

# Memoria compartida distribuida

## Arquitectura de Computadores

J. Daniel García Sánchez (coordinador)  
David Expósito Singh  
Javier García Blas  
J. Manuel Pérez Lobato

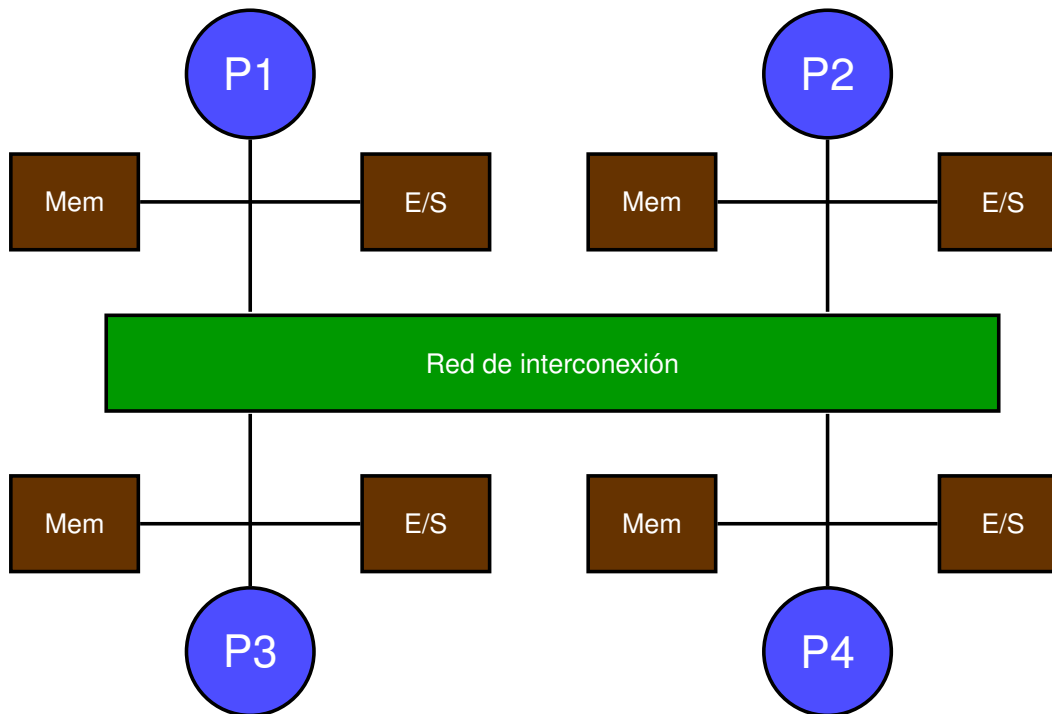
Grupo ARCOS  
Departamento de Informática  
Universidad Carlos III de Madrid

- 1 Introducción a memoria compartida distribuida
- 2 Bases del protocolo de directorio
- 3 Protocolo basado en directorio
- 4 Conclusión

# Protocolos de espionaje y escalabilidad

- **Problemas** de protocolos de espionaje.
  - Requiere comunicación con todas las cachés:
    - En cada fallo de caché.
    - En cada escritura de dato compartido.
  
- ¿Qué **ventaja** tienen los protocolos de espionaje?
  - Ausencia de estructura de datos centralizada.
    - Bajo coste de implementación.
  
- ¿Qué **inconveniente** tiene los protocolos de espionaje?
  - Ausencia de estructura de datos centralizadas.
    - Comunicaciones limitan la escalabilidad.

# Modelo básico de DSM



- **Necesidad de eliminar tráfico de coherencia.**

# Clases de protocolos de coherencia

## ■ **Espionaje** (*snooping*):

- Cada caché mantiene el estado de compartición de cada bloque que tiene.
- Las cachés accesibles mediante medio de multidifusión (bus).
- Todas las cachés monitorizan si tienen una copia del bloque.

## ■ **Basados en directorio**:

- El estado de compartición se mantiene en un directorio.
- **SMP**: Directorio centralizado en memoria o en caché de más alto nivel.
- **DSM**: Para evitar cuello de botella se usa un directorio distribuido (más complejo).

