ARQUITECTURA DE COMPUTADORES II

AUTORES:
David Expósito Singh
Florin Isaila
Daniel Higuero Alonso-Mardones
Javier García Blas
Borja Bergua Guerra

Área de Arquitectura y Tecnología de Computadores
Departamento de Informática
Universidad Carlos III de Madrid

Julio de 2012

TEMA 4:
MP DE MEMORIA DISTRIBUIDA

Índice (I)

1. Introducción
   - Definición
   - Mercado
   - Uso
   - Nomenclatura

2. Problemática
   - 'Escalabilidad'
   - Comunicación
   - Organización de los nodos
   - Modelo de computación

Índice (y II)

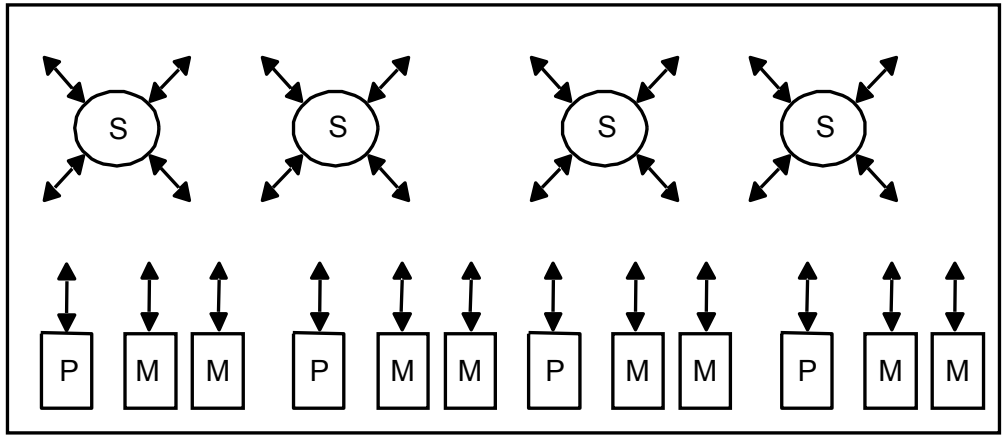
3. Programación
   - Primitivas de comunicación
   - Implementaciones
4. Ejemplos

Introducción

- Definición:
  - **MP de Memoria Distribuida** (*Distributed Memory MP*): un procesador no puede acceder a la memoria de otro procesador.
  - Cada procesador: **propio mapa de direcciones.** Las direcciones que genera accederán a su **memoria local.**
  - Los procesos que se ejecutan en procesadores distintos sólo se pueden **comunicar** mediante el **paso de mensajes.**
  - Red de **interconexión no** es nunca un **bus**, sino estructuras más sofisticadas: hipercubos, mallas 2-D y 3-D, etc.
- Ventajas:
  - 'Escalabilidad': (pero no más que los NUMAs)

Problemática (I)

- 'Escalabilidad'
  - Redes de Interconexión.
  - Límite de los buses: ancho de banda limitado: un único conjunto común de "elementos conductores".
  - ¿Cómo conseguir 'escalabilidad' en la red de comunicación?
    - Conjunto de caminos ("elementos conductores") independientes: *switches* entre ellos.

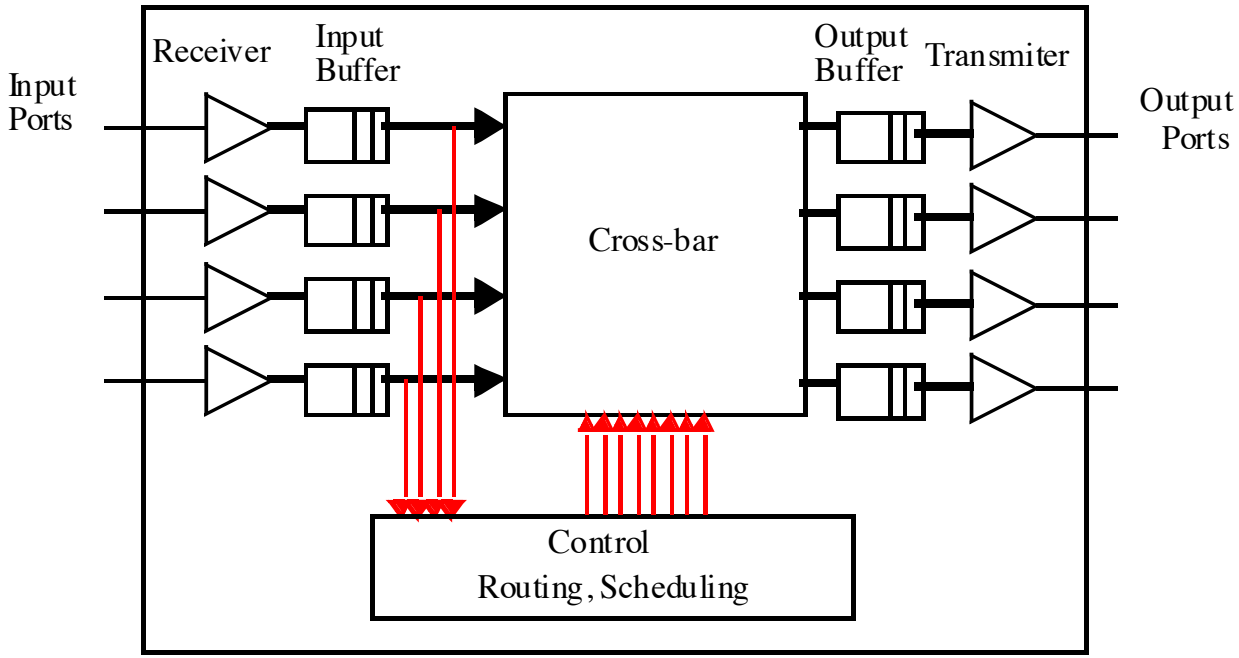


Problemática (II)

- Cuestión crucial: reducir la **frecuencia de comunicación** (costosa) entre procesadores?
  - **Particionamiento** de los programas.
  - **Asignación** de procesos para minimizar la comunicación inter-procesador → Problema NP-Completo.
- Sistema de comunicación:
  - Crucial conseguir una minimización del tiempo de comunicación
    - Red de interconexión
    - Soporte *hardware* para el paso de mensajes entre procesadores.
  - **Red de interconexión**: conjunto de conexiones directas *punto-a-punto*, o **enlaces** o **canales**, entre nodos.
  - Los mensajes divididos en **paquetes.**
  - Emisor → Camino de comunicación → Receptor.
  - **Camino de comunicación**: Procesadores intermedios + enlaces directos.

Problemática (III)

- Estructura de los *switch*:



