

2º curso / 2º cuatr.

Grado en
Ing. Informática

Arquitectura de Computadores

Tema 1

Arquitecturas Paralelas: Clasificación y Prestaciones

Material elaborado por los profesores responsables de la asignatura:

Mancia Anguita – Julio Ortega

Licencia Creative Commons



ugr

Universidad
de Granada

ETSIIT

Escuela Técnica Superior
de Ingenierías Informática
y de Telecomunicación



ATC

Departamento de Arquitectura
y Tecnología de Computadores
UNIVERSIDAD DE GRANADA



Lecciones

- Lección 1. Clasificación del paralelismo implícito en una aplicación
- Lección 2. Clasificación de arquitecturas paralelas
 - Computación paralela y computación distribuida
 - Clasificaciones de arquitecturas y sistemas paralelos
 - Nota histórica
- Lección 3. Evaluación de prestaciones

Objetivos Lección 2

- Clasificar los computadores según segmento del mercado.
- Distinguir entre procesamiento o computación paralela y distribuida.
- Distinguir entre las diferentes clases de arquitecturas de la clasificación de Flynn.
- Diferenciar un multiprocesador de un multicomputador.
- Distinguir entre NUMA y SMP.
- Distinguir entre arquitecturas DLP, ILP y TLP.
- Distinguir entre arquitecturas TLP con una instancia de SO y TLP con varias instancias de SO.

Bibliografía

➤ Fundamental

- Secciones 1.3, 7.3 y 7.2.2. J. Ortega, M. Anguita, A. Prieto. *Arquitectura de Computadores*, Thomson, 2005. ESIIT/C.1
ORT arq

➤ Complementaria

- T. Rauber, G. R nder. *Parallel Programming: for Multicore and Cluster Systems*. Springer 2010. Disponible en l nea (biblioteca UGR): <http://dx.doi.org/10.1007/978-3-642-04818-0>

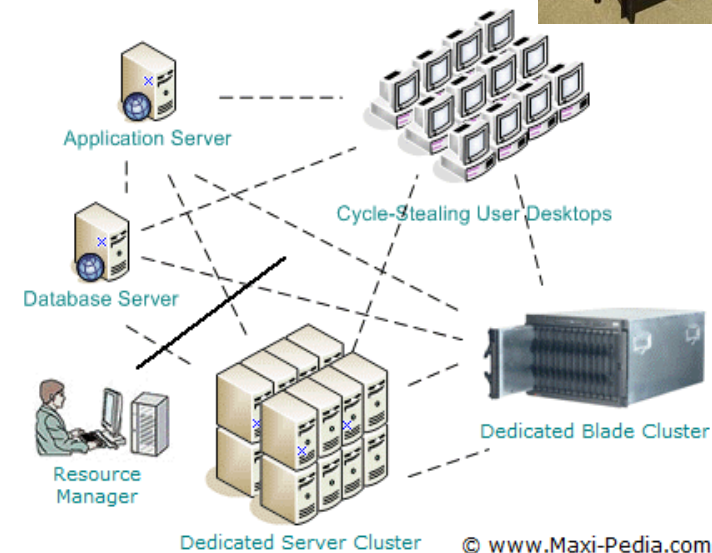
Contenidos

- Computación paralela y computación distribuida
- Clasificaciones de arquitecturas y sistemas paralelos
- Nota histórica

Computación paralela – Computación distribuida



- Computación paralela (se estudia en AC)
 - Estudia los aspectos hardware y software relacionados con el desarrollo y ejecución de aplicaciones en un sistema de cómputo compuesto por **múltiples cores/procesadores/computadores** que es visto externamente como una **unidad autónoma** (multicores, multiprocesadores, multicomputadores, cluster)
- Computación distribuida
 - Estudia los aspectos hardware y software relacionados con el desarrollo y ejecución de aplicaciones en un **sistema distribuido**; es decir, en una **colección de recursos autónomos** (PC, servidores -de datos, aplicaciones, ...-, supercomputadores ...) situados en **distintas localizaciones físicas**

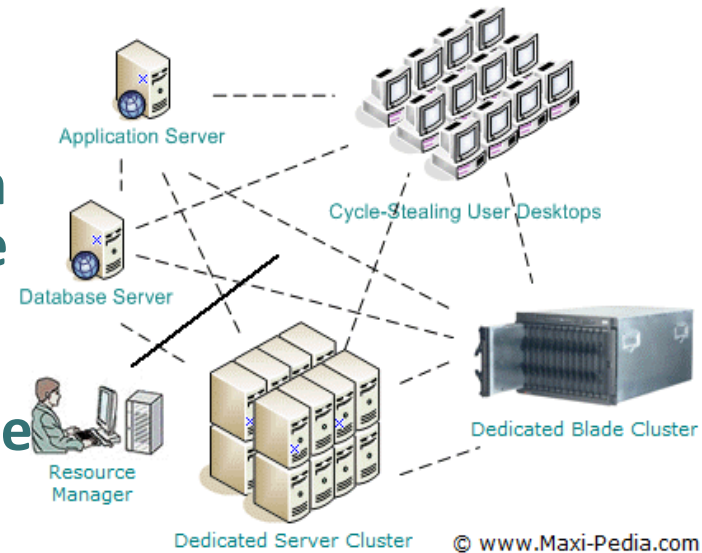


© www.Maxi-Pedia.com

Computación distribuida a gran escala: Computación grid

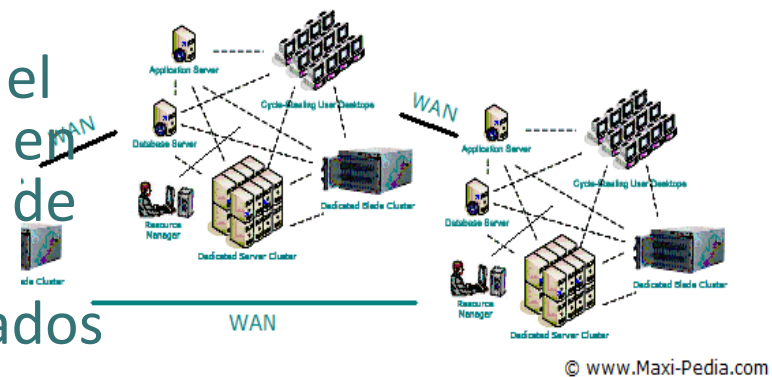
➤ Computación distribuida baja escala

- Estudia los aspectos relacionados con el desarrollo y ejecución de aplicaciones en una colección de recursos autónomos **de un dominio administrativo** situados en **distintas localizaciones físicas** conectados a través de **infraestructura de red local**



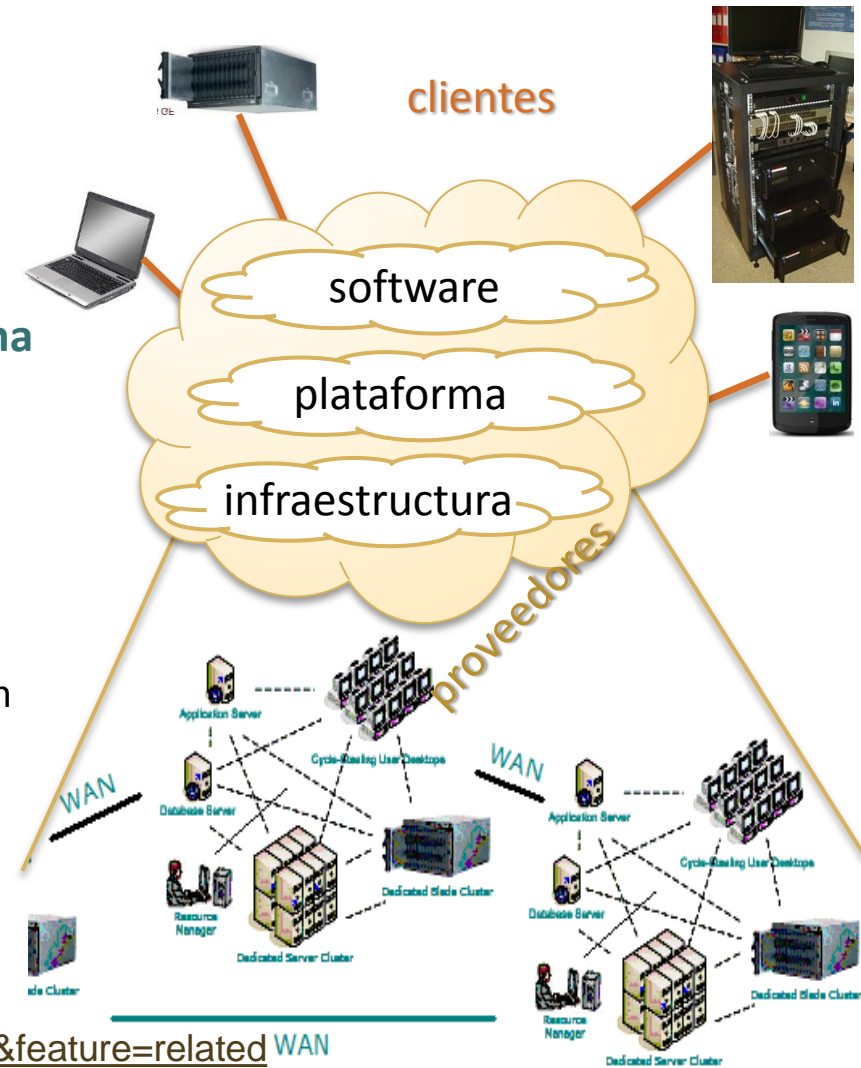
➤ Computación grid

- Estudia los aspectos relacionados con el desarrollo y ejecución de aplicaciones en una colección de recursos autónomos **de múltiples dominios administrativos** **geográficamente distribuidos** conectados con **infraestructura de telecomunicaciones**



Computación distribuida a gran escala: Computación nube o *cloud*

- Computación *cloud*
 - Comprende los aspectos relacionados con el desarrollo y ejecución de aplicaciones en un **sistema cloud**
- Sistema *cloud*
 - Ofrece **servicios de infraestructura, plataforma y/o software**, por los que se paga cuando se necesitan (**pay-per-use**) y a los que se accede típicamente a través de una **interfaz (web) de auto-servicio**
 - Consta de **recursos virtuales** que
 - son una **abstracción** de los recursos físicos
 - parecen **ilimitados** en **número y capacidad** y son reclutados/liberados de forma **inmediata** sin interacción con el proveedor
 - soportan el acceso de múltiples clientes (**multi-tenant**)
 - están conectados con métodos **estándar independientes** de la **plataforma de acceso**.



