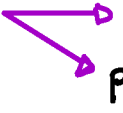


PROTOCOLOS DE ESPIONAJE- TEMA 4 AC - Juan Villegas

Se van a describir 4 protocolos de mantenimiento de coherencia,

- Dos para multiprocesadores UMA con red bus → MSI, MESI
- Dos para multiprocesador NUMA en una placa → MSI con difusión.
MSI sin difusión.
- Todos ellos usan posescritura  para actualización de memoria e invalidación.
para propagación de escritura a otras cachés.

Los protocolos para buses se llaman de espionaje o sondes porque todos los controladores, al estar conectados al bus, pueden espiar los accesos del resto sondeando el bus.

Protocolo MSI de espionaje ← multiprocesadores NUMA

MSI es el protocolo con menor número de estados que utiliza posescritura e invalidación.

Estados de un bloque en caché:

MODIFICADO (M): - Un bloque en este estado en caché es la **única copia de bloque válida** en todo el sistema de memoria.

- El controlador de caché **debe responder con el bloque** si le llega una **petición con lectura de bloque**, y deberá **escribirlo en memoria** si se **reemplaza** por otro bloque.

COMPARTIDO (S): - Un bloque en este estado **es válido**, también **está válido en memoria** y puede que haya **copias válidas en otras cachés**.

- El controlador de caché **invalida esta copia de bloque** si le llega una **petición de invalidación del bloque**.

INVÁLIDO (I): - Está físicamente en la caché **en estado inválido** o bien **no está físicamente**.
- Si el procesador de la caché **accede a una dirección de este bloque**, se genera un **fallo de caché** y el controlador **enviará un paquete de petición del bloque por el bus**.

Estados de un bloque de memoria:

VÁLIDO

- El bloque está actualizado en memoria principal y puede haber una copia válida en una o varias cachés.
- El controlador de memoria responde con el bloque si ve en el bus una petición de lectura de un bloque en estado válido.

INVÁLIDO

→ El bloque no está actualizado en memoria principal, hay una copia válida en una caché.

Transferencias generadas por un nodo con caché. ← tipos de paquetes

Son los paquetes que el controlador de caché puede enviar por el bus ante los eventos que se puedan producir en el procesador del nodo (lectura y escritura), en el controlador de caché o que llegan en forma de paquete a otros nodos. → un controlador de caché genera paquetes de petición (P+) y respuesta (R+)

- PETICIÓN LECTURA DE UN BLOQUE: → $P_{Lec}(B)$

- Lectura con fallo de caché del procesador del nodo.
- El nodo intenta leer un dato en caché, y si no está lo pide a memoria.
- Como respuesta a esta petición recibirá un paquete de respuesta con el bloque $R_{P_{Bloque}}(B)$ de memoria o de la caché que lo tiene en estado modificado (si la hubiera).
- Esto hará que el bloque pase de MODIFICADO a COMPARTIDO.

- PETICIÓN DE ACCESO EXCLUSIVO A UN BLOQUE CON LECTURA DE BLOQUE

- Consecuencia de una escritura del procesador ($P_{rEsc}(B)$) en un bloque compartido o inválido de la caché.
- Como respuesta a esta petición → Se invalidarán las copias del bloque en otras cachés.
Recibirá un paquete de respuesta con el bloque ($R_{P_{Bloque}}(B)$)

- PETICIÓN DE ACCESO EXCLUSIVO A UN BLOQUE: \longrightarrow $PtEx(B)$

- Consecuencia de escritura del procesador ($PrEsc(B)$)
- Si el bloque estuviera válido y en estado modificado no se necesita pedir acceso exclusivo porque ya lo tiene.
- Como respuesta a esta petición \longrightarrow Se invalidarán las copias del bloque en otras cachés.

- PETICIÓN DE POSESCRITURA DE UN BLOQUE \longrightarrow $PtPEsc(B)$

- Reemplazo de un bloque en estado modificado.