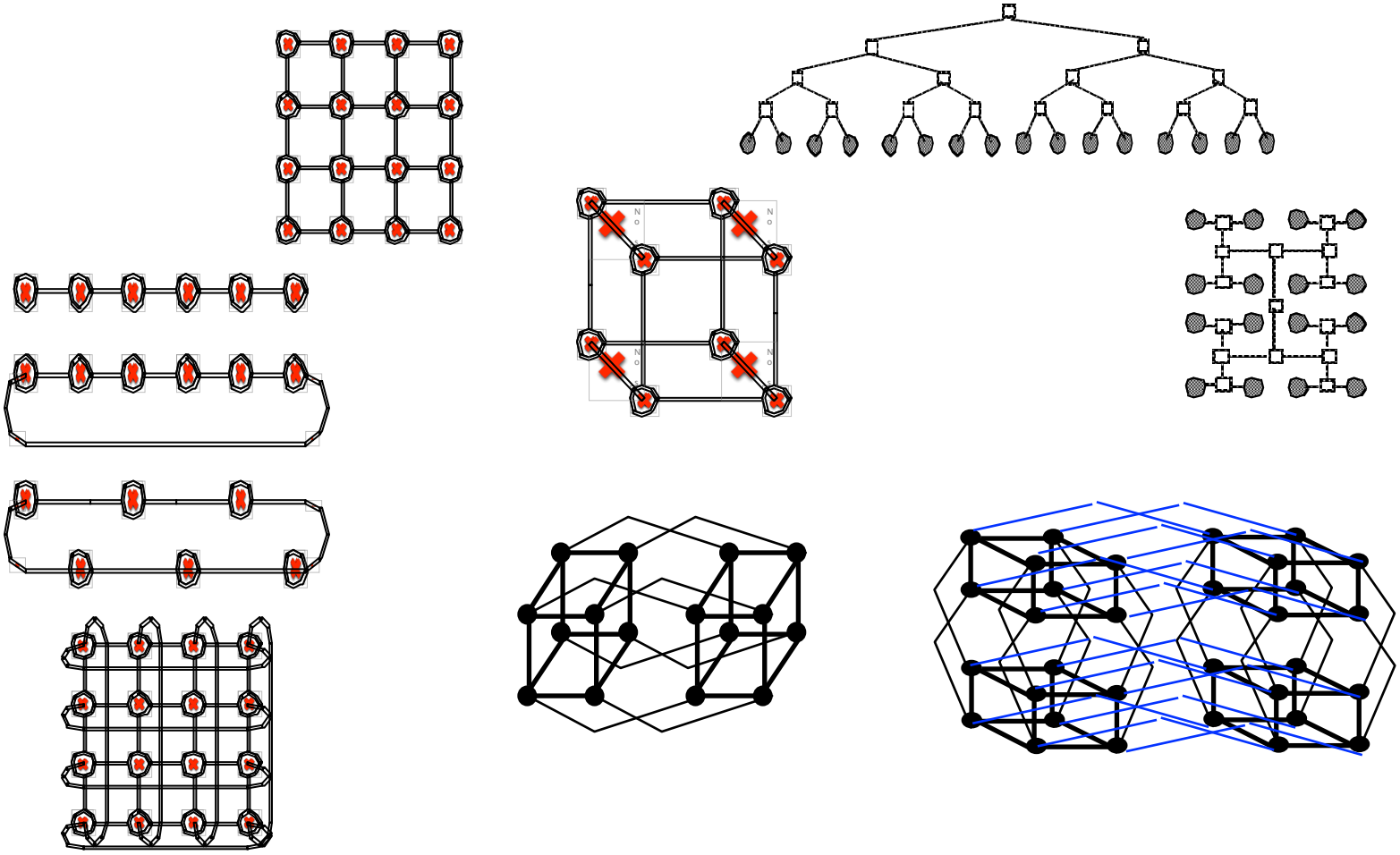
Problemática (V)

- La comunicación viene determinada por tres elementos principales:
  - La **topología** de la red.
  - La técnica de **conmutación** (*switching*).
  - El protocolo de **encaminamiento** (*routing*).

- 1. Topología

  Ejemplos: array lineal, anillos, estrellas, árboles, mallas 2-D cubo 3-D, hipercubos, etc.

Topología: ejemplos

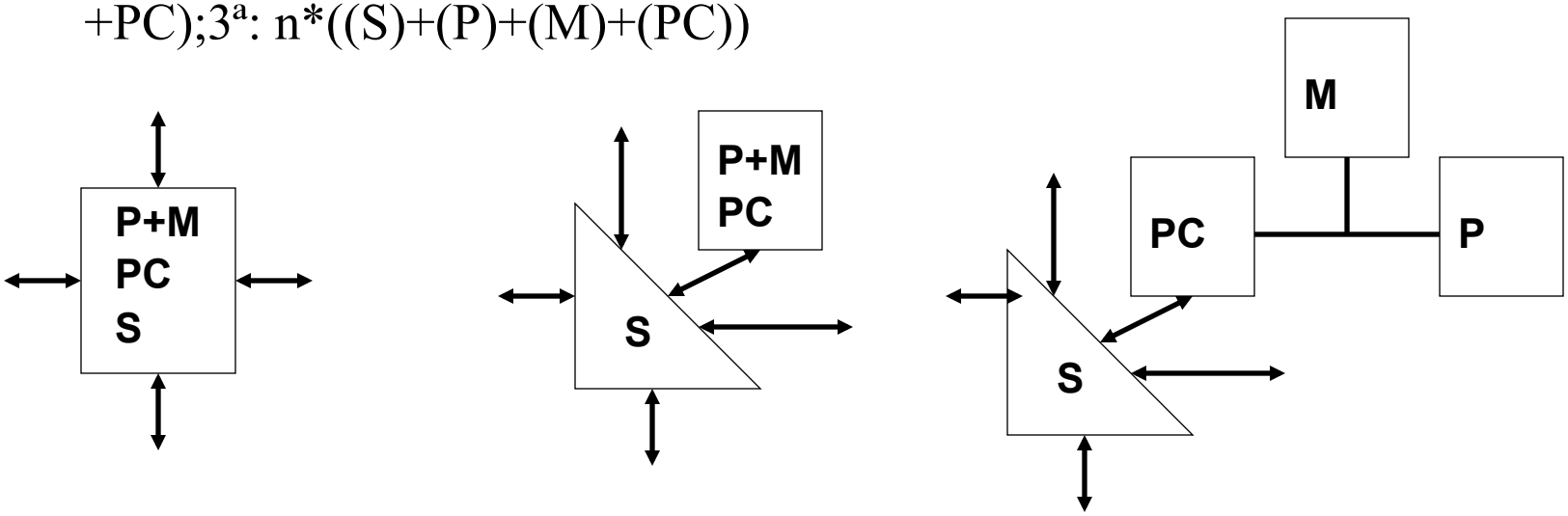


Problemática (VI)

- 2. Técnicas de conmutación
  - Modo en que los mensajes se retiran del buffer de entrada y se llevan al de salida.
  - Principales modalidades:
    - *Store-and-forward* o conmutación de paquetes.
    - Conmutación de circuitos: fase de establecimiento; no empaquetado, único mensaje; enlaces dedicados (un solo mensaje).
    - *Wormhole*: mensaje descompuesto en *flits* (*flow control digits*), varias versiones del mismo mensaje. Se consigue evitar conflictos, una misma línea compartida por varios mensajes (sus flits) a la vez.
- 3. Encaminamiento (*routing*)
  - Búsqueda del camino entre el origen y el destino.
  - Modalidades:
    - Determinista / Adaptativo.
    - Minimal / No minimal .

Problemática (VII)

- Organización de cada nodo:
  - Elementos:
    - **P**: Procesador de "cómputo" (el de siempre) + M: Memoria.
    - **PC**: Procesador de comunicación: mensajes.
    - **S**: Switch.
  - Evolución: generaciones:
    - 1ª: n*(P+M+PC+S);2ª: n*(S+(P+M+PC)) / Conm Central.+ n*(P+M+PC);3ª: n*((S)+(P)+(M)+(PC))



Problemática (VIII)

- Modelo de computación en la arquitectura
  - Lo más habitual, lenguajes convencionales de programación + librerías de comunicaciones.
  - No convencionales:
    - Basado en **CSP** de Hoare (78): **Occam** transputers: Supernodos: Gigacube
    - Otras (experimentales): Caltech Cosmic Cube, Mosaic C, J-Machine (MIT), Gigacube
- Clasificación: organización del nodo + modelo de computación: CSP