Para ampliar: <http://www.youtube.com/watch?v=SgujalzkwrE&feature=related> WAN
<http://soa-eda.blogspot.com/2010/05/cloud-computing-explained.html>

Contenidos

- Computación paralela y computación distribuida
- Clasificaciones de arquitecturas y sistemas paralelos
- Nota histórica

Criterios de clasificación de computadores

➔ Comercial

- Segmento del mercado
 - embebidos, servidores gama baja ...
- Educación, investigación (también usados por fabricantes y vendedores)
 - Flujos de control y flujos de datos: clasificación de Flynn (1972)
 - Sistema de memoria
 - Flujos de control (propuesta de clasificación de arquitecturas con múltiples flujos de control)
 - Nivel del paralelismo aprovechado (propuesta de clasif.)

Segmento del mercado

MATERIAS DEL GRADO

Supercomputadores

Servidores de gama alta
(500.000\$ < *high-end*)

Servidores de gama media
(25.000\$ < *mid-range* < 500.000\$)

Servidores de gama baja
(*entry-level* < 25.000\$)

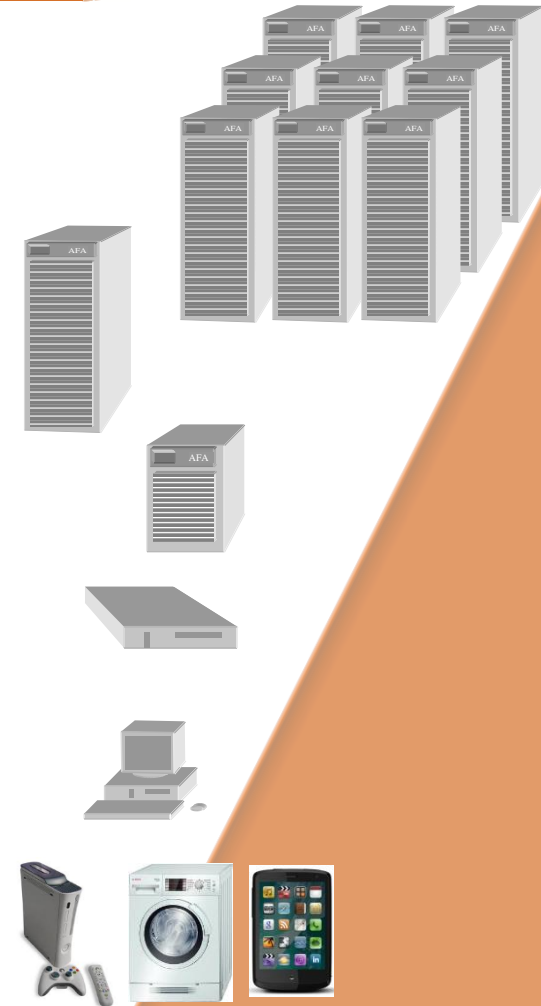
PC/WS

Mercado de computadores
empotrados

Sistemas de Cómputo
de Altas Prestaciones
(IC.SCAP)

Estructura y
Arquitectura de
Computadores
(R.EAC)

Sistemas de Cómputo
para Aplicaciones
Específicas (IC.SCAE)



Clasificación de Computadores según segmento

Externo (*desktop, laptop, server, cluster ...*)
- R.EAC, IC.SCAP

Para todo tipo de aplicaciones:

- Oficina, entretenimiento, ...
- Procesamiento de transacciones o OLTP, sistemas de soporte de decisiones o DSS, e-commerce, ...
- Científicas (medicina, biología, predicción del tiempo, etc.) y animación (películas animadas, efectos especiales, etc.), ...

Empotrado (oculto) - IC.SCAE

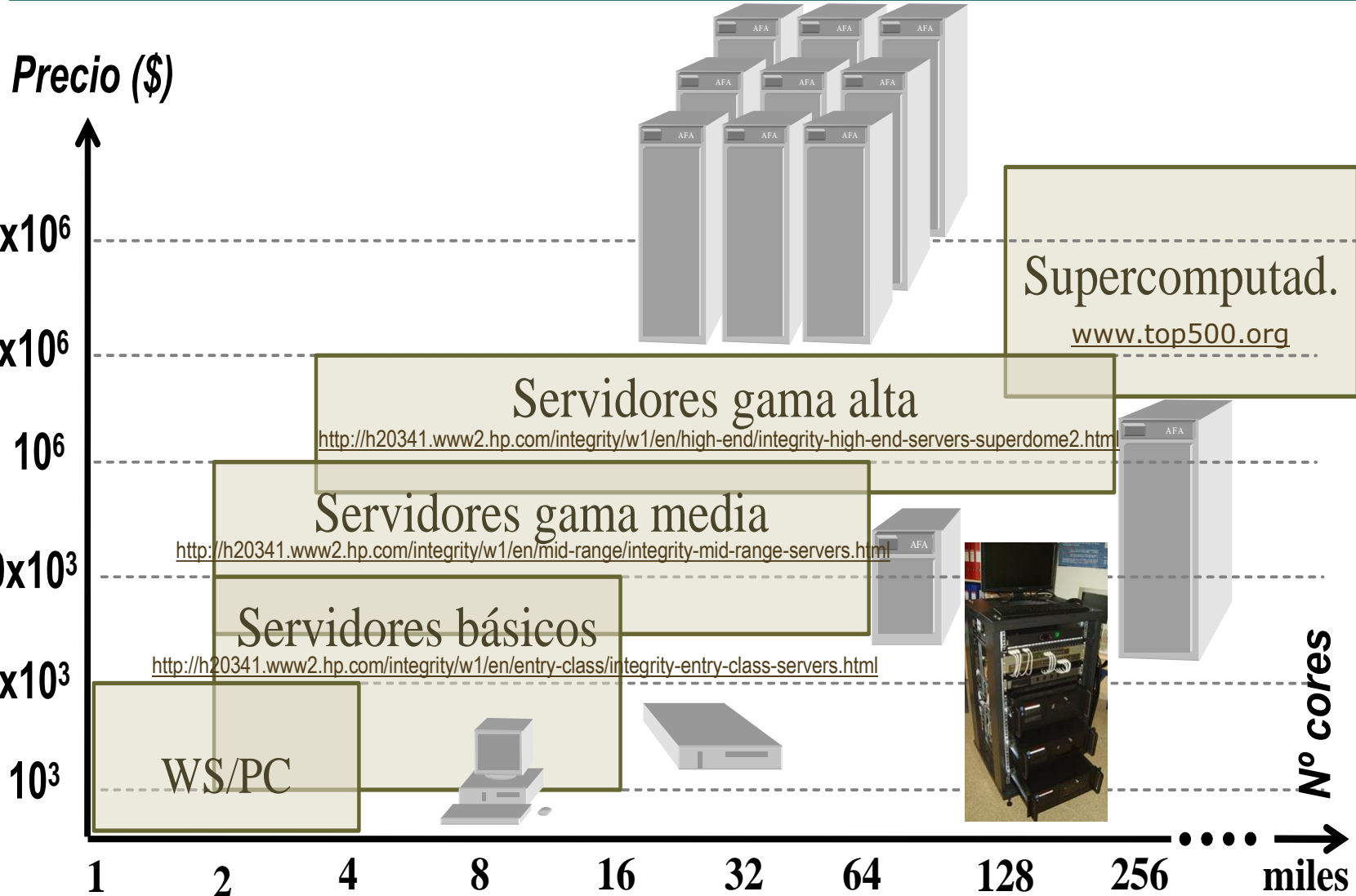
Aplicaciones de propósito específico

- Videojuegos, teléfonos, coches, electrodomésticos, ...

Restricciones típicas:

- Consumo de potencia, precio, tamaño reducidos
- Tiempo real

Clasificación de Comp. (externos) según segmento del mercado



Criterios de clasificación de computadores

➤ Comercial

➤ Segmento del mercado

- embebidos, servidores gama baja ...

➔ Educación, investigación (también usados por fabricantes y vendedores)

- Flujos de control y flujos de datos (clasificación de Flynn 1972)
- Sistema de memoria
- Flujos de control (propuesta de clasificación de arquitecturas con múltiples flujos de control)
- Nivel del paralelismo aprovechado (propuesta de clasif.)

