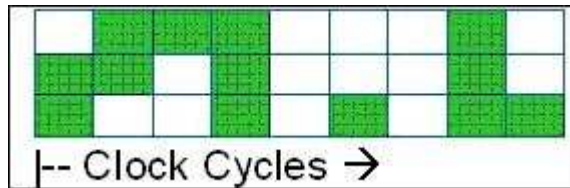
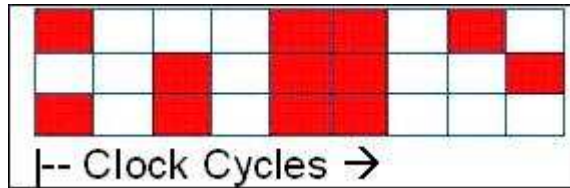
(Terminal Service)

zVM zOS

VMS

- Hyper Threading (HT) ¿?

# Hyper Threading

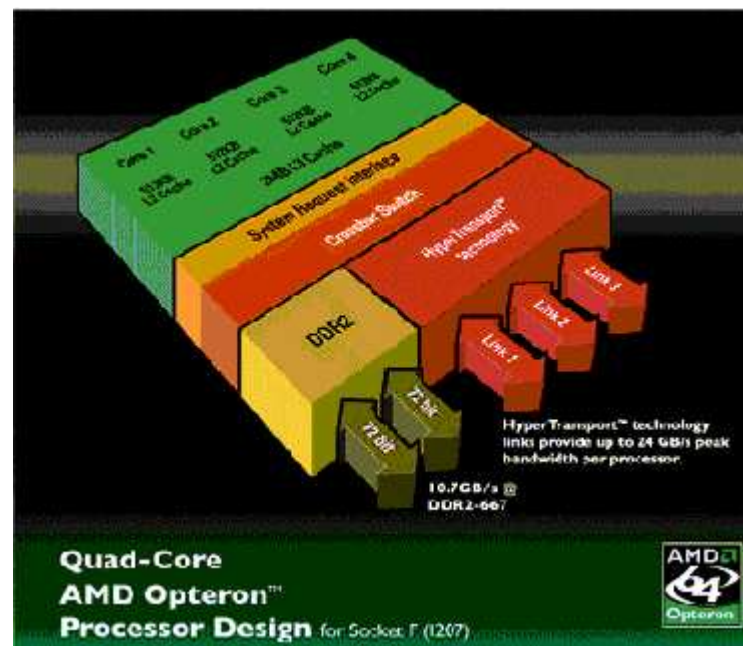


# Repaso Vocabulario

- Multiprocesador  
Varios Procesadores  
Independientes o  
Independientes con  
Cores (Dual Quad Octo etc.)  
→ Multiprogramación  
SMP vs MP