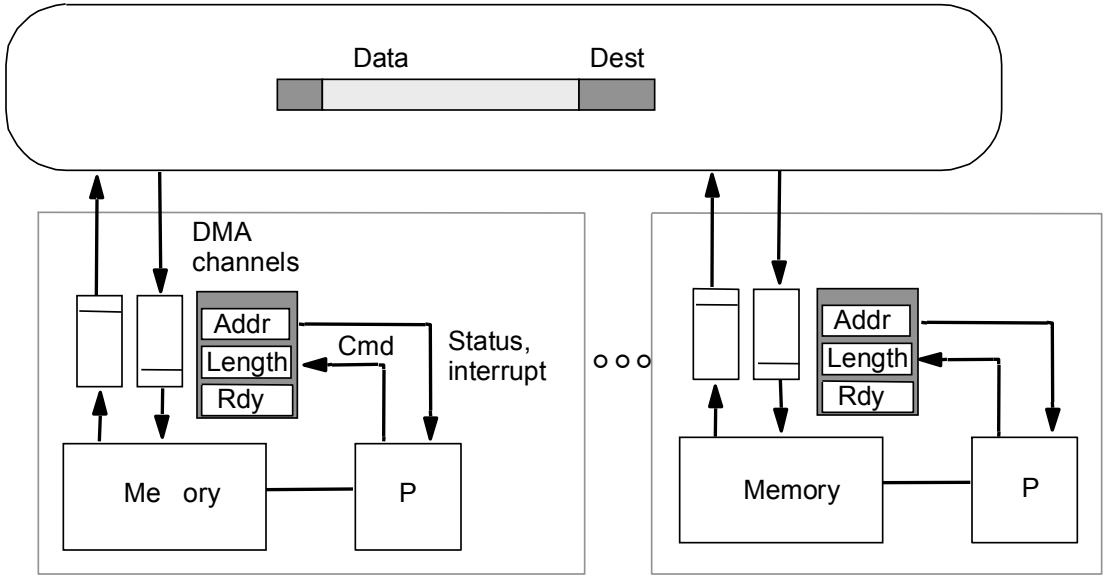
Modelo de computación en la arquitectura

Supernodos

SN 9800

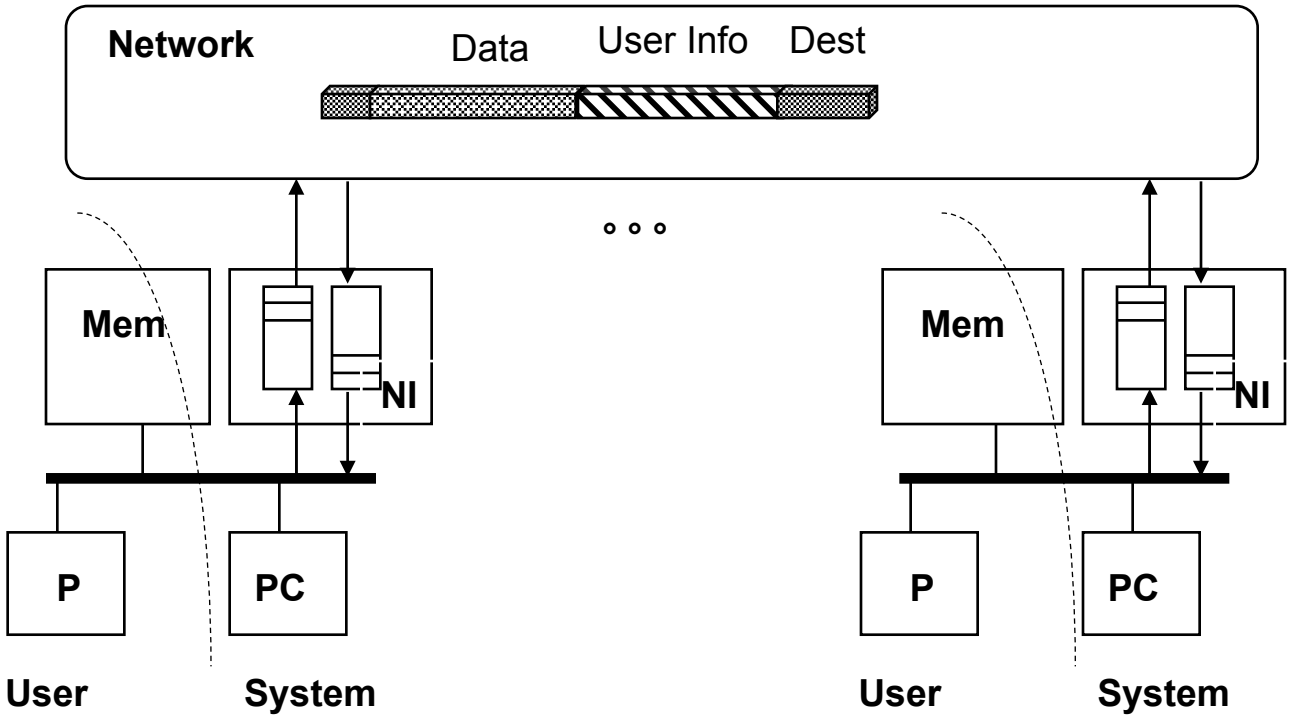
Problemática (IX)

- Tipo de organización determina la información "de control" en los mensajes: protección. Sistema/Usuario: véase Culler *et al.*: muy complejo.
- Ejemplo 1:
  - 1ª generación
  - Los PC se llaman hardware para "DMA" [no E/S]



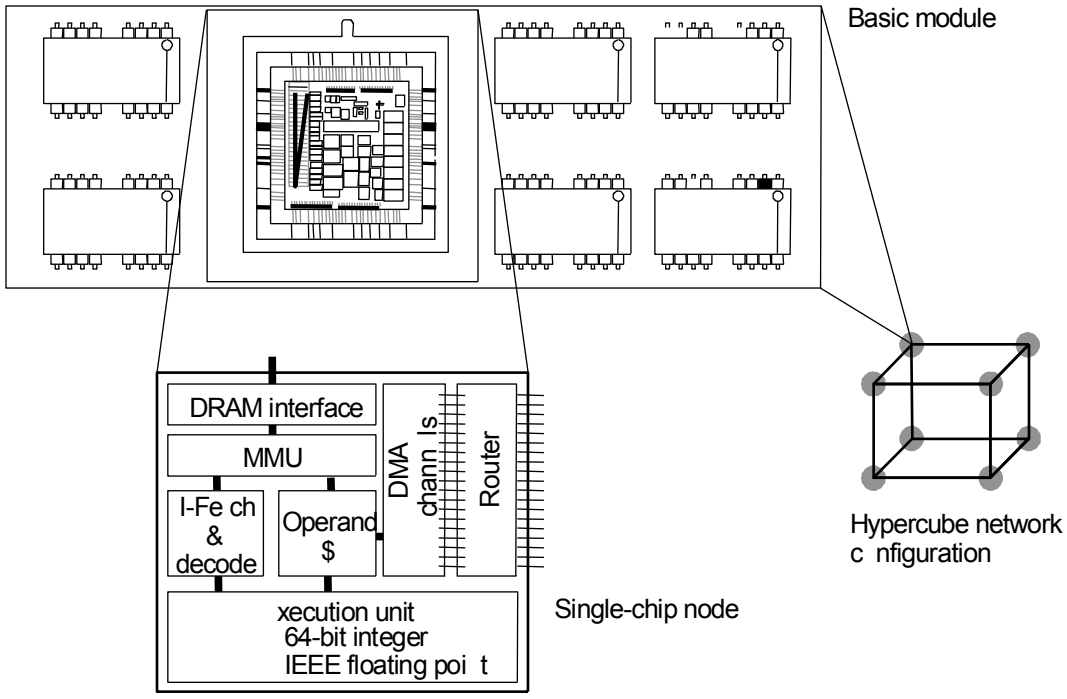
Problemática (X)

- Ejemplo 2:
  - 3ª generación
  - El PC es otro procesador "estándar"