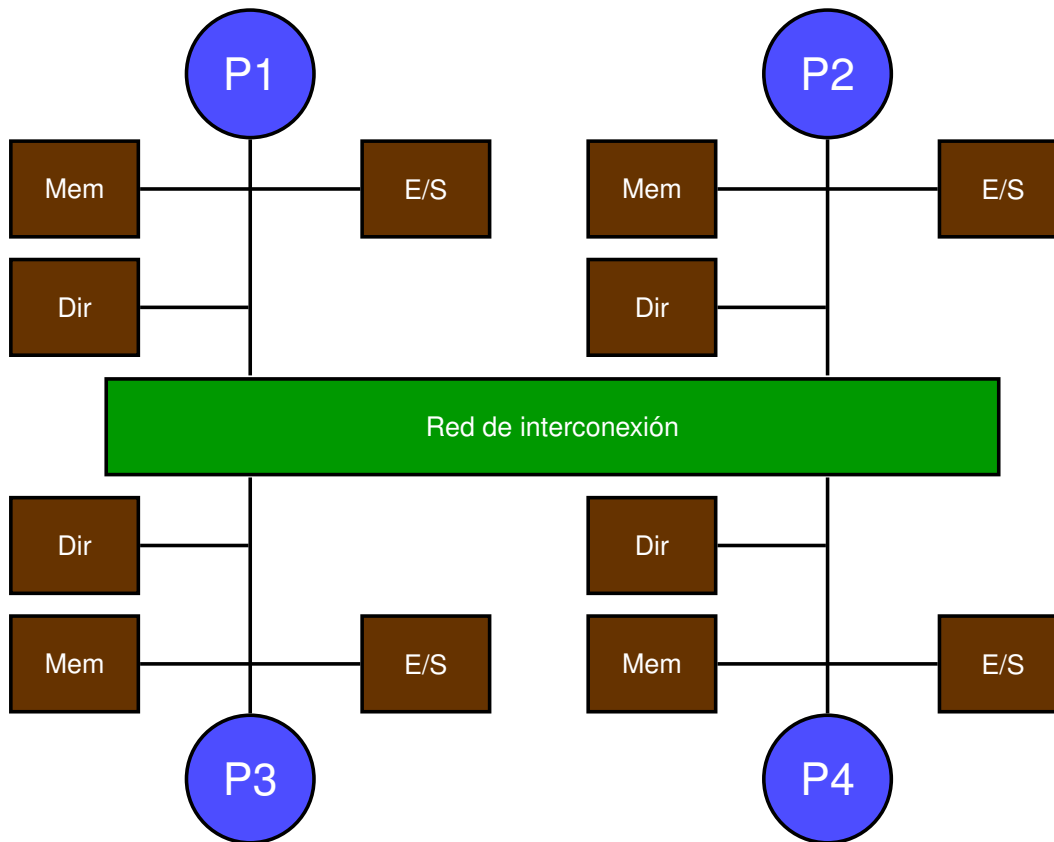
# Protocolo basado en directorio

- **Idea**: Mantener el **estado** de cada bloque de caché.
  - ¿Qué cachés tiene copia del bloque?
  - Bits de estado del bloque.
- Multicores con caché externa compartida.
  - **Vector de bits** de longitud igual a número de cores.
    - Indica que cachés privadas pueden tener copia del bloque.
    - Solamente se envía invalidación a cachés marcadas en mapa de bits.
  - Esquema funciona bien dentro de un único multicore.
  - **Ejemplo**: **Intel Core i7**.

# Directorio centralizado y escalabilidad

- Un directorio centralizado evita broadcast pero
  - Se convierte en cuello de botella.
  - Problema de escalabilidad con número de procesadores.
  
- **Solución: Directorio distribuido.**
  - Distribuir el directorio con la memoria.
  - Cada directorio tiene información de la memoria local asociada.
    - Siempre se sabe a qué directorio ir.
  - Distintas peticiones de coherencia van a distintos directorios.