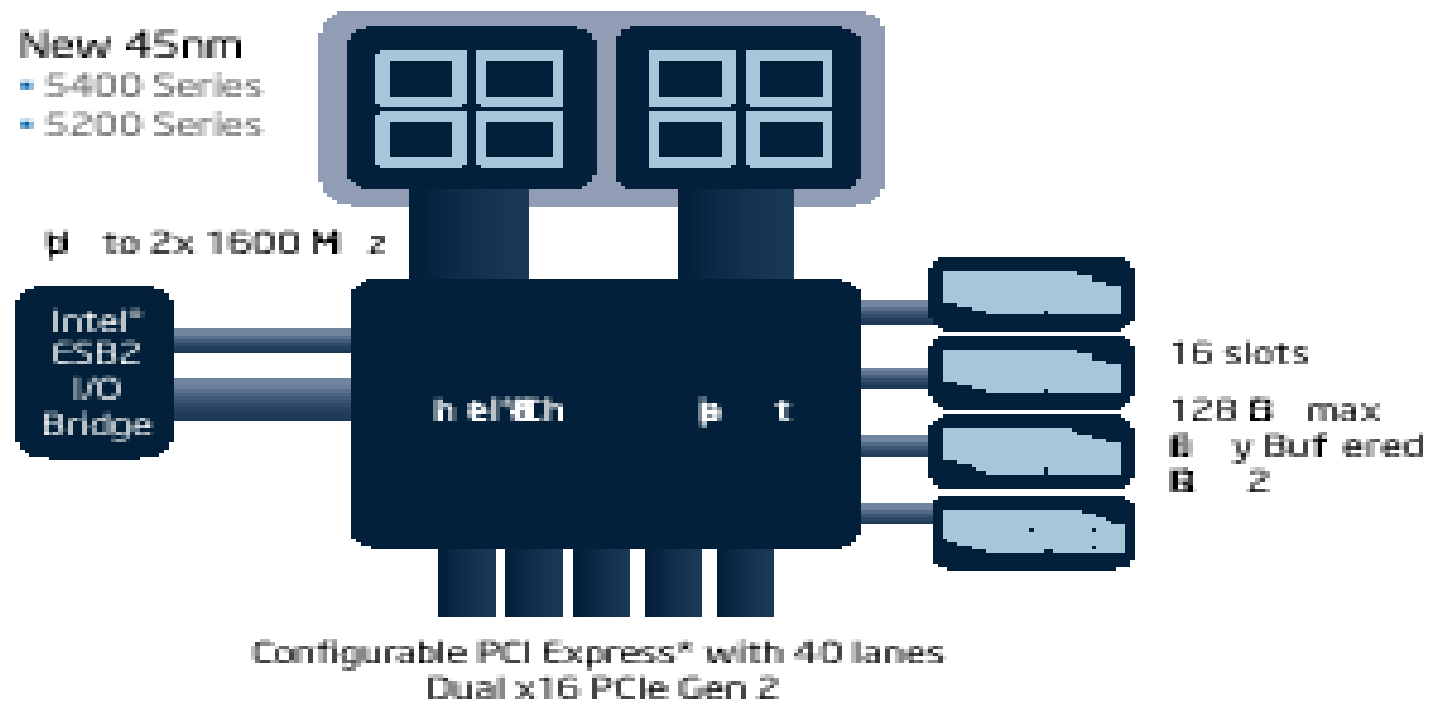


# Cores

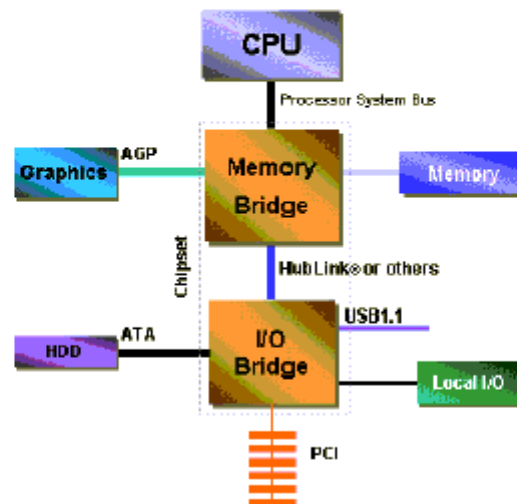


# Cores

## Quad-Core or Dual-Core

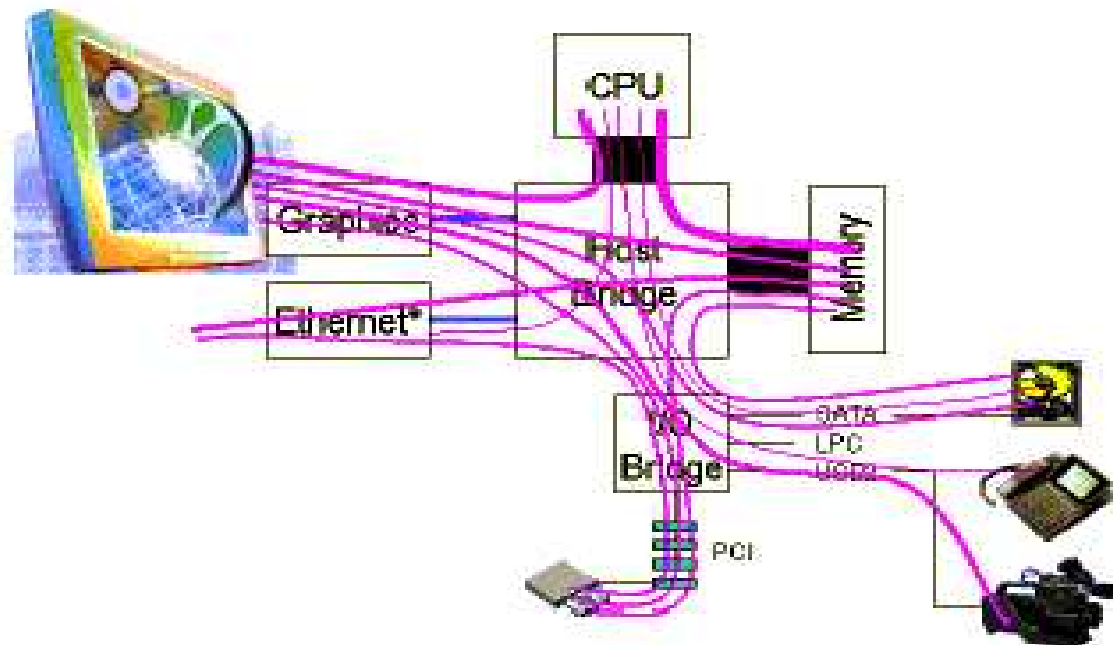


# PCIs



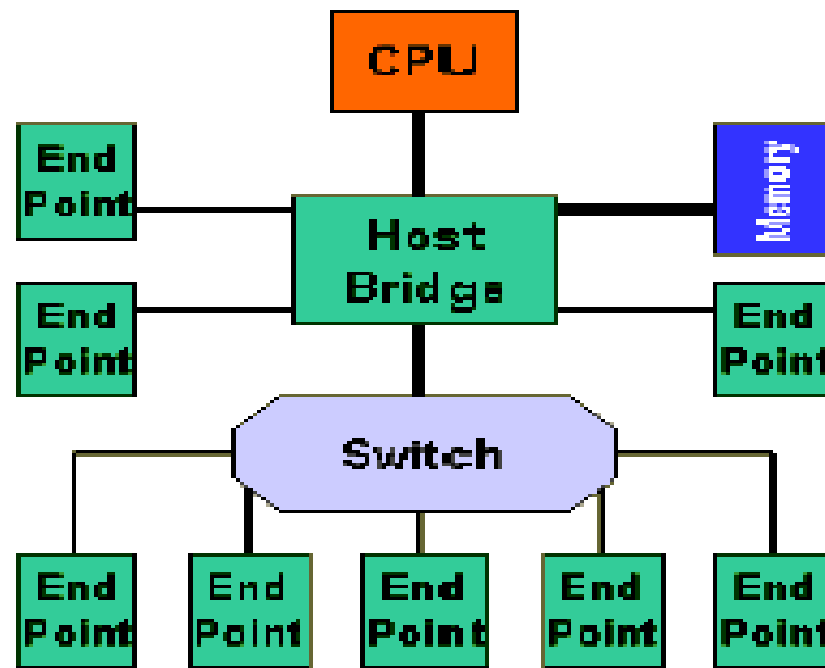
*Today's PC has multiple local buses with different requirements*

# PCIs



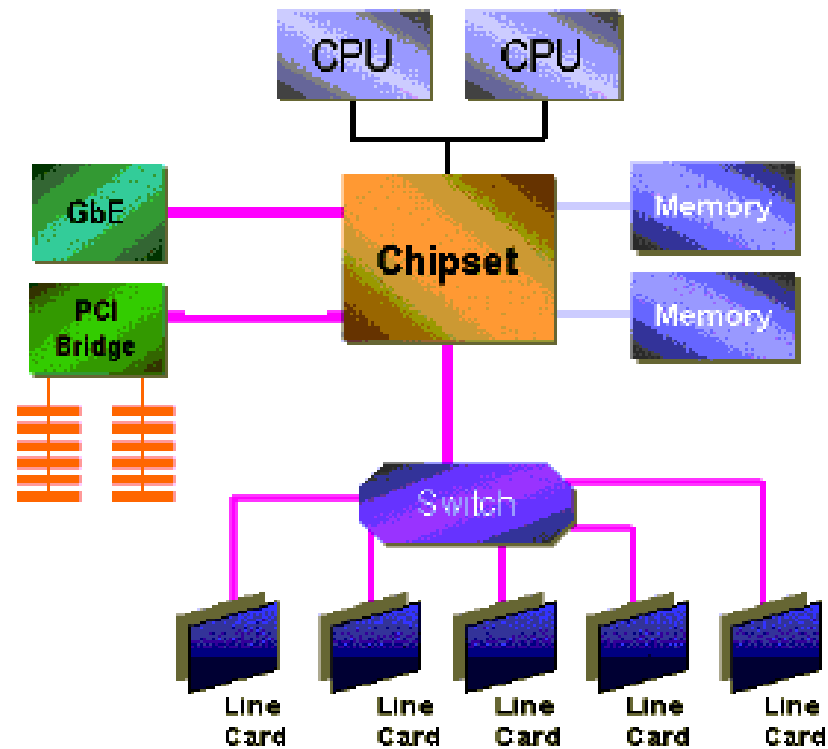
*Multiple concurrent data transfers.*

# Mejora de Comunicación (Switch)



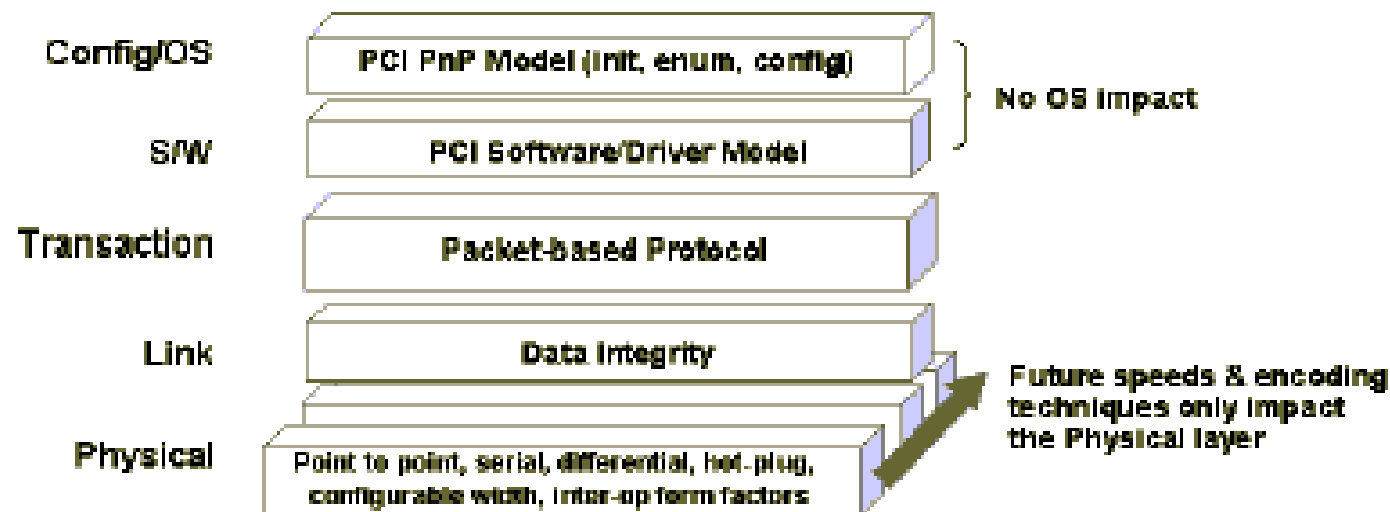
*A switch is added to the system topology*

# PCI Express



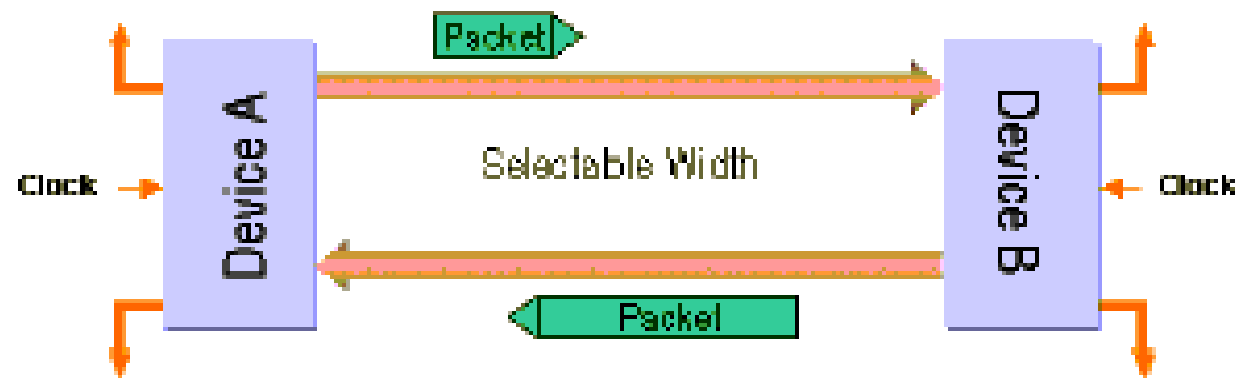
*PCI Express-based Networking Communications System*

