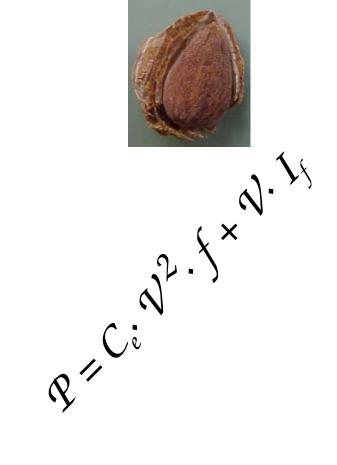




Multiprocesadores



J.M. Llabería

© Copyright 2014, 2015 los autores, Universidad Politécnica de Cataluña

Contenido

Capítulo 8	Protocolo de directorio con una red escalable						
	El CM es el receptor de las respuestas	524 525					
	Peticiones y respuestas del CM se encaminan por la misma red Organización del multiprocesador	526 526 527 528 536 538 539					
	Peticiones y respuestas del CM se encaminan por redes distintas Organización del multiprocesador	540 541 542 543					
	El solicitante es el receptor de las respuestas Descripción funcional de la secuencia de mensajes en una transacción Organización del multiprocesador Mensajes del protocolo Recepción de respuestas en el solicitante Estados y transiciones Inferencia en un CC del orden entre peticiones y respuestas emitidas desde CM Controlador de memoria Controlador de coherencia. Gestión de cruces de peticiones	554 556 556 558 559 e un 568 571 575					
	Red generica	576 577					

	Organización del multiprocesador	578 579 579 583 589
Apeı	ndice A: Tablas de transiciones entre estados	591 591 591 592 594 595
	Recepción de respuestas de los CC en el CM. Peticiones y respuestas del 0 encaminan por la misma red	596
Eiero	cicios	609

Capítulo 8 Protocolo de directorio con una red escalable

• • • • •

Una red crossbar es costosa en recursos de cableado y la escalabilidad está limitada cuando se tiene en cuenta el número de puertos¹. Por otro lado, la longitud de los cables se incrementa al incrementar el número de puertos y con ello el retardo de propagación de la señal, en los mismos, se incrementa². Otra característica es la baja utilización de los recursos dedicados en la implementación de la red.

El objetivo es utilizar otro tipo de redes de interconexión que tengan un menor coste y sean más escalables. La idea es que la topología sea regular y se incremente la utilización de los recursos dedicados. En particular, interesa que la longitud de los enlaces que conectan dos nodos no dependa del número de nodos. La desventaja de este tipo de redes es que no facilitan el establecimiento u observación de un orden lógico global de los accesos a memoria. Por ello, es necesario añadir mecanismos, en los protocolos descritos en el Capítulo 7, para determinar cuando todas las caches, involucradas en una acción de coherencia, han observado una escritura. Esto es, una petición debe serializarse individualmente con respecto a todas las caches que tienen copia. Por ello, son necesarios mensajes explícitos para notificar al agente gestor del bloque que la petición de invalidación ha sido serializada en cada cache que tiene copia.

En el protocolo descrito en el Capítulo 7 no es necesario responder a las peticiones de invalidación, ya que la red crossbar entre el CM y los CC, mediante el arbitraje, mantiene un orden lógico global de los mensajes. En este capítulo las redes de interconexión lógicas mantienen, en algunos casos, el

- 1. El coste se incrementa cuadraticamente.
- 2. Una posibilidad es segmentar la propagación para incrementar la concurrencia. Ahora bien, se incrementa la latencia de propagación.

orden de los mensajes entre un emisor y un receptor. En otros casos no existen garantías de recibir en un receptor los mensajes en el mismo orden con el cual son emitios desde un emisor.

En estas condiciones y de forma generica, como no existe un orden global de las peticiones de los CM, es necesario que una petición de invalidación de un CM sea respondida de forma explícita por el CC que la recibe.

Las respuestas de invalidación deben ser recolectadas en un punto o agente. Una vez han sido recolectadas todas las respuestas, se tiene constancia de que la escritura ha sido observada por los CC implicados en la acción de coherencia. Esto es, la escritura está consolidada.

En este capítulo se presentan en primer lugar dos diseños donde las respuestas de invalidación para un bloque son recolectadas en el CM que ha emitido las peticiones de invalidación. En uno de los diseños los mensajes de petición y respuesta de los CM utilizan la misma red de comunicación con los CC y en esta red existe un orden entre un emisor y un receptor. En el otro diseño los mensajes de petición y respuesta de los CM a los CC utilizan redes distintas. Cada una de estas redes mantiene orden entre un emisor y un receptor, pero entre ellas un CC debe inferir el orden de los mensajes que recibe.

En segundo lugar, con el objetivo de reducir la latencia de algunas transacciones, que requieren la colaboración de terceros, se presenta un diseño donde el recolector de las respuestas de invalidación es el CC cuya petición ha inducido que el CM emitiera las peticiones de invalidación.

En todos los protocolos que se ha comentado que se describen, la red que se utiliza para transmitir mensajes de petición de los CM a los CC mantiene el orden entre un emisor y un receptor. El siguiente diseño, que se describe en este capítulo, no mantiene esta propiedad. Los mensajes emitidos desde un CM pueden llegar en un orden distinto al de emisión en el CC destinatario.

En la exposisión que se efectúa en este capítulo se utiliza el protocolo de coherencia MLI.

EL CM ES EL RECEPTOR DE LAS RESPUESTAS

Teniendo en cuenta el encaminamiento de los mensajes desde los CM a los CC podemos distinguir dos diseños: a) los mensajes de petición y respuesta de los CM se encaminan por la misma red y b) los mensajes de petición y respuesta se encaminan por redes distintas.

En los dos casos una red mantiene el orden de los mensajes entre un emisor y un receptor. Entonces, en el caso a) un CC observa el orden de emisión de peticiones y respuestas desde un CM a este CC. En cambio, en el caso b) un CC no observa el orden entre peticiones y respuestas emitidas desde un CM a este CC. Sólo observa orden entre las peticiones y orden entre las respuestas. Por ello, en el protocolo hay que disponer de un mecanismo que permita a un CC inferir el orden entre peticiones y respuestas emitidos por un CM a este CC.

Descripción funcional de la secuencia de mensajes en una transacción

En la Figura 8.1 se muestra la descripción funcional de protocolo del Capítulo 7 y la del protocolo que se describe en el siguiente apartado.

Cuando no hay copia del bloque en otras cache o la transacción es de lectura y una cache no tiene el bloque en exclusividad, la secuencia de mensajes y su serialización es la misma en los dos protocolos (2 pasos, parte superior de la Figura 8.1).

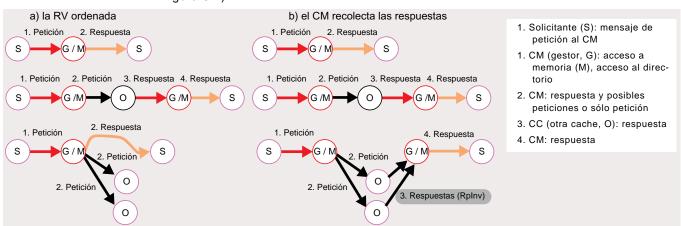


Figura 8.1 Flujos de mensajes en dos protocolo de directorio MLI: a) la red entre los CM y los CC mantiene un orden lógico global de los mensajes (Capítulo 7) y b) el CM recolecta las respuestas de los CC a peticiones de invalidación. El sombreado muestra la diferencia básica con la especificación del Capítulo 7.

Cuando una cache tiene el bloque en exclusividad la secuencia de mensajes es la misma en los dos protocolos (4 pasos). El solicitante recibe el bloque del CM, después de que el CC, que tiene el bloque en exclusividad, ha suministrado el bloque al CM (centro de la Figura 8.1).

Cuando hay caches con copias del bloque y una transacción solicita obtener la exclusividad, las respuestas de invalidación las recolecta el CM (parte inferior de la Figura 8.1). Una vez las respuestas de los CC han sido recolectadas, el CM responde al CC solicitante.

Un CM da por completada o consolidada una transacción una vez emite el mensaje de respuesta al CC solicitante.

PETICIONES Y RESPUESTAS DEL CM SE ENCAMINAN POR LA MISMA RED

El protocolo que se describe es básicamente el mismo que en el Capítulo 7. Ahora bien, el arbitraje en la red que comunica los CM con los CC, y por la que se transmiten peticiones y respuestas del CM, no garantiza que los CC observen el mismo orden lógico global de los mensajes. Sólo se garantiza orden entre un emisor y un destino.

En primer lugar se describe la organización del multiprocesador. Seguidamente se describe el protocolo de coherencia.

Organización del multiprocesador

El protocolo utiliza tres redes lógicas (Figura 8.2). Los CC utilizan la RI para emitir peticiones y la RCM para emitir respuestas al CM. Los CM utilizan la RPR para emitir peticiones y respuestas a los CC.

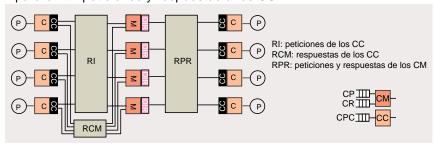


Figura 8.2 Misma red para peticiones y respuestas del CM. Organización del multiprocesador.

La RPR mantiene un orden punto a punto de los mensajes emitidos desde un emisor a un receptor³.

3. En esta organización la RPR puede corresponderse con una red crossbar, donde el arbitraje es independiente en cada puerto de entrada. También puede corresponderse con una red en malla donde el encaminamiento de los mensajes entre emisor y destino está prefijado. El orden de los mensajes generados por un CM para ser encaminados a un CC se mantiene en los dos casos.

La cola de mensajes de respuestas en un CM se denomina CR. La cola de mensajes de petición y respuesta en un CC se denomina CPC.

Suponemos que un CM genera los mensajes de petición y respuesta en paralelo. Entonces, si los árbitros de cada destino seleccionan los mensajes en el mismo ciclo, los mensajes se propagan en paralelo⁴.

Para representar retardo en una red se replica el acrónimo de la red en ciclos consecutivos.

En este capítulo el arbitraje para acceder a un nodo no se puede considerar centralizado. En estas condiciones, en un diagrama temporal no se replicará el acrónimo arb para representar una llegada en serie de los mensajes a un nodo. En su lugar, después del acrónimo arb se utiliza el acrónimo de la cola correspondiente para indicar que el mensaje no se procesa. Ello puede ser debido a dos causas. Una de ellas sería un riesgo estructural. La otra causa sería debida a que el protocolo de coherencia no procesa el mensaje, que está en la cabeza de la cola correspondiente, debido al estado actual del bloque al que hace referencia el mensaje.

En la Figura 8.3 se muestra un ejemplo. En la parte izquierda se muestra la representación utilizada hasta este capítulo y en la parte derecha la representación que se utilizará a partir de ahora.



Figura 8.3 Representación en un diagrama temporal de la espera de un mensaje. Ejemplo en la cola de peticiones (CP) del CM.

Cuando varios mensajes llegan en el mismo instante a un destinatario se encolan en paralelo en la cola correspondiente. Un destinatario sólo procesa un mensaje en un instante dado.

Mensajes del protocolo

En la Figura 8.4 se muestra un esquema genérico de la transmisión de mensajes con peticiones y respuestas entre los CC y el CM y viceversa.

En la descripción del protocolo se distinguen los siguientes tipos de mensajes:

4. En los ejemplos y ejercicios usualmente, para simplificar, supondremos que un destinatario puede recibir varios mensajes en un ciclo. Por ejemplo, esta circunstancia es factible en una red en malla. Ahora bien, el procesado en el destinatario es en secuencia.

- Peticiones de un CC a un CM (Pt, PtIm, PtXm, PtXI)
- Peticiones de un CM a los CC (PtObL, PtObE)
- Respuestas de un CM a un CC (RpD, RpX)
- Respuestas de un CC a un CM (RpDc, RpInv)

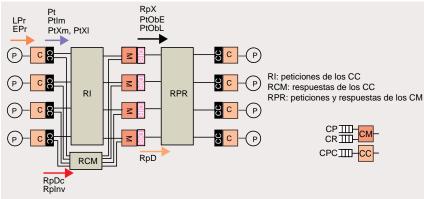


Figura 8.4 Tipos de mensajes en un protocolo MLI.

Los mensajes que se han añadido o modificado respecto al protocolo del Capítulo 7 se muestran en la Tabla 8.1.

Controlador de coherencia Respuestas	Comentario				
RpInV: respuesta a una petición de invalidación	El CC que recibe una petición PtObE a un bloque en el estado L, invalida el bloque y responde con RpInv.				
RpDc: respuesta con el bloque	El CC que recibe la petición PtObE tiene el bloque en exclusividad. Emite una respuesta con el bloque y la acción de invalidación está implícita en el tipo de respuesta. Esta respuesta también se utiliza si la petición del CM es PtObL. En este caso no se invalida el bloque al responder.				

Tabla 8.1 Respuesta de un CC a un CM.

Estados y transiciones

Cuando un CM procesa una petición de exclusividad y hay copias del bloque en varias caches, el CM debe esperar las respuestas de invalidación de los CC correspondientes, antes de responder al CC solicitante. Para identificar la espera de las respuestas de invalidación de los CC se utiliza un estado transitorio en el directorio.

Directorio. Los estados estables de un bloque en el directorio son los mismos que en el Capítulo 7 (Figura 8.5).

El número de estados transitorios en el directorio es tres (ML, MM, LM). Se utilizan para identificar la espera de la respuesta con el bloque o respuestas de invalidación de los CC. Respecto al protocolo del Capítulo 7 se ha añadido el estado LM, cuyo objetivo es esperar las respuestas de invalidación.

Cache. En una cache se utilizan los mismos estados estables que en el protocolo descrito en el Capítulo 7 (Figura 8.5). En cuanto a los estados transitorios, se añade otro estado transitorio para gestionar de forma canónica los cruces de peticiones al expulsar un bloque en el estado M.

Para describir el protocolo se utilizan las dos peticiones del procesador que requieren acceder al directorio (fallo en una instrucción load, store o una instrucción store que accede a un bloque sin permiso de exclusividad) y las dos posibilidades de ubicación del bloque solicitado, en memoria o en una cache. Posteriormente se detallan las transiciones entre estados en una expulsión de un bloque de una cache.

En la descripción se muestran las transiciones entre estados de un bloque en el CC que efectúa la petición (agente procesador), en el CM y en otros CC (agente observador).

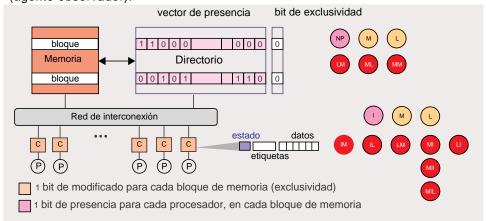


Figura 8.5 Protocolo de directorio MLI y recepción de respuestas de invalidación de los CC en el CM. Estados de un bloque en el directorio y en un contenedor de cache.

Fallo en lectura

Memoria tiene actualizado el bloque. En la parte izquierda de la Figura 8.6 se muestra el flujo de mensajes en un fallo de lectura. El CC emite un mensaje con una petición de lectura de bloque (Pt). El CM accede al direc-

torio para leer el VP y el BE y determina que el estado del bloque es NP o L. El CM emite un mensaje de respuesta (RpD), que incluye el bloque, al CC que ha efectuado la petición.

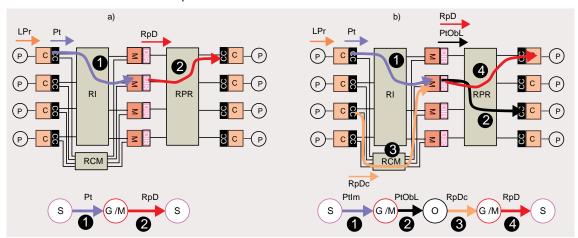


Figura 8.6 Recepción de respuestas de los CC en el CM. Flujo de mensajes en un fallo de lectura: a) memoria está actualizada y b) una cache tiene el bloque en exclusividad.

Una cache tiene el bloque en exclusividad. En la parte derecha de la Figura 8.6 se muestra el flujo de mensajes entre el CC que efectúa la petición y el CM y entre el CM y el CC que tiene el bloque en exclusividad.

El CM emite una petición de observación de lectura (PtObL) al CC que tiene el bloque en exclusividad, para que suministre el bloque. Este CC responde al CM con un mensaje que incluye el bloque (RpDc). El CM actualiza el directorio y la memoria al recibir la respuesta del CC que tiene el bloque en exclusividad. Finalmente el CM responde con el bloque al CC solicitante.

Diagrama de transiciones entre estados. En la Figura 8.7 se muestran las transiciones entre estados en un fallo de lectura en: a) la cache del CC que efectúa la petición (solicitante), b) el directorio (CM) y c) otros CC. El CC envía el mensaje de petición de lectura y establece IL como estado transitorio del bloque. El bloque permanece en este estado hasta que recibe la respuesta del CM (RpD). Entonces, el CC establece como estado estable del bloque el estado L.

El CM al recibir el mensaje de petición de lectura accede al directorio y especulativamente a memoria. Si el estado del bloque en el directorio es NP o L, el CM determina que puede responder a la petición utilizando el bloque leído de memoria. Entonces, el CM, además de enviar la respuesta, actualiza el VP

añadiendo el identificador del CC que ha efectuado la petición. Los otros CC no reciben peticiones de coherencia desde el CM. En la parte izquierda de la Figura 8.8 se muestra un diagrama temporal.

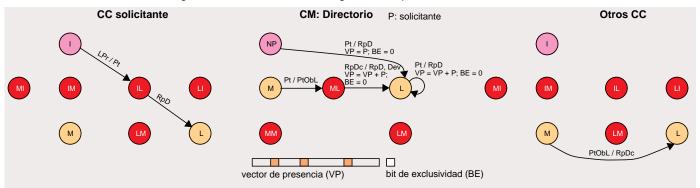


Figura 8.7 Recepción de respuestas de los CC en el CM. Protocolo de directorio MLI. Transiciones entre estados en un fallo de lectura.

Si el estado del bloque en el directorio es M, el CM establece el estado transitorio ML. El CM emite una petición al CC que tiene el bloque en exclusividad (PtObL). Este CC, al recibir la petición, responde al CM con un mensaje, que incluye el bloque (RpDc) y establece como nuevo estado del bloque el estado L. El CM al recibir la respuesta (RpDc) actualiza la memoria con el bloque y modifica el VP y el BE, estableciendo en el directorio como estado estable del bloque el estado L. En la parte derecha de la Figura 8.8 se muestra un diagrama temporal.