- RESPUESTA CON BLOQUE \longrightarrow $RpBloque(B)$

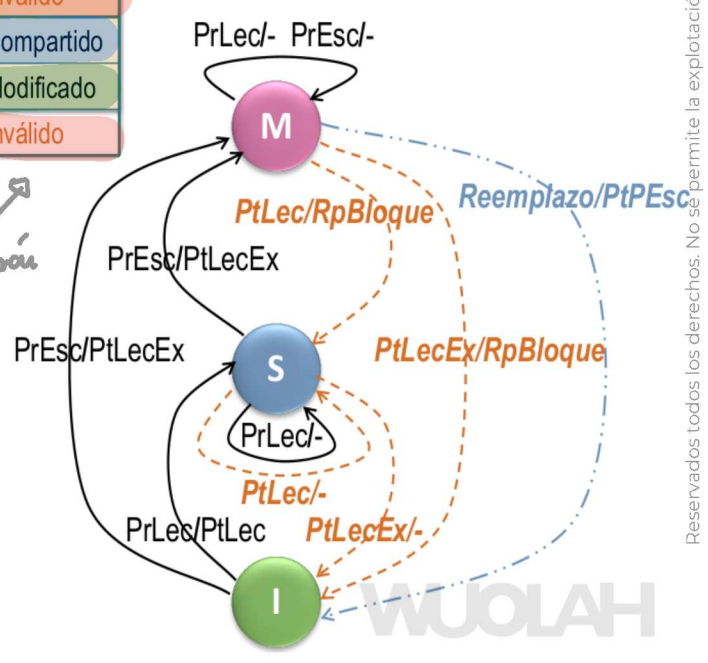
- Un nodo devuelve un bloque.
- Se genera cuando el bloque B en estado MODIFICADO es solicitado por un paquete de petición de lectura B \longrightarrow $PtLec(B)$
 \longrightarrow $PtLecEx(B)$

EST. ACT.	EVEN TO	ACCIÓN	SIGUIENTE
Modificado (M)	PrLec/PrEsc		Modificado
	PtLec	Genera paquete respuesta (RpBloque)	Compartido
	PtLecEx	Genera paquete respuesta (RpBloque) Invalida copia local	Inválido
	Reemplazo	Genera paquete posescritura (PtPEsc)	Inválido
Compart. (S)	PrLec		Compartido
	PrEsc <small>posescr</small>	Genera paquete PtLecEx (PtEx)	Modificado
	PtLec		Compartido
	PtLecEx <small>inválido</small>	Invalida copia local	Inválido
Inválido (I)	PrLec	Genera paquete PtLec	Compartido
	PrEsc	Genera paquete PtLecEx	Modificado
	PtLec/PtLecEx		Inválido

↑
Están en este estado

↑
Lecturas del procesador (PrLec)
Escrituras del procesador (PrEsc)
Peticiones (Pt...)

↑
respuesta
Se ponen en este estado



Protocolo MESI de espionaje

← Multiprocesadores NUMA

Diferencias frente a MSI

Con el protocolo MSI siempre que se escribe en la copia de un bloque en una caché se genera un paquete de petición exclusiva de bloque

Para evitar este tráfico innecesario, el protocolo MESI divide el estado S en dos estados

EXCLUSIVO (E)

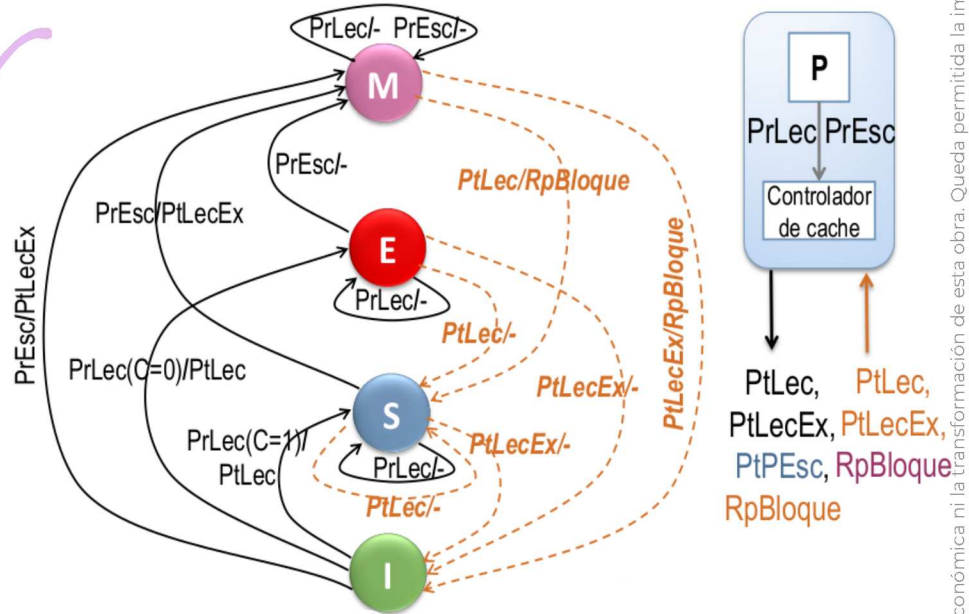
- No hay copias en otras cachés
- Sólo está en dicha caché y en memoria

COMPARTIDO (S)

- lo tienen dos o más cachés y en memoria

Hay cambios en las acciones generadas

Explicadas más abajo.



También tenemos un bit que nos indica si un bit es compartido o no

C=1 → si el bit es compartido — más de un nodo tiene el bloque.