# PCI Express Layers



*The PCI Express Architecture is specified in layers*

# PCI Express



*PCI Express link uses transmit and receive signal pairs*



# Usa más caminos

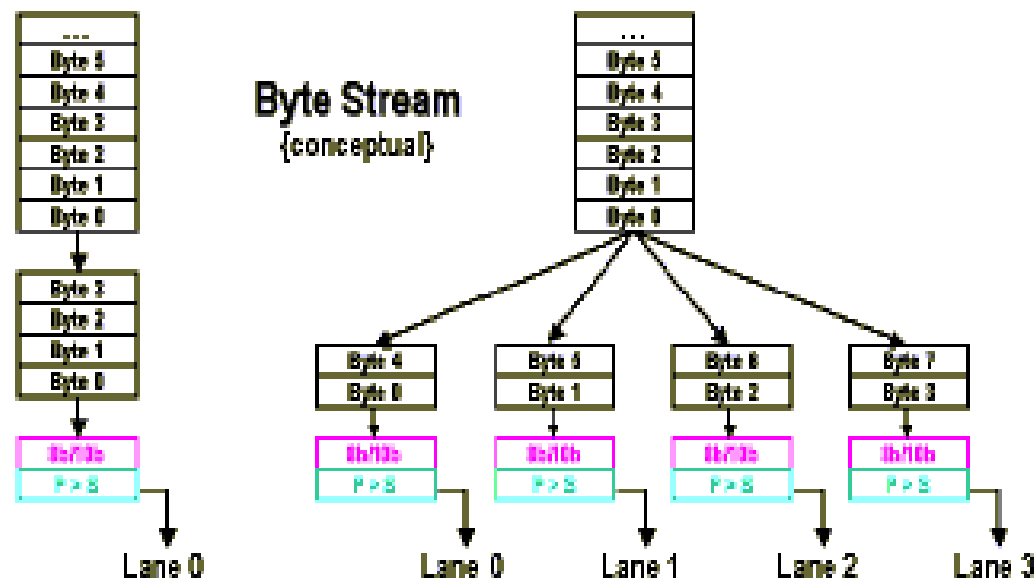
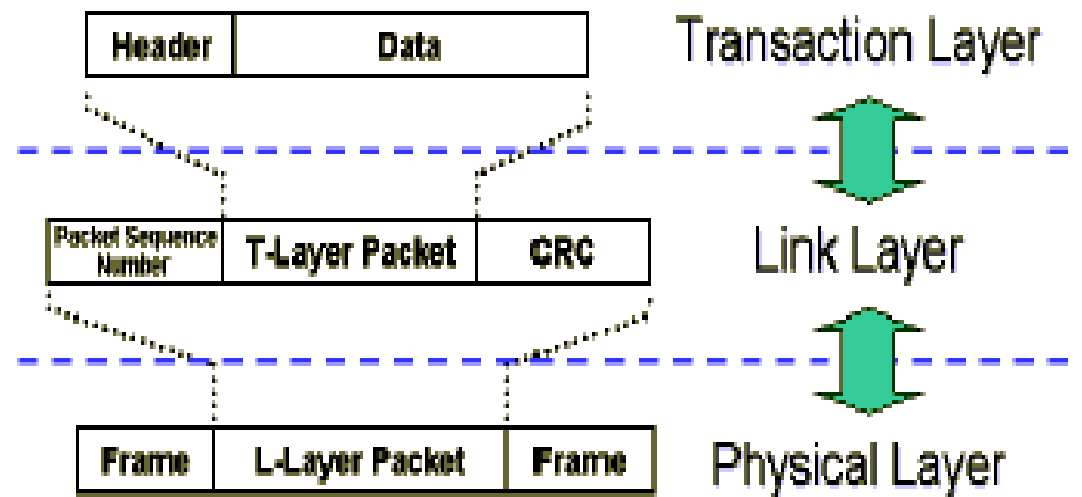


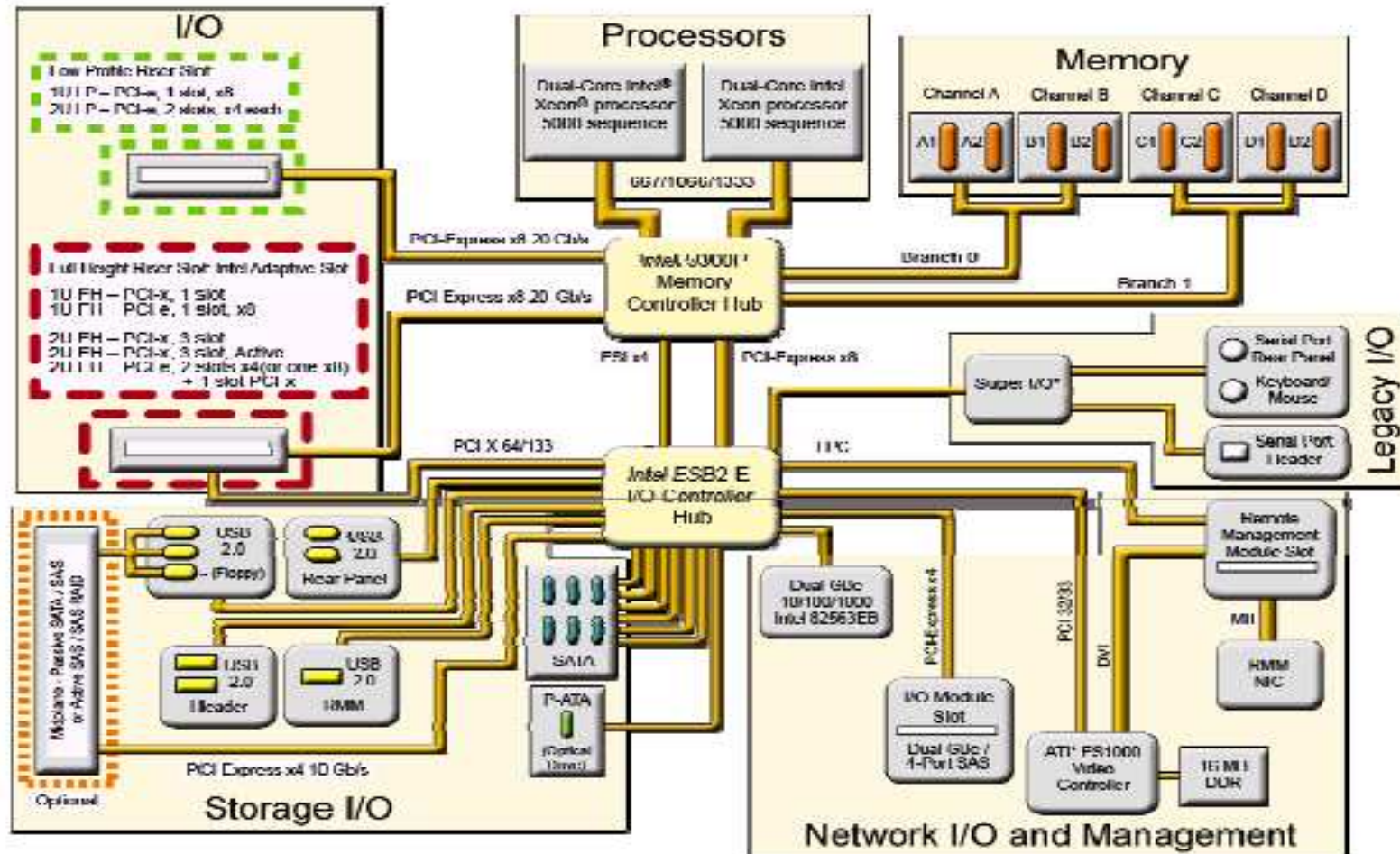
Figure 9. A PCI Express Link consists of one or more lanes

# PCI Express Link Layer



*The Link Layer adds data integrity features*

# INTEL S5000PAL



# Repaso Vocabulario

- Multiprocesador  
Memoria Única  
(Fuertemente Acoplado)

Memoria Distribuida  
(Debilmente Acoplado) (en Red)

Sistemas Distribuidos o Multicomputadoras

# Repaso Vocabulario

- Memorias

UMA

NUMA

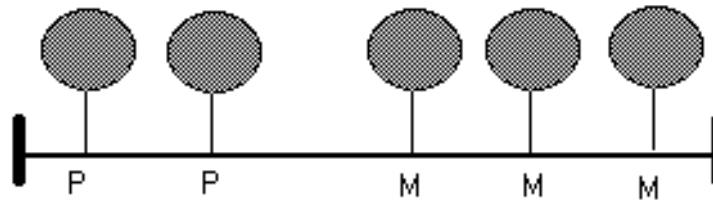
NORMA

COMA

cc-NUMA o SMP2

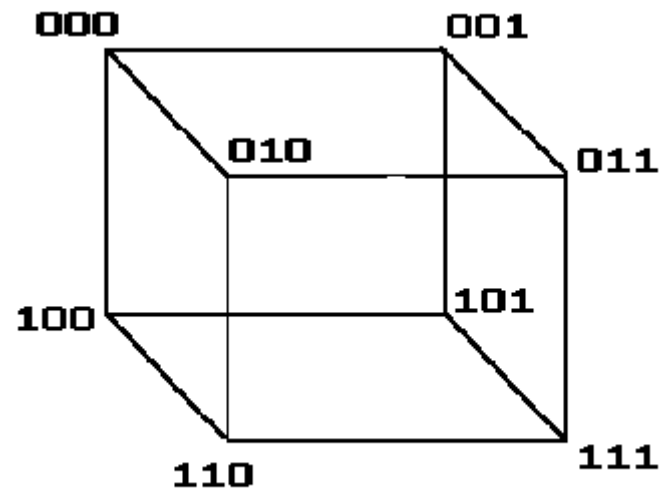
# Repaso Vocabulario

- UMA



# Repaso Vocabulario

- NUMA/NORMA/ccNUMA



N-cubo de grado 3.