Clasificación de Flynn de arquitecturas, 1972 (Flujo instr./flujo de datos)

Clasificación de Flynn

Flujo de instrucciones

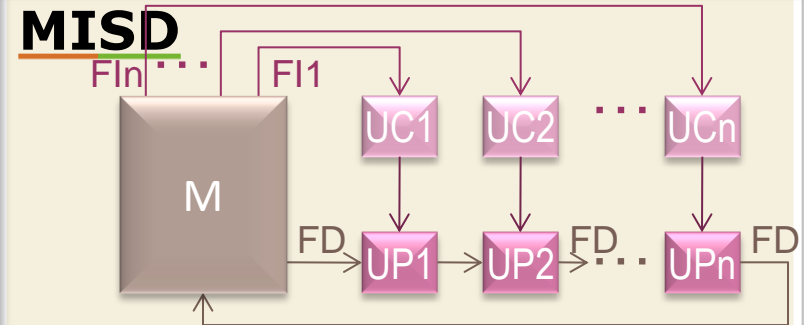
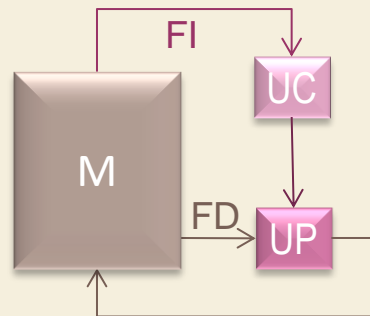
Único (Single)

Múltiple

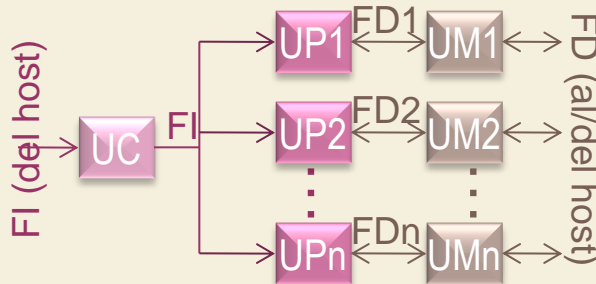
Flujo de datos

Único

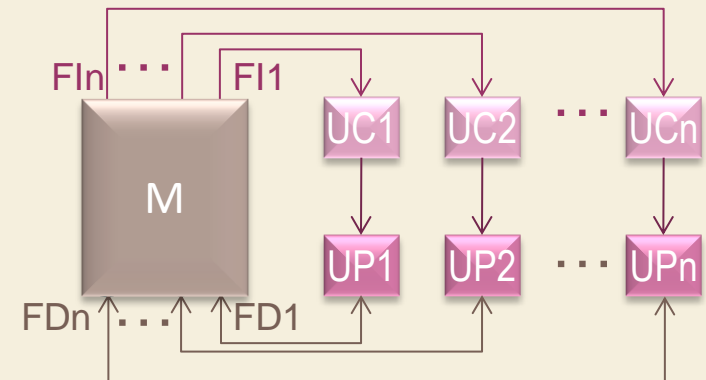
SISD
(Un núcleo procesador)



SIMD
(GPU, Procesadores matriciales)

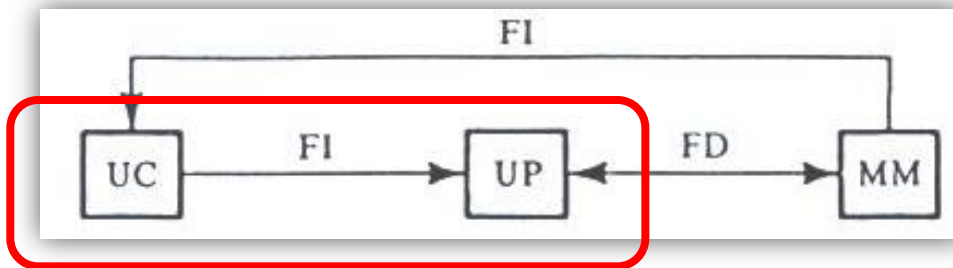


MIMD (multicores, multiprocesadores, multicomputadores)



Arquitecturas SISD

Descripción Estructural

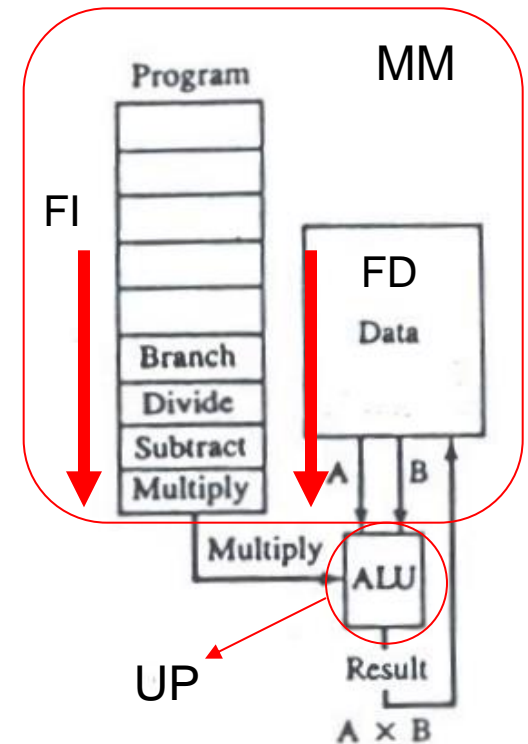


CPU

Corresponde a los computadores uni-procesador (un núcleo o core procesador)

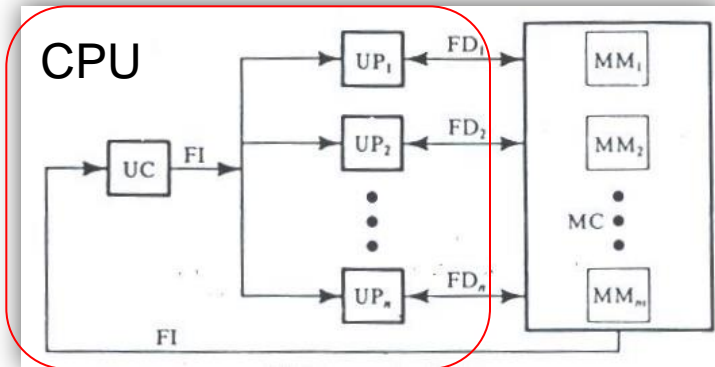
```
for i:=1 to 4 do
begin
    C[i]:=A[i]+B[i];
    F[i]:=D[i]-E[i];
    G[i]:=K[i]*H[i];
end;
```

Descripción Funcional

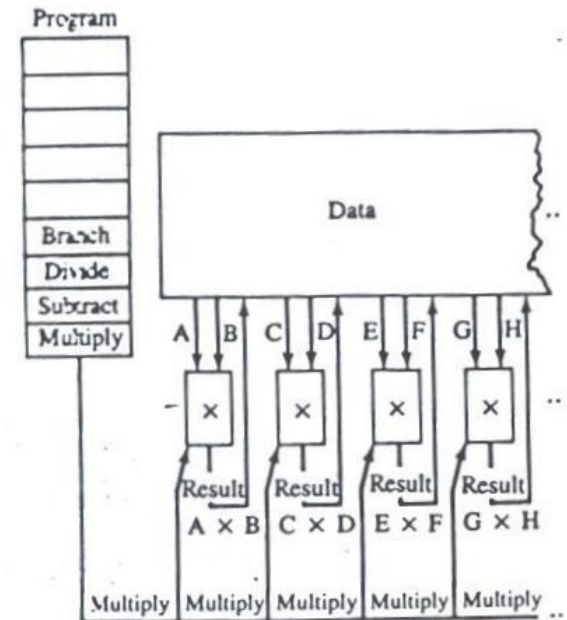


Arquitecturas SIMD

Descripción Estructural



Descripción Funcional



Aprovechan **paralelismo de datos**

```
for all EPI(i:=1 to 4) do
begin
    C[i]:=A[i]+B[i];
    F[i]:=D[i]-E[i];
    G[i]:=K[i]*H[i];
end;
```