# Direcctrio distribuido





- 1 Introducción a memoria compartida distribuida
- 2 Bases del protocolo de directorio
- 3 Protocolo basado en directorio
- 4 Conclusión

# Directorio

- Operaciones básicas.
  - Tratamiento de fallo de lectura.
  - Tratamiento de escritura en un bloque compartido limpio.
- El directorio debe mantener el **estado de cada bloque**:
  - **Compartido**: Uno o más nodos tienen el bloque en caché y el valor en memoria está actualizado.
  - **No cacheado**: Ningún nodo tiene una copia del bloque.
  - **Modificado**: Solamente un nodo tiene copia del bloque en caché y lo ha escrito.
    - Valor en memoria no actualizado.
- **Además**:
  - Mapa de bits con información de nodos que tienen copias del bloque.

# Mensajes

Mensaje	Fuente	Destino	Contenido	Función
Fallo lectura	Caché local	Directorio local	P,A	P tiene fallo de lectura en A. Pedir dato y P compartidor.
Fallo escritura	Caché local	Directorio local	P,A	P tiene fallo de escritura en A. Pedir dato y P propietario.
Invalidación	Caché local	Directorio local	A	Invaldar A en todas las cachés.
Invalidación	Caché local	Directorio remota	A	Invaldar copia compartida.
Captación	Directorio local	Caché remota	A	Capta bloque. Estado a compartido.
Captación/ Innvalidación	Directorio local	Caché remota	A	Capta bloque. Invalida bloque.
Respuesta valor dato	Directorio local	Caché local	D	Devolver valor a directorio.
Post-escritura dato	Caché remota	Directorio local	A,D	Post-escritura de dato.

**P** → **Nodo**, **A** → **Dirección**, **D** → **Dato**

- 1 Introducción a memoria compartida distribuida
- 2 Bases del protocolo de directorio
- 3 Protocolo basado en directorio
- 4 Conclusión

# Transición de estados

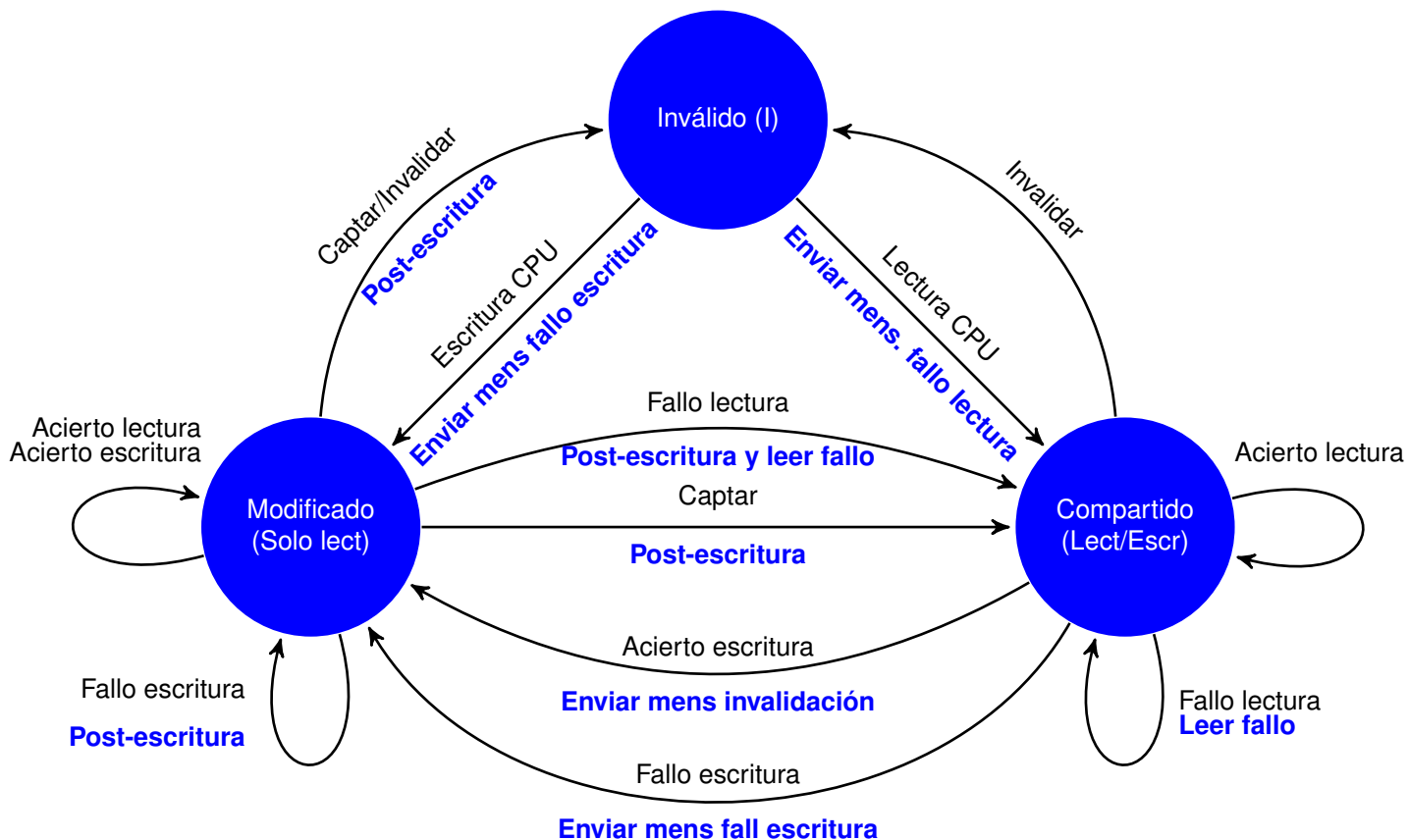
## ■ En chips multicore:

- La **coherencia interna** se mantiene mediante **directorio centralizado**.
- El **mismo directorio** puede actuar como **directorio local** en **DSM**.

## ■ Implementación del protocolo:

- Transición de estados de **caché local**.
  - Envían peticiones a **directorio local**.
- Transición de estados del **directorio**.

# Transición de estados de caché individual



# Entrada no cacheada

- El valor de memoria está actualizado.
- **Peticiones:**
  - **Fallo de lectura:**
    - Se envía dato de memoria a nodo peticionario.
    - Nodo peticionario es el único en estado compartido.
    - Estado pasa a compartido.
  - **Fallo de escritura:**
    - Se envía dato de memoria a nodo peticionario.
    - El bloque se pasa a estado exclusivo.
    - Nodo peticionario es el propietario.

# Entrada compartida

- El valor de memoria está actualizado.
- **Peticiones:**
  - **Fallo de lectura:**
    - Se envía el dato de memoria al nodo peticionario.
    - El nodo peticionario se añade al conjunto de nodos de la entrada.
  - **Fallo de escritura:**
    - Se envía el dato de memoria al nodo peticionario.
    - Se envían mensajes de invalidación al conjunto de nodos de la entrada.
    - Se activa en el conjunto solamente el nodo peticionario.
    - Se pasa a estado exclusivo.



# Entrada exclusiva

- El valor del bloque se encuentra en caché en el nodo identificado por el conjunto (nodo propietario).
  
- **Peticiones:**
  - **Fallo de lectura:**
    - Se envía mensaje de captación a propietario.
    - Se escribe dato en memoria.
    - Se envía dato a nodo peticionario.
    - Se añade nodo peticionario a conjunto de nodos.