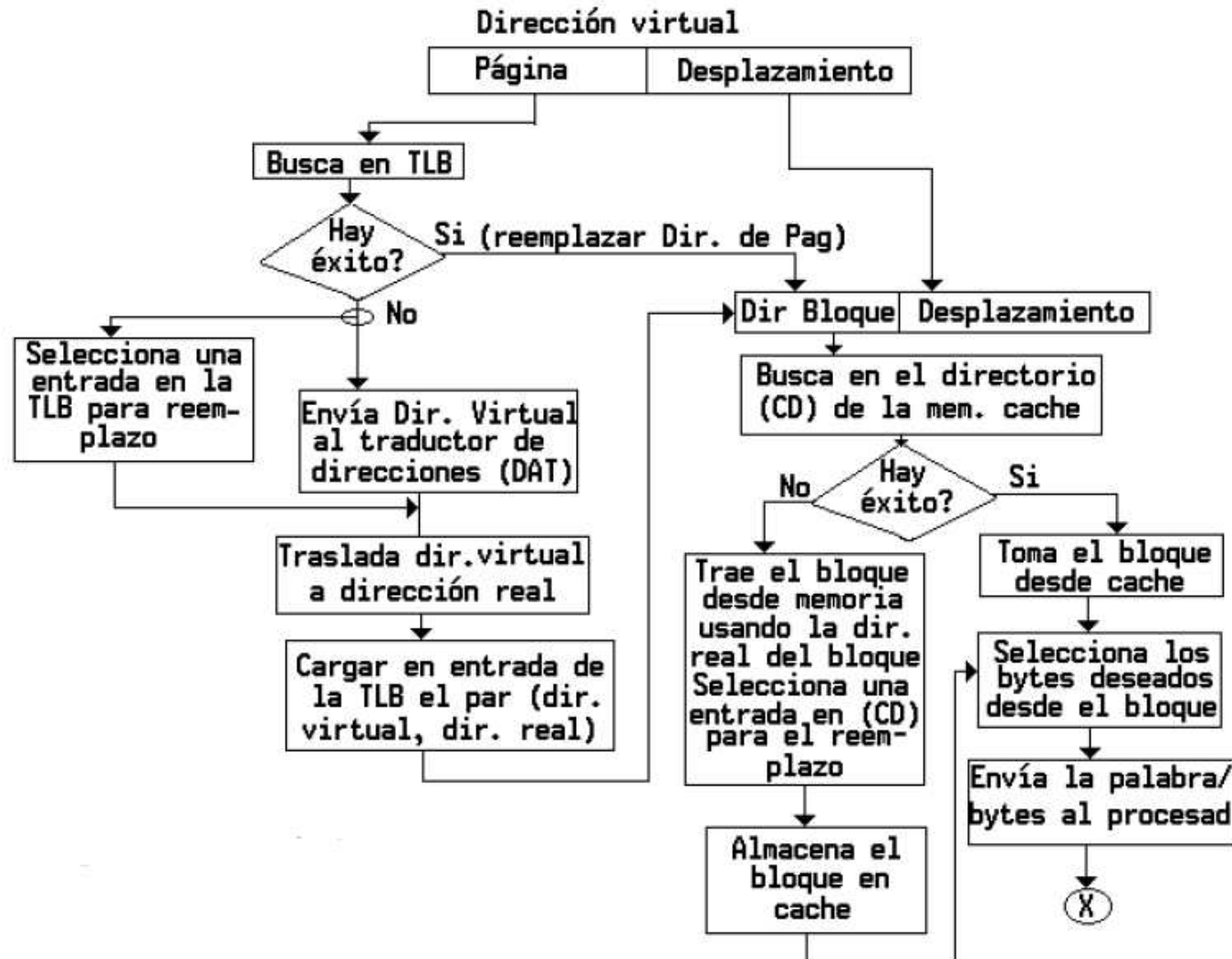
# Repaso Vocabulario

- CACHE (bus compartido)  
Snoopy - Write-through  
Consistencia  
00 Limpia  
01 Compartida  
10 Sucia por este procesador  
11 Inválida en este procesador

¿Qué pasa sino se comparte bus?



# Repaso Vocabulario: Memoria Cache (TLB)



# Repaso Vocabulario

- Sistemas Distribuidos (sirven para ...)

Cooperar (MPI, PVM, DSM, etc.)

Compartir (archivos, directorios, etc.)

Servidores (Modelo Cliente/Servidor  
RPC,)