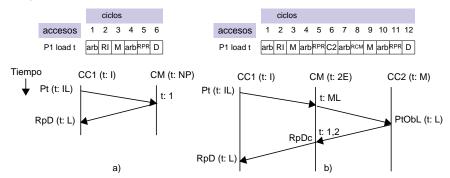


Figura 8.8 Recepción de respuestas de los CC en el CM. Diagrama temporal: a) estado NP o L en el directorio y b) estado M en el directorio.

Fallo de escritura o petición de exclusividad

Memoria tiene el bloque actualizado. En la parte izquierda de la Figura 8.9 se muestra el flujo de mensajes en un fallo de escritura cuando no hay copias del bloque en otras caches. El CC del solicitante emite un mensaje con una petición de bloque con intención de modificación (PtIm). El CM, al recibir el mensaje, lee el estado del bloque en el directorio y accede especulativamente a memoria. El CM determina que el estado del bloque es NP y responde con el bloque al solicitante. Además, actualiza el VP y el BE.

Hay copias del bloque en caches. En la parte derecha de la Figura 8.9 se muestra el flujo de mensajes en un fallo de escritura cuando otras caches tiene copia del bloque. El CM, utilizando la información de estado del bloque en el directorio, determina los CC a los que es necesario enviar una petición de invalidación (PtObE). Una vez el CM ha recolectado las respuestas de los CC (RpInV), el CM responde al CC solicitante con el bloque (RpD). Además, modifica el VP y el BE.

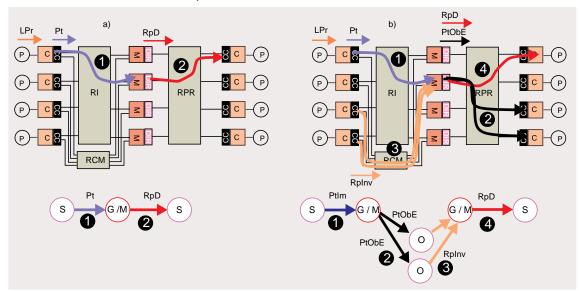


Figura 8.9 Recepción de respuestas de los CC en el CM. Flujo de mensajes en un fallo de escritura: a) no hay copias en caches y b) hay copias en caches.

Una cache tiene el bloque en exclusividad. En la Figura 8.10 se muestra el flujo de mensajes en un fallo de escritura cuando otra cache tiene el bloque en exclusividad. El CM, emite un mensaje PtObE al CC que tiene el bloque en exclusividad. Como el estado del bloque es M, el CC infiere que además de

invalidar el bloque debe suministrarlo (RpDc). El CM, al recibir la respuesta, actualiza el VP para identificar al CC que ahora tiene el bloque en exclusividad y responde al CC solicitante.

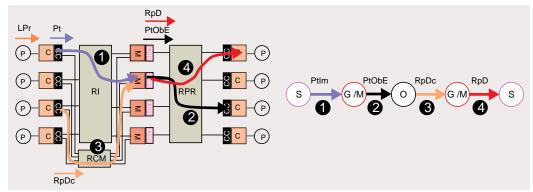


Figura 8.10 Recepción de respuestas de los CC en el CM. Flujo de mensajes en un fallo de escritura y una cache tiene el bloque en exclusividad.

Diagrama de transiciones entre estados. En la Figura 8.11 se muestran las transiciones entre estados en la cache del CC que efectúa la petición, el directorio y otros CC. El CC del solicitante establece un estado transitorio (IM o LM), en función del estado estable inicial, esperando la respuesta.

Cuando no hay copias del bloque o sólo el CC solicitante tiene copia del bloque, el estado del bloque en el directorio pasa de NP a M o de L a M. El VP se actualiza en consecuencia y el BE se activa. Además, el CM emite un mensaje de respuesta al solicitante (RpD). En la parte izquierda de la Figura 8.12 se muestra un diagrama temporal.

Cuando un CC emite una petición PtIm, estando el bloque en estado L, el bit de presencia del CC está activado en el VP. Por tanto, hay que excluirlo de la lista de CC que reciben una petición de observación de escritura PtObE (VP - P). El CM inicializa el contador de respuestas (|VP - P|) y emite las peticiones, para invalidar el bloque, a los CC involucrados en la acción de coherencia. Estos CC al recibir la petición PtObE invalidan la copia del bloque y responden al CM (RpInv). El CM recolecta todas las respuestas de invalidación antes de responder al CC solicitante (RpD). Este CC, al recibir la respuesta siempre almacena el bloque recibido, en el contenedor correspondiente, tanto si el estado estable inicial es L como I. En el centro de la Figura 8.12 se muestra un diagrama temporal. Suponemos que el CM genera todos los mensajes en paralelo. Estos mensajes, cuando no hay conflictos en la RPR para acceder a

los destinatarios, se transmiten en paralelo. Cuando varios mensajes llegan en el mismo instante a un destinatario se encolan. Un destinatario sólo procesa un mensaje en un instante determinado.

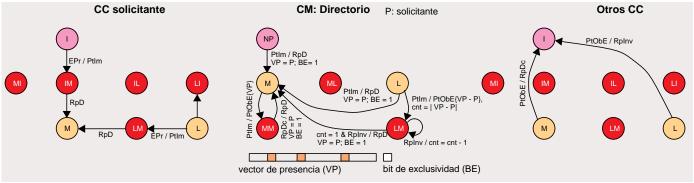


Figura 8.11 Recepción de respuestas de los CC en el CM. Transiciones entre estados en un fallo de escritura.

Cuando el estado del bloque en el directorio es M, el CM emite una petición PtObE al CC que tiene el bloque en exclusividad. Este CC responde con el bloque y establece como estado del bloque en cache el estado I⁵. El CM al recibir la respuesta del CC, que tenía el bloque en exclusividad, responde al CC solicitante. Además, actualiza el VP con el identificador del CC solicitante. En la parte derecha de la Figura 8.12 se muestra un diagrama temporal.

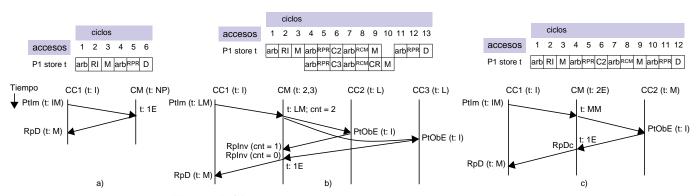


Figura 8.12 Recepción de respuestas de los CC en el CM. Diagrama temporal simplificado: a) estado NP en el directorio o L y VP = P, b) estado L en el directorio y c) estado M en el directorio. El acrónimo CR indica cola de respuestas en un CM.

5. Un CC al recibir una petición PtObE del CM determina, en función del estado del bloque en la cache, si es necesario una acción de suministro del bloque.

Expulsión

El directorio es preciso y una petición de expulsión de un CC espera una respuesta del CM. En la Figura 8.13 se muestra el flujo de mensajes cuando se expulsa un bloque de cache.

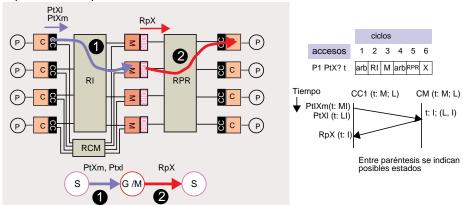


Figura 8.13 Recepción de respuestas de los CC en el CM. Expulsión de un bloque de cache.

El CC efectúa una petición de expulsión (PtXm, PtXI), que va acompañada del bloque si el bloque está en el estado M (PtXm). El CM en cualquier caso actualizada la entrada correspondiente en el directorio. Además, el CM actualiza la memoria cuando la petición es PtXm. El CM responde al CC confirmando que se ha realizado la acción solicitada.

En la Figura 8.50 se muestran las transiciones entre estados en: a) la cache del CC que expulsa el bloque, b) el directorio y c) otros CC. En otros CC no se producen transiciones entre estados.

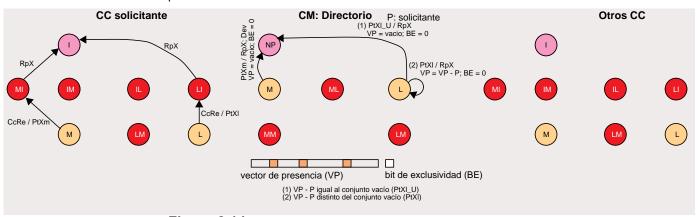


Figura 8.14 Recepción de respuestas de los CC en el CM. Transiciones entre estados cuando se expulsa un bloque.

Para distinguir entre la expulsión de un bloque en el estado M o en el estado L se utilizan distintos tipos de peticiones y también estados transitorios distintos; MI y LI respectivamente. Cuando el CM procesa la petición actualiza memoria si la petición es PtXm y en cualquiera de los dos casos se actualiza la entrada correspondiente del bloque en el directorio (VP). Finalmente el CM envia una respuesta de confirmación. El CC al recibir la respuesta cambia el estado transitorio del bloque al estado I.

En el diagrama del CM (Figura 8.14) hay que distinguir dos casos al procesar la petición PtXI de un CC (PtXI, PtXI_U). El estado final depende de si el CC, que efectúa la petición, es el único que está en el vector de presencia.

Diagrama completo de estados y transiciones

En las Figura 8.15 se muestran los diagramas de transiciones entre estados de un bloque en una cache y en un directorio. En un CC se distinguen, en diagramas separados, las transiciones iniciadas por el agente procesador (CC solicitante) y el agente observador (otros CC). En el diagrama etiquetado como CC solicitante están incluidas las peticiones del procesador que no requieren iniciar transacciones explícitas de coherencia.

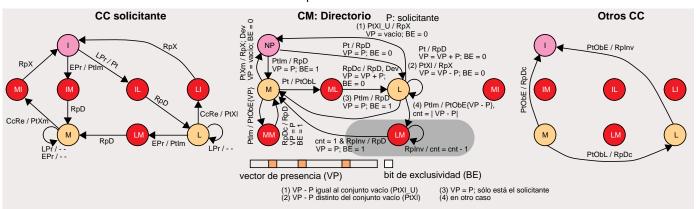


Figura 8.15 Recepción de respuestas de los CC en el CM. Estados y transiciones entre estados de un bloque en una cache y en el directorio. El sombreado muestra la diferencia básica con la especificación del Capítulo 7.

Ejemplo

Los dos trozos de código que se utilizan son el esqueleto del algoritmo de Dekker para exclusión mútua (parte izquierda de la Figura 8.16). Las variables aviso1 y aviso2 están contenidas en bloques que se ubican en los módulos de memoria M1 y M2 respectivamente. Antes de iniciarse la ejecución, la cache

C1 tiene copia del bloque que contiene la variable aviso2 y la cache C2 tiene copia del bloque que contiene la variable aviso1. El valor de las dos variables es cero.

En la Figura 8.16 se muestra un diagrama temporal del ejemplo utilizando el protocolo de este apartado. Suponemos que la petición del CM2 al CC1 experimenta un retardo significativo en la red (varios ciclos RPR).

Las instrucciones store de los procesadores P1 y P2 son fallos en cache. Las peticiones que emiten CC1 y CC2 acceden a CM distintos. Por tanto progresan de forma paralela hasta el CM correspondiente. El CM1 emite una petición PtObE al CC2 y espera la respuesta. En paralelo el CM2 emite una petición PtObE al CC1 y espera la respuesta.

Cuando el CM1 recibe la respuesta de CC2 (RpInv) emite una respuesta al CC1. Seguidamente el procesador P1 ejecuta la instrucción load y accede a la región crítica.

El procesado de la petición de CM2 al CC1 se retrasa por conflictos en la red. Cuando el CM2 recibe la respuesta del CC1 (RpInv) responde al CC2. El procesador P2 ejecuta la instrucción load y detecta un fallo.



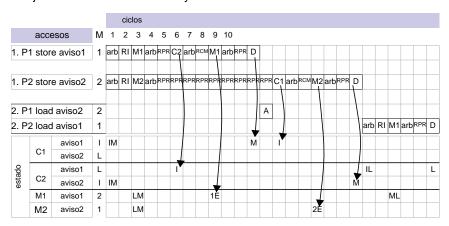


Figura 8.16 Protocolo de directorio MLI. Recepción de respuestas de los CC en el CM. Esqueleto del algoritmo de Dekker.

En la Figura 8.17 se muestra, utilizando el ejemplo previo, una comparación del flujo de mensajes del protocolos MLI del Capítulo 7 y el protocolo de este apartado. En el Capítulo 7 las peticiones de invalidación de un CM no son respondidas por los CC que las reciben. El algoritmo de arbitraje de la RV garantiza un orden lógico global de los mensajes de los CM (Figura 8.17). En el protocolo descrito en este apartado un CC responde a una petición de invalida-

ción del CM. El algoritmo de arbitraje de la red RPR no garantiza un orden lógico global de los mensajes emitidos desde los CM. Sólo garantiza un orden entre un emisor y un receptor.

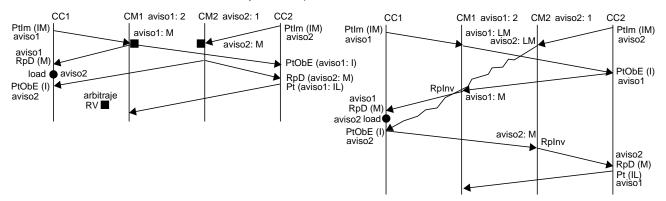


Figura 8.17 Protocolo de directorio MLI. Comparación entre el diseño del Capítulo 7 y el diseño de este apartado (recepción de respuestas de los CC en el CM). Esqueleto del algoritmo de Dekker.

Controlador de memoria. Gestión de cruces de peticiones

Ventana de vulnerabilidad. Mientras el CM está esperando una respuesta puede analizar una petición al mismo bloque (estados transitorios MM, ML y LM).

El procesado en el CM de peticiones a bloques en un estado transitorio es el mismo que en el Capítulo 7. El CM bloquea el análisis de la CP hasta que el CM recibe la respuesta o respuestas necesarias para efectuar la transición a un estado estable (Figura 8.18).

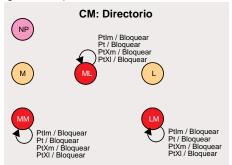


Figura 8.18 Recepción de respuestas de los CC en el CM. Bloqueo del procesado de peticiones en los estados transitorios del directorio.

El CM no procesa peticiones de los CC a un bloque en un estado transitorio. Entonces, el resto de cruces entre peticiones del CM y un CC se observan (infieren) en estados estables. Suponemos que la RI es de tipo crossbar y se utiliza un arbitraje por antigüedad⁶. En estas condiciones, los cruces de peticiones son los mismos que los mostrados en el Capítulo 7. En particular, en el estado estable NP no se infieren cruces de peticiones. En la Figura 8.19 se muestran las transiciones entre estados en las cuales se infieren cruces de peticiones, teniendo en cuenta los estados del protocolo de este apartado.

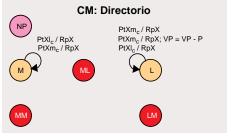


Figura 8.19 Recepción de respuestas de los CC en el CM. Inferencia de cruces en el directorio.

Controlador de coherencia. Gestión de cruces de peticiones

Funcionalmente el conjunto de cruces son los mostrados en el Capítulo 7. La diferencia es que un CC debe responder a todas las peticiones del CM. En la Figura 8.20 se muestran estos cruces de peticiones. Recordemos que la RPR se utiliza para transmitir peticiones y respuestas de los CM a los CC y se mantiene el orden de los mensajes entre un emisor y un receptor.

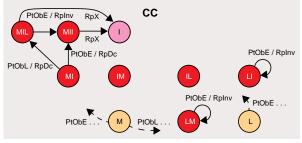


Figura 8.20 Recepción de peticiones de un CM en un CC . Inferencia de cruces en un CC. También se muestran peticiones del CM en estados estables.

6. En el próximo protocolo del capítulo se analiza el caso de utilizar una red de tipo malla.

En el estado LI un CC responde (RpInv) a una petición de observación de escritura (PtObE) y se espera la respuesta de expulsión del bloque (RpX). En el estado LM un CC responde (RpInv) y espera la respuesta del CM con el bloque (RpD).

En el estado MI la actuación depende del tipo de petición que se recibe del CM. En cualquier caso se espera la respuesta del CM a la petición de expulsión. Cuando se recibe una petición PtObE se suministra el bloque al CM (RpDc) y se espera la respuesta del CM en el estado MII. Si la petición es PtObL, posteriormente se puede recibir una petición PtObE. Entonces se utiliza un estado transitorio para identificar el hecho. Este estado se denomina MIL. Si, antes de recibir la respuesta del CM (RpX), se recibe una petición PtObE del CM, el CC responde y establece MII como estado transitorio del bloque⁷.

PETICIONES Y RESPUESTAS DEL CM SE ENCAMINAN POR REDES DISTINTAS

El protocolo que se describe básicamente es el mismo que el descrito en el apartado previo. La diferencia reside en que un CC, en un cruce de peticiones, debe inferir si la petición del CM debe procesarse: a) suponiendo el estado estable inicial de la transacción pendiente en el CC o b) hay que esperar a que finalice la transacción pendiente en el CC (procesado en el estado estable final). Esta diferencia es debida a la organización del multiprocesador. En concreto, las redes de interconexión utilizadas.

En la Figura 8.21 se muestra la diferencia básica, en la descripción funcional de la secuencia de mensajes en una transacción, entre el apartado previo y el actual. En la parte izquierda se muestra el apartado previo y en la parte derecha el apartado actual. Para efectuar la inferencia en un CC, cuando se produce un cruce de peticiones, se añade una nueva petición (PtOblnv).

^{7.} El diagrama de transiciones entre estados es funcionalmente equivalente al del Capítulo 7, pero el presentado es este apartado es más ortodoxo o canónico.

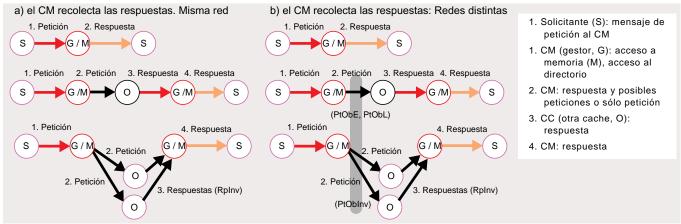


Figura 8.21 Flujos de mensajes en dos protocolo de directorio MLI donde el CM recolecta las respuestas: a) misma red para peticiones y respuestas de los CM a los CC (apartado previo) y b) redes distintas para peticiones y respuestas de los CM a los CC. El sombreado muestra la diferencia básica.