# Entrada exclusiva

## ■ Peticiones:

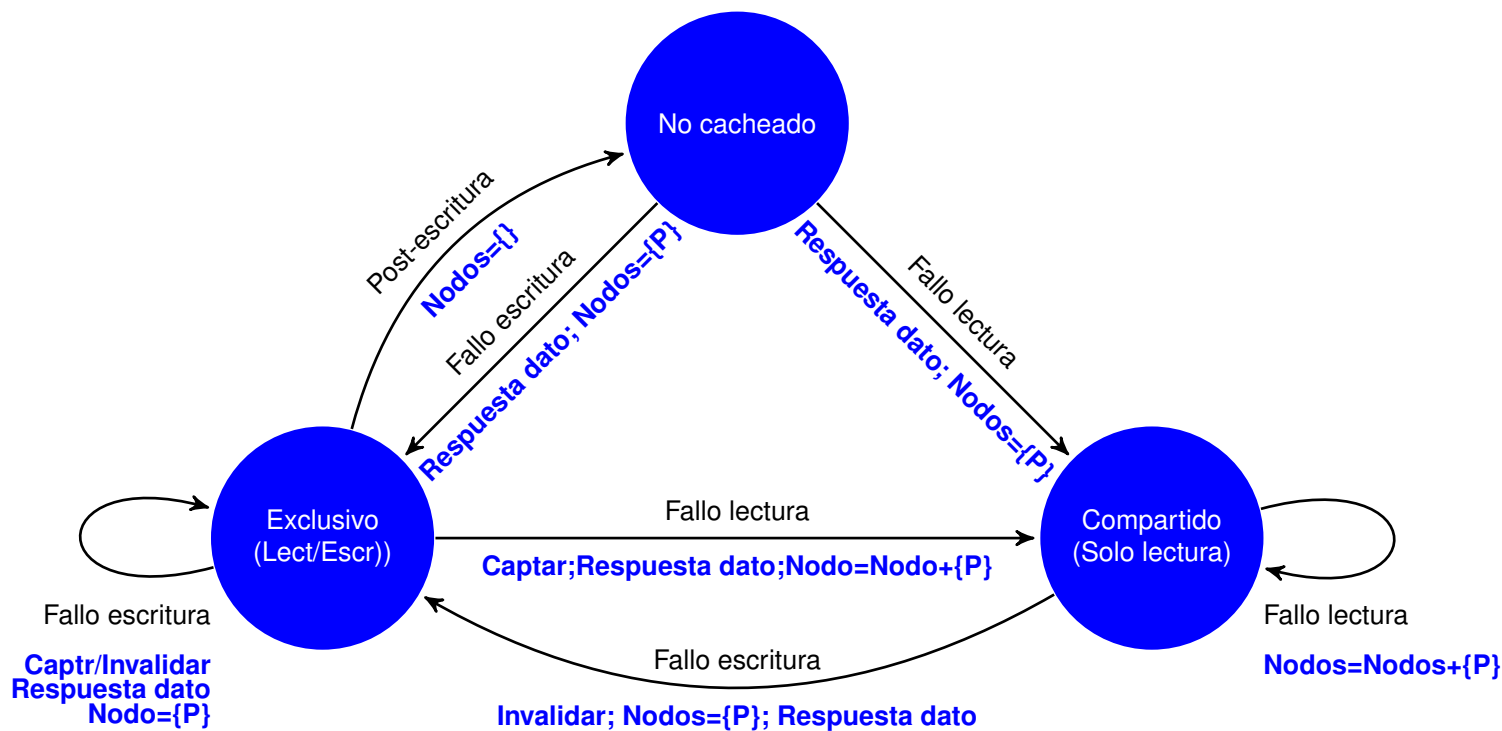
### ■ Post-escritura:

- Ocurre cuando el propietario hace post-escritura del bloque.
- El bloque pasa a estado no cacheado.
- Se vacía el conjunto de la entrada.

### ■ Fallo de escritura:

- El bloque tiene nuevo propietario.
- Se invalida bloque en antiguo propietario y se obtiene valor.
- Se envía valor a nodo peticionario.
- Se activa en el conjunto solamente el nuevo peticionario.

# Transición de estados del directorio



- 1 Introducción a memoria compartida distribuida
- 2 Bases del protocolo de directorio
- 3 Protocolo basado en directorio
- 4 Conclusión

# Resumen

- Problemas de escalabilidad en protocolos de espionaje.
- Alternativas de protocolos basados en directorio:
  - Directorio centralizado en SMP.
  - Directorio distribuido en DSM.
- En chips multicore:
  - Coherencia interna mediante directorio centralizado.
  - Usado como directorio local en DSM.

# Referencias

- **Computer Architecture. A Quantitative Approach**  
5th Ed.  
Hennessy and Patterson.  
**Secciones:** 5.4.
  
- **Ejercicios recomendados:**
  - 5.9, 5.10, 5.11, 5.12.

# Memoria compartida distribuida

## Arquitectura de Computadores

J. Daniel García Sánchez (coordinador)  
David Expósito Singh  
Javier García Blas  
J. Manuel Pérez Lobato

Grupo ARCOS  
Departamento de Informática  
Universidad Carlos III de Madrid