Clusters → GRID

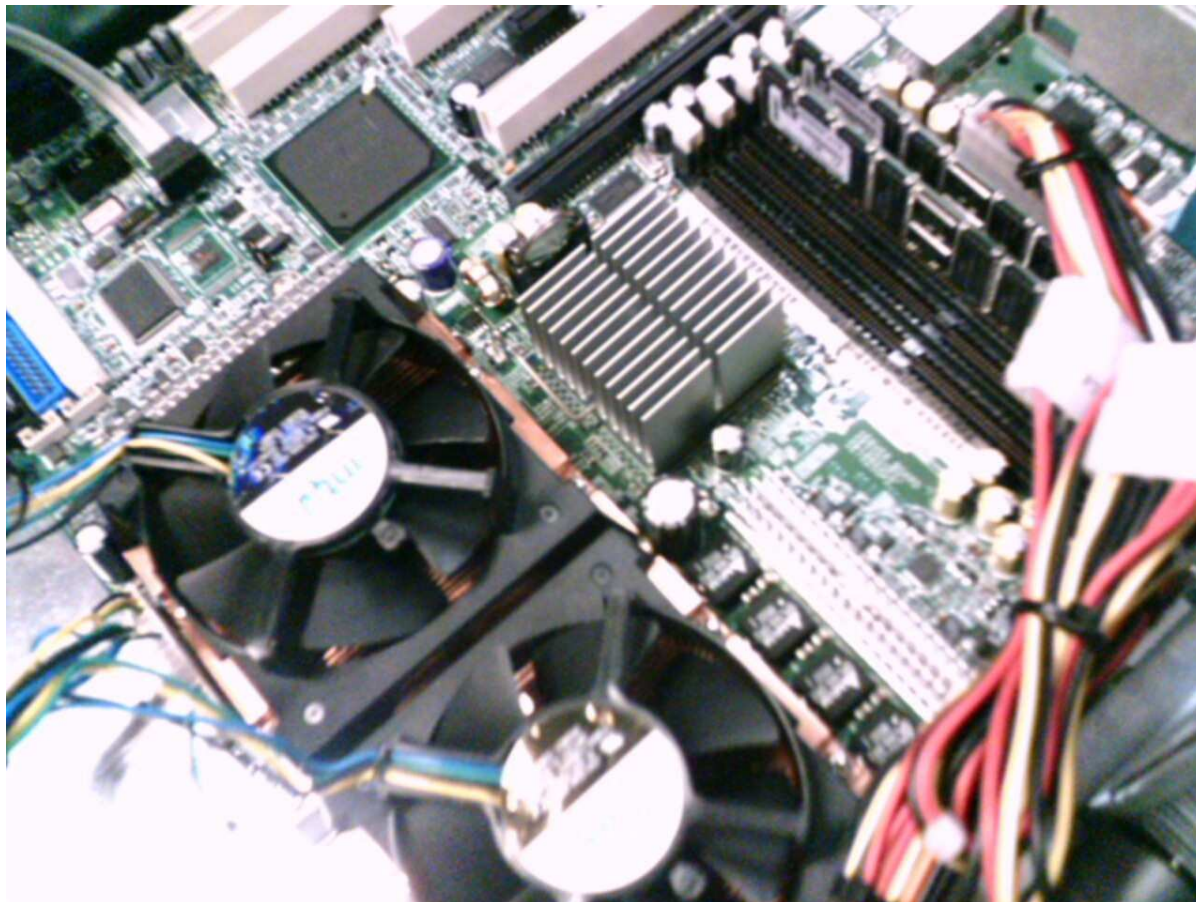
# Cluster

- Sheldon



# Cluster

- Nodo

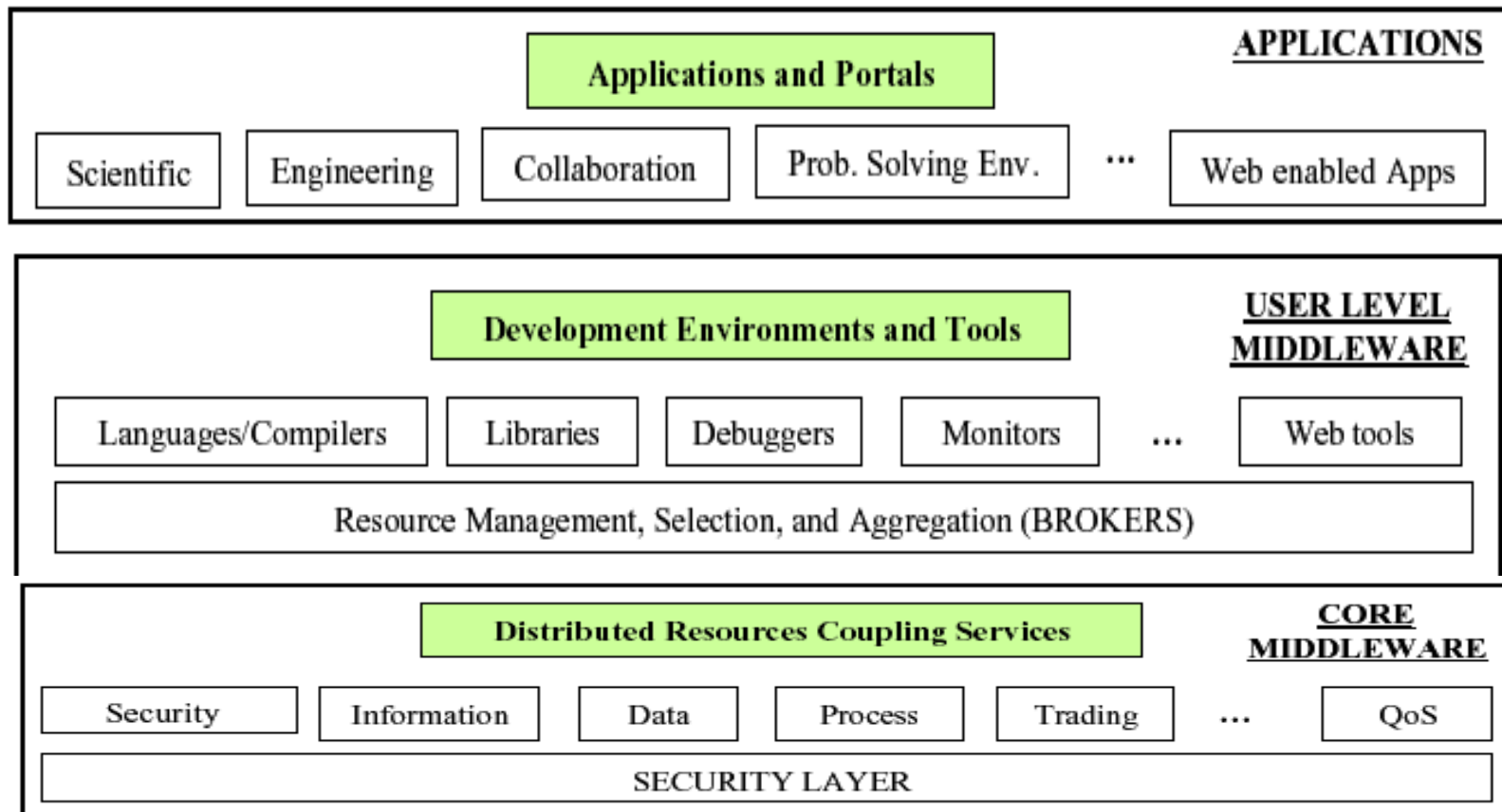




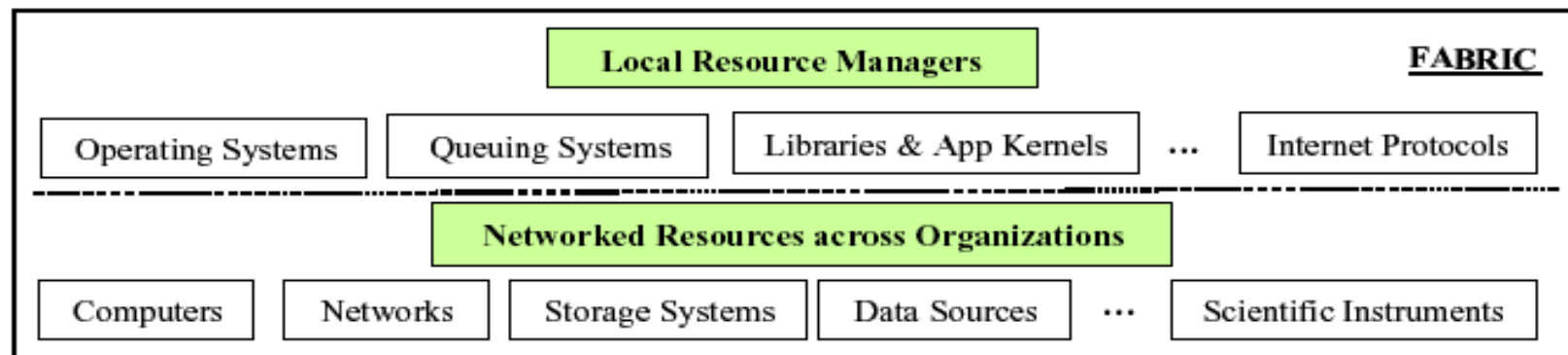
# Cluster



# GRID



# GRID



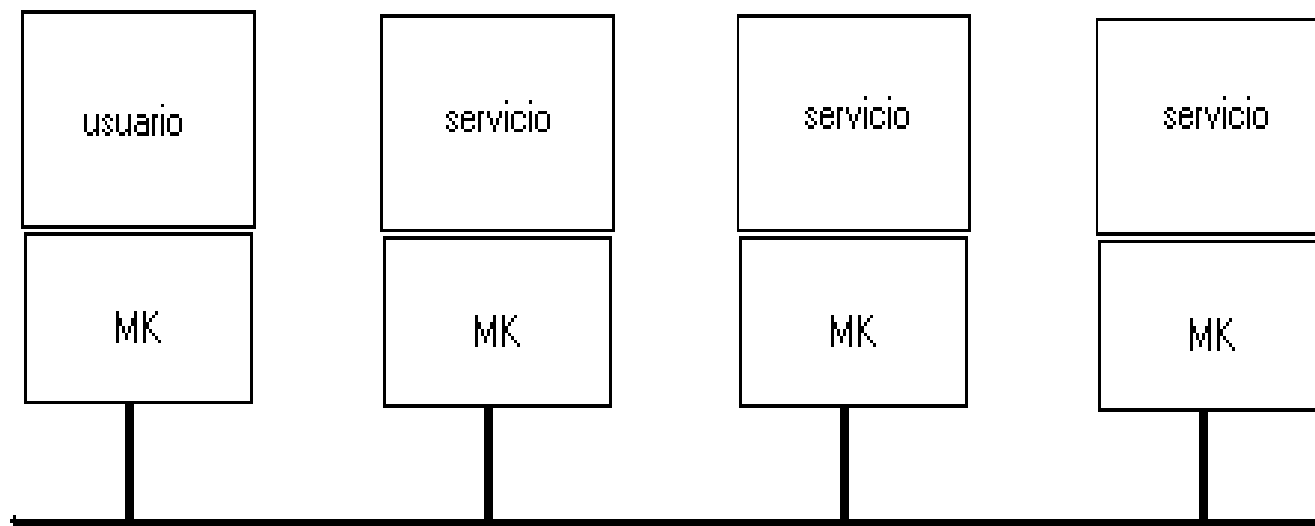
# Repaso Vocabulario

- ... Y que pasa con Sistemas Operativos?  
Monolíticos – Modular  
Centralizados – en Red – de Red  
Microkernels - Exokernels  
Centralizados (UNIX)  
en Red (NFS – Novell)  
de RED (SOD – basados en Microkernel)  
Microkernel (mach – chorus – ameoba)  
(WNT – AIX – IRIX basados en mach)



# Repaso Vocabulario

- Microkernel



# Repaso Vocabulario

- Comparación Paradigmas SO

	<b>Sistemas Operativos Mltiprocador</b>	<b>Sistemas Operativos en Red</b>	<b>Sistemas Operativos de Red (Distribuido)</b>
<b>Se ve como Uniprocador Virtual</b>	<b>Si</b>	<b>NO</b>	<b>Si</b>
<b>Igual SO</b>	<b>Si</b>	<b>No</b>	<b>Si</b>
<b># Copias</b>	<b>1</b>	<b>n</b>	<b>1/n</b>
<b>Comunicación</b>	<b>Memoria</b>	<b>Archivos Compartidos</b>	<b>Mensajes</b>
<b>Protocolos de Red</b>	<b>No</b>	<b>Si</b>	<b>Si</b>
<b>Única cola de Ejecución</b>	<b>Si</b>	<b>No</b>	<b>Si/No</b>

# Introducción Arquitectura

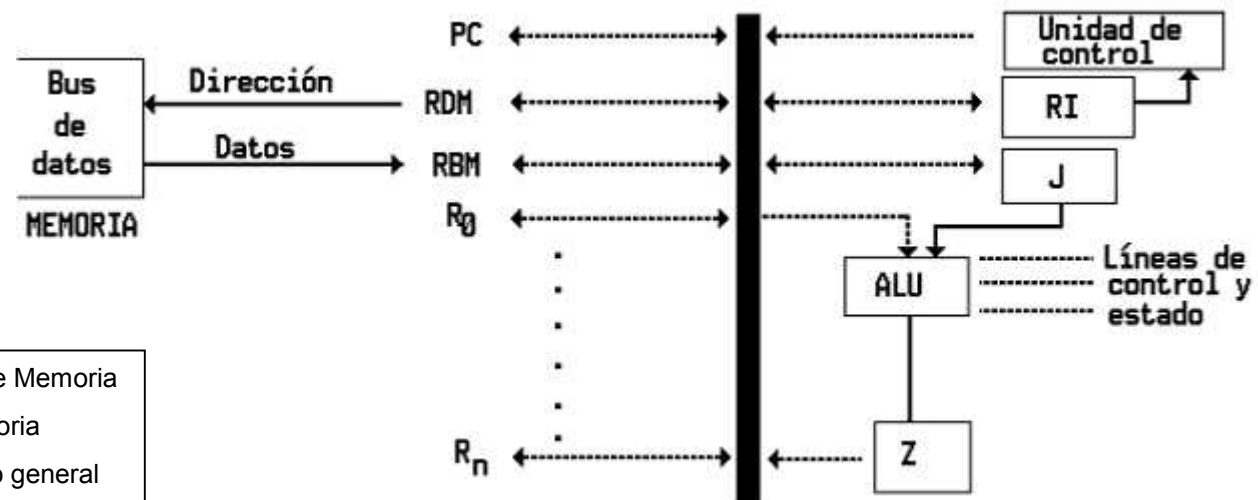
- Estructura abstracta con un set fijo de instrucciones
- Determinar componentes, funciones de los componentes y reglas de interacción entre ellos (N. Prasad – 1981)
- Otras ...
- Nos quedamos con una mezcla ...