Organización del multiprocesador

El protocolo utiliza cuatro redes lógicas (Figura 8.22): RI, RCM, RP y RMC.

Los CC utilizan la RI para emitir peticiones y la RCM para emitir respuestas al CM. Los CM utilizan la RP para emitir peticiones y la RMC para emitir respuestas a los CC⁸.

En esta organización un CC no observa el orden de emisión de peticiones y respuestas desde un CM al CC. Esto es debido a que se utilizan redes distintas para transmitir las peticiones y respuestas de los CM a los CC.

La cola de mensajes de petición en un CC se denomina CPC y la cola de mensajes de respuesta se denomina CRC. Para representar retardo en una red se replica el acrónimo de la red en ciclos consecutivos.

^{8.} Las redes pueden ser de tipo crossbar, anillo o malla. El encaminamiento entre un emisor y un receptor debe estar preestablecido.

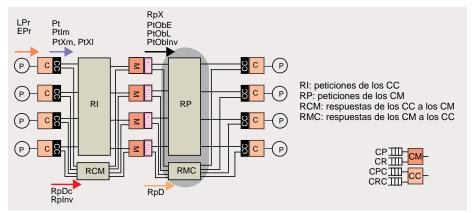


Figura 8.22 Redes distintas para peticiones y respuestas del CM. Organización del multiprocesador. Redes lógicas y mensajes. En sombreado se muestra la diferencia básica con el apartado previo.

Mensajes del protocolo

En la Figura 8.22 se muestra un esquema genérico de la transmisión de mensajes con peticiones y respuestas entre los CC y el CM y viceversa.

En la descripción del protocolo se distinguen los siguientes tipos de mensajes:

- Peticiones de un CC a un CM (Pt, PtIm, PtXm, PtXI)
- Peticiones de un CM a los CC (PtObL, PtObE, PtObInv)
- Respuestas de un CM a un CC (RpD, RpX)
- Respuestas de un CC a un CM (RpDc, RpInv)

Los mensajes que se han añadido o modificado respecto al protocolo del Capítulo 7 se muestran en la Tabla 8.2. Posteriormente se describe la necesidad de la petición PtOblnv.

Controlador de memoria	Controlador de coherencia	Comentario			
Mensajes de petición a CC y acciones	Respuestas				
PtOblnv: petición de invalidación op dirección Id	RpInV: respuesta a una petición de invalidación	El CC que recibe una petición PtOblnv en el estado L invalida el bloque y responde con RpInv.			
PtObE: petición de observación de escritura op dirección Id	RpDc: respuesta con el bloque	El CC que recibe la petición PtObE tiene el bloque en exclusividad. Emite una respuesta con el bloque y la acción de invalidación está implícita en el tipo de respuesta. Esta respuesta también se utiliza si la petición del CM es PtObL. En este caso no se invalida el bloque al responder.			

Tabla 8.2 Petición del CM a los CC y respuesta de estos al CM.

Estados y transiciones

Los estados en el directorio y en la cache son los mismos que en el protocolo previo de este capítulo (Figura 8.5).

La descripción funcional de la secuencia de mensajes también es la misma que en el protocolo previo de este capítulo (Figura 8.6, Figura 8.9, Figura 8.10, Figura 8.13).

La diferencia básica, con el protocolo previo de este capítulo, es que los mensajes de petición y los mensajes de respuesta de los CM a los CC se transmiten por redes distintas. Entonces, en un CC no se observa un orden entre peticiones y respuestas emitidas desde un mismo CM. Cuando sea necesario conocer este orden hay que inferirlo en el CC, a partir del estado del bloque en la cache y la petición del CM. Para ello se introduce la petición PtOblnv, la cual utilizan los CM para solicitar la invalidación de un bloque en una cache⁹.

Diagrama completo de estados y transiciones

En las Figura 8.23 se muestran los diagramas de transiciones entre estados de un bloque en cache y en el directorio. En el CC se distinguen, en diagramas separados, las transiciones iniciadas por el agente procesador (CC solicitante) y el agente observador (otros CC). En el diagrama etiquetado como CC solicitante están incluidas las peticiones del procesador que no requieren iniciar transacciones explícitas de coherencia. Observe que, en el CM, la petición al efectuar la transición del estado L al estado LM es PtOblnv en lugar de PtObE (Figura 8.15). Por tanto, la petición que recibe un CC para efectuar la transición del estado L al estado I es PtOblnv.

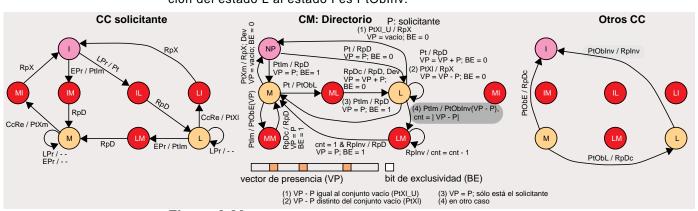


Figura 8.23 Redes distintas para peticiones y respuestas del CM. Estados y transiciones entre estados de un bloque en una cache. En sombreado se muestra la diferencia básica con el apartado previo.

9. Recordemos que en el protocolo descrito previamente en este capítulo se utiliza la petición

En el diagrama del CM (Figura 8.23) hay que distinguir dos casos al procesar la petición PtXI de un CC (PtXI, PtXI_U). El estado final depende de si el CC, que efectúa la petición, es el único que está en el vector de presencia.

Inferencia en un CC del orden entre peticiones y respuestas emitidas desde un CM

Un CC tiene que inferir el orden en que un CM ha procesado una petición de este CC respecto de una petición de otro CC al mismo bloque.

En un directorio sólo se identifican estados estables de los bloques en los CC. En consecuencia, la emisión, por parte de un CM, de una petición a un CC, al procesar el CM una petición de otro CC, está determinada por el estado estable del bloque en el CC del cual tiene constancia el CM.

Misma red para transmitir mensajes de petición y mensajes de respuesta desde un CM a un CC. Cuando se utiliza la misma red para transmitir peticiones y respuestas de los CM a los CC, y la red mantiene el orden entre emisor y receptor, una petición del CM a un CC está ordenada respecto de la respuesta previa del CM a este CC. Esta respuesta del CM ha determinado un estado estable en el CC y este estado es del cual tiene constancia el CM. El CM procesa la petición teniendo en cuenta este estado.

Esta propiedad es la que se ha utilizado en el protocolo del apartado previo para inferir que las peticiones de un CM, que se reciben en un CC, estando el bloque en un estado transitorio, han sido ordenadas por el CM antes que la petición pendiente del CC (Figura 8.24). Por tanto, el CC responde al CM como si el bloque estuviera en el estado estable del que proviene¹⁰.

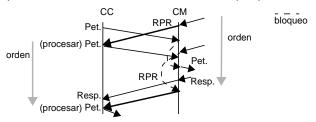


Figura 8.24 Cruce de peticiones en un CC. Comunicación entre un CM y los CC: una red para transmitir mensajes del CM.

En resumen, un CC puede iniciar una nueva transacción cuando un bloque está en un estado estable. Entonces, el CC establece un estado transitorio para el bloque. Si este CC recibe una petición del CM, esta petición ha sido

10. Si en un CC se puede recibir una secuencia de peticiones del CM hay que tener en cuenta las respuestas previas. En el CC hay que recordar que el CM ha ordenado previamente peticiones de otros CC.

emitida por el CM teniendo en cuenta el estado estable del que se proviene para llegar al estado transitorio. Por tanto, el CC tiene que procesar la petición del CM.

Redes distintas para transmitir mensajes de petición y mensajes de respuesta de un CM a un CC. En un CC no existe orden en la recepción de mensajes provenientes de redes distintas, aunque exista orden en la recepción de mensajes provenientes de una misma red. En estas condiciones puede ser factible que (Figura 8.25)

- Una respuesta del CM adelante a una petición previa del CM al mismo CC.
- Una petición del CM adelante a una respuesta previa del CM al mismo CC.



Figura 8.25 Cruce de peticiones en un CC: a) una respuesta adelanta a una petición previa y b) una petición adelanta a una respuesta previa.

Una respuesta del CM no puede adelantar a una petición previa del CM al mismo bloque

En este protocolo no es posible que una respuesta adelante a una petición previa del CM al mismo bloque. Un CM no procesa peticiones que acceden a un bloque en un estado transitorio en el directorio. Además, un CC siempre responde al CM una petición del mismo. Entonces, en la situación a) de la Figura 8.25 el CM procesa, como muy pronto, la petición del CC después de recibir la respuesta a su petición (Figura 8.26).

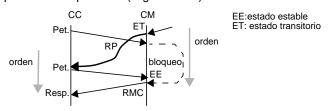


Figura 8.26 Una respuesta del CM no puede adelantar a una petición previa del CM al mismo bloque.

Una petición del CM adelanta a una respuesta previa del CM al mismo bloque

La situación b) de la Figura 8.25 es factible en este protocolo. El CM procesa una petición de otro CC en un estado estable que induce una petición a un CC. El CM ha emitido previamente una respuesta al mismo CC que accede al mismo bloque. Como los mensajes se transmiten por redes distintas el retardo que experimentan puede ser distinto. En estas condiciones, la petición del CM puede llegar antes que la respuesta al CC.

Entonces, un CC al procesar una petición del CM, es un estado transitorio, tiene que determinar si el CM ha emitido la petición teniendo en cuenta el estado inicial o final de la transacción pendiente en el CC. Esto es, la petición del CM ha sido emitida antes que la respuesta o al contrario (Figura 8.27).

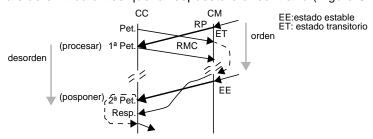


Figura 8.27 Cruce de peticiones en un CC. Comunicación entre los CM y los CC: una red para transmitir peticiones del CM y otra red para transmitir respuestas del CM

Ejemplo. Un bloque en el estado L puede recibir una petición del CM para que se invalide el bloque. Un bloque en el estado M puede recibir una petición para que suministre el bloque y el próximo estado del bloque sea inválido. Cuando el CC está en el estado L y emite una petición para obtener la exclusividad, el próximo estado es el estado transitorio LM.

En el protocolo que se describe, un CM utiliza redes distintas para transmitir las peticiones y las respuesta a un CC. Entonces, un CC puede recibir una petición del CM, para un bloque en el estado LM, que ha sido inducida por una petición de un segundo CC, que el CM ha procesado antes que la transacción pendiente del CC (1ª petición en la Figura 8.27). También, la petición del CM puede haber sido inducida por una petición de un tercer CC, que el CM ha procesado después de la transacción pendiente del CC (2ª petición en la Figura 8.27). Ahora bien, al recibir el CC esta petición del CM, la transacción del CC sigue pendiente, debido a que la respuesta se recibe por otra red (RMC). Por tanto, el procesado en el CC de la petición del CM debe posponerse.

En consecuencia, para distinguir cuando se procesa una petición del CM en un CC, son necesarias peticiones distintas desde el CM¹¹. Por ello, se ha añadido la petición PtOblnv en el protocolo.

Bloque en el estado L en el directorio. Cuando un CC solicita la exclusividad del bloque, el CM utiliza la petición PtOblnv para invalidar las copias del bloque en otras cache (parte izquierda de la Figura 8.28).

Bloque en el estado M en el directorio. Cuando un CC solicita la exclusividad, el CM utiliza la petición PtObE para solicitar el suministro del bloque y la invalidación del bloque a una cache (parte derecha de la Figura 8.28).

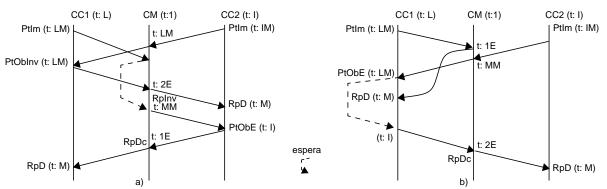


Figura 8.28 Identificación de cruces de peticiones en el CC1: a) El CM ordena la petición de otro CC antes, b) el CM ordena la petición de otro CC después.

En estas condiciones, un CC infiere que una petición PtObInv del CM debe procesarse en el estado transitorio LM y el procesado de una petición PtObE debe posponerse hasta que llegue la respuesta de la petición pendiente (Figura 8.29).

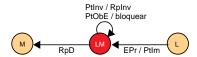


Figura 8.29 En un estado transitorio de un bloque, identificación en un CC del orden de procesado, respecto a la petición propia, de una petición de otro CC en el CM, la cual ha inducido una petición del CM al CC.

^{11.} Recordemos que, en todos los protocolos MLI descritos hasta ahora, la petición PtObE se interpreta en función del estado del bloque en la cache. Si el estado de bloque es L se interpreta como invalidación. Si el estado del bloque es M se interpreta como suministro e invalidación.

Identificación de posibles cruces de peticiones

Funcionalmente los posibles cruces de peticiones, en los cuales el CC infiere que la petición del CM debe procesarse, teniendo en cuenta el estado inicial de la transacción pendiente en el CC, han sido descritos en el Capítulo 7.

Desorden en la recepción de peticiones y respuestas del CM

Los estados transitorios de un bloque, en los cuales un CC está esperando recibir una respuesta de un CM, que incluye el bloque son: IL, IM y LM. Los estados del bloque en la cache al finalizar la transacción pendiente son L, M y M respectivamente.

Para que se produzca un cruce, el CM debe procesar una petición que modifique el estado del bloque en la cache.

Bloque en el estado L. El estado transitorio del que proviene es IL. La petición del CC1 al CM ha sido Pt. El estado del bloque en el directorio es L (Figura 8.30).

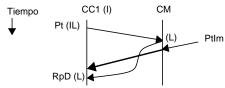


Figura 8.30 Bloque en estado L en la cache C1. Peticiones de otros CC.

El CM puede procesar una petición PtIm de otro CC, la cual induce que el CM emita una petición PtObInv al CC1.

Bloque en el estado M. El estado transitorio del que proviene es IM o LM. La petición del CC1 al CM ha sido PtIm. El estado del bloque en el directorio es M (Figura 8.31).

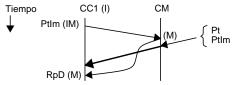


Figura 8.31 Bloque en estado M en la cache C1. Peticiones de otros CC.

El CM puede procesar una petición Pt o PtIm de otro CC, las cuales inducen respectivamente que el CM emita una petición PtObL o PtObE al CC1.

En la Figura 8.32 se muestra el orden de procesado en el CM de varias secuencia de accesos.

Orden de procesado en el CM							
I	J	K					
P1 load t	P1 store t	P1 store t					
P2 store t	P2 load t	P2 store t					

Figura 8.32 Bloque en estado L o M en la cache C1 al finalizar la primera transacción. Orden de procesado en el CM de una secuencia de accesos. La petición del CM, inducida por la petición del CC2, llega al CC1 antes que la respuesta.

En la RI no hay arbitraje por antigüedad: Identificación de cruces de peticiones

En el Capítulo 7 se han analizado los cruces teniendo en cuenta que el arbitraje en la RI mantiene el orden de antigüedad de las peticiones de los CC. Además, el CM extrae de la CP las peticiones en el orden de llegada.

En el protocolo descrito en este apartado no se cumple esta propiedad. Por ejemplo, la latencia efectiva del camino entre un CC y un CM puede ser significativamente distinta, en función del CC y del CM que se comunican. Por otro lado, en una red en malla puede existir congestión en algunos enlaces de comunicación y no existe un arbitraje centralizado.

En este contexto, hay que tener en cuenta la posibilidad de nuevos cruces de peticiones.

El CM puede procesar la petición de exclusividad de un CC2 y la petición de expulsión del bloque por este CC2¹², antes que la petición de expulsión de un CC1, que está identificado en el VP, al procesar la petición de exclusividad del CC2. El CC1 ha emitido la petición al CM antes de recibir la petición del CM, la cual ha sido inducida por la petición de exclusividad del CC2.

Partiendo de los ordenes de procesado en el CM de las secuencias de acceso del Capítulo 7 (Figura 7.30 y Figura 7.32) se pueden construir los ordenes de secuencias de acceso de la Figura 8.33¹³.

- 12. Recordemos que un CC sólo tiene en curso de procesado una petición a un mismo bloque. Por ejemplo, en el orden AX de la Figura 8.33, el procesador P2 no emite la petición de expulsión antes de que haya consolidado la instrución previa: a) llegue el dato o b) recibido la confirmación de que ha consolidado la instrucción store. Si el algoritmo de reemplazo selecciona un contenedor, que aún no contiene el bloque, se suspende la interpretación de la instrucción que determina el reemplazo.
- 13. Podemos suponer que la expulsión es voluntaria o forzada por un fallo. En este último caso, el bloque al que se accede está almacenado en otro módulo de memoria. Por esta razon no se muestra en el entrelazado. Recordemos que la petición para servir el fallo se puede emitir antes o después de la petición de expulsión.

Orden de procesado en el CM													
AX	ВХ		СХ	D	DX EX		FX	GX		GX		НХ	
P2 store t	P2 store t	P2 store t	P2 store t	P2 store t	P2 store t	P2 store t	P2 load t	P2 store t	P2 store t	P3 load t	P3 load t		
P2 PtXm t	P3 load t	P3 load t	P2 PtXm t	P3 load t	P3 load t	P2 PtXm t	P2 PtXI t	P3 load t	P3 load t	P2 store t	P2 store t		
P1 PtXI t	P2 PtXm t	P2 PtXI t	P1 store t	P2 Ptxm t	P2 PtXI t	P1 PtXm t	P1 PtXm t	P2 PtXm t	P2 PtXI t	P3 PtXI t	P3 PtXI t		
	P3 PtXI t	P3 PtXI t		P3 PtXI t	P3 PtXI t			P3 PtXI t	P3 PtXI t	P2 PtXm t	P2 PtXI t		
	P1 PtXI t	P1 PtXI t		P1 store t	P1 store t			P1 PtXm t	P1 PtXm t	P1 PtXm t	P1 PtXm t		

Figura 8.33 Orden de procesado en el CM de secuencias de accesos a memoria. Una secuencia Zx ha sido construida a partir de la secuencia Z de la Figura 7.30 y de la Figura 7.32.