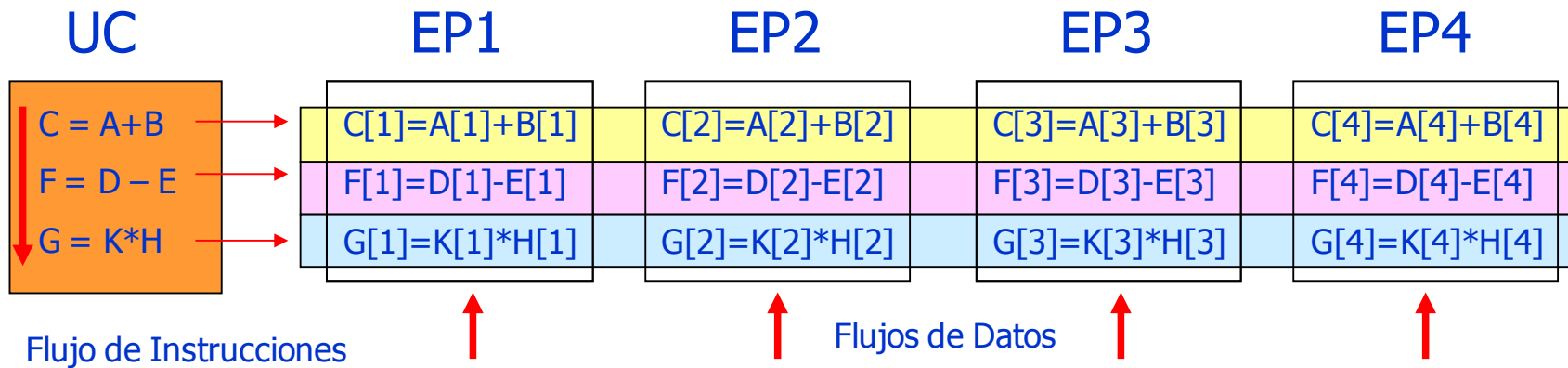
Procesadores
Matriciales

Procesadores
Vectoriales

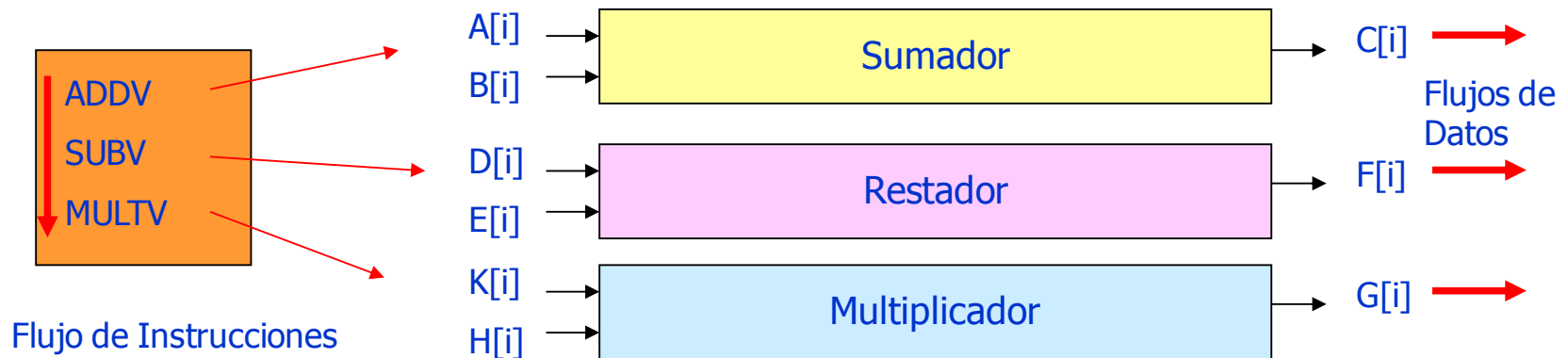
```
ADDV C,A,B
SUBV F,D,E
MULV G,K,H
```

Arquitecturas SIMD: Ejemplo

Procesador Matricial

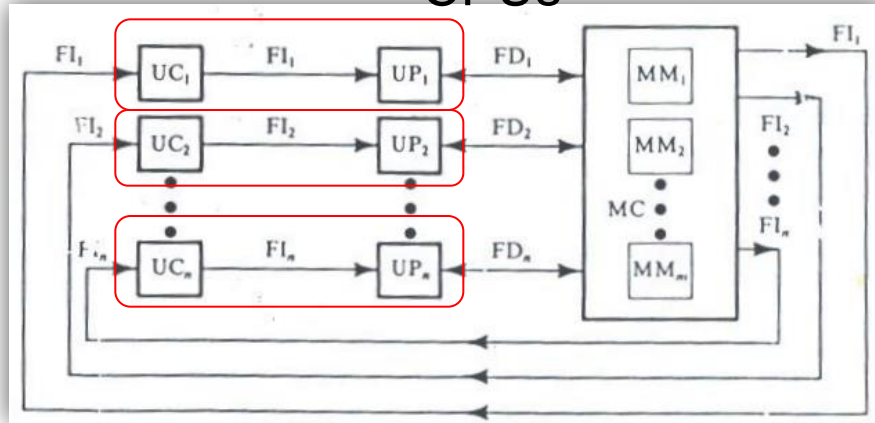


Procesador Vectorial

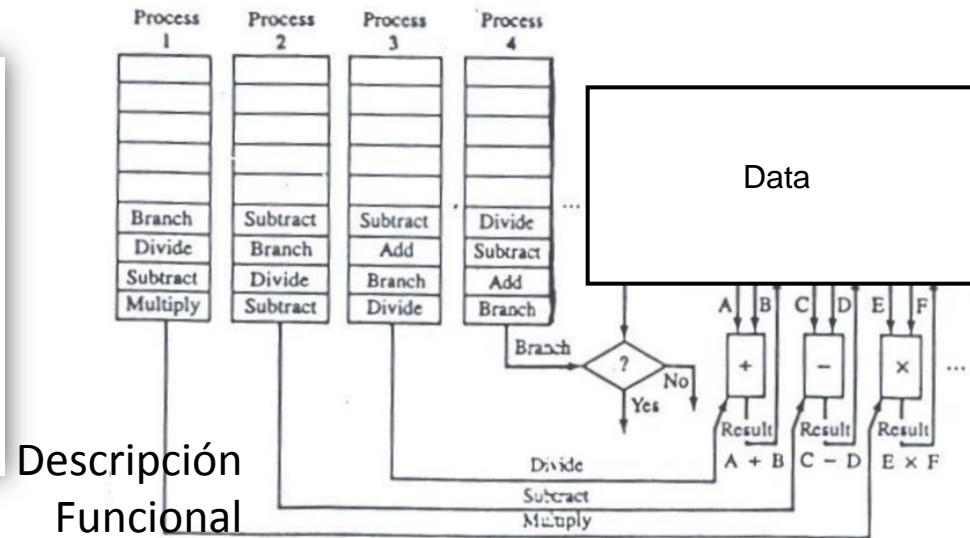


Arquitecturas MIMD

CPU's



Descripción Estructural



Descripción Funcional

Corresponde con Multinúcleos, Multiprocesadores y Multicomputadores: Puede aprovechar, además, **paralelismo funcional**

```
for i:=1 to 4 do
begin
  C[i]:=A[i]+B[i];
end;
```

Proc 1

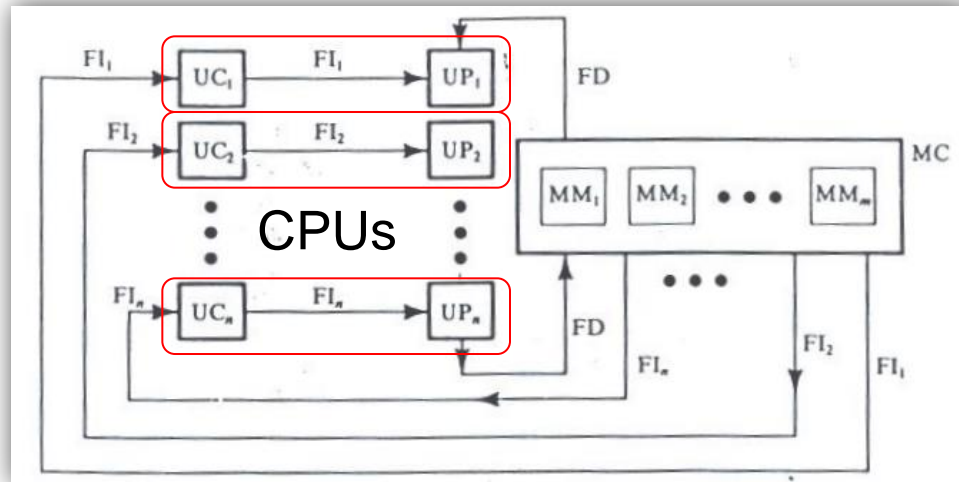
```
for i:=1 to 4 do
begin
  F[i]:=D[i]-E[i];
end;
```

Proc 2

```
for i:=1 to 4 do
begin
  G[i]:=K[i]*H[i];
end;
```

Proc 3

Arquitecturas MISD



- No existen computadores que funcionen según este modelo
- Se puede simular en un código este modelo para aplicaciones que procesan una secuencia o flujo de datos

Criterios de clasificación de computadores

➤ Comercial

➤ Segmento del mercado

- embebidos, servidores gama baja ...

➤ Educación, investigación: también usados por fabricantes y vendedores

➤ Flujos de control y flujos de datos (clasificación de Flynn 1972)

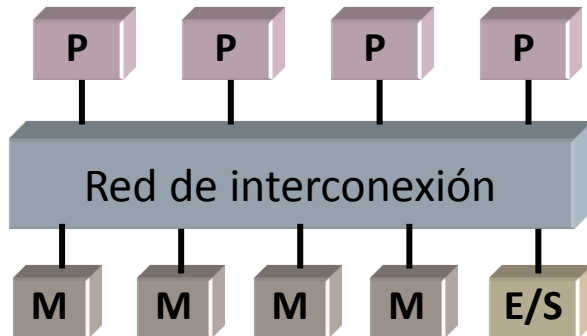
➡ Sistema de memoria

➤ Flujos de control (propuesta de clasificación de arquitecturas con múltiples flujos de control)

➤ Nivel del paralelismo aprovechado (propuesta de clasif.)