# Introducción Arquitectura

- Definimos una Arquitectura por:
  - Componentes
  - Interconexión entre componentes
  - Interacción entre componentes
  - Implementación (materiales, con o sin pipeline, etc.)
  - Set de instrucciones

# Introducción Arquitectura

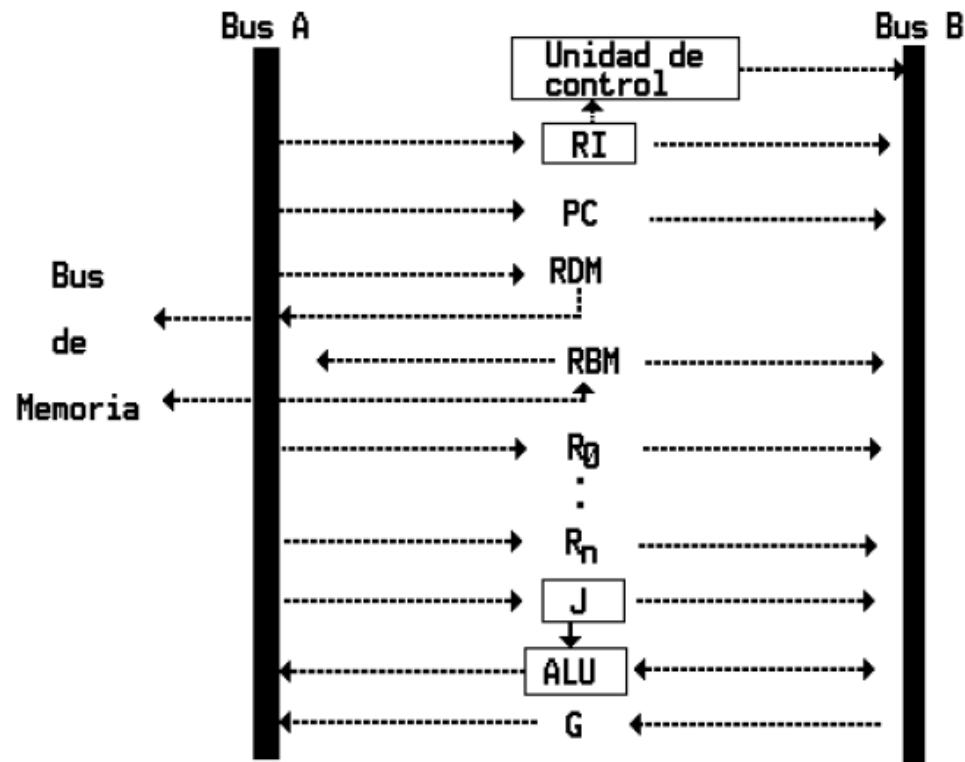
- Interconexión
  - Procesador de un Bus



RDM : Registro de dirección de Memoria  
RBM : Registro buffer de memoria  
R<sub>0</sub> a R<sub>n</sub> = son registros de uso general  
PC = Program Counter  
J = es un registro  
Z = es un registro acumulador  
RI = Registro de instrucción

# Introducción Arquitectura

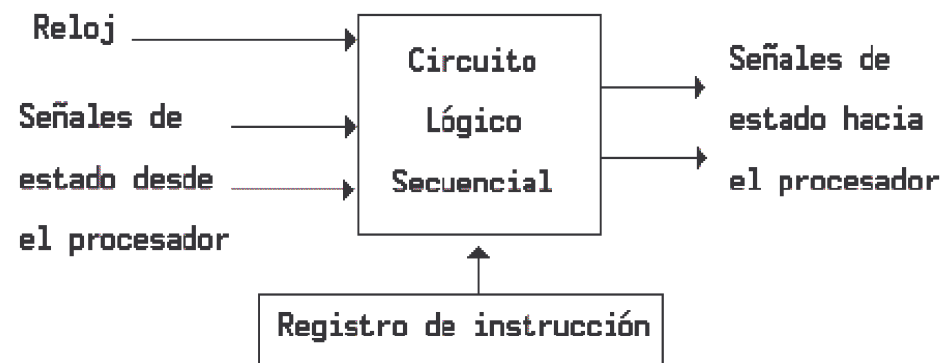
- Interconexión
  - Procesador MultiBus



# Introducción Arquitectura

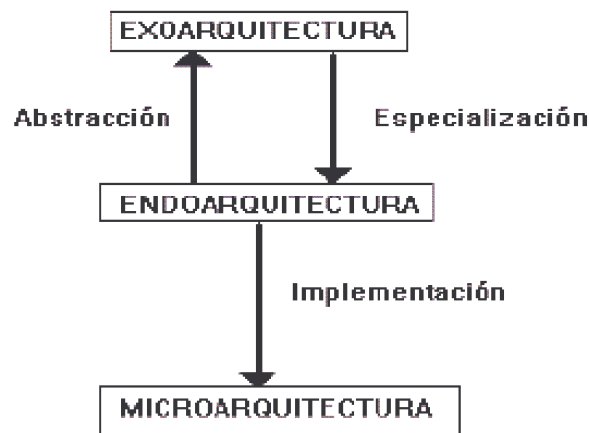
## ■ Unidad de Control

- La unidad de control deberá **no** sólo decodificar un código de operación, sino también emitir señales (códigos de condición) según se vayan ejecutando las instrucciones (overflow, carry -acarreo-, cero, negativo, etc).
- Es decir la Unidad de Control recibe de la Unidad Aritmético-Lógica señales que indican el estado que ha resultado de la ejecución de la operación indicada.



# Arquitectura

- Tomando en cuenta lo anterior se puede hablar de niveles de arquitectura:



- Exoarquitectura: Es la estructura y capacidad funcional de la arquitectura visible al programador
- Endoarquitectura: Las capacidades funcionales de los componentes físicos, las estructuras lógicas de sus interconexiones, las interacciones, los flujos de información, sus controles e implementación.
- Microarquitectura: Que componentes se abran o cierran durante la ejecución de una instrucción.



# Arquitectura Secuenciales

- Von Neuman (1946/1947)
  - Programa Almacenado (Control Flow)
  - Datos Almacenados
  - Representación binaria
  - Flujo interno paralelo (byte o palabra)
  - No paralelismo de operaciones
  - Unidades Funcionales

# Arquitecturas Secuenciales

- Unidades Funcionales
  - En el computador se encuentran Unidades Funcionales interrelacionadas que interactúan.
  - Unidades Funcionales:
    - CPU
    - Memoria
    - Sistema E/S

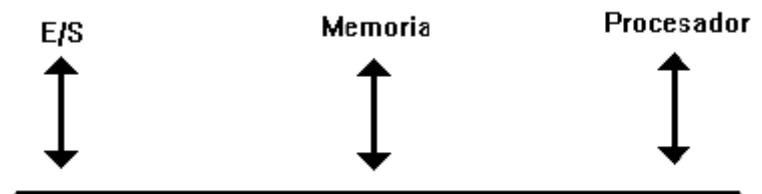
# Arquitecturas Secuenciales

## ■ Estructura Bus

- La distancia física que separa los componentes puede ser grande de manera tal que sea imposible lograr una transmisión sincrónica de la información entre ambos. La transmisión asincrónica se implementa muy frecuentemente con un mecanismo que se denomina "**handshaking**" (acuerdo).

## ■ Un Bus

- Una Transferencia x vez



# Arquitecturas Secuenciales

- 2 Buses

- Por Procesador



- Canales

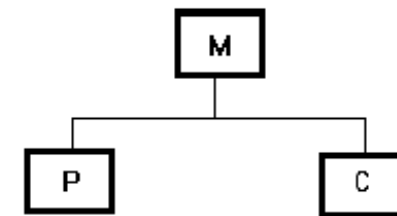
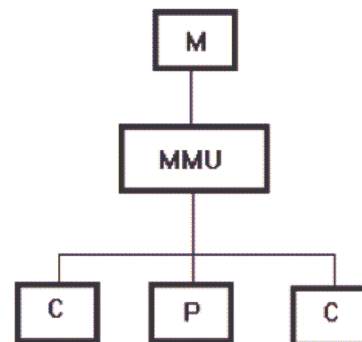
- Colisiones

- Unica Entrada

- (Robo Ciclos al Proc.)