Cruces de peticiones en un CM y en un CC

Controlador de memoria

En un CM, los estados con ventana de vulnerabilidad y los cruces de peticiones en estados estables L y M son los mismos que en el protocolo previo de este capítulo. Las acciones y respuestas del CM también son las mismas (Figura 8.35).

Ahora bien, debido a que el árbitraje en la RI no mantiene la antigüedad de las peticiones, se pueden inferir cruces en el estado NP.

Dado el orden AX de secuencia de accesos, en el CM se infiere un cruce de peticiones en el estado NP (Figura 8.34). Dados los dos ordenes de secuencias de accesos etiquetados como BX, el cruce de peticiones, al procesar la petición de CC1, se infiere en el mismo estado.

En los ordenes CX y los dos DX no se infiere ningún cruce en el CM.

Dado el orden EX de secuencia de accesos, en el CM se infiere un cruce de peticiones en el estado NP. Dados los dos ordenes de secuencias de accesos etiquetados como GX, el cruce de peticiones, al procesar la petición de CC1, se infiere en el mismo estado

Dado el orden FX de secuencia de accesos, en el CM se infiere un cruce de peticiones en el estado L. El CC1 está en el VP, pero el tipo de petición no es esperado.

Dados los dos ordenes de secuencia de accesos HX, en el CM se infiere un cruce de peticiones, al procesar la petición de CC1, en el estado NP.

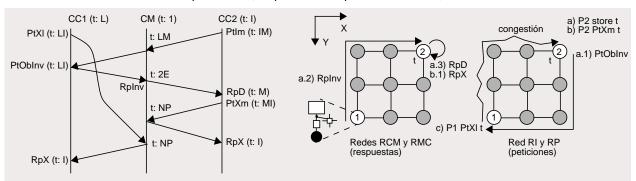


Figura 8.34 Ejemplo de cruce de peticiones. Orden AX y redes en malla. La variable t está ubicada en el módulo de memoria del nodo 2. No se muestra la respuesta RpX al CC1.

Controlador de coherencia.

CC1 infiere un cruce y pospone el procesado.

En un CC, dado un bloque en un estado transitorio, existe un conjunto de peticiones del CM que el CC responde infiriendo que el CM ha procesado la petición de otro CC antes que la petición del CC (Figura 8.35). Estos cruces de peticiones se corresponden con los descritos en el protocolo previo de este capítulo (Figura 8.20).

Identificación de nuevos cruces de peticiones. El otro conjunto de peticiones que un CC infiere debe responderse una vez finaliza la transacción pendiente (IM, IL, LM). Para efectuar la inferencia se utiliza el tipo de petición y el estado transitorio donde se recibe¹⁴. Para facilitar la comprensión en el siguiente desarrollo, se replica la Figura 8.32 en el margen izquierdo.

Dado el orden I de secuencias de acceso (Figura 8.32), el CM procesa la petición del CC2 cuando el bloque está en el estado L en el directorio. El CC1 está identificado en el VP. En consecuencia el CM emite una petición PtOblnv al CC1. El CC1 recibe esta petición cuando el bloque está en el estado transitorio IL en la cache C1. Teniendo en cuenta el estado y el tipo de petición, el

Dado el orden J de secuencias de acceso, el CM procesa la petición de CC2 cuando el bloque está en el estado M en el directorio. El CC1 está identificado en el VP y el BE = 1. En consecuencia el CM emite una petición PtObL al CC1.



^{14.} Este cruce de peticiones se denomina temprano, ya que la petición llega antes de disponer del bloque.

El CC1 recibe esta petición cuando el bloque está en el estado transitorio IM o LM en la cache C1. Teniendo en cuenta el estado y el tipo de petición el CC1 infiere un cruce y pospone el procesado.

Dado el orden K de accesos a memoria, el razonamiento es similar teniendo en cuenta que la petición del CM, inducida por la petición de CC2, es PtObE.

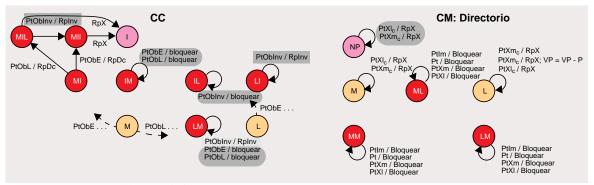


Figura 8.35 Redes distintas para peticiones y respuestas del CM. Diagrama de estados y transiciones de un CM y un CC cuando se producen cruces de peticiones. En sombreado se muestran las diferencias básicas con el protocolo del apartado previo. En sombreado rectangular se muestra el cambio de petición del CM. También se muestran las peticiones del CM en estados estables.

En resumen, en los estados IM y LM un CC puede recibir las peticiones PtObE y PtObL. En los dos casos, el CC infiere un cruce y debe esperar a que llegue la respuesta a su transacción pendiente. Se bloquea el análisis de la cola de peticiones de los CM al CC (CPC). En las dos peticiones el CM está solicitando que el CC suministre el bloque (Figura 8.35)¹⁵.

Un CC, con un bloque en el estado IL, puede recibir una petición PtOblnv del CM a este bloque. El CC infiere un cruce de peticiones y pospone el procesado. El CM está solicitando que se invalide el bloque.

Ejemplo. En la Figura 8.36 se muestra un ejemplo donde se observa el procesado, en el CC1, de una petición PtOblnv del CM en un estado transitorio y el bloqueo del procesado de una petición PtObE hasta que se dispone del bloque. En el CC1 la petición PtOblnv del CM se procesa cuando se recibe (ciclo 6). La petición PtObE del CM llega antes que la respuesta a la petición de CC1 (ciclo 20). El procesado de la misma en el CC se posterga (ciclo 22). La petición del CC3, que ha inducido la petición PtObE del CM al CC1, ha sido procesada en el CM después de la petición de CC1.

15. El bloque lo recibe el CC por la red de respuestas (RMC) y una respuesta del CM siempre se procesa. Las peticiones del CM llegan por la red RP.

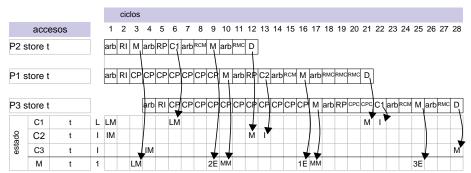


Figura 8.36 Diagrama temporal donde se muestra la inferencia en un CC, cuando un bloque está en un estado transitorio: a) una petición PtOInv se procesa y b) una petición PtObE se posterga hasta disponer del bloque. Varios ciclos en RMC indican retardo en la red. El acrónimo CPC se corresponde con la cola de peticiones en un CC.

En la Figura 8.37 se muestra un ejemplo donde se compara el protocolo del apartado previo (parte izquierda de la figura) con el protocolo de este apartado (parte derecha de la figura).

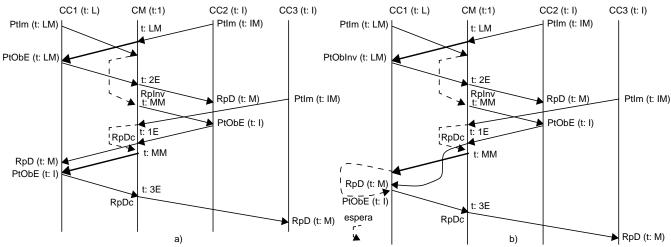


Figura 8.37 Cruces de peticiones en un CC cuando el bloque está en el estado LM. Comunicación entre los CM y los CC: a) una red para transmitir mensajes y b) una red para transmitir peticiones y otra red para transmitir respuestas.

En la parte izquierda de la Figura 8.37 las peticiones y respuestas del CM se reciben por el mismo canal de comunicación (RPR), el cual mantiene el orden de emisión. En el CC1 la segunda petición del CM llega después de la respuesta a la petición del CC1. En la parte derecha de la Figura 8.37

peticiones y respuestas del CM utilizan redes distintas (RP, RCM). En el CC1 la segunda petición del CM llega antes que la respuesta a la petición del CC1. Sin embargo, la petición del CC3, que ha inducido la petición PtObE del CM al CC1, ha sido procesada en el CM después de la petición del CC1.

EL SOLICITANTE ES EL RECEPTOR DE LAS RESPUESTAS

En los protocolos descritos previamente, para obtener la exclusividad de acceso a un bloque, pueden ser necesarios cuatro mensajes en secuencia (4 saltos, Figura 8.1). También son necesarios estos cuatro mensajes cuando una cache tiene el bloque en exclusividad y una transacción accede al bloque. En estos protocolos los CC, involucrados en una acción de coherencia, responden al CM y éste al CC solicitante.

Una forma de reducir la latencia de algunas transacciones es que los CC, involucrados en una acción de coherencia, respondan al CC solicitante en lugar de al CM. En algunas transacciones el número total de mensajes es el mismo, pero la serialización es de tres mensajes (3 saltos).

Por otro lado, se descongestiona al CM. Cuando un CM recibe las respuestas de los CC, involucrados en una acción de coherencia, debe de gestionar hasta P de ellas¹⁶. Cuando las respuestas de los CC, involucrados en una acción de coherencia, las recibe el CC solicitante, éste debe gestionar una acción de coherencia¹⁷.

Descripción funcional de la secuencia de mensajes en una transacción

En la Figura 8.38 se muestra la descripción funcional de protocolo previo y la del protocolo que se describe en este apartado.

Cuando no hay copia del bloque en otras cache o la transacción es de lectura y una cache no tiene el bloque en exclusividad, la secuencia de mensajes y su serialización es la misma en los dos protocolos (2 pasos, parte superior de la Figura 8.38).

Cuando una cache tiene el bloque en exclusividad la serialización de los mensajes se reduce en un paso. El solicitante recibe el bloque del CC que lo tiene en exclusividad. Si la transacción es de lectura, el CC que suministra el bloque también envia una copia al CM (centro de la Figura 8.38).

- 16. Si un procesador puede tener k accesos pendientes (por ejemplo, prebúsqueda), el número total es k x P.
- 17. Si un procesador puede tener k accesos pendientes, el número total es k.

Cuando hay caches con copias del bloque y una transacción solicita obtener la exclusividad, las respuestas de invalidación las recolecta el CC solicitante (parte inferior de la Figura 8.38).

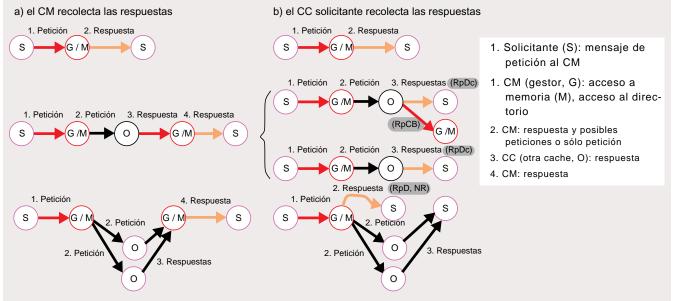


Figura 8.38 Flujos de mensajes en dos protocolo de directorio MLI. a) el CM recolecta las respuestas de los CC y b) el CC solicitante recolecta las respuestas de los CC. En sombreado las diferencias básicas con el protocolo del apartado previo.

En la transacción de la parte inferior de la Figura 8.38, el CM, en la respuesta que emite al solicitante, indica el número de respuestas que debe recolectar para dar por completada la transacción.

El CM da por completada una transacción una vez emite los mensajes de petición y respuesta, excepto en una transacción de lectura y una cache tiene el bloque en exclusividad. En este caso, el CM espera la respuesta, con el bloque, del CC cuya cache tenía el bloque en exclusividad (segundo caso en el centro de la Figura 8.38). El CM utiliza la respuesta para actualizar la memoria.

Notemos que un CM puede empezar a procesar la siguiente transacción a un bloque antes de que los CC involucrados en la transacción previa, al mismo bloque, hayan procesado las peticiones del CM¹⁸. En estas condiciones, debido a que el CC tiene que inferir el orden entre peticiones y respuestas del CM hay que analizar posibles cruces de mensajes¹⁹. Estos cruces pueden requerir la modificación del protocolo o la asignación de la red por la cual se transmite un mensaje. En caso contrario el protocolo puede bloquearse.

18. Dos últimos flujos de mensajes de la Figura 8.38.

Organización del multiprocesador

El protocolo requiere que exista una comunicación lógica directa entre los CC. La función de esta comunicación es la transmisión del mensaje de respuesta de cada CC, que participa en la acción de coherencia, al CC solicitante (RDR, Figura 8.39). Esta red lógica también es utilizada por los CM para emitir las respuestas.

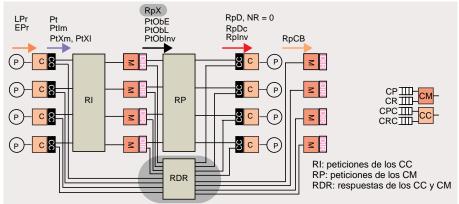


Figura 8.39 Recepción de respuestas de los CC en el solicitante. Organización del multiprocesador. Redes lógicas y mensajes en un protocolo MLI. En sombreado se muestran las diferencias básicas respecto del protocolo del apartado previo.

Un CC genera las respuestas en paralelo. Los mensajes pueden propagarse en paralelo en función de la ocupación de la red.

La cola de mensajes de respuesta en un CC se denomina CRC.

Mensajes del protocolo

En la Figura 8.39 se muestra un esquema genérico de la transmisión de mensajes entre los CC y los CM y viceversa.

En la descripción del protocolo se distiguen los siguientes tipos de mensajes:

- Peticiones de un CC a un CM (Pt, PtIm, PtXm, PtXI)
- Peticiones de un CM a los CC (PtObL, PtObE, PtOInv, RpX)
- Respuestas de un CM a un CC (RpD, NR). La respuesta RpD incluye el bloque y el número de respuestas que debe esperar (NR = K)

19. Un enlace punto a punto mantiene el orden de los mensajes entre emisor y receptor. Sin embargo, las peticiones y respuestas del CM utilizan redes distintas. La red puede ser de tipo crossbar, anillo o malla. El encaminamiento entre un emisor y un receptor debe estar preestablecido.

- Respuestas de un CC a un CC solicitante (RpDc, RpInv)
- Respuesta de un CC a un CM (RpCB)

Los mensajes que se han añadido o modificado respecto al protocolo del Capítulo 7 se muestran en la Tabla 8.3 y en la Tabla 8.4.

Controlador de coherencia (CC)	Controlador de memoria (CM)	Comentario	
Mensajes de petición al CM	Mensaje de respuesta		
Pt: petición de bloque op dirección Id	RpD, NR: respuesta con el bloque y el número de respuestas que debe esperar Id bloque NR	Se lee el bloque de la memoria, se actualiza el directorio, se suministra el bloque y se indica el número de respuestas que el CC solicitante debe recibir (NR).	

Tabla 8.3 Petición del CC a un CM y respuesta del CM.

Aunque RpX es una respuesta de un CM a un CC se utiliza la red RP para transmitir el mensaje. Posteriormente se justifica la decisión. El objetivo es que un CC pueda liberar el contenedor al procesar la respuesta del CM, teniendo la seguridad de que posteriormente no recibirá una petición del CM al mismo bloque, que ha sido emitida por el CM antes que la respuesta.

Controlador de memoria Mensajes de petición a los CC y acciones	Controlador de coherencia Respuestas	Comentario
PtObE: petición de observación de escritura op dirección Id DEST	RpDc: respuesta con el bloque e invalidación Id bloque DEST	El CC emite una respuesta al solicitante (DEST) con el bloque e invalida el bloque.
PtObL: petición de observación de lectura op dirección Id DEST	RpCB: respuesta con el bloque al CM Id bloque RpDc: respuesta con el bloque al CC Id bloque DEST	El CC emite una respuesta al solicitante (DEST) con el bloque y cambia el estado del bloque para indicar que no hay exclusividad. El CC suministra el bloque al CM.
PtOblnv: petición de invalidación op dirección Id DEST	RpInv: respuesta a una petición de invalidación Id DEST	El CC que recibe la petición invalida el bloque y responde al solicitante (DEST).

Tabla 8.4 Peticiones de un CM a los CC y respuestas de un CC al CC solicitante y al CM.

Recepción de respuestas en el solicitante

En una transacción donde se solicita la exclusividad, un CC puede recibir dos conjuntos de respuestas²⁰: a) la respuesta del CM que, además del bloque, indica el número de respuestas (NR) que deben esperarse y b) las respuestas de confirmación de invalidación de los CC participantes en la acción de coherencia.

La respuesta del CM y las respuestas de los CC se transmiten por la RDR donde, como mucho, se garantiza orden de los mensajes entre emisor y destino. Por tanto, la respuesta del CM y las respuestas de los CC pueden llegar en cualquier orden al CC solicitante. Para gestionar el desorden, en la recepción de las respuestas, se utilizan estados transitorios en el CC.

En la Figura 8.40 se muestra un ejemplo del diagrama de transiciones entre estados. Las respuestas de los CC son RpInv y hay que contabilizar su número. La respuesta del CM es RpD e indica el número de respuestas que deben esperarse (NR).

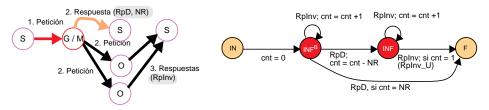


Figura 8.40 Estados transitorio cuandos se esperan varios conjuntos de respuestas.

A medida que llegan las respuestas de invalidación de los CC se contabilizan (INF^B). Cuando llega la respuesta del CM se calcula el número de respuestas de invalidación que aún no han llegado. Cuando han llegado todas las respuestas de invalidación se efectúa la transición a un estado estable (F). En caso contrario, se efectúa una transición a un estado transitorio (INF), que identifica que sólo se esperan respuestas de invalidación. Cuando llega la última respuesta de invalidación (RpInv_U) se efectúa una transición a un estado estable (F). El superíndice B en un estado indica que aún no ha llegada la respuesta del CM, la cual es esperada en la transición entre los estados estables inicial (IN) y final (F).

En resumen, en un CC es necesario utilizar, en ocasiones, un reconocedor de secuencias de respuestas. En este caso se puede expresar como (RpInv*, RpD, RpInv*).

20. El estado del bloque en el directorio es L.

Estados y transiciones

Cuando un CC tiene el bloque en exclusividad y el CM procesa una transacción de lectura hay que actualizar la memoria. En estas condiciones, el CM debe esperar a que responda el CC, con el bloque, antes de procesar otra transacción al mismo bloque. Para identificar la espera de una respuesta de un CC se utiliza un estado transitorio en el CM.