Clasificación de Computadores Paralelos

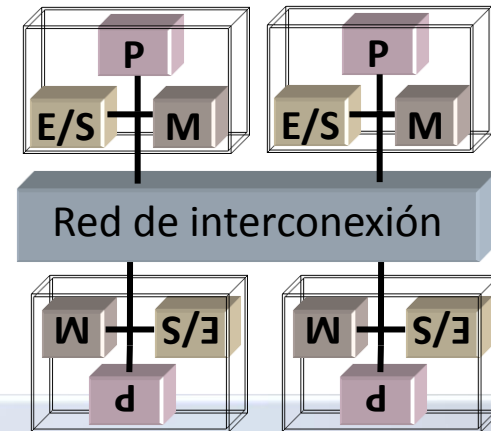
MIMD según el sistema de memoria



Multiprocesadores

Todos los procesadores comparten el **mismo espacio de direcciones**

El programador NO necesita conocer dónde están almacenados los datos

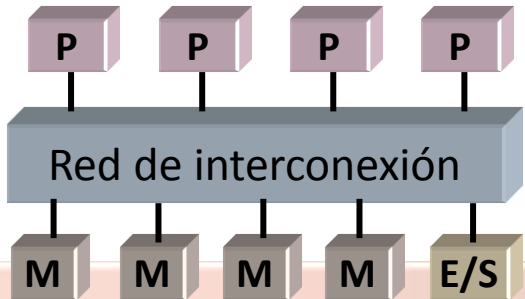


Multicomputadores

Cada procesador tiene su **espacio de direcciones propio**

El programador necesita conocer dónde están almacenados los datos

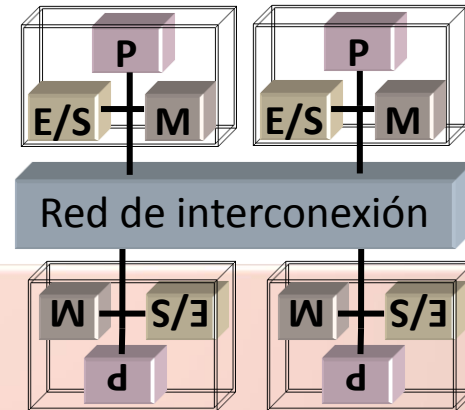
Comparativa SMP (*Symmetric Multi-Processor*) y multicomputadores I



Multiprocesador con memoria centralizada (SMP)

Mayor latencia - Poco escalable

Comunicación implícita mediante variables compartidas. Datos no duplicados en memoria principal

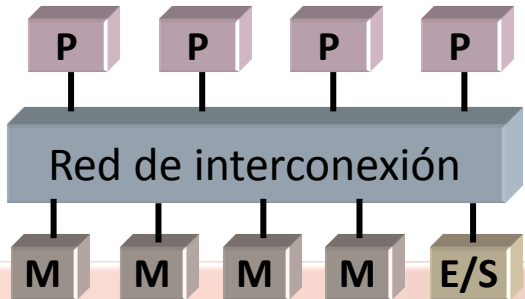


Multicomputador

Menor latencia – Más escalable

Comunicación explícita mediante software para paso de mensajes (send/receive). Datos duplicados en memoria principal, copia datos

Comparativa SMP (*Symmetric Multi-Processor*) y multicomputadores II

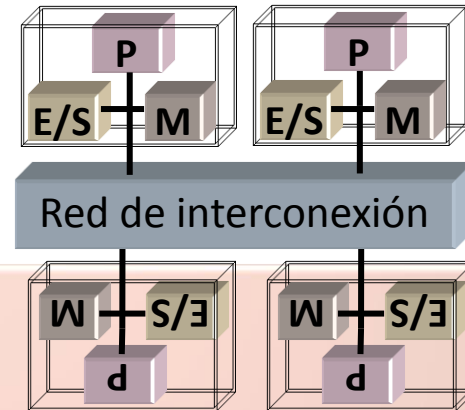


Multiprocesador con memoria centralizada (SMP)

Necesita implementar primitivas de sincronización

Distribución código y datos entre procesadores: no necesaria

Programación, generalmente, más sencilla



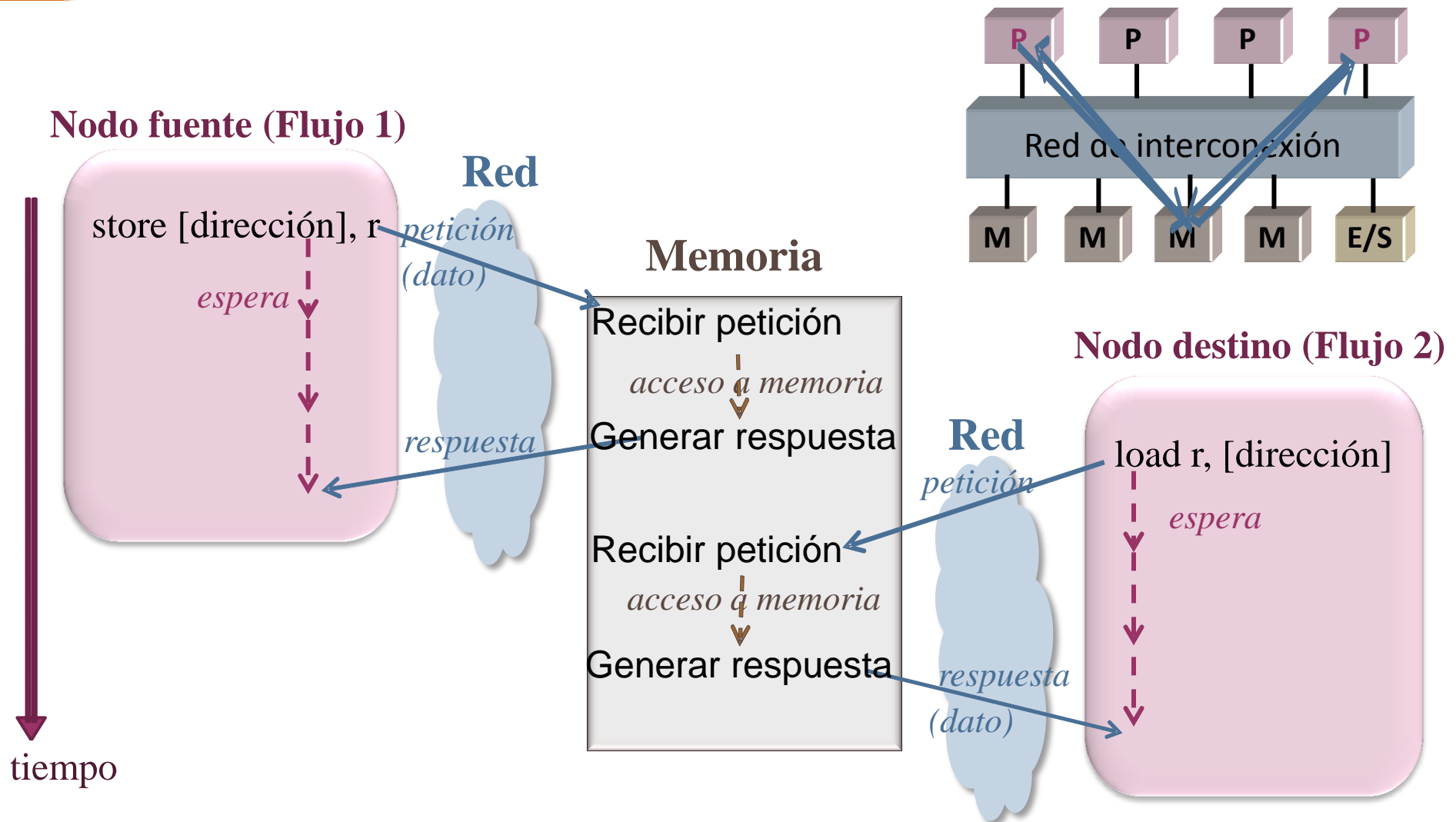
Multicomputador

Sincronización mediante software de comunicación

Distribución código y datos entre procesadores: necesaria=> herramientas program. más sofisticadas

Programación generalmente más difícil

Comunicación uno-a-uno en un multiprocesador



Comunicación uno-a-uno en un multiprocesador

Secuencial	Paralelo	
...	<u>F1</u>	<u>F2</u>
A=valor;
...	A=valor;	copia=A;
copia=A;
...		

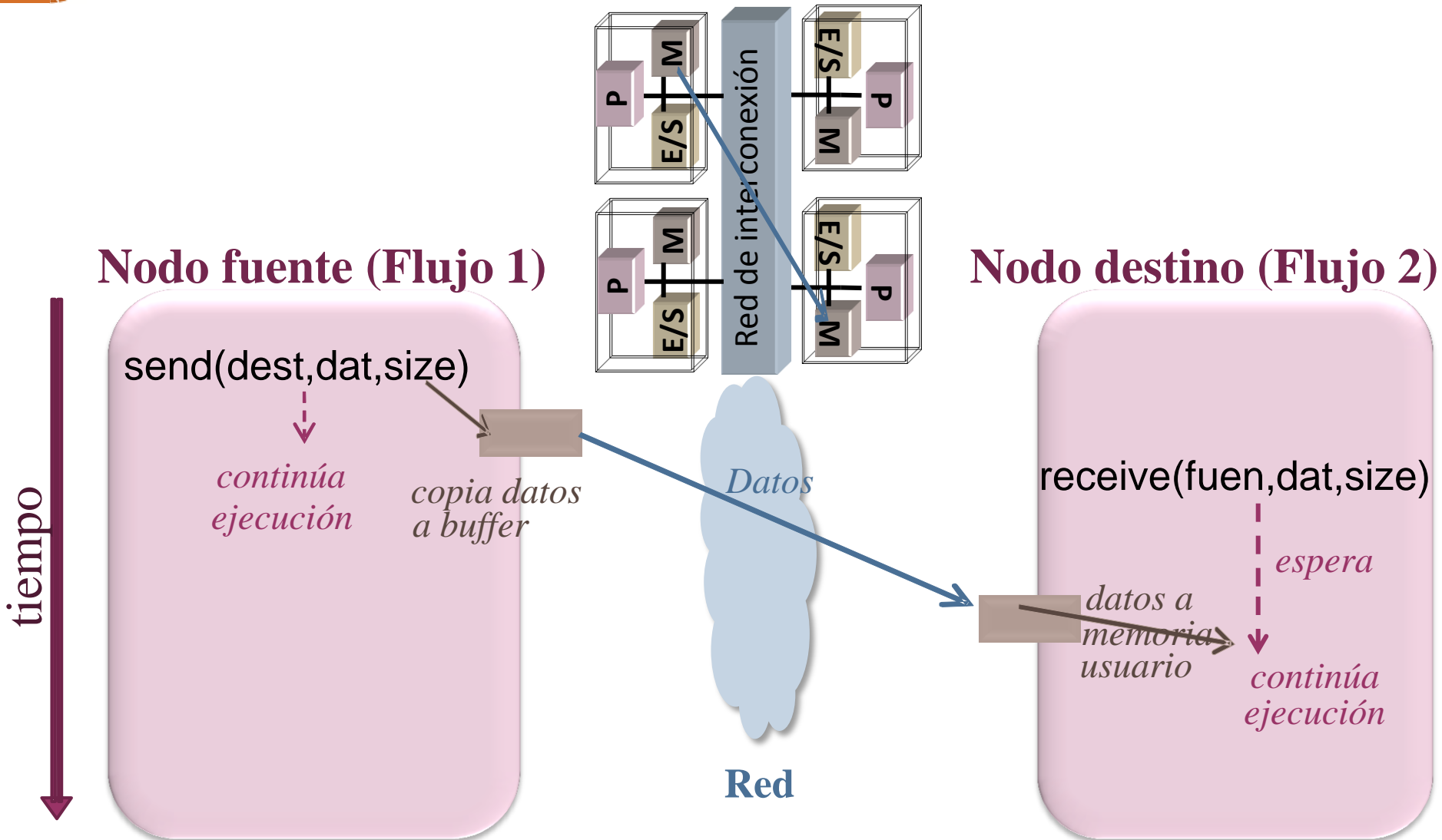
F1 es el flujo de control productor del dato (*envía* el dato)
F2 es el flujo de control consumidor del dato (*recibe* el dato)