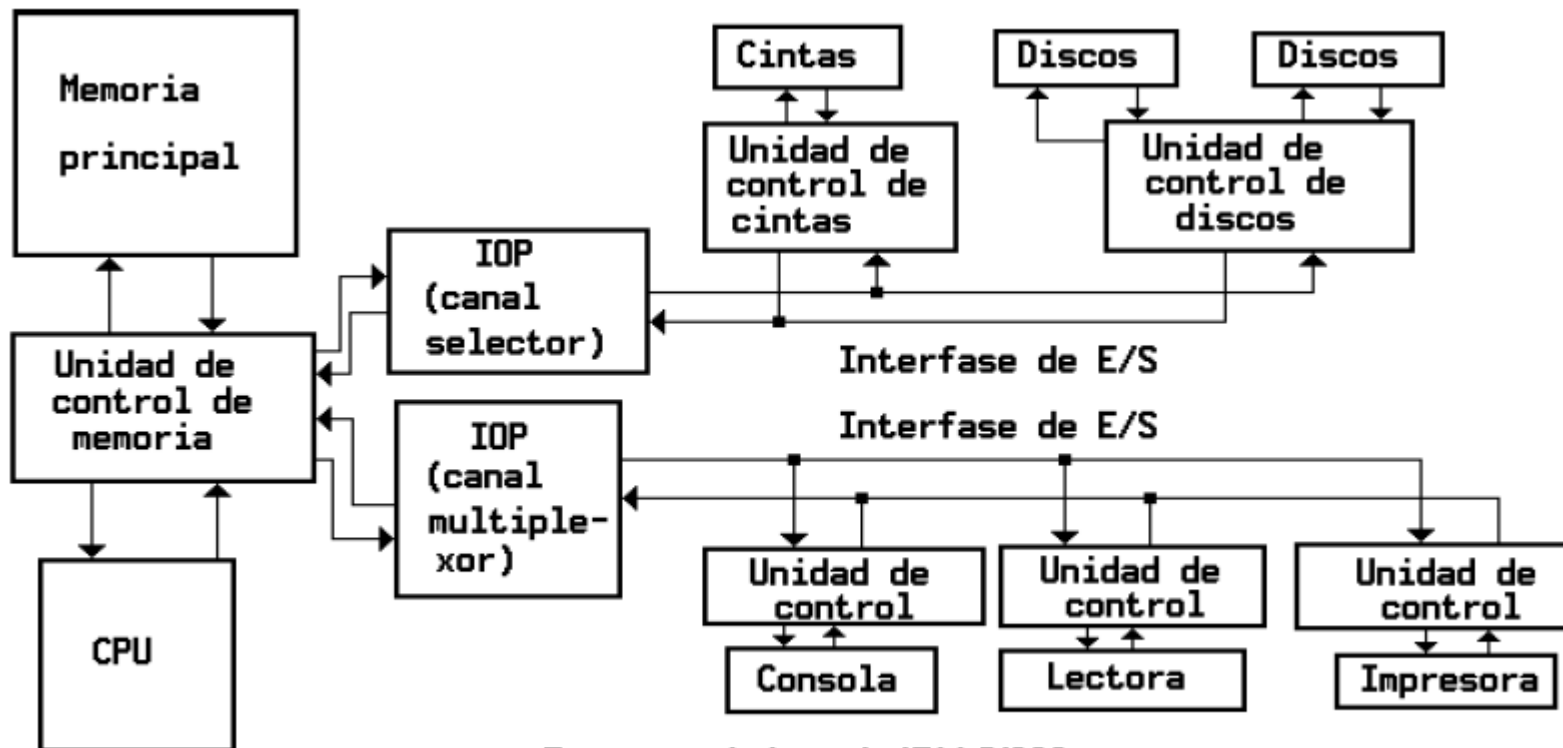
- MMU

- Arbitro



# Arquitecturas Secuenciales

## ■ Unidades Funcionales

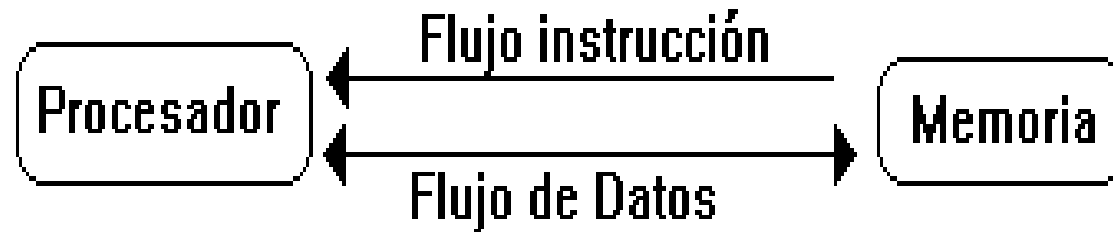


Estructura de la serie IBM S/360.

# Etapas de una Instrucción

- Generar Dir. Próxima Instrucción
- Buscar Instrucción
- Decodificar Instrucción (genera DPI)
- Generar Dir. Operandos
- Buscar Operandos
- Ejecutar Instrucción
- Almacenar Resultados

# Arquitecturas Secuenciales

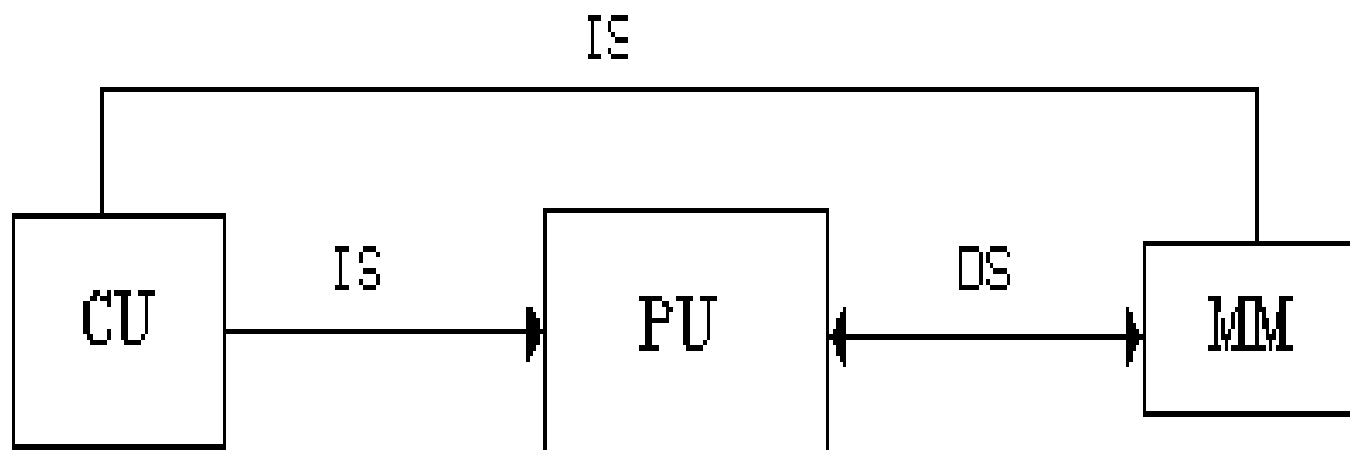


# Clasificación de Arquitecturas

- Genericamente todos los computadores tendran los componentes funcionales mencionados (CPU, Memoria y E/S) interconectados de alguna forma y la cantidad necesaria.
- Clasificación de Flynn
  - Forma de mostrar la interconexión de los componentes y sus cantidades



# Clasificación de Flynn



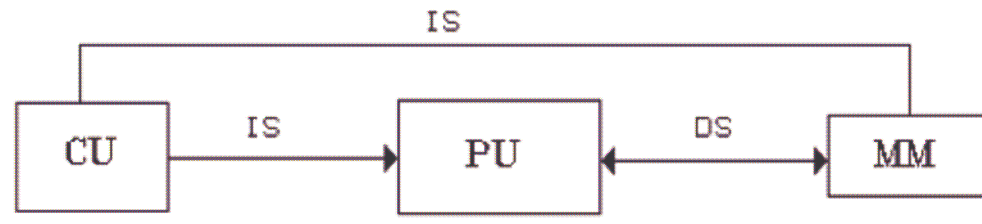
Computador SISD.

# Clasificación de Flynn

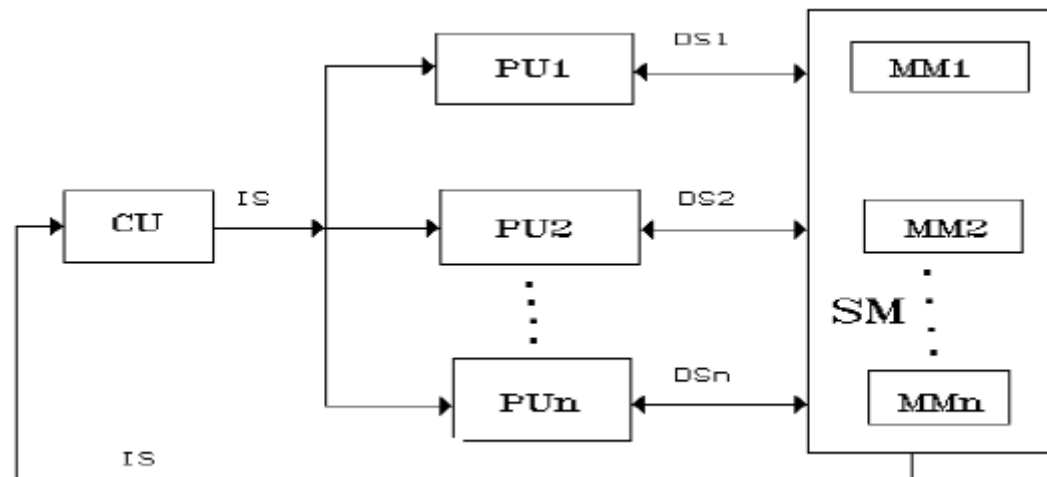
- Flujos de Datos e Instrucciones
  - Son los parámetros básicos para la clasificación
- Clasificación
  - SISD (Single Instruction Single Data)  
[Paradigma von Newman o Harvard]
  - SIMD (Single Instruction Multiple Data)
  - MISD (Multiple Instruction Single Data)
  - MIMD (Multiple Instruction Multiple Data)  
[Paradigma Paralelo]