El directorio se utiliza para detectar accesos concurrentes al mismo bloque y gestionarlos (ventana de vulnerabilidad). Esto es, cuando la petición en la cabeza de la CP accede a un bloque en un estado transitorio, el CM efectúa la detección. La gestión utilizada, en todos los protocolos descritos previamente, es bloquear el procesado de esta petición, hasta que el estado del bloque es estable. En este protocolo, en un estado transitorio se procesan peticiones de expulsión.

Directorio. Los estados estables de un bloque en el directorio son los mismos que en el Capítulo 7 (Figura 8.41).

Las respuestas en una transacción de exclusividad son recolectadas por el CC solicitante. Entonces, no son necesarios estados transitorios en el CM para este menester. Ahora bien, en el CM se utiliza un estado transitorio, denominado ML, para identificar la espera de la respuesta del CC que debe suministrar el bloque con el que se actualiza la memoria.

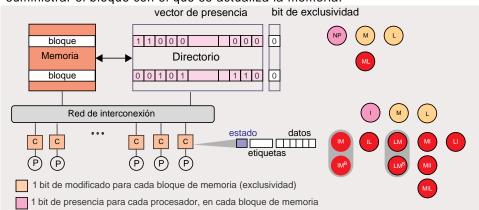


Figura 8.41 Recepción de respuestas de los CC en el solicitante. Estados de un bloque en el directorio y en un contenedor de cache.

Cache. Los estados estables de un bloque en una cache son los mismos que en el Capítulo 7 (Figura 8.41). Para identificar la espera de la respuesta del CM y las respuestas de los CC, a una petición del CC, se utilizan siete estados transitorios (IM^B, IM, IL, LM^B, LM, MI, LI). Los dos nuevos estados transitorios, respecto a los utilizados en el protocolo del apartado previo, se

utilizan para gestionar el desorden en la recepción de la respuesta del CM y las respuestas de los CC, participantes en la acción de coherencia, al CC solicitante.

Para describir el protocolo se utilizan las dos peticiones del procesador que requieren acceder al directorio (fallo en una instrucción load, store o una instrucción store que accede a un bloque sin permiso de exclusividad) y las dos posibilidades de ubicación del bloque solicitado, en memoria o en una cache. Posteriormente se detallan las transiciones entre estados en una expulsión de un bloque de cache.

En la descripción se mostrarán las transiciones entre estados de un bloque, en el CC que efectúa la petición (agente procesador), en el CM y en otros CC (agente observador).

Fallo en lectura

Memoria tiene actualizado el bloque. En la parte izquierda de la Figura 8.42 se muestra el flujo de mensajes en un fallo de lectura. El CC emite un mensaje con una petición de lectura de bloque (Pt). El CM accede al directorio para leer el VP y el BE (estado del bloque) y determina que el estado del bloque es NP o L. El CM emite un mensaje de respuesta (RpD, NR = 0), que incluye el bloque y el número adicional de respuestas que debe esperar, al CC que ha efectuado la petición.

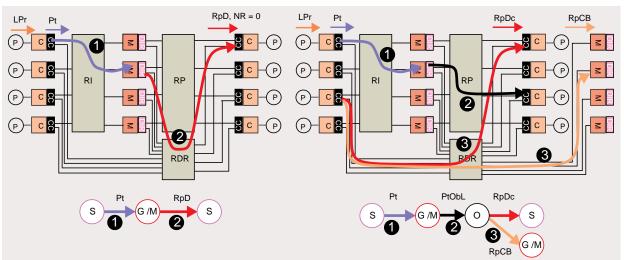


Figura 8.42 Recepción de las respuestas de los CC en el solicitante. Flujo de mensajes en un fallo de lectura.

Una cache tiene el bloque en exclusividad. En la parte derecha de la Figura 8.42 se muestra el flujo de mensajes entre el CC que efectúa la petición y el CM, entre el CM y el CC que tiene el bloque en exclusividad y entre este último y el CC solicitante y el CM.

El CM emite una petición PtObL al CC que tiene el bloque en exclusividad para que suministre el bloque. Este CC responde, con sendos mensajes que incluyen el bloque, al CC solicitante (RpDc) y al CM (RpCB). El CM actualiza el directorio y la memoria al recibir la respuesta del CC.

Diagramas de transiciones entre estados. En la Figura 8.43 se muestran las transiciones entre estados en un fallo de lectura en: a) la cache del CC que efectúa la petición (solicitante), b) el directorio (CM) y c) otros CC. El CC emite el mensaje de petición de lectura y establece IL como estado transitorio del bloque. El bloque permanece en este estado hasta que recibe la respuesta del CM (RpD, Nr = 0) o del CC que tiene el bloque en exclusividad (RpDc), la cual es procesada por el CC que establece como estado estable del bloque el estado L.

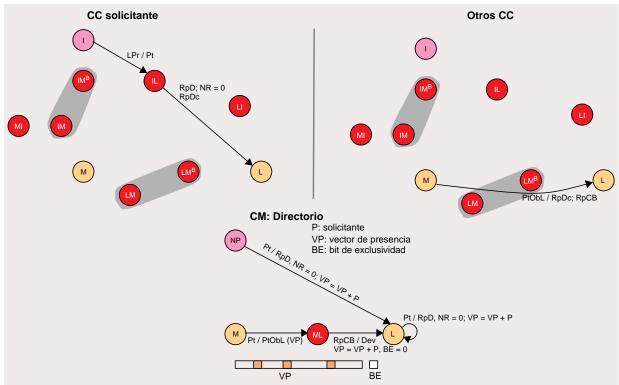


Figura 8.43 Recepción de las respuestas de los CC en el solicitante. Transiciones entre estados en un fallo de lectura.

El CM al recibir el mensaje de petición de lectura accede al directorio y especulativamente a memoria. Si el estado del bloque en el directorio es NP o L, el CM determina que puede responder a la petición utilizando el bloque leído de memoria. El CM, además de enviar la respuesta, actualiza el VP añadiendo al CC, que ha efectuado la petición, en el VP. Los otros CC no reciben peticiones de coherencia desde el CM. En la parte izquierda de la Figura 8.44 se muestra el diagrama temporal.

Si el estado del bloque en el directorio es M, el CM establece el estado transitorio ML. El CM emite una petición al CC que tiene el bloque en exclusividad (PtObL). Este CC, al recibir la petición, responde con mensajes, que incluyen el bloque, tanto al CC (RpDc) como al CM (RpCB) y establece como nuevo estado del bloque el estado L. El CM al recibir la respuesta (RpCB) actualiza la memoria con el bloque y modifica el VP y el BE, estableciendo en el directorio como estado del bloque el estado L. En la parte derecha de la Figura 8.44 se muestra el diagrama temporal²¹.

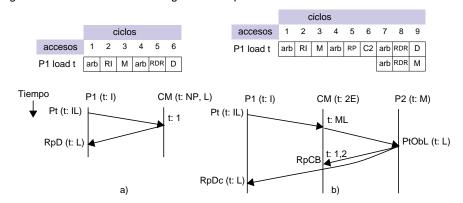


Figura 8.44 Recepción de las respuestas de los CC en el solicitante. Diagrama temporal: a) estado NP o L en el directorio y b) estado M en el directorio.

Fallo de escritura o petición de exclusividad

Memoria tiene el bloque actualizado. En la parte izquierda de la Figura 8.45 se muestra el flujo de mensajes en un fallo de escritura cuando no hay copias del bloque en otras caches. El CC solicitante emite un mensaje con una petición de bloque con intención de modificación (Ptlm). El CM, al recibir el mensaje, lee el estado del bloque en el directorio y accede especulativamente a memoria. El CM determina que el estado del bloque es NP y responde con el bloque al CC solicitante. Además, actualiza el VP y el BE.

21. Un CM o un CC generan los mensajes en paralelo. La transmisión de los mismos en paralelo depende de los conflictos en la red correspondiente.

Hay copia del bloque en otras caches. En la parte derecha de la Figura 8.45 se muestra el flujo de mensajes en un fallo de escritura cuando otras caches tiene copia del bloque. El CM, utilizando la información de estado del bloque en el directorio, determina los CC a los que es necesario enviar una petición de invalidación. El CM responde al CC solicitante con el bloque y el número de respuestas que debe recolectar (RpD, Nr = K). Por otro lado, el CM emite un mensaje de invalidación (PtObInv) a cada uno de los CC involucrados en la acción de coherencia. Posterormente, el CM modifica el VP y el BE. Las respuestas a las peticiones de invalidación las recolecta el CC solicitante.

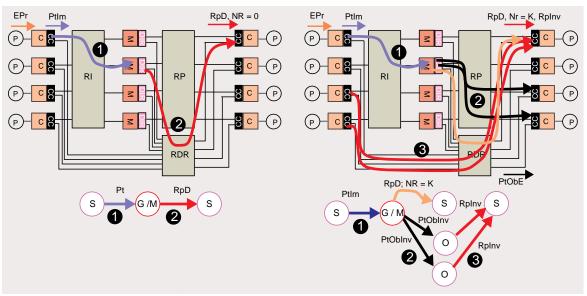


Figura 8.45 Recepción de las respuestas de los CC en el solicitante. Flujo de mensajes en un fallo de escritura.

Una cache tiene el bloque en exclusividad. En la Figura 8.46 se muestra el flujo de mensajes en un fallo de escritura cuando otra cache tiene el bloque en exclusividad. El CM emite un mensaje, de suministro del bloque e invalidación (PtObE), a la cache que tiene el bloque en exclusividad. El CC correspondiente responde al CC solicitante con el bloque, además invalida el bloque en su cache. El CM actualiza el VP para identificar la cache que ahora tiene el bloque en exclusividad.

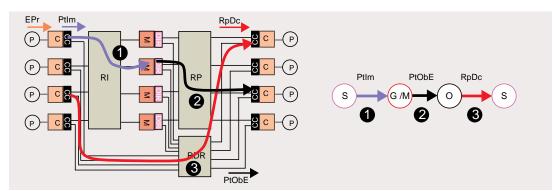


Figura 8.46 Recepción de las respuestas de los CC en el solicitante. Flujo de mensajes en un fallo de escritura cuando una cache tiene el bloque en exclusividad.

Diagramas de transiciones entre estados. En la Figura 8.47 se muestran las transiciones entre estados en la cache del CC que efectúa la petición, el directorio y los otros CC. El CC solicitante establece un estado transitorio (IM o LM), en función del estado estable inicial, esperando la respuesta o respuestas. El patrón de transición entre estados sigue la exposición efectuada en el apartado "Recepción de respuestas en el solicitante".

En el directorio el estado del bloque pasa de L a M o de NP a M, el VP se actualiza en consecuencia y el BE se activa. Además, el CM emite mensajes de respuesta (RpD, NR = K) y de petición de invalidación (PtObInv) o de petición de observación de escritura (PtObE).

Cuando un CC emite una petición PtIm, estando el bloque en estado L en la cache, el bit de presencia del CC está activado en el VP. Por tanto, hay que excluirlo de la lista de los CC que reciben una petición PtObInv (VP - P). El CC, al recibir la respuesta RpD, NR = K, siempre almacena el bloque recibido en el contenedor correspondiente, tanto si el estado estable inicial es L como I (centro de la Figura 8.48). Un caso particular es que no haya copias del bloque o el CC solicitante tenga la única copia (parte izquierda de la Figura 8.48).

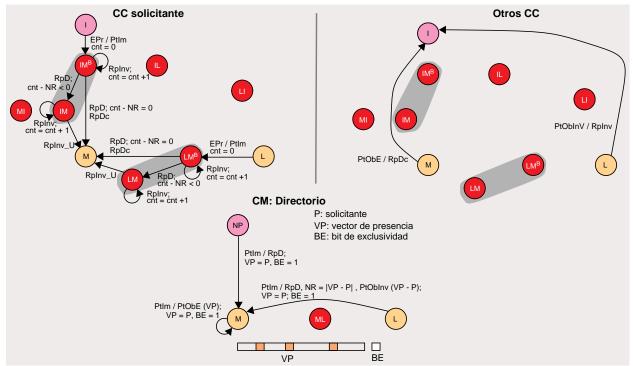


Figura 8.47 Recepción de las respuestas de los CC en el solicitante. Transiciones entre estados en un fallo de escritura o solicitud de exclusividad. Rplnv_U indica cnt = 1 y recepción de un mensaje Rplnv.

Un CC que tiene copia del bloque, al recibir la petición PtObInv invalida la copia del bloque y responden al CC solicitante (RpInv). Cuando un CC tiene el bloque en exclusividad, el CM, al procesar la petición PtIm, emite una petición PtObE. El CC que tiene el bloque en exclusividad suministra el bloque e invalida el contenedor que almacena el bloque (RpDc, parte derecha de la Figura 8.48).

Un CM genera todas las peticiones y respuestas a los CC, inducidas por una petición de un CC, en paralelo. Su transmisión en paralelo depende de los conflictos en la red correspondiente. Un destinatario de mensajes sólo puede recibir un mensaje en cada instante de tiempo.

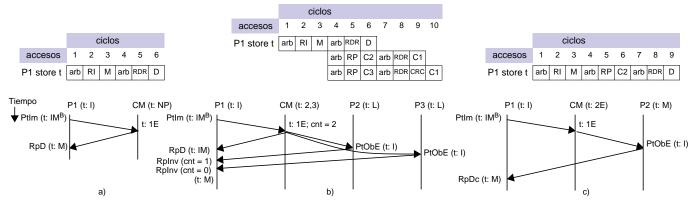


Figura 8.48 Recepción de las respuestas de los CC en el solicitante. Diagrama temporal: a) estado NP en el directorio, b) estado L en el directorio y c) estado M en el directorio. El acrómino CRC indica cola de respuestas en un CC.

Expulsión

El directorio es preciso y las peticiones de expulsión esperan una respuesta del CM. En la Figura 8.49 se muestra el flujo de mensajes cuando se expulsa un bloque de cache.

El CC efectúa una petición de expulsión (PtXm, PtXI) que va acompañada del bloque si el estado es M (PtXm). El CM en cualquier caso se actualiza la entrada correspondiente en el directorio. Además, se actualiza memoria si el bloque expulsado estaba en el estado M en la cache. El CM responde al CC confirmando que se ha realizado la acción solicitada. Esta respuesta se transmite por la red RP.

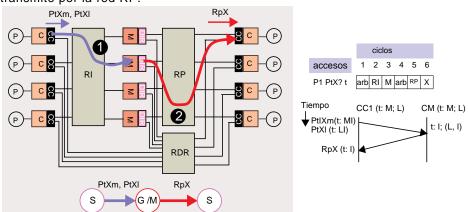


Figura 8.49 Recepción de respuestas de los CC en el solicitante. Expulsión de un bloque de cache.

En la Figura 8.50 se muestran las transiciones entre estados en: a) la cache del CC que expulsa el bloque, b) el directorio y c) otros CC. En otros CC no se producen transiciones entre estados.

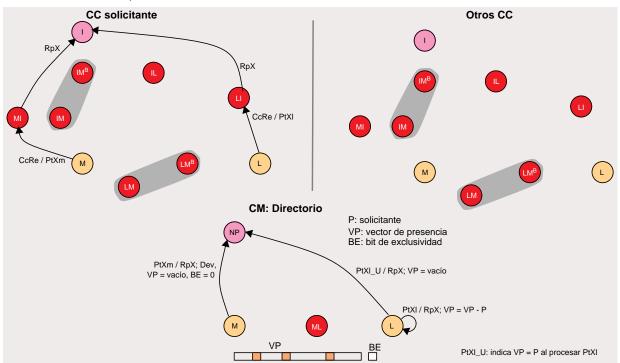


Figura 8.50 Recepción de las respuestas de los CC en el solicitante. Transiciones entre estados cuando se expulsa un bloque.

En un CC, para distinguir entre la expulsión de un bloque en el estado M o en el estado L, se utilizan distintos tipos de peticiones y también estados transitorios distintos; MI y LI respectivamente. Cuando el CM procesa la petición actualiza la memoria si la petición es PtXm y en cualquiera de los dos casos, se actualiza la entrada correspondiente del bloque en el directorio (VP). Finalmente el CM envia una respuesta de confirmación. El CC al recibir la respuesta cambia el estado del bloque al estado I.

Diagrama completo de estados y transiciones

En las Figura 8.51 y Figura 8.52 se muestran, respectivamente, los diagramas de transiciones entre estados de un bloque en cache y en el directorio. En el primer diagrama están incluidas las peticiones del procesador que no requieren iniciar transacciones explícitas de coherencia. En el CC se distinguen, en diagramas separados, las transiciones iniciadas por el agente procesador (CC solicitante) y el agente observador (otros CC).

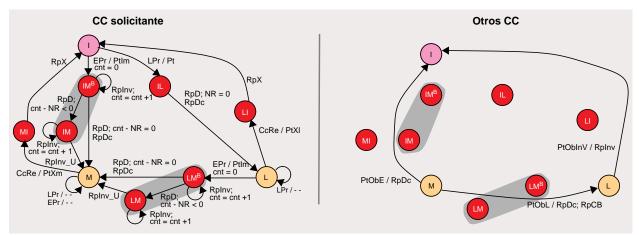


Figura 8.51 Recepción de las respuestas de los CC en el solicitante. Estados y transiciones entre estados de un bloque en una cache.

En el diagrama del CM (Figura 8.52) hay que distinguir dos casos al procesar la petición PtXI de un CC (PtXI, PtXI_U). El estado final depende de si el CC, que efectúa la petición, es el único que está en el vector de presencia.

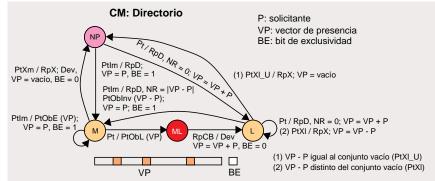


Figura 8.52 Recepción de las respuestas de los CC en el solicitante. Estados y transiciones entre estados de un bloque en el directorio

Inferencia en un CC del orden entre peticiones y respuestas emitidas desde un CM

En este protocolo, en ocasiones, el CM no conoce el instante en que una transacción ha finalizado²². Las respuestas a las peticiones del CM las recolecta el CC solicitante. Esta decisión permite reducir la ocupación del CM,

22. En concreto, en una transición desde los estados L y M al estado M.

ya que el directorio sólo se visita dos veces en un tipo de transacción. En las otras transacciones el directorio se visita una vez. Entonces, el CM puede iniciar el procesado de una petición de un segundo CC antes de que la transacción, al mismo bloque, de un primer CC haya finalizado.