- Se debe garantizar que el flujo de control consumidor del dato lea la variable compartida (A) cuando el productor haya escrito en la variable el dato

Paralelo multiproc. (K=0)	
<u>F1</u>	<u>F2</u>
...	...
A=valor;	while (K=0) { };
K=1;	copia=A;
...	...

NOTA: La programación paralela de multiprocesadores se estudia en AC

Comunicación uno-a-uno en multicomputador (*receive* bloqueante)



Comunicación uno-a-uno en un multicomputador

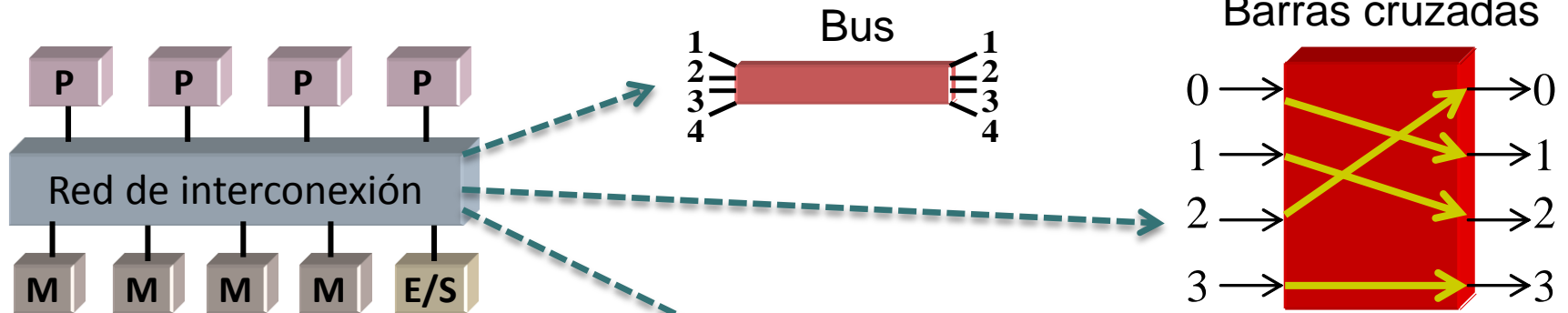
Secuencial	Paralelo	
...	<u>F1</u>	<u>F2</u>
A=valor;
...	A=valor;	copia=A;
copia=A;
...		

F1 es el flujo de control productor del dato (*envía* el dato)
F2 es el flujo de control consumidor del dato (*recibe* el dato)

Paralelo multicomputador (size = 4 byte)	
<u>F1</u>	<u>F2</u>
...	...
send(F2, valor, 4);	receive(F1,copia,4);
...	...

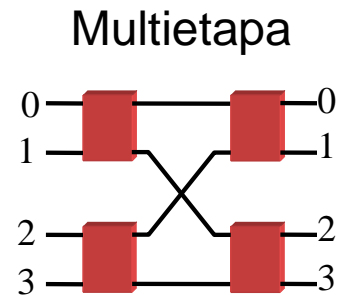
NOTA: La programación paralela de multicomputadores se estudia en la asignatura: **Arquitecturas y Computación de Altas Prestaciones** (IC.SCAP.ACAP – Especialidad (IC), Materia (SCAP), Asignatura (ACAP))

Incremento de escalabilidad en multiprocesadores y red de interconexión

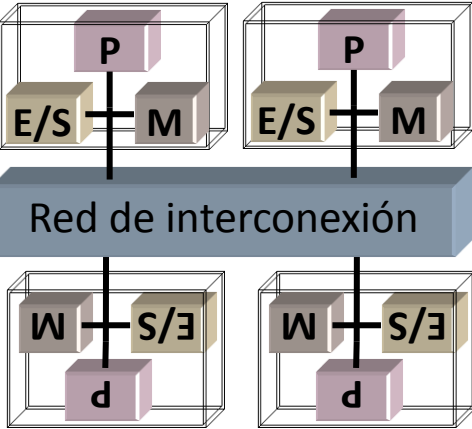
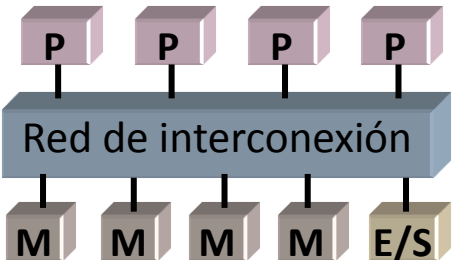


➤ Incremento escalabilidad multiprocesadores:

- Aumentar cache del procesador
- Usar redes de menor latencia y mayor ancho de banda que un bus (jerarquía de buses, barras cruzadas, multietapa)
- Distribuir físicamente los módulos de memoria entre los procesadores (pero se sigue compartiendo espacio de direcciones)

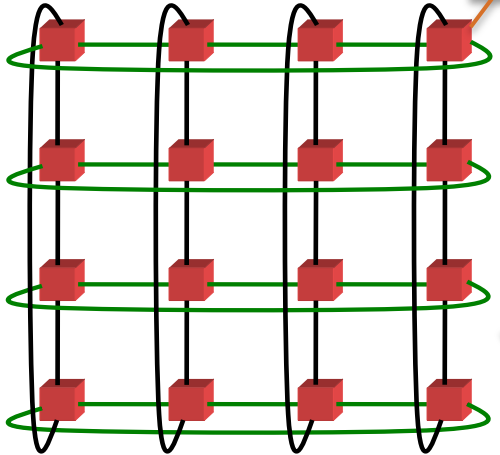


Clasificación completa de computadores según el sistema de memoria

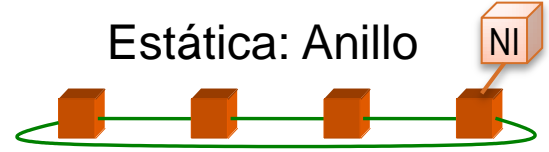
Multi-computadores Memoria no compartida	NORMA <i>No Remote Memory Access</i>	<i>ej. cluster, red de computadores</i>	<p>Memoria físicamente distribuida</p>  <p>Red de interconexión</p>	<div>+</div> <div>+</div> <div>Nivel de empaquetamiento y conexión</div> <div>Escalabilidad</div> <div>-</div> <div>-</div>
Multi-procesadores Memoria compartida Un único espacio de direcciones	NUMA <i>Non-Uniform Memory Access</i>	NUMA		
		CC-NUMA		
		COMA		
	UMA <i>Uniform Memory Access</i>	SMP <i>Symmetric MultiProcessor</i>	<p>Memoria físicamente centralizada</p>  <p>Red de interconexión</p>	

Red en sistemas con memoria físicamente distribuida (NI: *Network Interface*)