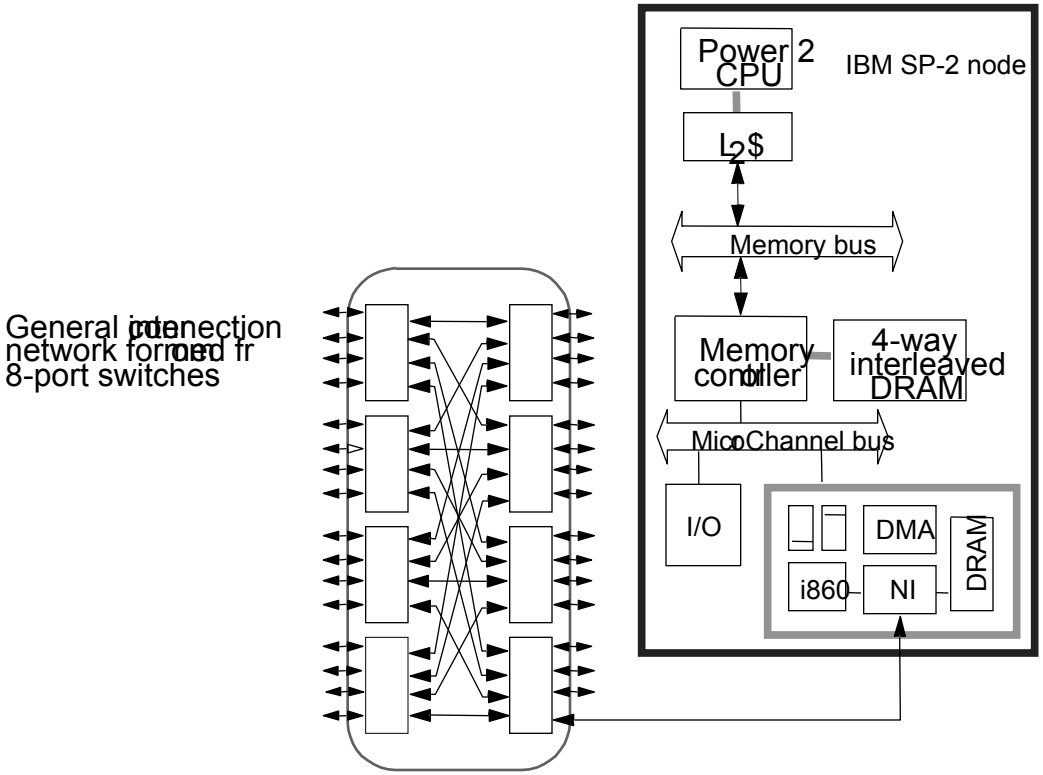
Ejemplo: nCUBE/2

- Escalabilidad: 1024 nodos
- S + (P+M+PC)
- P+ PC: *ad hoc*



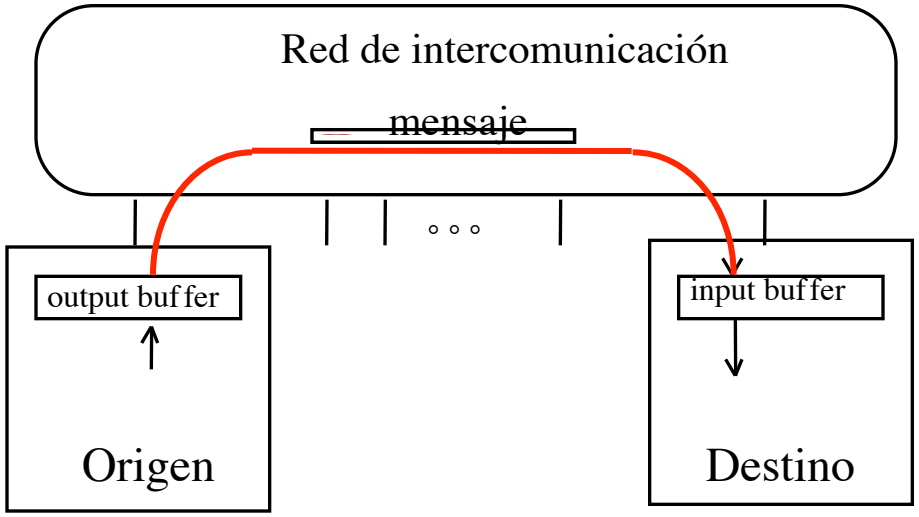
Ejemplo: IBM SP 2

- (S) + (P)+(M)+(PC)
- P: Power 2
- PC: Intel 860



Programación

- **Primitiva de comunicación** (*network transaction primitive*)
  - Leer del buffer de salida de un procesador y escribir en el buffer de entrada de otro.



Programación (II): Implem. M Compartida

- Implementación de modelo de programación de M Compartida:
  - Mensajes de petición/respuesta: escrituras, mensajes de ACK.

(1) Inicio

(2) Trad. Dirección

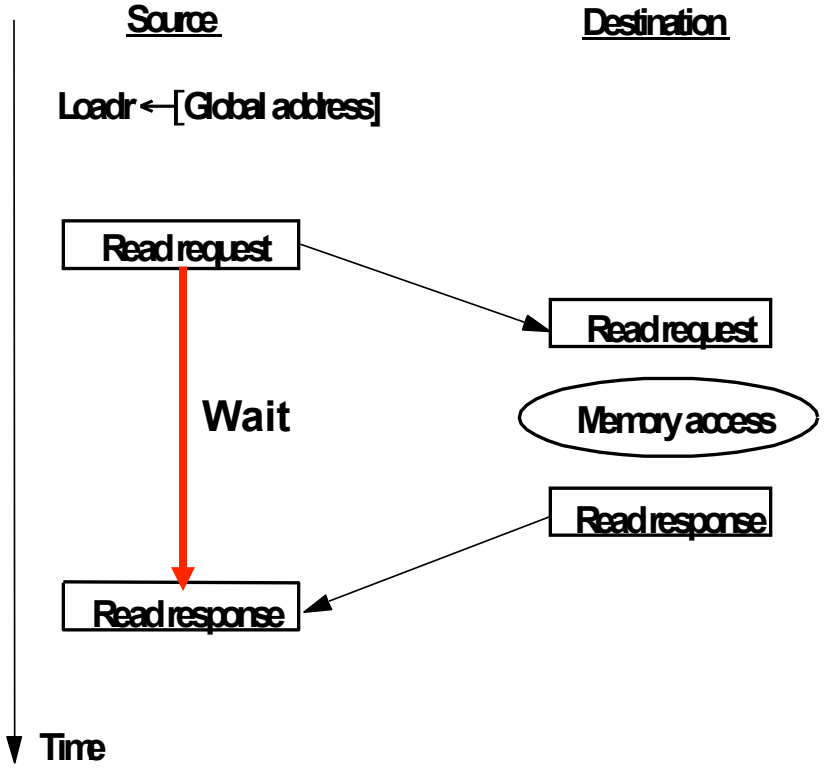
(3) Comprob. rem./local

(4) Transac. de Req.

(5) Acceso remoto

(6) Transac. de reply

(7) Fin



Programación (III): Impl. Paso Msg

- Implementación de modelo de programación de Paso de Mensajes:
  - Síncrono
  - Asíncrono