Esta circunstancia puede dar lugar a que la respuesta de un CM a un CC se adelante a una petición que ha efectuado previamente el CM al mismo CC (Figura 8.25 a). Recordemos que la peticiones y respuestas del CM se transmiten por redes distintas.

Una respuesta del CM puede adelantar a una petición previa del CM al mismo bloque

En primer lugar supondremos que un CM no procesa peticiones a un bloque en un estado transitorio en el directorio (ML)²³. El CM espera la respuesta del CC involucrado en la acción de coherencia para establecer un estado estable del bloque en el directorio. Posteriormente, puede procesar otra petición que accede al bloque.

Otro CC está involucrado en la acción de coherencia. En estados estables el CM procesa peticiones de los CC. En una transición de un bloque del estado L al estado M, el CM no espera respuestas de los CC involucrados en la acción de coherencia (Figura 8.53). El encargado de recolectar las respuestas es el CC que solicita el bloque en exclusividad (CC2). Este CC2 no procesa ninguna petición del CM al mismo bloque hasta que ha recolectado todas las respuestas de los CC involucrados en la acción de coherencia. Por tanto, si el CC1 está incluido en el conjunto de CC que deben responder al CC2, el CC1 no puede recibir una respuesta a su petición, si el CM ha procesado la petición del CC1 después de la petición del CC2, el cual está esperando la respuesta del CC1 entre otras²⁴.

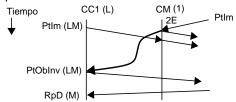


Figura 8.53 Petición de un CC que requieren la colaboración de terceros.

23. Posteriormente se relaja la hipótesis y el protocolo procesa peticiones de expulsión en el estado transitorio.

24. El CC2, que debe emitir la respuesta, está esperando la respuesta del CC1, entre otras, para pasar a un estado estable. Una vez esté en el estado estable responderá al CC1, si es el caso.

Para que se produzca un cruce, el CM debe procesar una petición que modifique el estado del bloque en la cache²⁵. Por otro lado, no debe estar involucrado otro CC en la acción de coherencia.

Para que el CM pueda responder a una petición de un CC sin ser necesaria, en ningún caso, la colaboración de otros CC, las peticiones del CC deben ser de expulsión (PtXm y PtXI).

Bloque en el estado L. El estado transitorio del bloque es LI. La petición del CC1 al CM es PXI (Figura 8.54).

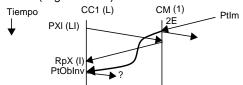


Figura 8.54 Bloque en estado L en la cache C1. Petición de otro CC.

El CM ha procesado una petición PtIm de otro CC, la cual induce que el CM emita una petición PtObInv al CC1. Al procesar el CM la petición PtXI el estado del bloque en el directorio es un estado estable (M). Como la petición PtXI no es esperada, el CM infiere un cruce de peticiones y responde. El CC1 procesa la respuesta RpX antes que la petición PtObInv.

El CC1 al procesar la respuesta RpX libera el contenedor de cache y el estado del bloque en la cache es inválido. Entonces, puede pensarse en que el CC1 responda al CC que ha inducido la petición del CM, ya que la acción de invalidar ha sido efectuada al recibir la respuesta RpX del CM.

Bloque en el estado M. El estado transitorio del bloque es MI. La petición del CC1 al CM es PtXm. El estado del bloque en el directorio es M (Figura 8.55).

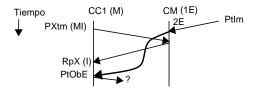


Figura 8.55 Bloque en estado M en la cache C1. Petición de otro CC.

25. Un CM no conoce cuando finaliza una transacción (dos últimos flujos de mensajes de la Figura 8.38). Recordemos que es el CC solicitante el que recolecta las respuestas o respuesta de los CC involucrados en la acción de coherencia. Por otro lado, las peticiones y respuestas del CM utilizan redes distintas. Entonces, una petición de un CM a un CC puede llegar al CC después de que el CC haya recibido una respuesta del CM y esta respuesta haya sido emitida por el CM después de la petición.

El CM ha procesado una petición PtIm de otro CC, la cual induce que el CM emita una petición PtObE al CC1. Al procesar el CM la petición PtXm el estado del bloque en el directorio es un estado estable (M). La petición PtXm no es esperada, ya que el CC1 no está identificado en el VP. Entonces, el CM infiere un cruce de peticiones y responde. El CC1 procesa la respuesta RpX antes que la petición PtObE.

El CC1 al procesar la respuesta RpX libera el contenedor de cache y el estado del bloque en la cache es inválido. Entonces, cuando procesa la petición PtObE no dispone del bloque y no puede responder al CC que ha inducido la petición del CM²⁶.

En estas condiciones, una alternativa es que la respuesta RpX se transmita por la misma red que las peticiones de los CM. Mediante esta decisión, como la RP mantiene el orden entre emisor y receptor, una petición de un CM y la respuesta a una petición de expulsión de un CC, al mismo bloque, llegan de forma ordenada al CC que efectúa la petición de expulsión (Figura 8.56).

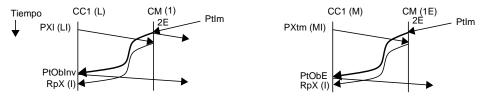


Figura 8.56 Transmisión de la respuesta RpX por la misma red que las peticiones de los CM.

Controlador de memoria

El procesado de una petición en un CM puede necesitar solicitar el bloque a una cache y esperar la respuesta (petición pendiente; ventana de vulnerabilidad). Durante este periodo de tiempo el estado del bloque es transitorio (ML, Figura 8.57).

Respecto de los protocolo de apartados previos se ha incrementado el tipo de transacciones donde el directorio sólo se visita una vez. Sólo es necesario visitar el directorio dos veces cuando una cache tiene el bloque en exclusividad y la petición es de lectura. En este caso, se ha reducido el número de pasos de la transacción (serialización).

Peticiones concurrentes a bloques distintos

Mientras un CM está esperando la respuesta de un CC, puede extraer otra petición de la CP. Si el bloque de memoria, al que accede esta segunda petición, es distinto de los bloques que están en un estado transitorio, el CM puede iniciar el procesado de la petición y por tanto existe concurrencia.

En la Figura 8.57 se muestra la concurrencia en el procesado de peticiones en el CM. Las dos primeras peticiones compiten por acceder al CM. La primera petición accede a un bloque que tiene otra cache en exclusividad. La segunda petición es para obtener la exclusividad y hay copias del bloque en otras caches. Como el CM está ocupado, la petición se espera en la CP. El CC2 recibe el bloque del CM y las respuestas de los CC participantes en la acción de coherencia. La tercera petición es para obtener la exclusividad. Como el CC2 está procesando un mensaje de respuesta de otros CC se espera en la CPC²⁷. La cuarta petición accede a un bloque que está actualizado en la memoria.

	ciclos														
accesos	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
P1 load t	arb	RI	М	arb	RP	C9	arb	RDR	D						
							arb	RDR	М						
P2 store v	arb	RI	СР	М	arb	RDR	D								
					arb	RP	C1	arb	RDR	C2					
					arb	RDR	C6	arb	RDR	CRC	C2				
P3 store u					arb	RI	СР	arb	RP	CPC	CPC	C2	arb	RDR	D
P4 load r								arb	RI	М	arb	RDR	D		

Figura 8.57 Concurrencia en un CM. Las peticiones referencian bloques distintos. Los acrónimos CP, CPC y CRC indica respectivamente cola de peticiones en el CM, cola de peticiones en un CC y cola de respuestas en un CC.

Peticiones concurrentes al mismo bloque

Ventana de vulnerabilidad. Mientras el CM está esperando una respuesta puede analizar una petición al mismo bloque (estado transitorio ML).

El directorio es el punto de ordenación de las peticiones al mismo bloque. En otras palabras es el encargado de serializar las peticiones al mismo bloque²⁸. Para ello, el CM bloquea el análisis de la CP, cuando la petición que hay en la cabeza accede a un bloque en un estado transitorio (detección). El bloqueo perdura hasta que el CM recibe la respuesta relacionada con el bloque que determina el bloqueo (gestión). Esto es, cuando el bloque está en un estado estable en el directorio.

- 27. En un CM y en un CC el procesado de las respuestas es prioritario frente a las peticiones.
- 28. En este protocolo, el CC que tiene el bloque en exclusividad también colabora.

Un CM no procesa peticiones de los CC (Pt y Ptlm) a un bloque en un estado transitorio (Figura 8.58). Ahora bien, en el estado transitorio ML se procesan peticiones de expulsión.

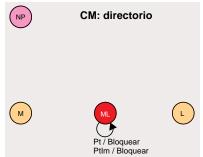


Figura 8.58 Recepción de respuestas de los CC en el solicitante. Bloqueo del procesado de peticiones en el estado transitorio del directorio.

Identificación de cruces de peticiones

En el protocolo del Capítulo 7 y en los protocolos previos de este capítulo, en un CM se bloquea el procesado de peticiones a un bloque en un estado transitorio. La petición bloqueada se procesa al finalizar la transacción pendiente en el CM. Esto es, cuando el bloque está en un estado estable.

En el protocolo descrito en este apartado se procesan peticiones de expulsión (PtXm, PtXl) cuando un bloque está, en el directorio, en el estado transitorio ML. El paso por el estado ML es debido a una petición Pt de un CC, cuando el bloque está en el estado M en el directorio. Esto es, cuando una cache tiene el bloque en exclusividad y el CC de otra cache emite una transacción de lectura.

Desde el estado transitorio ML, un CM efectúa una transición al estado estable L cuando recibe la respuesta del CC involucrado en la acción de coherencia. Entonces, es suficiente analizar los cruces de peticiones que en el Capítulo 7 se infieren cuando un bloque está en el estado L en el directorio. La inferencia del cruce de peticiones en el estado ML o en el estado L depende de la temporalidad del procesado de las peticiones en el CM. En la Figura 8.59 se indican los ordenes de procesado en un CM de las secuencias de acceso a memoria utilizadas en el Capítulo 7.

Orden de procesado en el CM					
В	F	G			
P2 store t	P2 load t	P2 store t			
P3 load t	P1 PtXm t	P3 load t			
P1 PtXI t		P1 PtXm t			

Figura 8.59 Cruces de peticiones en el CM.

Dado el orden B de accesos a memoria, se puede inferir un cruce en el estado ML o en el estado L. En este último caso, la petición del CC1 llega al CM cuando ha finalizado la transacción de CC3 (parte derecha de la Figura 8.60). En los dos estados (ML y L) el CC1 no está identificado en el VP. Dado el orden G se produce la misma inferencia. Además, el tipo de petición del CC1 no es esperada.

Dado el orden F de accesos a memoria, se puede inferir un cruce en el estado ML o en el estado L. En este último caso, la petición del CC1 llega al CM cuando ha finalizado la transacción del CC2. En los dos estados (ML y L) el CC1 está identificado en el VP, pero el tipo de petición no es esperada.

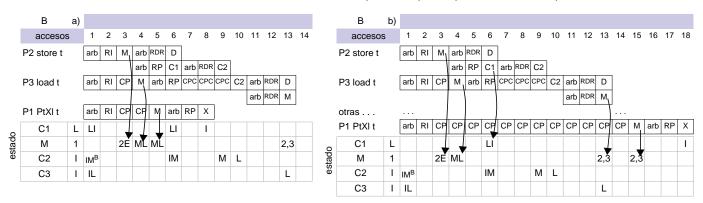


Figura 8.60 Temporalidad en un cruce de peticiones: a) estado ML, b) estado L. En el estado L el CM procesa otras peticiones o el encaminamiento por la RI requiere varios ciclos.

El resto de cruces que se infiere en el directorio son los mismos que en los protocolos previos. En Figura 8.61 se muestran todos los cruces de peticiones.

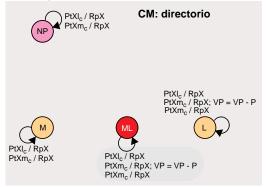


Figura 8.61 Recepción de respuestas de los CC en el solicitante. Cruces de peticiones en el directorio.

Controlador de coherencia. Gestión de cruces de peticiones

Los conjuntos de cruces de peticiones en un CC son los mismos que los identificados en el protocolo descrito previamente en este capítulo, donde se utiliza una red para los mensajes de petición de los CM y otra red para los mensajes de respuesta.

La diferencia es que en el protocolo previo de este capítulo hay un estado transitorio entre los estados I y M y entre los estados L y M. En el protocolo de este apartado hay dos estados transitorios²⁹. El cruce se puede inferir en cualquiera de ellos dos y la respuesta es la misma que en el estado equivalente del protocolo previo (Figura 8.62).

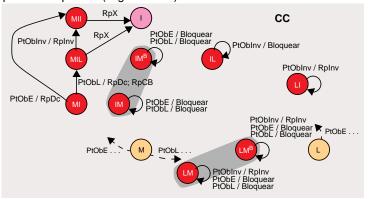


Figura 8.62 Recepción de respuestas de los CC en el solicitante. Diagramas de transiciones en un CC cuando se produce un cruce entre peticiones a un bloque. También se muestran peticiones del CM en estados estables.

^{29.} Los estados ${\rm IM}^{\rm B}$ e ${\rm IM}$ son equivalente al estado ${\rm IM}$ en el protocolo previo. De la misma forma los estados ${\rm LM}^{\rm B}$ y ${\rm LM}$ son equivalente al estado ${\rm LM}$ del protocolo previo.

RED GENERICA

Por red genérica se entiende una red que no mantiene ningún orden entre los mensajes transmitidos entre un emisor y un receptor.

En los protocolos descritos previamente en este capítulo la red mantiene el orden de los mensajes de petición emitidos desde un CM a un CC. En este apartado la red no garantiza ningún orden punto a punto. Esto es, la red no garantiza orden en la transmisión de mensajes entre un emisor y un receptor. Recordemos que un CC tampoco dispone de información temporal (orden) entre los mensajes de petición del CM y los mensajes de respuesta.

En un protocolo donde el recolector de las respuestas es el CM, siempre que el CM responde al solicitante éste tiene constancia de que todos los CC, involucrados en la acción de coherencia, la han observado. Por otro lado, durante el lapso de tiempo que requiere el procesado de una transacción, el CM no procesa otras transacciones al mismo bloque. En consecuencia, un CC no puede recibir una segunda petición del CM al mismo bloque sin haber respondido a la primera. Tampoco un CC puede recibir una respuesta a una petición, sin haber respondido antes a una petición del CM, inducida por una petición de otro CC, que el CM ha emitido antes. Un CM antes de procesar la petición de un CC espera las respuestas de la petición previa.

En un protocolo donde el recolector de las respuestas, de una acción de coherencia, es el CC solicitante, es necesario añadir mensajes en el protocolo de coherencia para establecer una ordenación. Estos mensajes aumentan el tráfico en las redes y en ocasiones introducen serialización. Esto es, una transacción tiene más pasos (Figura 8.63).

En este protocolo, donde el CC solicitante recolecta las respuestas, un CM sólo conoce cuando ha finalizado su trabajo correspondiente a una acción de coherencia. Sin embargo, un CM no conoce, en ocasiones, cuando los otros CC, involucrados en la acción de coherencia, la han observado. Es el CC solicitante el que disponde de este conocimiento. En estas condiciones, el CC solicitante puede notificar, mediante un mensaje, al CM la finalización de la transacción. En ocasiones, puede ser el único CC que participa en la acción de coherencia el que emite el mensaje al CM.

El desarrollo se efectúa utilizando un protocolo MLI donde el recolector de las respuestas es el CC solicitante.

Descripción funcional de la secuencia de mensajes en una transacción

En la Figura 8.63 se muestra la descripción funcional del protocolo previo de este capítulo y la del protocolo que se describe en este apartado.

Cuando no hay copia del bloque en otras cache o la transacción es de lectura y una cache no tiene el bloque en exclusividad, la secuencia de mensajes y su serialización es la misma en los dos protocolos (2 pasos, parte superior de la Figura 8.63).

Cuando una cache tiene el bloque en exclusividad el número de pasos es el mismo. El solicitante recibe el bloque del CC que lo tiene en exclusividad. Sin embargo, tanto si la transacción es de lectura como si es de escritura, el CC, que tenía el bloque en exclusividad, notifica al CM la emisión de la respuesta al CC solicitante. En particular, si la transacción es de lectura el CC envia una copia del bloque al CM junto con la notificación (centro de la Figura 8.63).

Cuando hay caches con copias del bloque y una transacción solicita obtener la exclusividad, las respuestas de invalidación las recolecta el CC solicitante (parte inferior de la Figura 8.63). Una vez recolectados todos los mensajes de respuesta, el CC solicitante envia una notificación al CM.

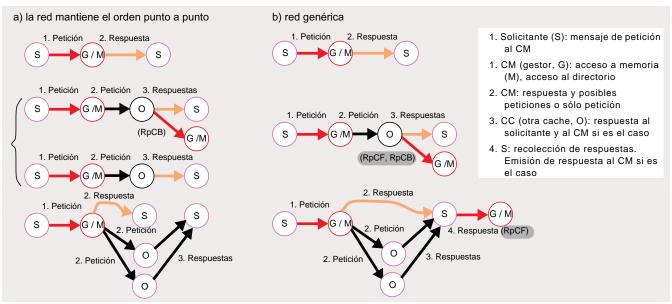


Figura 8.63 Flujos de mensajes en dos protocolo de directorio MLI. a) la red mantiene el orden punto a punto y b) red genérica. En sombreado se muestran las diferencias con el protocolo del apartado previo.

El CM da por consolidada o completada una transacción una vez emite el mensaje de respuesta o recibe la notificación del CC que tiene el bloque en exclusividad o del CC solicitante (casos segundo y tercero en la Figura 8.63)³⁰. El CM utiliza la notificación para actualizar la memoria, si es el caso.

Notemos que mediante este protocolo un CM conoce, a falta de la recepción de la respuesta por el CC solicitante, cuándo todas las caches involucradas en la acción de coherencia han procesado la petición del CM. En estas condiciones, en el multiprocesador sólo existe una transacción en curso que hace referencia a un bloque y si es el caso, la respuesta de la transacción previa al mismo bloque.

Organización del multiprocesador

La organización del multiprocesador es la misma que en el protocolo previo. La diferencia es que las redes no mantienen orden entre un emisor y un receptor de mensajes (Figura 8.64)³¹.