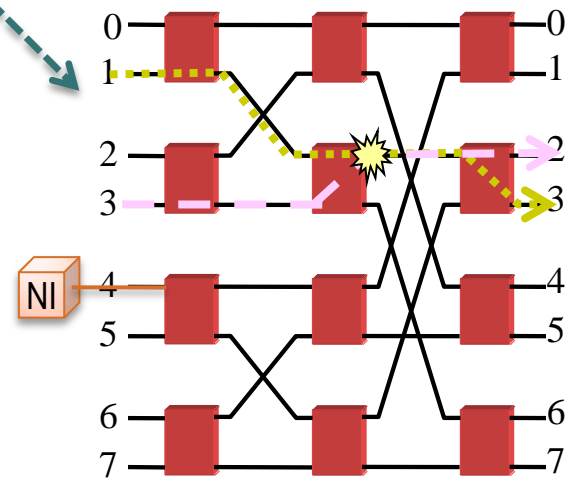
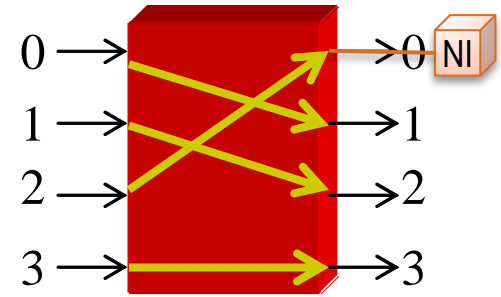
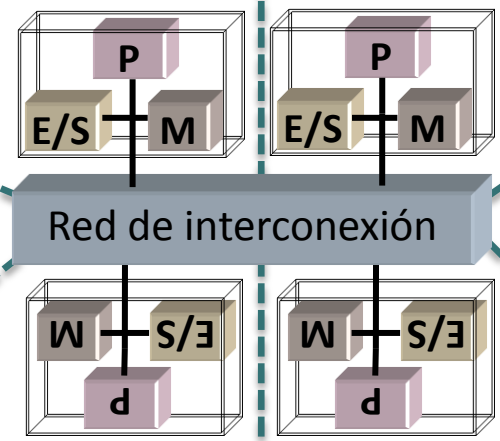
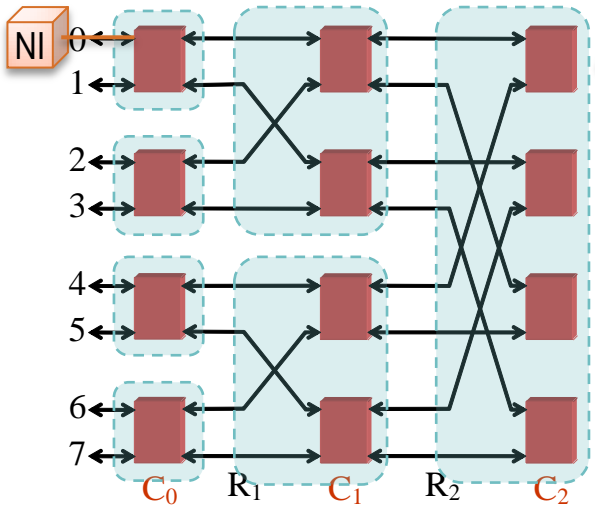
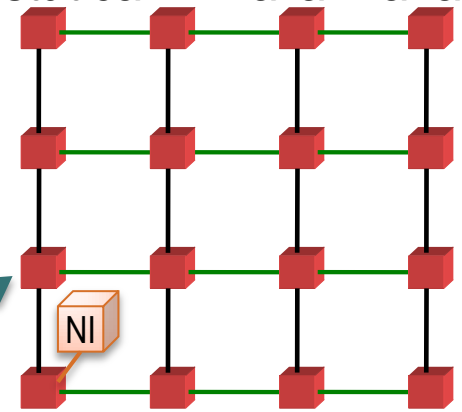
Estática: 2-cubo 4-ario NI



Estática: Anillo NI



Estática: 2-malla 4-aria NI



Dinámica: árbol grueso

Dinámica: barras cruzadas

Dinámica: multietapa

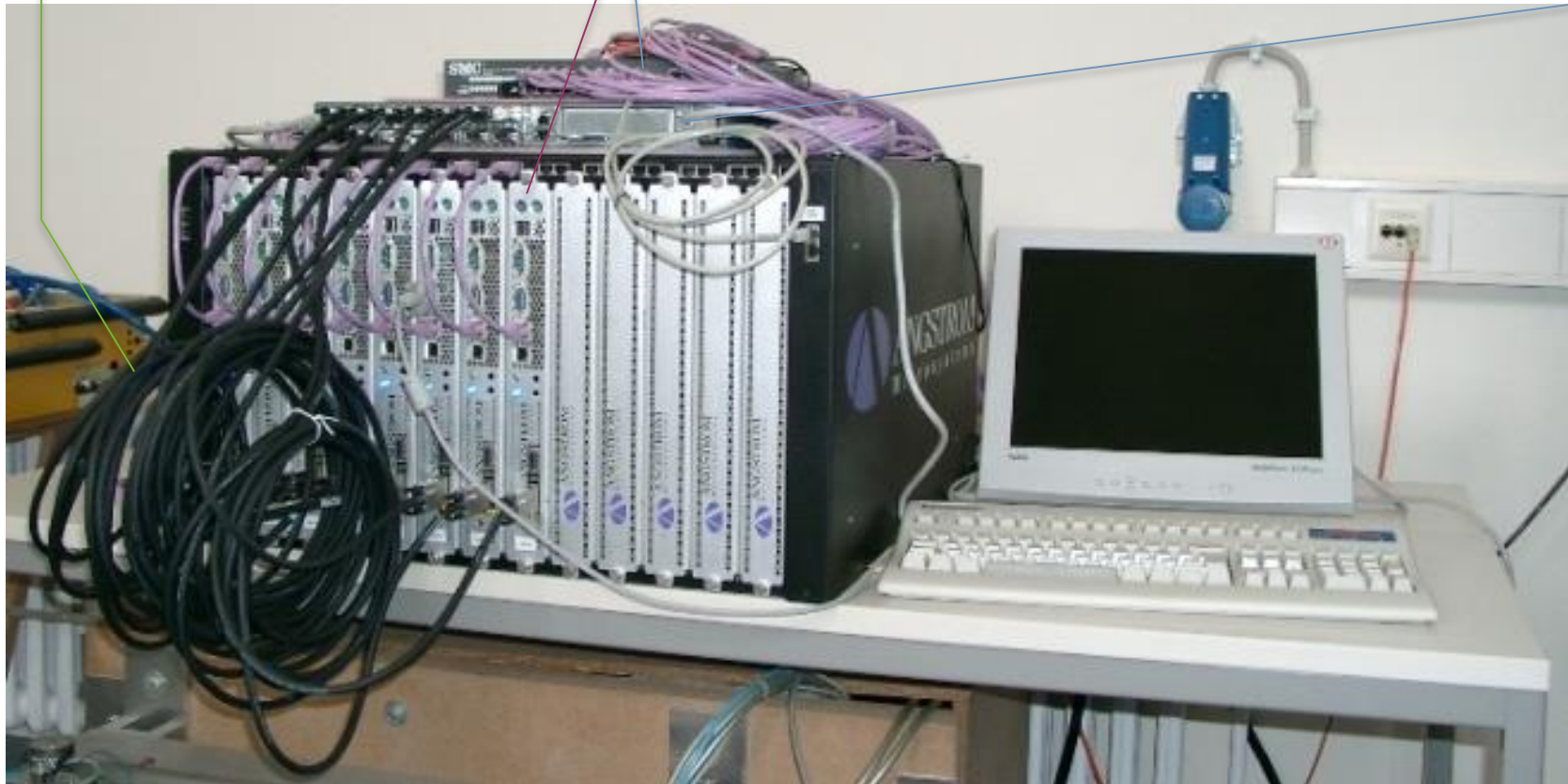
Ejemplo: Red (con conmutador o *switch*) de barras cruzadas

nodos

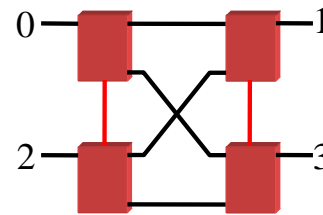
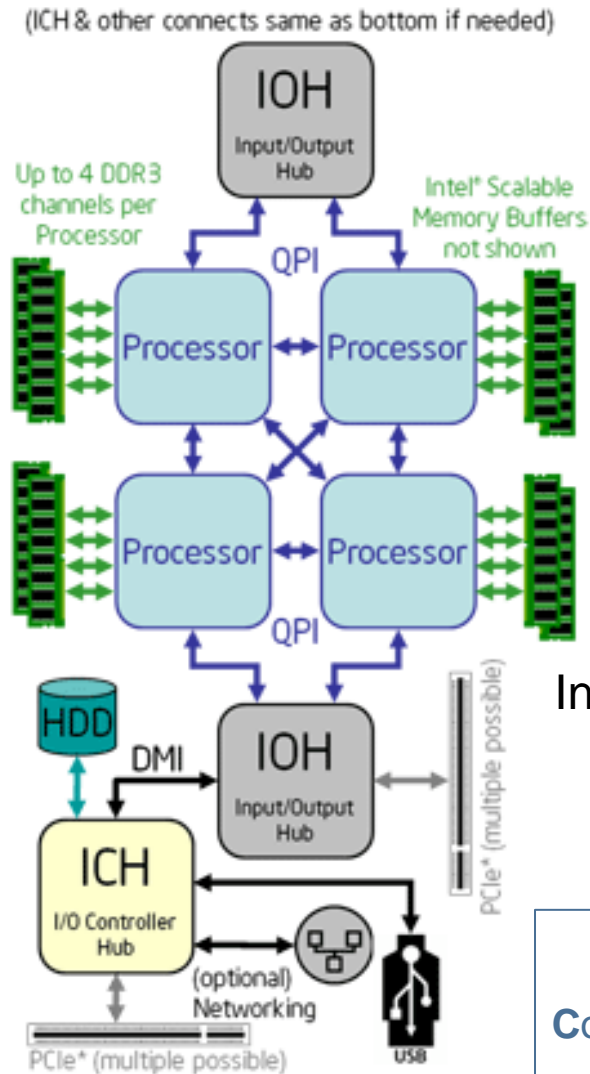
enlaces

Conmutador Gigabit Ethernet (para administración)

Conmutador Infiniband (para aplicaciones)