1. Síncrono

(1) Send de inicio

(2) Trad. Dirección en Psrc

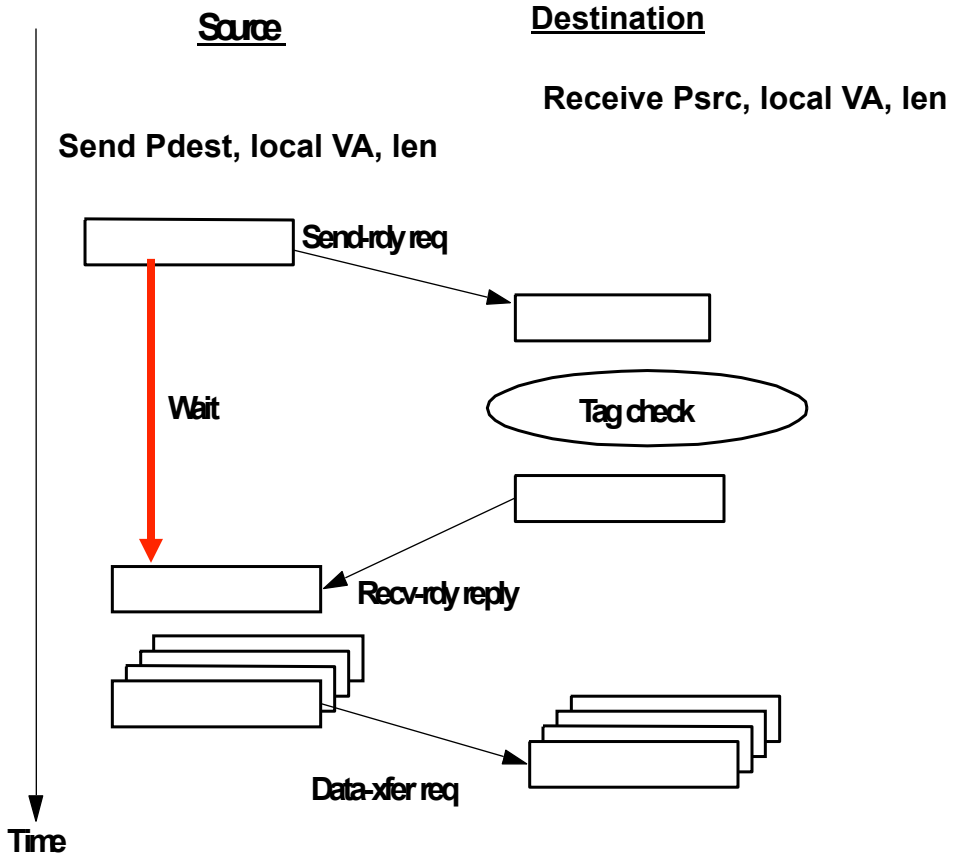
(3) Comprob. rem./local

(4) Envío req, Send-ready

(5) Comprob. remota de "match" con el Receive

(6) Transac. de reply

(7) Transf. de Datos: Source VA -> Dest VA



Programación (IV): Impl. Paso Msg

2. Asíncrono

(1) Send de inicio

(2) Trad. Dirección en Psrc

(3) Comprob. rem./local

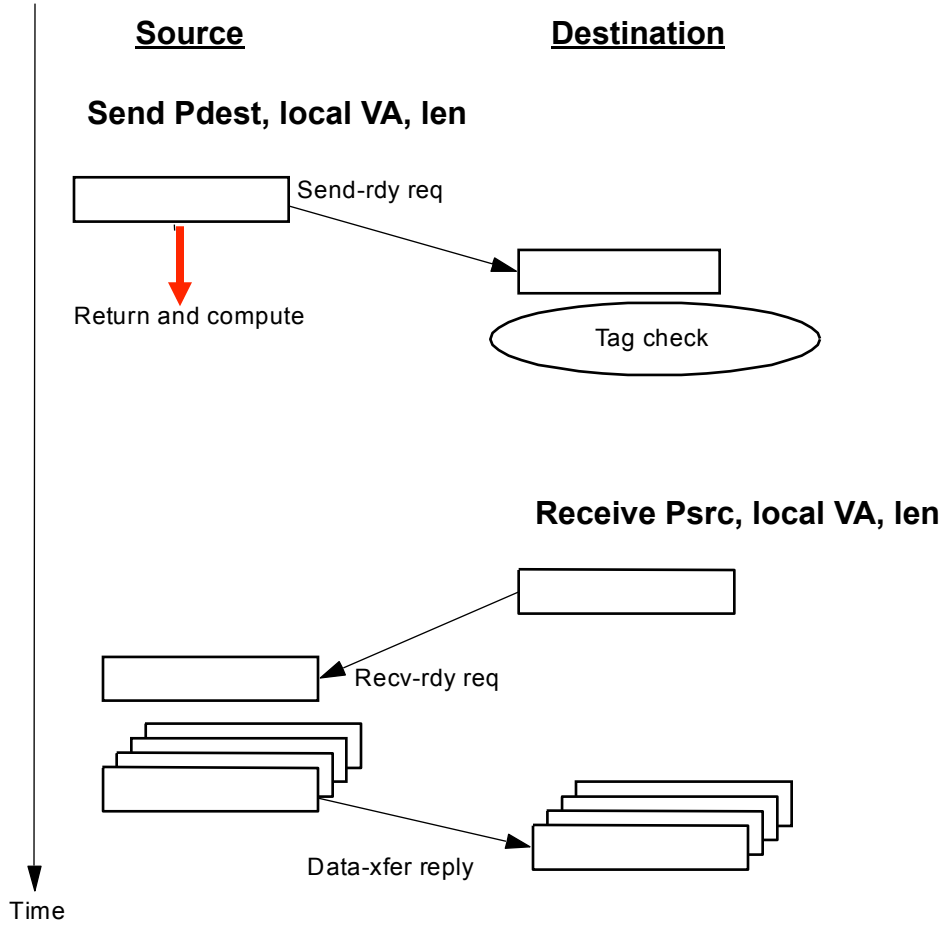
(4) Envío req, Send-ready

(5) Comprob. remota de receive pendiente: fallo, registra send-rdy

(6) Receive

(7) Req. de Receive-ready

(8) Transf. de Datos: Source VA -> Dest VA



Ejemplos

Máquina	P	PC	Topología
nCUBE 2	propio		hipercubo
iPSC-2	I 386		hipercubo
Paragon	I 860	I 860	malla 2-D
Meiko CS-2	Sparc	propio	árbol plano
IBM SP 2	Power 2	I 860	árbol plano
SN 9800	T9000	T9000	switch jerar.

Supernodo

Transputer 9000

Arquitecturas escalables

- Escalabilidad: capacidad de un sistema de incrementar añadiendo procesadores, memoria, E/S, etc.
- Aspector importantes
  - Ancho de banda aumenta con el número de procesadores
  - Latencia no aumenta (o lo hace lentamente)
  - Coste aumenta lentamente con número de procesadores.
  - Ubicación física de los recursos.

Escalabilidad de un bus

Characteristic	Bus
Physical Length	~ 1 ft
Number of Connections	fixed
Maximum Bandwidth	fixed
Interface to Comm. medium	extended memory interface
Global Order	arbitration
Protection	virtual -> physical
Trust	total
OS	single
comm. abstraction	HW

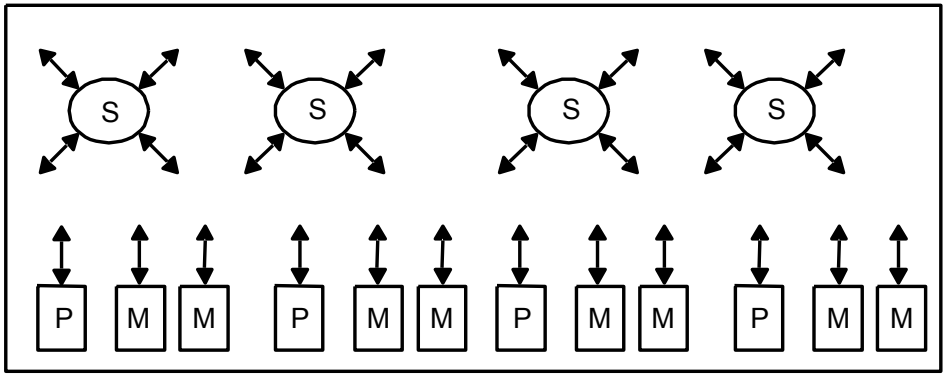
- Configuraciones reducidas son económicas

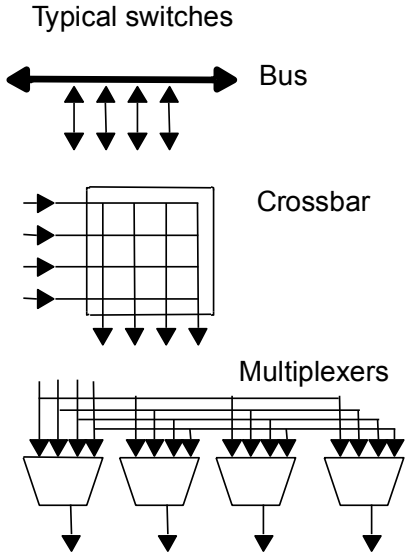
Ordenadores conectados en red LAN

Characteristic	Bus	LAN
Physical Length	~ 1 ft	KM
Number of Connections	fixed	many
Maximum Bandwidth	fixed	???
Interface to Comm. medium	memory interface	peripheral
Global Order	arbitration	???
Protection	Virtual -> physical	OS
Trust	total	none
OS	single	independent
comm. abstraction	HW	SW