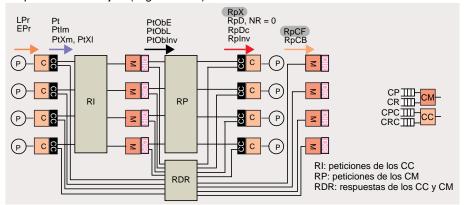


Figura 8.64 Red genérica. Organización del multiprocesador. En sombreado se muestran las diferencias básicas respecto del protocolo del apartado previo.

^{30.} Notemos que la notificación al CM, en el caso de que una cache tenga el bloque en exclusividad, también la puede efectuar el CC solicitante. Sin embargo este diseño incrementa el número de pasos de la transacción.

^{31.} Las redes pueden ser de tipo malla con encaminamiento adaptativo. Esto es, el encaminamiento entre un emisor y un receptor no está preestablecido.

Mensajes del protocolo

En la Figura 8.64 se muestra un esquema genérico de la transmisión de mensajes entre los CC y los CM y viceversa.

En la descripción del protocolo se distiguen los siguientes tipos de mensajes:

- Peticiones de un CC a los CM (Pt, PtIm, PtXm, PtXI)
- Peticiones de un CM a los CC (PtObL, PtObE, PtOInv)
- Respuestas de los CM a un CC (RpD, RpX). La respuesta RpD incluye el bloque y el número de respuestas que debe esperar (NR = K)
- Respuestas de un CC a un CC solicitante (RpDc, RpInv)
- Respuestas de un CC a un CM (RpCB, RpCF)

Los mensajes que se han añadido o modificado respecto del protocolo previo de este capítulo se muestran en la Tabla 8.5.

Controlador de coherencia	Comentario
Respuestas	
RpCF: notificación al CM de que los CC participantes en la acción de coherencia la han observado	El CC que tenía el bloque en exclusividad notifica al CM que ha emitido la respuesta al solicitante.
ld bloque	El CC solicitante notifica al CM que ha recibido todas las respuestas.

Tabla 8.5 Repuesta de un CC a un CM.

A diferencia del protocolo previo la respuesta RpX se transmite por la red RDR.

Mensajes para ordenar las transacciones

En los protocolos descritos en este capítulo el gestor de un bloque (CM) conoce, o en algunos casos es informado, de cuándo su trabajo para procesar una petición ha finalizado. Sin embargo, este hecho no significa que la transacción haya consolidado respecto a todos los nodos involucrados en la acción de coherencia³². En estos protocolos hemos analizado la serialización de acciones de coherencia en un CC. En la Figura 8.66 se muestran dos ejemplos utilizando el protocolo descrito previamente en este capítulo, donde el receptor de las respuestas de los CC, participantes en una acción de

^{32.} Recordemos que en los protocolos descritos un CM no procesa una petición (Pt, PtIm) a un bloque que está en un estado transitorio. Un CM inicia el procesado de una petición cuando el bloque está en un estado estable. El procesado de esta petición puede inducir que el CM genere una petición a un CC, cuya petición acaba de procesar el CM, el cual aún espera la respuesta o algunas respuestas a su petición.

coherencia, es el CC solicitante. Las secuencias de accesos a memoria que analizaremos se muestran en la Figura 8.65. En la parte izquierda de la Figura 8.66 se muestra la secuencia X y en la parte derecha la secuencia Y.

Orden de procesado en el CM		
X	Y	Comentario
P1 load t	P1 store t	X: no hay copias del bloque en las caches
P2 store t	P4 store t	Y: hay copia del bloque en las caches C2 y C3

Figura 8.65 Secuencias de accesos a memoria.

En los dos casos de la Figura 8.66 el CC1 recibe una petición del CM, inducida por una petición de otro CC, antes de la respuesta a su petición que el CM ha procesado previamente. El desorden es entre peticiones y respuestas del CM, debido a que se utilizan redes distintas para transmitirlas.

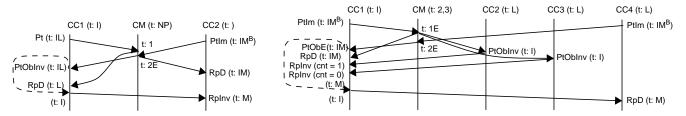


Figura 8.66 Serialización en el CC. Diagrama temporal simplificado.

En el protocolo descrito en este apartado la red de peticiones entre los CM y los CC no mantiene el orden punto a punto. Entonces, un CC puede recibir peticiones de un CM en un orden distinto al de emisión o procesado. Sin embargo, debido a la simplicidad del protocolo MLI veremos que no es posible que se produzca³³.

En el protocolo MLI un CC puede recibir la secuencia de peticiones PtObL y PtObInv a un bloque en estado M (Figura 8.67). Si el bloque está en el estado L sólo puede recibir la petición PtObInv.

En la Figura 8.67 la petición PtIm se queda encolada en la CP hasta que el CM recibe la respuesta a la petición que ha emitido previamente (estado transitorio). Notemos que hasta ese instante no dispone del bloque. Esta serialización en el CM garantiza que el CC2 reciba las peticiones PtObL y PtObInv en el

33. Si el protocolo dispone de un estado, distinto del M, desde el que suministra a otra cache es sencillo observar el desorden. Por ejemplo, un protocolo donde los bloques en cache disponen de un estado desde el que se sigue suministrando el bloque, después de que el bloque haya sido suministrado estando en el estado M. La memoria no está actualizada.

orden en el cual el CM ha procesado las peticiones que las inducen. El CM, para poder emitir la petición PtObinv al CC2, debe esperar a que éste responda a su petición anterior.

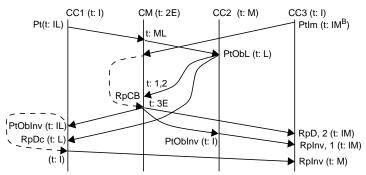


Figura 8.67 Serialización en las peticiones del CM que recibe un CC con un bloque en estado M.

Por otro lado, recordemos que la respuesta RpX del CM se transmite, en el protocolo previo del capítulo, por la red de peticiones de los CM a los CC. El objetivo es que la respuesta RpX no llegue antes al CC, que la petición que el CM ha emitido previamente al mismo CC. Como la red de peticiones no mantiene el orden punto a punto, la solución adoptada no es válida en la organización del multiprocesador de este apartado. Seguidamente analizamos el flujo de mensajes en varias situaciones.

En la parte izquierda de la Figura 8.68 se utiliza el protocolo previo de este capítulo, pero la respuesta RpX se transmite por la red de respuestas. En la figura se muestra el procesado en el CM de la petición PtXm de CC1, después de procesar la petición PtIm de CC2. La respuesta (RpX) puede adelantar a la petición (PtObE). Notemos que el CM procesa la petición de CC1 antes de que la transacción de CC2 haya finalizado³⁴. Esto es, antes de que el CC1 haya emitido la respuesta al CC2. El CC1 debe suministrar el bloque, para responder a la petición PtObE, pero no dispone del mismo. El contenedor que almacenaba el bloque ha sido liberado al procesar el CC1 la respuesta RpX.

Para que no se produzca este adelantamiento el CM debe conocer cuándo el CC1 ha procesado la petición inducida por el CC2. Para ello se añade al protocolo un mensaje de respuesta del CC1 al CM (RpCF) cuando el CC procesa una petición PtObE (centro de la Figura 8.68 y centro de la Figura 8.63). El bloque en el directorio está en un estado transitorio (MM),

^{34.} El estado del bloque en el directorio es estable. Estado M. El CM al procesar la petición PtXm detecta un cruce de peticiones, ya que el CC1 no está identificado en el VP.

mientras espera la respuesta del CC1. Una vez llega la respuesta, el CM al procesar la petición PtXm detecta un cruce de peticiones, ya que el CC1 no está identificado en el VP. El CM responde.

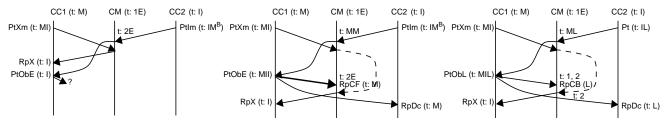


Figura 8.68 Una respuesta RpX adelanta a una petición del CM. La respuesta se corresponde con una petición PtXm. Mensaje de finalización del CC, que tiene el bloque en exclusividad, al CM para establecer orden.

Notemos que cuando la petición de CC2 es Pt no es necesario añadir este mensaje, ya que está en el protocolo de partida (parte derecha de la Figura 8.68 y centro de la Figura 8.63). El CM después de emitir la petición PtObL espera una respuesta del CC1. Durante esta espera el bloque está en el estado ML en el directorio. El CM al procesar la petición PtXm detecta un cruce de peticiones, ya que una petición PtXm no es esperada cuando el bloque está en el estado L. El CM responde y elimina de VP al CC1.

Cuando la petición es PtXI, y se utiliza el protocolo previo de este capítulo, también se puede producir el mismo comportamiento (parte izquierda de la Figura 8.69)^{35 36}. Para establecer un orden de procesado en el CC1 es necesario que el solicitante (CC2) le notifique al CM que ha recibido todas las respuestas (parte inferior de la Figura 8.63 y parte derecha de la Figura 8.69). El CM procesa la petición PtXI una vez ha recibido el mensaje de notificación.

^{35.} El CM al procesar la petición PtXI detecta un cruce de peticiones, ya que el CC1 no está identificado en el VP.

^{36.} Procesar la petición RpX en el CC1 determina invalidar el contenedor. Posteriormente, el procesado de la petición PtOblnv requiere suponer que el directorio es impreciso, lo cual no es el caso. Esto es, la expulsión de un bloque en el estado L no ha sido notificada al directorio.

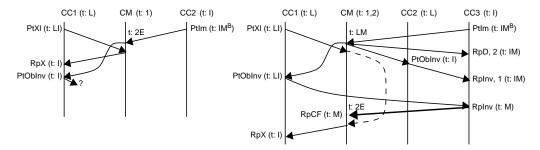


Figura 8.69 Una respuesta RpX adelanta a una petición del CM. La respuesta se corresponde con una petición PtXI. Mensaje de finalización del CC solicitante al CM para establecer orden.

Estados y transiciones

Cuando un CC tiene el bloque en exclusividad y el CM procesa una transacción de lectura hay que actualizar la memoria. En estas condiciones, el CM debe esperar a que responda el CC, con el bloque, antes de procesar otra transacción al mismo bloque. Para identificar la espera de una respuesta de un CC se utiliza un estado transitorio en el CM.

Cuando un CM procesa una petición de exclusividad hay que esperar la respuesta de finalización de la transacción. Esta respuesta la emite el CC que tiene el bloque en exclusividad o el CC solicitante. Para identificar las esperas se utilizan estados transitorios.

El directorio se utiliza para detectar accesos concurrentes al mismo bloque y gestionarlos (ventana de vulnerabilidad). Esto es, cuando la petición en la cabeza de la CP accede a un bloque en un estado transitorio, el CM efectúa la detección. La gestión utilizada en el protocolo de este apartado es bloquear el procesado de esta petición hasta que el estado del bloque sea estable.

Directorio. Los estados estables de un bloque en el directorio son los mismos que en el Capítulo 7 (Figura 8.70).

En el directorio se utilizan tres estados transitorios para: a) esperar a que el CC que tiene el bloque en exclusividad suministre el bloque al CM o notifique al CM el suministro del bloque al CC solicitante y b) esperar a que el CC solicitante notifique al CM la recepción de todas las respuestas de los CC participantes en la acción de coherencia.

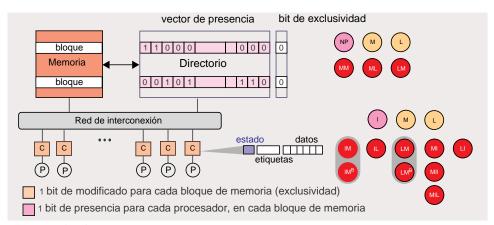


Figura 8.70 Red genérica. Estados de un bloque en el directorio y en un contenedor de cache.

Cache. Los estados estables de un bloque en una cache son los mismos que en el Capítulo 7 (Figura 8.70). Para identificar la espera de la respuesta del CM y las respuestas de los CC, a una petición del CC, se utilizan siete estados transitorios (IM^B, IM, IL, LM^B, LM, MI, LI).

Para describir el protocolo se utilizan las dos peticiones del procesador que requieren acceder al directorio (fallo en una instrucción load, store o una instrucción store que accede a un bloque sin permiso de exclusividad) y las dos posibilidades de ubicación del bloque solicitado, en memoria o en una cache. Posteriormente se detallan las transiciones entre estados en una expulsión de un bloque de cache.

En la descripción se mostrarán las transiciones entre estados de un bloque en el CC que efectúa la petición (agente procesador), en el CM y en otros CC (agente observador).

Fallo en lectura

El flujo de mensajes y transiciones entre estados en un CC y en un CM son idénticos a los descritos en el protocolo previo de este capítulo.

Fallo de escritura o petición de exclusividad

Memoria tiene el bloque actualizado. El flujo de mensajes y las transiciones entre estados son las mismas que en el protocolo previo de este capítulo.

Hay copia del bloque en otras caches. En la parte derecha de la Figura 8.71 se muestra el flujo de mensajes en un fallo de escritura cuando otras caches tiene copia del bloque. El CM, utilizando la información de estado

del bloque en el directorio, determina los CC a los que es necesario enviar una petición de invalidación. El CM responde al CC solicitante con el bloque y el número de respuestas que debe recolectar (RpD, Nr = K) y emite un mensaje de invalidación (PtObInv) a cada uno de los CC involucrados en la acción de coherencia. Posterormente modifica el VP y el BE. El CC solicitante, cuando ha recolectado todas las respuestas, emite un mensaje de notificación al CM (RpCF).

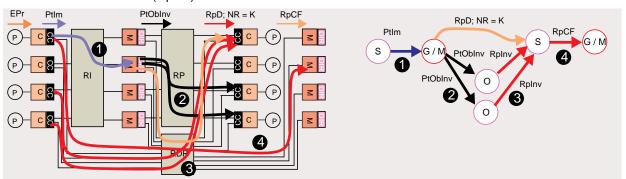


Figura 8.71 Red genérica. Flujo de mensajes en un fallo de escritura y hay copias del bloque en algunas caches.

Una cache tiene el bloque en exclusividad. En la Figura 8.72 se muestra el flujo de mensajes en un fallo de escritura cuando otra cache tiene el bloque en exclusividad. El CM, emite un mensaje, de suministro del bloque e invalidación (PtObE), a la cache que tiene el bloque en exclusividad. El CC correspondiente responde al CC solicitante con el bloque, invalida el bloque en su cache y envia un mensaje de notificación al CM (RpCF). El CM actualiza el VP para identificar la cache que ahora tiene el bloque en exclusividad.

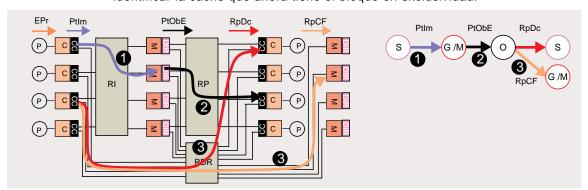


Figura 8.72 Red genérica. Flujo de mensajes en un fallo de escritura cuando una cache tiene el bloque en exclusividad.

Diagramas de transiciones entre estados. En la Figura 8.73 se muestran las transiciones entre estados en la cache del CC que efectúa la petición, el directorio y los otros CC. El CC del solicitante establece un estado transitorio (IM^B o LM^B), en función del estado estable inicial, esperando la respuesta o respuestas.

En el directorio el estado del bloque pasa de L a M, de NP a M o de M a M, el VP se actualiza en consecuencia y el BE se activa. Además, el CM emite mensajes de respuesta (RpD, NR = K) y de petición de invalidación (PtObInv) o de petición de observación de escritura (PtObE).

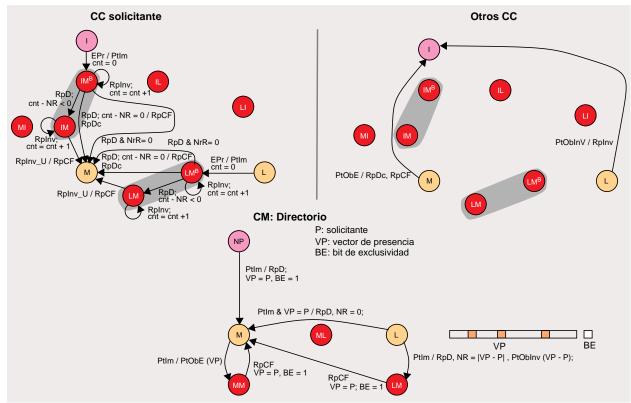


Figura 8.73 Red genérica. Transiciones entre estados en un fallo de escritura o solicitud de exclusividad. RpInv_U indica cnt = 1 y recepción de un mensaje RpInv.

Cuando un CC emite una petición PtIm, estando el bloque en estado L en la cache, el bit de presencia del CC está activado en el VP. Por tanto, hay que excluirlo de la lista de CC que reciben una petición PtObInv (VP - P). El CC, al recibir la respuesta RpD, NR = K, siempre almacena el bloque recibido en el contenedor correspondiente, tanto si el estado estable inicial es L como I

(centro de la Figura 8.74). El CC solicitante después de recibir todas las respuestas de los CC, involucrados en la acción de coherencia, emite una notificación al CM. Un caso particular es que no haya copias del bloque o el CC solicitante tenga la única copia (parte izquierda de la Figura 8.74).

Un CC que tiene copia del bloque, al recibir la petición PtOblnv invalida la copia del bloque y responden al CC solicitante (RpInv).

Cuando un CC tiene el bloque en exclusividad, el CM, al procesar la petición PtIm, emite una petición PtObE. El CC que tiene el bloque en exclusividad suministra el bloque, invalida el contenedor que almacena el bloque y emite una notificación al CM (RpDc, RpCF parte derecha de la Figura 8.74).

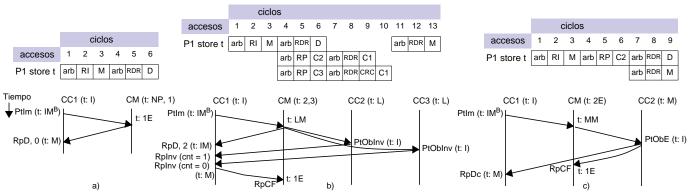


Figura 8.74 Red genérica. Diagrama temporal: a) estado NP o L en el directorio y VP = P, b) estado L en el directorio y c) estado M en el directorio. El acrómino CRC indica cola de respuestas en un CC.