C=0 → si el bit no es compartido — nadie tiene el bloque.

Modificado (M)	PrLec/PrEsc		Modificado
	PtLec	Genera RpBloque	Compartido
	PtLecEx	Genera RpBloque. Invalida copia local	Inválido
	Reemplazo	Genera PtPesc	Inválido
Exclusivo (E)	PrLec		Exclusivo
	PrEsc		Modificado
	PtLec		Compartido
	PtLecEx	Invalida copia local	Inválido
Compartido (S)	PrLec/PtLec		Compartido
	PrEsc	Genera PtLecEx	Modificado
	PtLecEx	Invalida copia local	Inválido
Inválido (I)	PrLec (C=1)	Genera PtLec	Compartido
	PrLec (C=0)	Genera PtLec	Exclusivo
	PrEsc	Genera PtLecEx	Modificado
	PtLec/PtLecEx		Inválido

Protocolos MSI basados en directorios con o sin difusión: ← NUMA

Protocolos MSI con posescritura e invalidación para NUMA

con difusión → Se difunde peticiones a todos los nodos.

sin difusión → Se envía petición solo al nodo que tiene el bloque en memoria principal más cercano al nodo.

Hay 3 tipos de nodos:

SOLICITANTE (S) → Es el nodo que genera una petición de bloque

- P+Lec
- P+Ex
- P+LecEx
- P+Esc

ORIGEN (O) → En un NUMA, la memoria está repartida físicamente en los nodos, así que cada procesador tiene "un trozo de memoria" más cercano. El nodo origen es aquel que tiene el bloque en su "trozo de memoria".

* Trozo del espacio de direcciones total del multiprocesador

PROPIETARIO (P) → Es un nodo que tiene copia del bloque en caché.

MSI con directorios SIN DIFUSIÓN

- Como no se usa difusión se necesita almacenar, en el directorio de memoria principal:
 - Estado del bloque en memoria.
 - Información sobre las cachés con copia del bloque.
- Se usa un vector de bits de presencia para almacenar esa información. Así, habrá un bit para cada caché conectada a la red NUMA.
 - Bit = 1 → Hay copia válida del bloque en esa caché.
- Todas las peticiones de bloque se envían al nodo origen del bloque.
- El nodo origen propagará las invalidaciones a los nodos que tengan copia de ese bloque.

Estados de un bloque en caché:

MODIFICADO INVÁLIDO
COMPARTIDO

Estados de un bloque en memoria:

VÁLIDO → Copias válidas en una o más cachés.

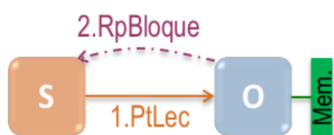
INVÁLIDO → Hay una copia válida en una caché en estado MODIFICADO.

El nodo solicitante solicita al origen ya que puede haber muchos nodos propietarios de un dato.

Cada nodo origen tiene su directorio de memoria en el que puede consultar quién tiene el dato y quién no.

Las entradas del directorio son el número de nodos más la entrada de memoria.

Estado inicial	Evento	Estado final
D) Válido S) Inválido P) Compartido Acceso remoto	Fallo de lectura	D) Válido S) Compartido P) Compartido



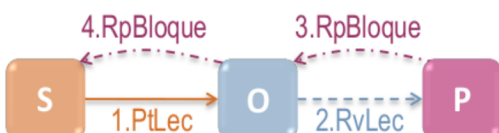
1. PtLec → lectura de un bloque
2. RpBloque → Respuesta con bloque.

Estado inicial	Evento	Estado final
D) Inválido S) Inválido P) Modificado Acceso remoto	Fallo de lectura	D) Válido S) Compartido P) Compartido

1. PtLec → lectura de un bloque
(El bloque no está, reenviamos petición.)

2. RvLec → Reenvío de lectura

3. RpBloque → Respuesta con bloque.



Estado inicial	Evento	Estado final
D) Inválido S) Inválido P) Modificado Acceso remoto	Fallo de escritura	D) Inválido S) Modificado P) Inválido

PIDES UN BLOQUE PARA MODIFICARLO

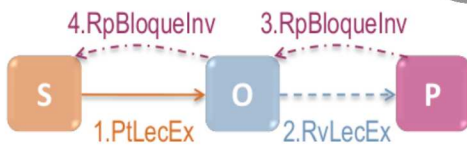
1. PtLecEx → lectura con acceso exclusivo.