- Límite no claro en escalabilidad física y seguridad.
- Baja fiabilidad

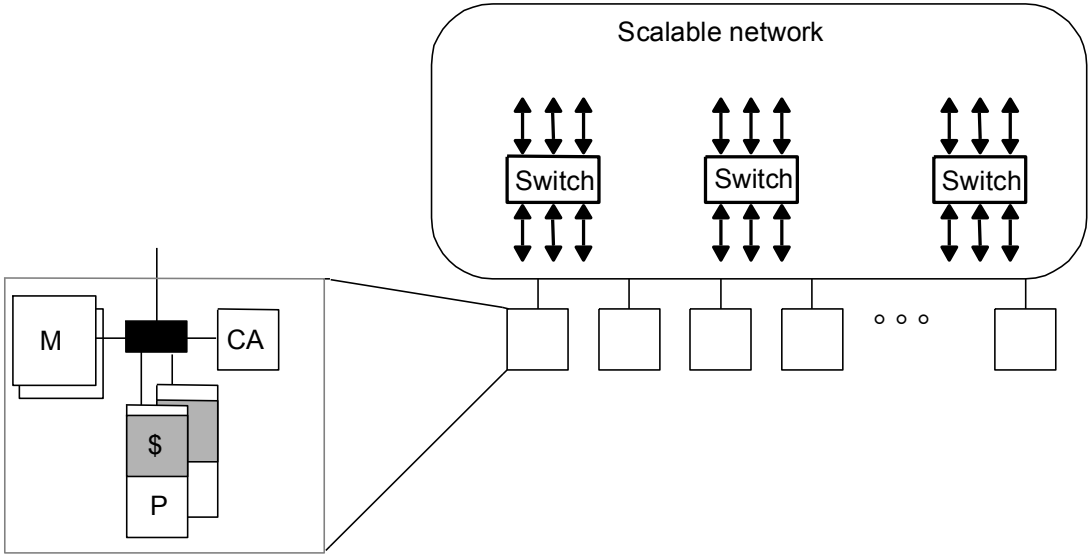
Escalabilidad de ancho de banda





- Limitación de ancho de banda: número simple de cables de red.
- Solución: añadir más switches.

Multiprocesador genérico



- Memoria local reduce latencia en acceso a memoria (alternativa cc-NUMA sin memoria local).

Escalabilidad ancho de banda

- Varias vías independientes de comunicación.
  - Más caminos de datos.
- Comunicaciones independientes
- No arbitración global.
- Effecto transacción sólo visible a nodos involucrados.
  - Dificultad para hacer broadcasts (operación trivial en buses).

Escalabilidad de latencias

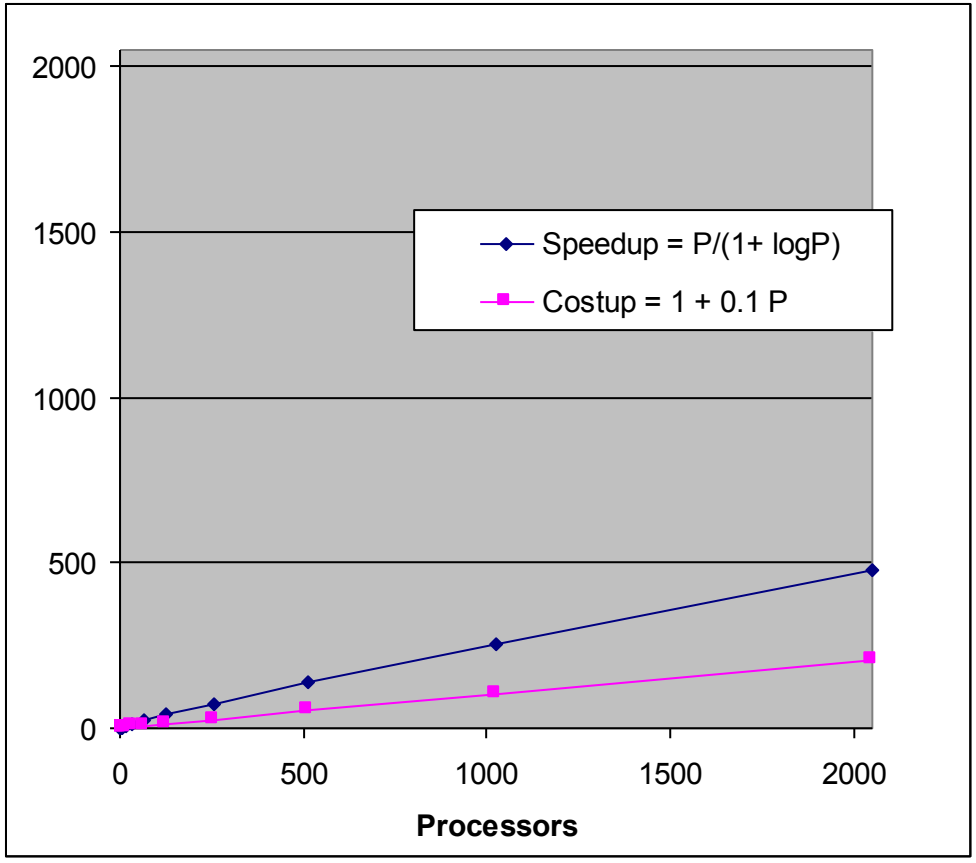
$$\text{latencia}(n)_{o \rightarrow d} = \textit{overhead} + \text{ocupación del canal} + \text{retardo de encaminamiento} + \text{retardo por “contención”}$$

- Reducción del retardo de encaminamiento

Escalabilidad coste

- Coste(p,m) = coste_fijo + coste_incremental(p,m)
- Arquitecturas basadas en bus
  - No hay coste adicional al aumentar número CPUs
- Arquitecturas escalables
  - Es necesario añadir elementos de red.
- Parallel efficiency(p) = Speedup(p) / p
- Costup(p) = Cost(p) / Cost(1)
- Cost-effective: Speedup(p) > Costup(p)

Cost Effective?



- 2048 procesadores: 475 aceleración con 206x coste

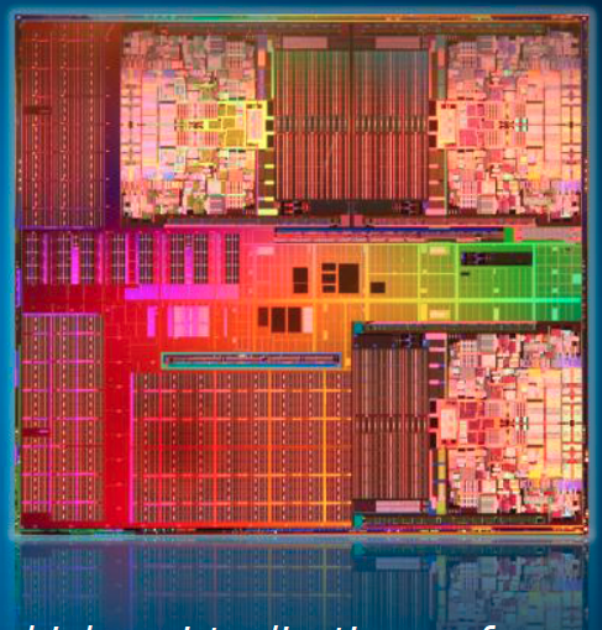
Escalabilidad física

- A nivel de chip
  - Multicore
- A nivel de nodo
  - Varios procesadores por placa
- A nivel de sistema
  - Clusters, supercomputadores