# Clasificación Flynn (Gráficos)

## ■ SISD



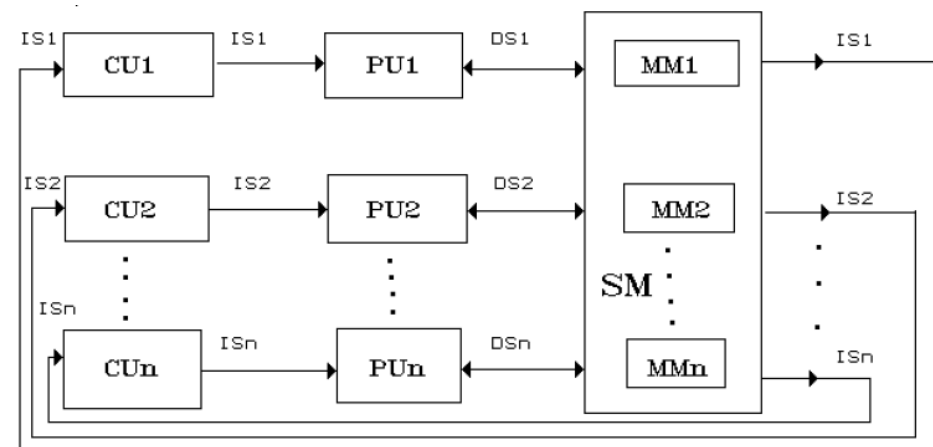
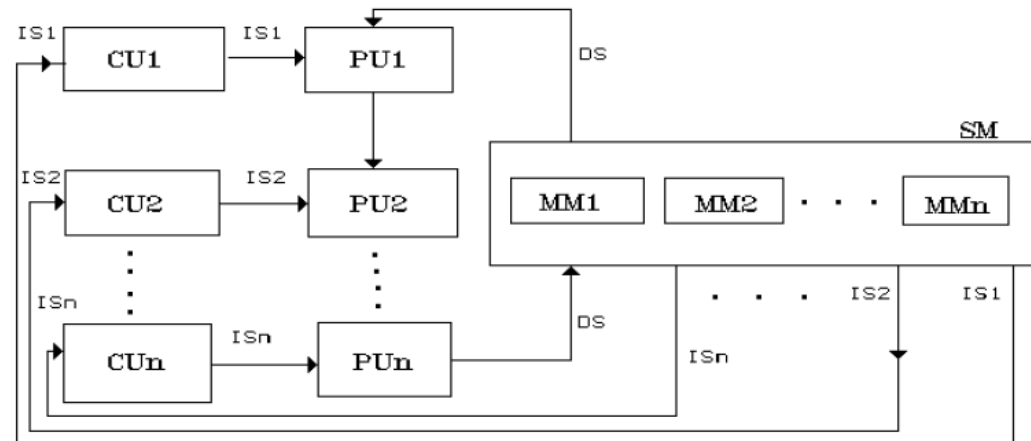
## ■ SIMD



# Clasificación Flynn (Gráficos)

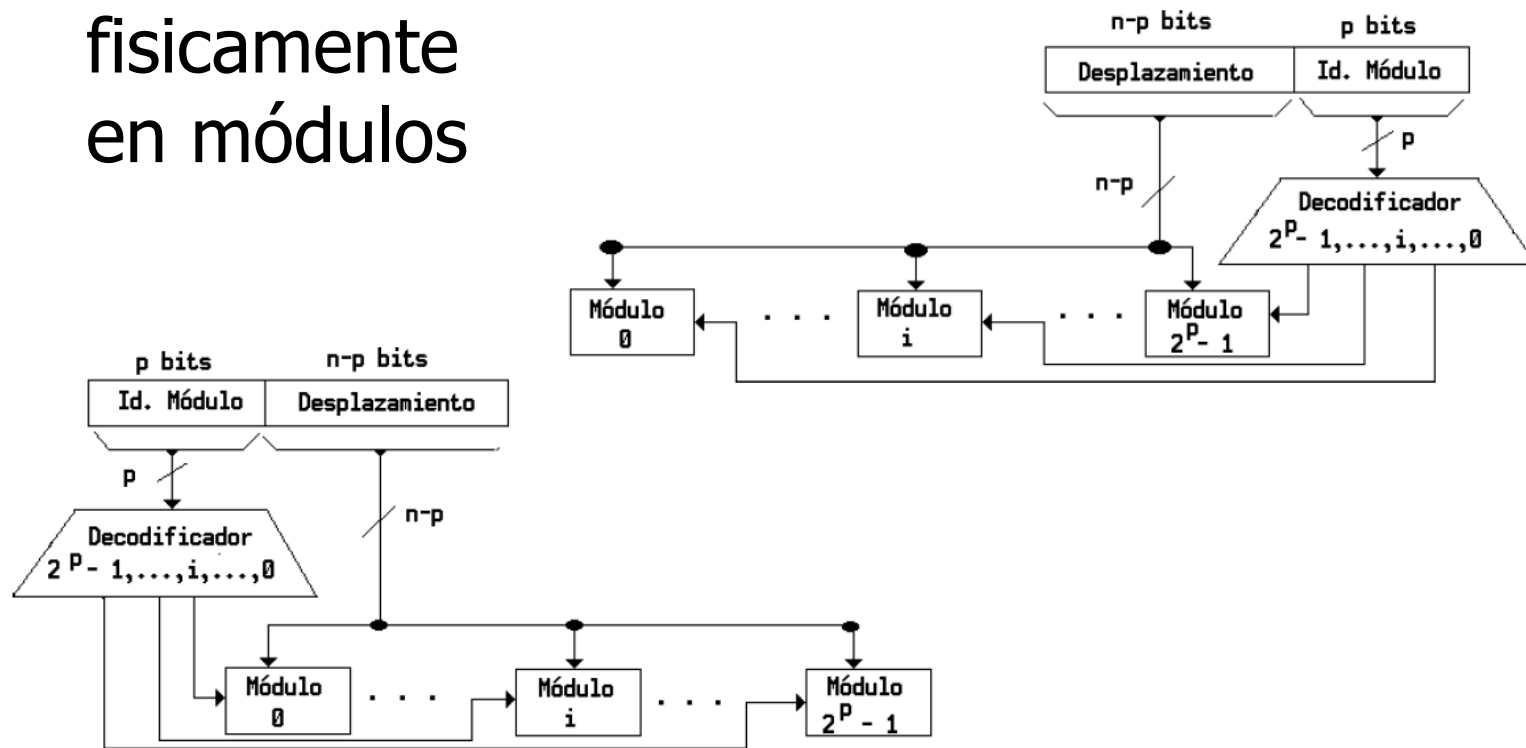
## ■ MISD

## ■ MIMD



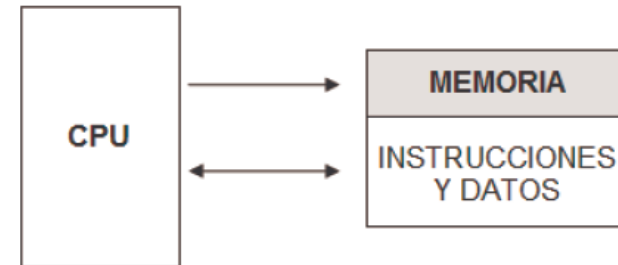
# Memoria para accesos múltiples

- Intercalación de Direcciones
  - La Memoria Principal se agrupa físicamente en módulos

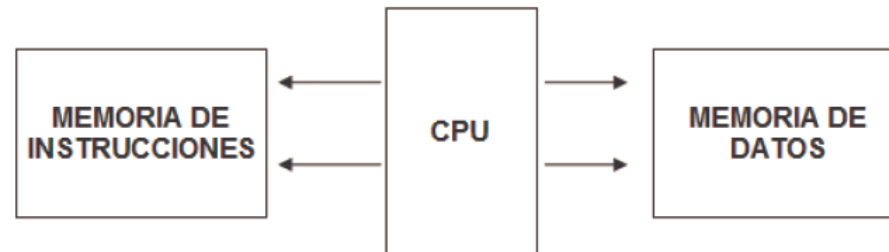


# Paradigmas

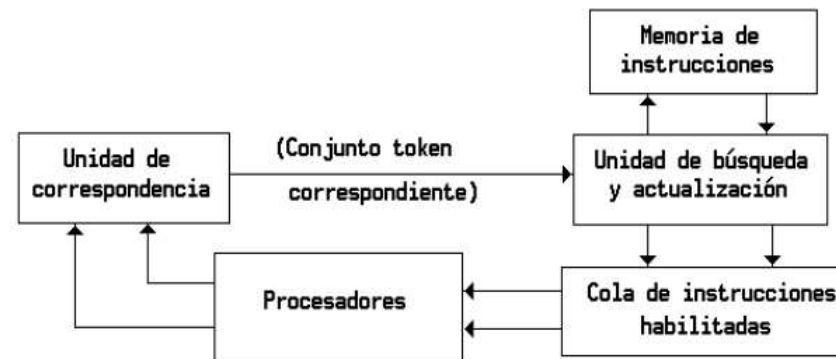
- Control Flow
  - de Von Newman



- de Harvard



- Data Flow
  - Dinámica



# Introducción Paralelo

- **La concurrencia implica paralelismo, simultaneidad y pipelining**
- **Sucesos Paralelos** ocurren en múltiples recursos durante el mismo intervalo de tiempo.
- **Sucesos Simultáneos** ocurren en el mismo instante.
- **Sucesos Pipeline** ocurren en lapsos superpuestos.

# Niveles de Paralelismo

***Multiprogramación, Multiprocesamiento:*** Estas acciones se toman a nivel de Programa o Trabajo.

***Tarea o Procedimientos:*** Acciones que se toman dentro de un mismo programa, ejecutándose procesos independientes en forma simultánea.

***Interinstrucciones:*** Acciones a nivel de instrucción, o sea, dentro de un mismo proceso o tarea se pueden ejecutar instrucciones independientes en forma simultánea.

***Intrainstrucciones:*** Acciones simultáneas que se pueden realizar para una misma instrucción, por ejemplo vectorización de operaciones escalares dentro de una instrucción compleja tipo DO, FOR, etc.



# ¿Cuándo se puede ejecutar paralelo?

- Condiciones de Bernstein
- Dada  $S_i$   $S_j$  instrucciones o conjunto de instrucciones
- 1.  $R(S_i) \cap W(S_j) = (\emptyset)$ .
- 2.  $W(S_i) \cap R(S_j) = (\emptyset)$ .
- 3.  $W(S_i) \cap W(S_j) = (\emptyset)$ .

# ¿Alcanza?

- $A = a$
- $B = b$
- $C = A + B$
- $D = C$

Falta determinar dependencias transitivas  
Grafos de Presidencia, los veremos en  
Concurrencia