⇒ (El bloque no está, reenviamos petición)

2. RvLecEx → Reenvío de lectura exclusiva.

3. RpBloqueInv → Respuesta con bloque
4. RvBloqueInv → Reenvío con bloque confirmando invalidación.

PORQUE QUIERES MODIFICARLO



Estado inicial	Evento	Estado final
D) Válido S) Compartido P) Compartido Acceso remoto	Fallo de escritura	D) Inválido S) Modificado P) Inválido

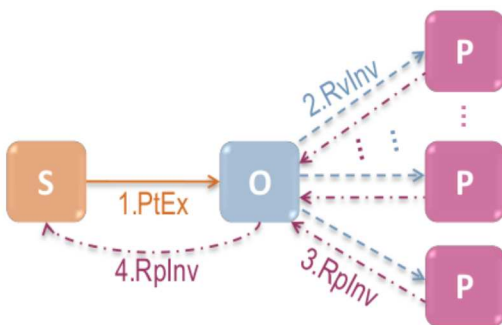
YA TIENES EL BLOQUE, AHORA VAS A MODIFICARLO

1. PtEx → petición de acceso exclusivo sin lectura.

⇒ (Ya tenemos el dato, reenviamos que va a estar inválido)

2. RvInv → Reenvío con invalidación.

3. RpInv → Respuesta confirmando invalidación.
4.



Estado inicial	Evento	Estado final
D) Válido S) Inválido P) Compartido Acceso remoto	Fallo de escritura	D) Inválido S) Modificado P) Inválido

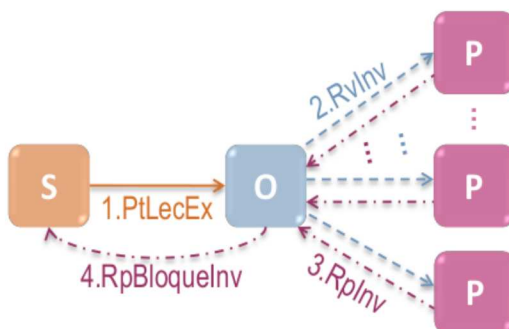
PIDES UN BLOQUE PARA MODIFICARLO

1. PtLecEx → lectura con acceso exclusivo.

2. RvInv → Reenvío con invalidación.

3. RpInv → Respuesta confirmando invalidación

4. RpBloqueInv → Respuesta con bloque confirmando invalidación.



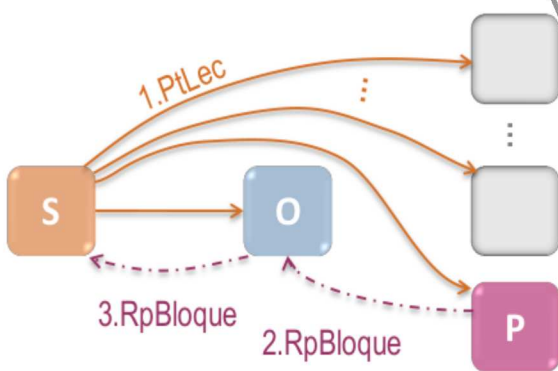
MSI con directorios CON DIFUSIÓN

- Al usar difusión:
 - Se envían los paquetes de petición tengan o no una copia del bloque.
 - No se requiere almacenar en el directorio de memoria principal información sobre las cachés con copia del bloque.

(Los estados de un nodo en caché y memoria coinciden con los MSI SIN DIFUSIÓN)

El solicitante cuando pide todos los nodos "se enteran" de su petición

Estado inicial	Evento	Estado final
D) Inválido S) Inválido P) Modificado Acceso remoto	Fallo de lectura	D) Válido S) Compartido P) Compartido

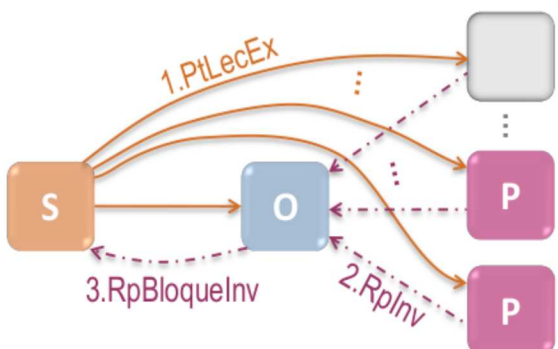


1. PtLec → Lectura de un bloque

(No está en nodo origen así que llegará antes aquí para que este se lo pase al nodo solicitante)

2. RpBloque → Respuesta con bloque.
3.

Estado inicial	Evento	Estado final
D) Válido S) Inválido P) Compartido Acceso remoto	Fallo de escritura	D) Inválido S) Modificado P) Inválido



1. PtLecEx → Lectura con acceso exclusivo.

(Como está en origen, el nodo origen solo tiene que esperar a recibir la confirmación de la invalidación del resto de nodos)

2. RpInv → Respuesta confirmando invalidación.
3. RpBloqueInv → Respuesta con bloque, confirmando invalidación.

PIDES UN BLOQUE PARA MODIFICARLO