Escalabilidad a nivel de chip

Intel® Xeon® 7400-based Server Platform Dunnington Extends Caneland Technology Leadership

- Latest Intel virtualization capabilities
- 6 cores,16 MB L3 cache
  - 4-core/large cache versions available
- Socket compatible with Caneland platform
- 45nm Hi-K technology
- 1.9 billion transistors
- Introduction Sep. 2008



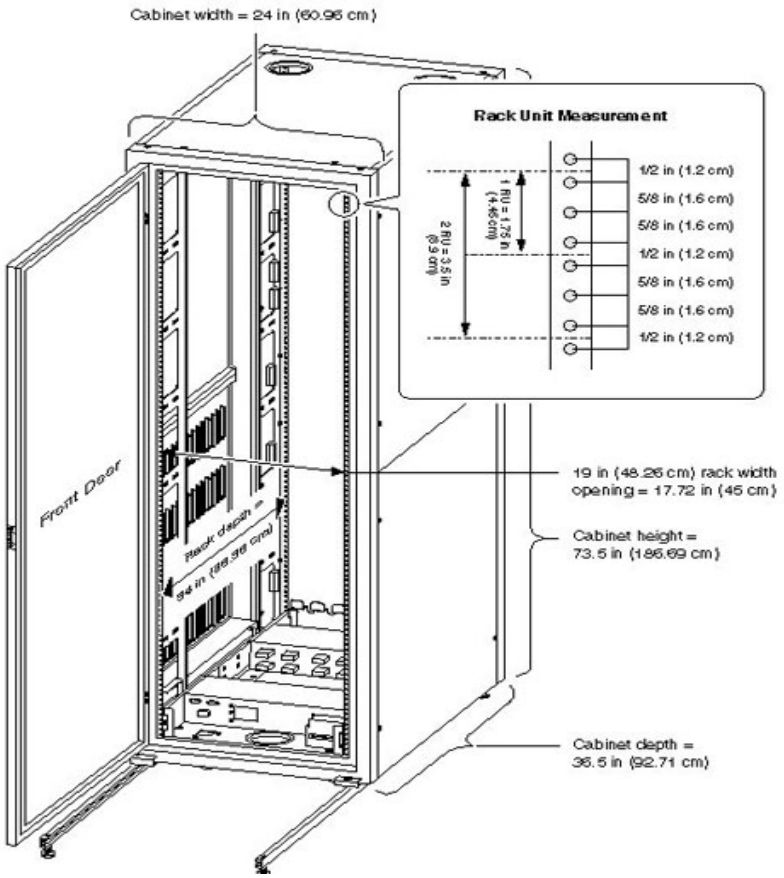
Caneland with Dunnington delivers higher virtualization performance for consolidation and data demanding applications offering more cores, cache and large memory footprint

intel

Clúster con baja escalabilidad física

Packaging/Physical Design

- Rack Mounted PC Boards



Packaging/Physical Design

- Blades
  - Shared power supply, cooling, etc.

(C) 2007 J. E. Smith ECE/CS 757

IBM BladeCenter

- **Motivation –**
  - Circa 1999 -- Rack mounted servers becoming difficult
    - 42 1u servers require a lot of cables in a very small space
    - No redundant power supplies/cooling make servicing difficult
- **Solution : Use server blades**
  - Similar to technology used in telecommunications/network switches and hubs
  - Chassis w/ backplane for interconnect (network and power)
  - Shared (and redundant) power supplies/ cooling

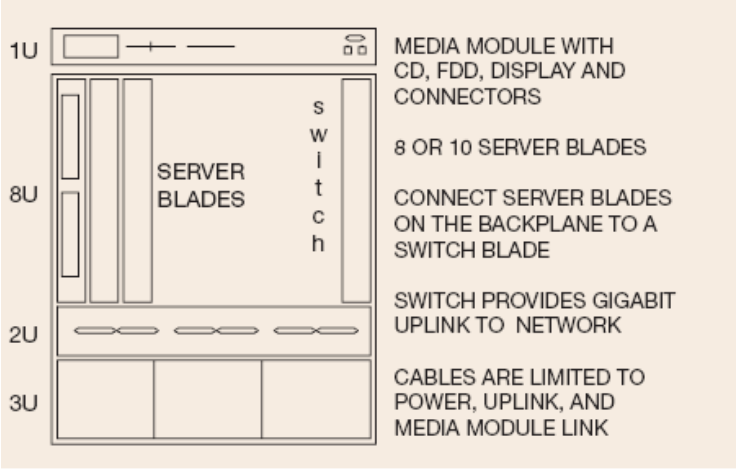
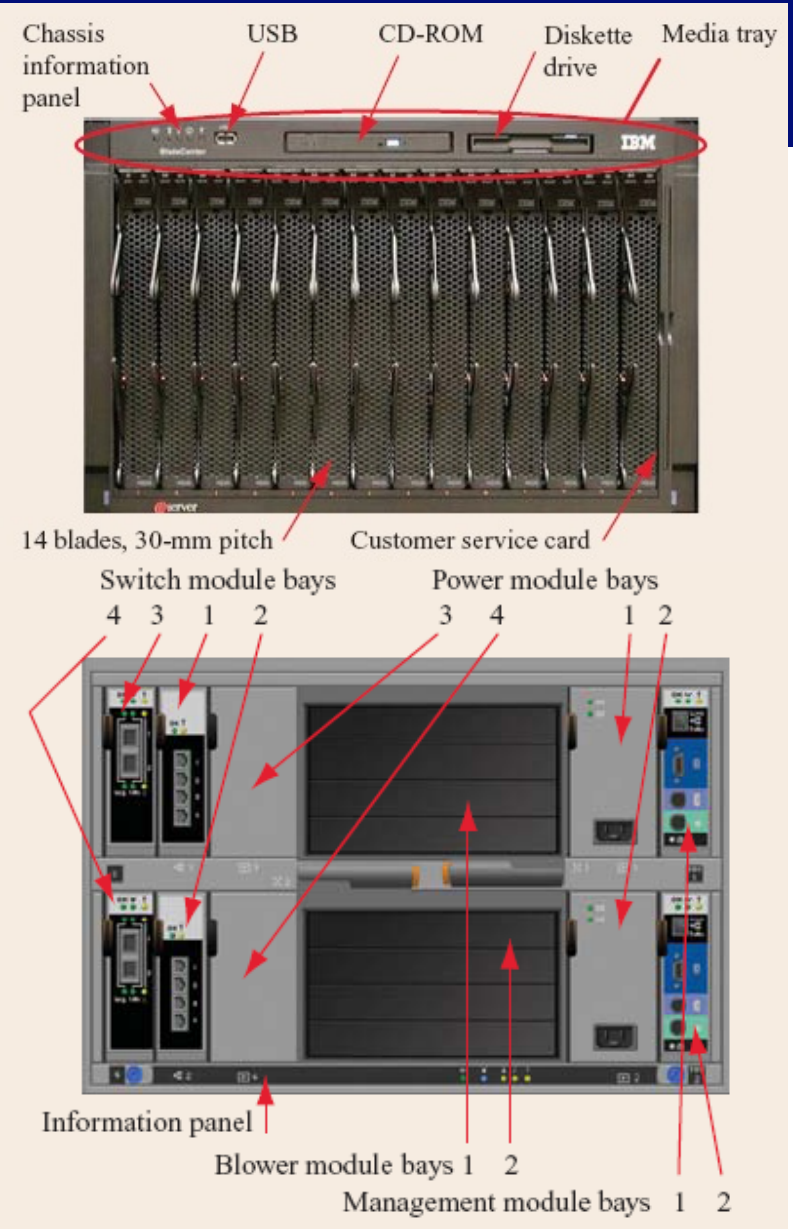


Figure 1

The earliest concept drawing of the BladeCenter architecture, created on October 20, 1999.