Ejemplo: Placa CC-NUMA con red estática



Intel Xeon 7500



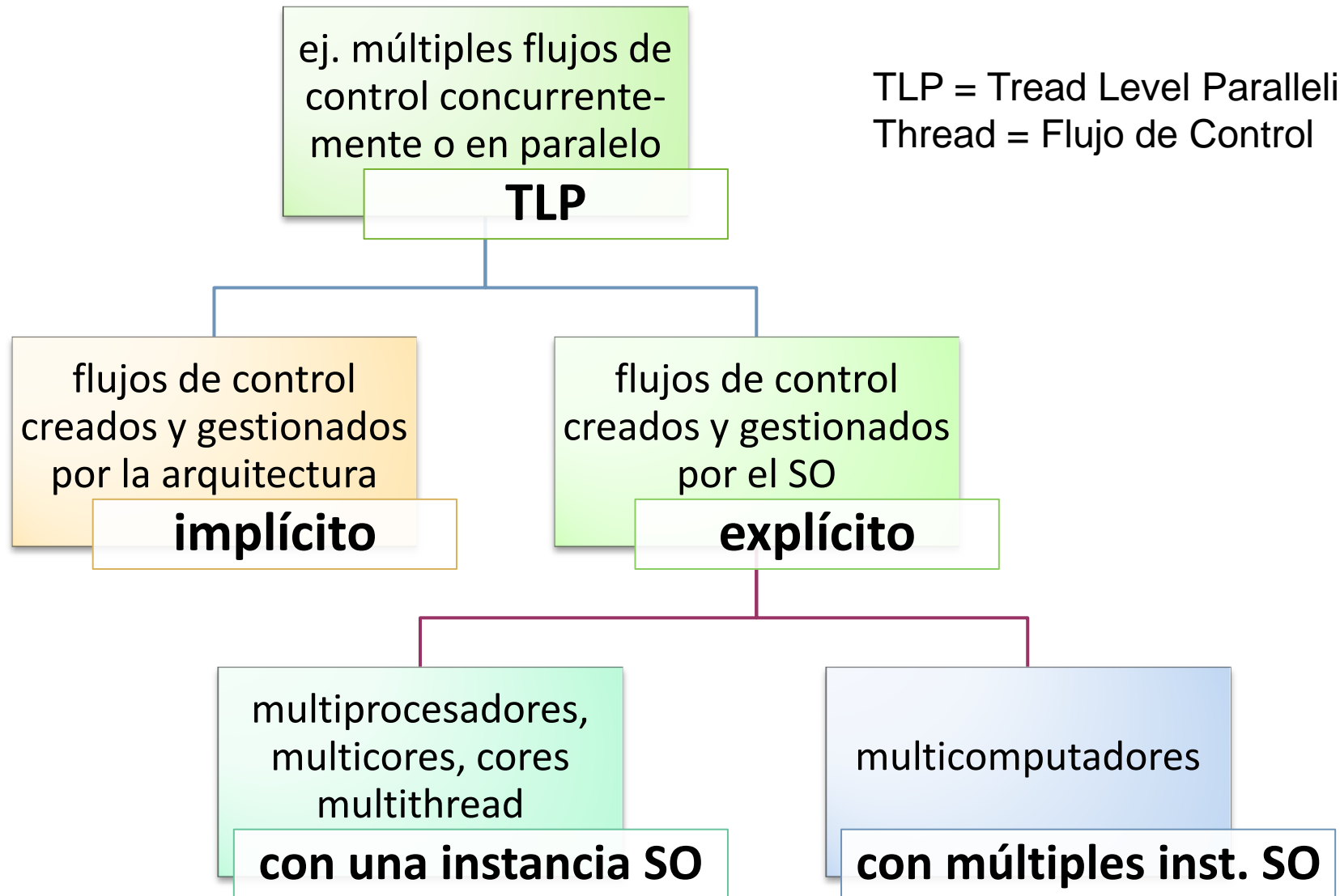
NOTA: Detalles de redes para multicomputadores y multiprocesadores se estudian en la asignatura: **Arquitecturas y Computación de Altas Prestaciones** (IC.SCAP.ACAP – Especialidad (IC), Materia (SCAP), Asignatura (ACAP))

Criterios de clasificación de computadores

- Comercial
 - Segmento del mercado
 - embebidos, servidores gama baja ...
- Educación, investigación: también usados por fabricantes y vendedores
 - Flujos de control y flujos de datos (clasificación de Flynn 1972)
 - Sistema de memoria
- ➡ Flujos de control (propuesta de clasificación de arquitecturas con múltiples flujos de control)
 - Nivel del paralelismo aprovechado (propuesta de clasif.)

Propuesta clasificación arquitecturas con múltiples flujos de control o threads

TLP = Tread Level Parallelism
Thread = Flujo de Control



Criterios de clasificación de computadores

- Comercial
 - Segmento del mercado
 - embebidos, servidores gama baja ...
- Educación, investigación: también usados por fabricantes y vendedores
 - Flujos de control y flujos de datos (clasificación de Flynn 1972)
 - Sistema de memoria
 - Flujos de control (propuesta de clasificación de arquitecturas con múltiples flujos de control)
- ➡ Nivel del paralelismo aprovechado (propuesta de clasif.)

Arquitecturas con DLP, ILP y TLP (thread=flujo de control)

Arq. con **DLP**
(*Data Level Parallelism*)

Tema 5

Ejecutan las operaciones de una instrucción **concurr.** o en **paralelo**

Unidades funcionales vectoriales o SIMD

Arq. con **ILP**
(*Instruction Level Parallelism*)

Tema 4

Ejecutan múltiples instrucciones **concurr.** o en **paralelo**

Cores escalares segmentados, superescalares o VLIW/EPIC

Arq. con **TLP** (*Thread Level Parallelism*) explícito y **una** instancia de SO

Temas 3, 5

Ejecutan múltiples flujos de control **concurr.** o en **paralelo**

Cores que modifican la archit. escalar segmentada, superescalar o VLIW/EPIC para ejecutar threads **concurr.** o en **paralelo**

Multi-procesadores: ejecutan threads en *paralelo* en un computador con múltiples cores (incluye **multicores**)

Arq. con **TLP** explícito y **múltiples** instancias SO

IC.SCAP

Ejec. múltiples flujos de control en **paralelo**

Multi-computadores: ejecutan threads en *paralelo* en un sistema con múltiples computadores

Contenidos

- Computación paralela y computación distribuida
- Clasificaciones de arquitecturas y sistemas paralelos
- Nota histórica

Nota histórica. DLP y ILP

➤ DLP (*Data Level Parallelism*)

TEMA 5

➤ Unidades funcionales (o de ejecución) SIMD (o multimedia)

- **1989** (*Intel i860*). **1991** (motorola M88110). **1993** (repertorio MAX en HP PA : PA7100LC). **1995** (repertorio VIS en Sun Sparc: Ultra I). **1997** (repertorio MMX en Intel x86: Pentium MMX). **1999** (repertorio SSE en Intel x86: Pentium III; repertorio AltiVec en IBM Power: PowerPC 8000)

➤ ILP (*Instruction Level Parallelism*) :

TEMA 4

➤ Procesadores/cores segmentados

- **1961** (*IBM 7030*). **1982** (chip Intel i286, Motorola 68020). **1986** (chip MIPS R2000). **1987** (chip AMD Am29000). **1988** (chip Sun Sparc)...

➤ Procesadores con múltiples unidades funcionales

- **1967** (IBM 360/91) ...

➤ Procesadores/cores superescalares

- **1989** (chip Intel 960CA (3)). **1990**: (chip IBM Power1 (4)). **1992**: (chips DEC α 21064 (2/4), HP PA 7100 (2/2), Sun SuperSparc (3/5)) ...

➤ Procesadores/cores VLIW

- **1990** (*chip DSP Intel i860 (2)*). **1997** (*chip DSP TMS320C6x (8)*). **2001** (chip Intel Itanium)...

NOTA: Destacado en cursiva las primeras implementaciones y en color más claro los chip de propósito específico

Nota histórica. TLP (*Thread Level Parallelism*)

TEMA 3

- TLP explícito con una instancia de SO:
 - Multithread grano fino (FGMT)
 - **1975** (*Denelcor HEP*). **2005** (chip Sun UltraSPARC T1) ...
 - Multithread grano grueso (CGMT)
 - **1990** (*MIT Alewife*). **2000** (chip IBM PowerPC RS64 IV (2)). **2006** (chip Intel Itanium Montecito (2)) ...
 - Multithread simultánea (SMT)
 - **2002** (chip Intel Pentium 4/Xeon Hyper-Threading). **2004** (chip IBM Power5) ...
 - Multiprocesadores en un chip (CMP) o multicores
 - **2001** (chip IBM Power4). **2004** (chip Sun UltraSPARC IV). **2006** (chip Intel Core Duo. **2008** (chip Intel Celeron Dual-core) ...
 - Multiprocesadores
 - **1962** (*Burroughs D825 - red barras cruzadas*). **1966** (*UNIVAC 1108 - red bus*). **1985** (*IBM RP3 - red multietapa*). **1996** (*SGI Origin 2000 -CC-NUMA, red estática+multietapa*). **2006** (SGI Altix 4000) ...
- TLP explícito con múltiples instancias del SO (multicomputadores) IC.SCAP
 - **1985** (*Intel iPSC1 - i286+red estática con Ethernet*) ... cualquier cluster

NOTA: Destacado en cursiva las primeras implementaciones y en color más claro los chip de propósito específico

Para ampliar

➤ Páginas Web:

- “Embedded Processors. 2010-14 Global Market Demand Analysis.” VDC Research Group.

<http://www.vdcresearch.com/Documents/proposal/pro-attachment-2637.pdf>

➤ Artículos de Revistas:

- Ranakrishna Rau, B.; Schlansker, M.S.: “Embedded Computer Architecture and Automation”. IEEE Computer, pp.75-82. Abril, 2001.