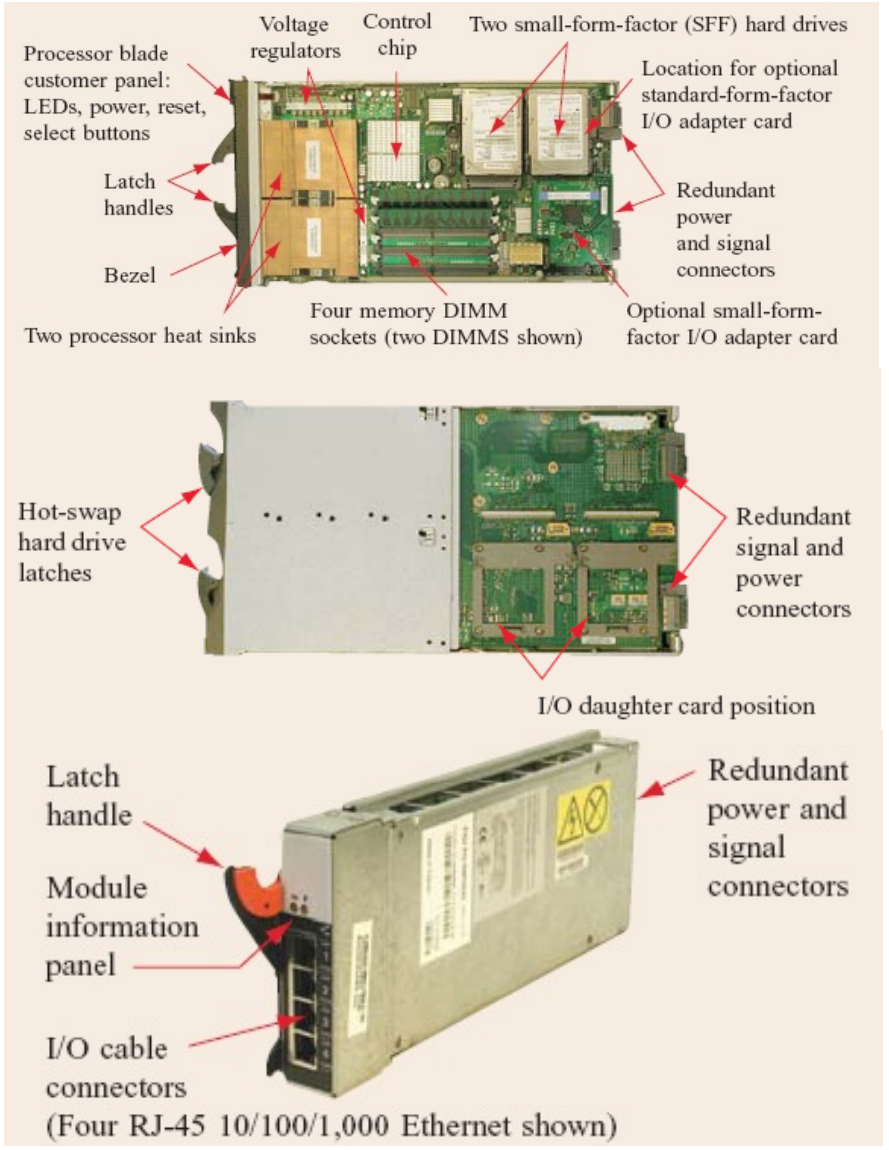
Hardware

- Midplane
- 14 server blades in front
  - Plus shared media stuff
  - Double density of 1U server
- Switch, Power supply modules, etc. in back
- Airflow front to rear
- Redundant power supplies



Hardware, contd.

- Processor Blade
- Storage Expansion Blade
- Switch Module

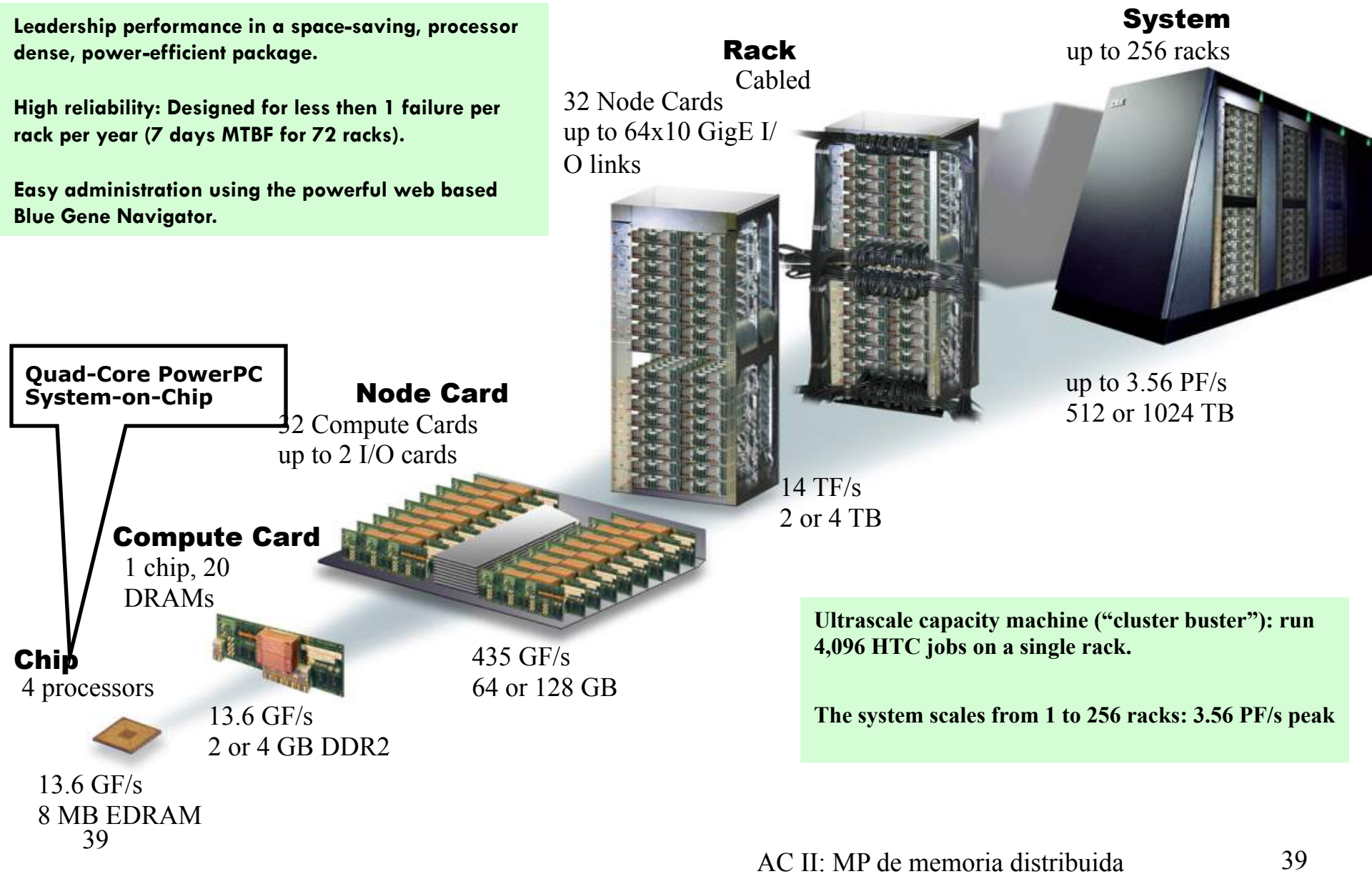


Blue Gene/P

Leadership performance in a space-saving, processor dense, power-efficient package.

High reliability: Designed for less then 1 failure per rack per year (7 days MTBF for 72 racks).

Easy administration using the powerful web based Blue Gene Navigator.



Ultrascale capacity machine ("cluster buster"): run 4,096 HTC jobs on a single rack.

The system scales from 1 to 256 racks: 3.56 PF/s peak

C. CC.: bibliografía

- Libros “de texto”:
  - El Culler *et al*. y el Sima *et al*.