Expulsión

En una expulsión de un bloque el flujo de mensajes y las transiciones entre estados son las mismas que en el protocolo previo de este capítulo.

Diagrama completo de estados y transiciones

En las Figura 8.75 y Figura 8.76 se muestran, respectivamente, los diagramas de transiciones entre estados de un bloque en cache y en el directorio. En el primer diagrama están incluidas las peticiones del procesador que no requieren iniciar transacciones explícitas de coherencia. En el CC se distinguen, en diagramas separados, las transiciones iniciadas por el agente procesador (CC solicitante) y el agente observador (otros CC).

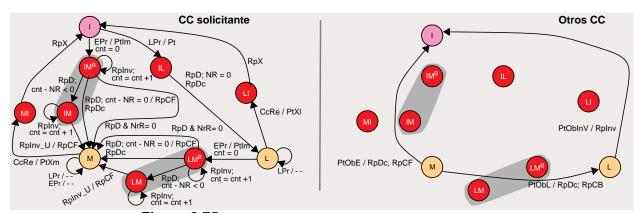


Figura 8.75 Red genérica. Estados y transiciones entre estados de un bloque en una cache.

En el diagrama del CM (Figura 8.76) hay que distinguir dos casos al procesar la petición PtXI de un CC (PtXI, PtXI_U). El estado final depende de si el CC, que efectúa la petición, es el único que está en el vector de presencia.

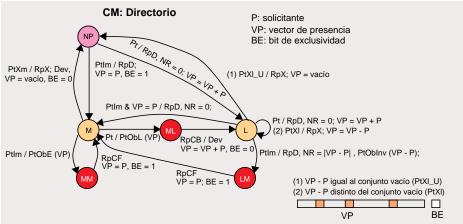


Figura 8.76 Red genérica. Estados y transiciones entre estados de un bloque en el directorio.

Gestión de cruces de peticiones

Controlador de memoria

Los estados transitorios representan una ventana de vulnerabiliad. Durante la estancia de un bloque en estos estados no se procesan peticiones. A diferencia del protocolo previo de este capítulo, en un estado transitorio no se procesan peticiones de expulsión (Figura 8.77).

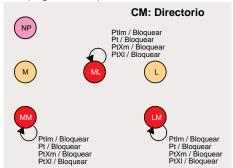


Figura 8.77 Red genérica. Bloqueo del procesado de peticiones en el estado transitorio del directorio.

Los cruces que se infieren en el directorio son los mismos que en los protocolos previos. En Figura 8.78 se muestran todos los cruces de peticiones.

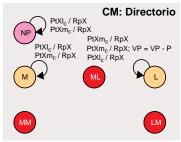


Figura 8.78 Red genérica. Cruce de peticiones en el directorio.

Controlador de coherencia

Los conjuntos de cruces de peticiones en un CC son los mismos que los identificados en el protocolo descrito previamente en este capítulo, donde la red de peticiones entre los CM y los CC mantiene el orden punto a punto (Figura 8.79).

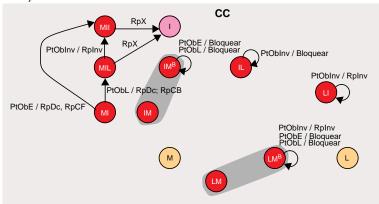


Figura 8.79 Red genérica. Diagramas de transiciones en un CC cuando se produce un cruce entre peticiones a un bloque.

Notemos que en los estados IM y LM no se puede producir un cruce de peticiones, ya que es este CC el que notifica al CM que ha finalizado la transacción (RpCF, Figura 8.63). El CM no procesa ninguna nueva transacción al bloque hasta que recibe RpCF. Por otro lado, en los estado IM^B y LM^B sólo puede haber cruces de peticiones si aún no se han recibido respuestas (RpInv). Si este es el caso, el CM está en un estado transitorio esperando la respuesta RpCF del CC.

APENDICE A: TABLAS DE TRANSICIONES ENTRE ESTADOS

Controlador de coherencia.

Se muestran en formato tabla los estados y las transiciones entre estados de un bloque en una cache. Las casillas que no contienen información indican un error. En un estado determinado no puede llegar el evento que determina la casilla correspondiente en el cruce.

Controlador de memoria.

Se muestran en formato tabla los estados y transiciones entre estados de un bloque en el directorio. En el evento PtXI se distingue el caso de que el CC sea el único que está en el vector de presencia o haya más CC.

Para identificar un cruce, se comprueba si el CC solicitante están en el vector de presencia o no. Además se comprueba si la petición recibida es esperada en el estado actual.

El CM recolecta las respuestas de invalidación

Peticiones y respuestas del CM se encaminan por la misma red

Controlador de coherencia

			Evente	os del proces reemplazo	ador y	Eventos externos (respuestas y peticiones					
			LPr	EPr	CcRe	RpD	RpX	PtObL	PtObE		
	S	I	Pt; IL	Ptlm; IM							
	Estables	L	; L	Ptlm; LM	PtXI; LI				; I		
	ш	М	; M	; M	PtXm; MI			RpDc: L	RpDc; I		
		IL				; L					
Estados		IM				; M					
Esta	ios	LM				; M			Rplnv; LM		
	transitorios	LI					; I		Rplnv; LI		
	tran	MI					;I	RpDc; MIL	RpDc; MII		
		MII					; I				
		MIL					;I		Rplnv; MII		

Figura 8.80 Tabla de estados y transiciones en un bloque de cache. Las casillas con fondo blanco indican cruces de peticiones.

Controlador de memoria (cuando {VP - P} está vacio no se emiten PtObE)

			Eventos del controlador de coherencia										
						PtXI			PtXm		RpDc	RpIn	ıv
			Pt	Ptlm	VP = P	VP≠P P∈ VP	VP≠P P∉ VP	VP = P	VP≠P P∉ VP	VP≠P P∈VP		cnt > 1	cnt = 1
		NP	RpD; L, VP = P	RpD; M VP = P, BE = 1									
	Estables	L	RpD; L, VP = VP + P	PtObE {VP - P}, RpD; M, VP = P, BE = 1	RpX; NP, VP = vacío	RpX; L VP = VP - P	RpX; L		RpX; L	RpX; L VP = VP - P			
Estados	E	M	PtObL; ML	PtObE {VP}; MM			RpX; M	RpX; NP, VP = vacío, BE = 0	RpX; M				
Ш	transitorios	ML	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	RpD, Dev; L VP = VP + P, BE = 0		
	trans	MM	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	RpD; M VP = P, BE = 1		
		LM	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo		cnt = cnt -1; LM	RpD; M VP = P, BE =1

Figura 8.81 Tabla de estados y transiciones de un bloque en el directorio. Las casillas con fondo blanco indican cruces de peticiones.

Peticiones y respuestas del CM se encaminan por redes distintas

Controlador de coherencia

	Eventos del procesador y reemplazo			Eventos externos (respuestas y peticiones							
			LPr	EPr	CcRe	RpD	RpX	PtObL	PtObE	PtOblnv	
	es	- 1	Pt; IL	Ptlm; IM							
	Estables	L	;L	Ptlm; LM	PtXI; LI				; I		
	Щ	М	; M	; M	PtXm; MI			RpDc: L	RpDc; I		
		IL				; L				Bloqueo	
sopi		IM				; M		Bloqueo	Bloqueo		
Estados	rios	LM				; M		Bloqueo	Bloqueo	RpInv; LM	
	transitorios	LI					; I			RpInv; LI	
	tran	MI					;I	RpDc; MIL	RpDc; MII		
		MII					; I				
		MIL					; I			Rplnv; MII	

Figura 8.82 Tabla de estados y transiciones en un bloque de cache. Las casillas con fondo blanco indican cruces de peticiones.

Controlador de memoria (cuando (VP - P) está vacio no se emiten PtObE)

						Eventos de	l controlad	or de cohere	ncia				
						PtXI	r controlad	01 40 0011010	PtXm		RpDc	RpIn	ıV
			Pt	Ptlm	VP = P	VP≠P P∈ VP	VP≠P P∉ VP	VP = P	VP≠P P∉ VP	VP≠P P∈VP		cnt > 1	cnt = 1
		NP	RpD; L, VP = P	RpD; M VP = P, BE = 1			RpX; NP		RpX; NP				
	Estables	L	RpD; L, VP = VP + P	PtOblnv {VP - P}, RpD; M, VP = P, BE = 1	RpX; NP, VP = vacío	RpX; L VP = VP - P	RpX; L		RpX; L	RpX; L VP = VP - P			
Estados	ES	М	PtObL; ML	PtObE {VP}; MM			RpX; M	RpX; NP, VP = vacío, BE = 0	RpX; M				
ES		ML	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	RpD, Dev; L VP = VP + P, BE = 0		
	transitorios	ММ	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	RpD; M VP = P, BE = 1		
		LM	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo	Bloqueo		cnt = cnt -1; LM	RpD; M VP = P, BE =1

Figura 8.83 Tabla de estados y transiciones de un bloque en el directorio. Las casillas con fondo blanco indican cruces de peticiones. El cruce en NP, de las peticiones PtXI y PtXm, se identifica en la columna $P \notin VP$ y $VP \neq P$

El solicitante recolecta las respuestas de invalidación

Controlador de coherencia

			Evento	s del proces reemplazo		Eventos externos (respuestas y peticiones										
			LPr	EPr	CcRe	RpD, NR	RpD, cnt - NR = 0	RpDc	RpInv cnt > 1	RpInv cnt = 1	RpX	PtObL	PtObE	PtOblnv		
	es	1	Pt; IL	PtIm; IM ^B												
	Estables	L	; L	PtIm; LM ^B	PtXI; LI								; I			
	ш	M	; M	exclu; M	PtXm; MI							RpDc: L	RpDc; I			
		IL				; L								Bloqueo		
		IM ^B				cnt - NR < 0; IM	; M	; M				Bloqueo	Bloqueo			
Estados		IM							cnt = cnt +1; IM	; M		Bloqueo	Bloqueo			
Est	rios	LM ^B				cnt - NR < 0; IM	; M	; M				Bloqueo	Bloqueo	RpInv; LM		
	transitorios	LM							cnt = cnt +1; IM	; M		Bloqueo	Bloqueo			
	tra	LI									; 1			RpInv; LI		
		MI									;1	RpDc; MIL	RpDc; MII			
		MII									;1					
		MIL									;I			RpInv; MII		

Figura 8.84 Tabla de estados y transiciones en un bloque de cache. Las casillas con fondo blanco indican cruces de peticiones.

Controlador de memoria (cuando {VP - P} está vacio no se emiten PtOblnv)

				Eventos del controlador de coherencia											
						PtXI			PtXm		RpCB				
			Pt	Ptlm	VP = P	VP≠P P∈ VP	VP≠P P∉ VP	VP = P	VP≠P P∉ VP	VP≠P P∈ VP					
		NP	RpD; L, VP = P	RpD; M VP = P, BE = 1			RpX; NP		RpX; NP						
	Estables	L	RpD; L, VP = VP + P	$\label{eq:ptobinv} \begin{aligned} \text{PtObinv} \{ \text{VP - P} \}, \text{RpD}, \text{NR}; \text{M}, \\ \text{VP = P}, \text{BE = 1} \end{aligned}$	RpX; NP, VP = vacío	RpX; L VP = VP - P	RpX; L		RpX; L	RpX; L VP = VP - P					
Estados	Ä	M	PtObL; ML	PtObE {VP}; MM			RpX; M	RpX; NP, VP = vacío, BE = 0	RpX; M						
	transitorios	ML	Bloqueo	Bloqueo			RpX; ML		RpX; ML	RpX; ML VP = VP - P	RpD, Dev; L VP = VP + P, BE = 0				

Figura 8.85 Tabla de estados y transiciones de un bloque en el directorio. Las casillas con fondo blanco indican cruces de peticiones. El cruce en NP, de las peticiones PtXI y PtXm, se identifica en la columna $P \notin VP$ y $VP \neq P$

Red genérica

Controlador de coherencia

			Evento	os del proce reemplazo		Eventos externos (respuestas y peticiones									
			LPr	EPr	CcRe	RpD, NR	RpD, cnt - NR = 0	RpDc	RpInv cnt > 1	RpInv cnt = 1	RpX	PtObL	PtObE	PtOblnv	
	es	- 1	Pt; IL	Ptlm; IM ^B											
	Estables	L	; L	PtIm; LM ^B	PtXI; LI								; I		
	Ш	М	; M	exclu; M	PtXm; MI							RpDc: L	RpDc; I		
		IL				; L								Bloqueo	
		IM ^B				cnt - NR < 0; IM	RpCF; M	RpCF; M				Bloqueo	Bloqueo		
Estados		IM							cnt = cnt +1; IM	RpCF; M		Bloqueo	Bloqueo		
Est	rios	LM ^B				cnt - NR < 0; IM	RpCF; M	RpCF; M				Bloqueo	Bloqueo	RpInv; LM	
	transitorios	LM							cnt = cnt +1; IM	RpCF; M		Bloqueo	Bloqueo		
	tra	LI									; I			RpInv; LI	
		MI									;I	RpDc; MIL	RpDc; MII		
		MII									;I				
		MIL									;I			RpInv; MII	

Figura 8.86 Tabla de estados y transiciones en un bloque de cache. Las casillas con fondo blanco indican cruces de peticiones.

Controlador de memoria (cuando {VP - P} está vacio no se emiten PtOblnv)

		(,										
					Eventos del d	controlador de	coherenci	a				
						PtXI			PtXm		RpCB	RpCF
			Pt	Ptlm	VP = P	VP≠P P∈ VP	VP≠P P∉ VP	VP = P	VP≠P P∉ VP	VP≠P P∈VP		
		NP	RpD; L, VP = P	RpD; M VP = P, BE = 1			RpX; NP		RpX; NP			
	Estables	L	RpD; L, VP = VP + P	PtObInv {VP - P}, RpD, NR; M o LM, $VP = P$, BE = 1		RpX; L VP = VP - P	RpX; L		RpX; L	RpX; L VP = VP - P		
Estados	Ë	М	PtObL; ML	PtObE {VP}; MM			RpX; M	RpX; NP, VP = vacío, BE = 0	RpX; M			
	transitorios	ML	Bloqueo	Bloqueo			Bloqueo		Bloqueo	Bloqueo	RpD, Dev; L VP = VP + P, BE = 0	
		ММ	Bloqueo	Bloqueo			Bloqueo		Bloqueo	Bloqueo		/ M VP = P, BE =1
		LM	Bloqueo	Bloqueo			Bloqueo		Bloqueo	Bloqueo		/ M VP = P, BE =1
				E: 007 -								

Figura 8.87 Tabla de estados y transiciones de un bloque en el directorio. Las casillas con fondo blanco indican cruces de peticiones. El cruce en NP, de las peticiones PtXI y PtXm, se identifica en la columna $P \notin VP$ y $VP \neq P$.

EJEMPLOS

Recepción de respuestas de los CC en el CM. Peticiones y respuestas del CM se encaminan por la misma red.

En el multiprocesador se ejecuta la siguiente secuencia de accesos a memoria

accesos	accesos
1. P1 load t	4. P1 load w
1. P2 store u	4. P3 load w
2. P1 load u	5. P2 store w
2. P3 load u	5. P1 store w
3. P1 load w	
3. P2 store u	

Pregunta 1: Represente mediante un diagrama temporal la secuencia de accesos previa. Las redes no disponen de un arbitraje centralizado. Cuando llegan varias peticiones o respuestas a un componente, el procesado de las mismas se serializa. Las peticiones que no se procesan se esperan en la cola correspondiente.

Respuesta: El primer par de accesos a memoria son debidos a un fallo de lectura y a un fallo de escritura. El estado transitorio de los bloques al emitir las peticiones es IL en la cache C1 (Pt) e IM en la cache C2 (PtIm). Las dos peticiones llegan a la par al CM. La transacción del CC1 se procesa y la transacción del CC2 se espera en la CP. En las dos transacciones el bloque lo suministra la memoria. El estados de los bloques al finalizar las transacciones es L en la cache C1 y M en la cache C2. En el directorio ha sido activado el BE del bloque que contiene la variable u.

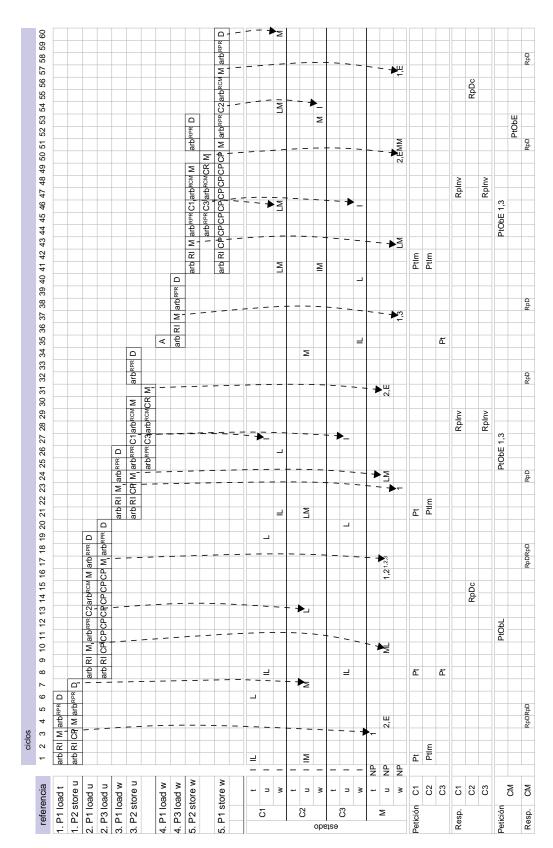
El segundo par de accesos a memoria son fallos de lectura. El CM procesa en primer lugar el mensaje emitido por CC1. El mensaje del CC3 se almacena en la CP. El procesado del mensaje del CC1 requiere que el CM solicite el bloque a la cache C2, mediante una petición de observación de lectura (PtObL). El segundo fallo hace referencia al mismo bloque. Como memoria ha sido actualizada en la transacción previa, el CM suministra directamente el bloque. Al finalizar las dos transacciones el VP del bloque identifica que las caches C1, C2 y C3 tienen copia del bloque.

El tercer grupo de accesos a memoria requiere dos transacciones. La primera petición es de lectura (Pt) y el bloque lo suministra la memoria. La segunda petición es de lectura con intención de modificación (PtIm). El CM emite

peticiones de observación de escritura del bloque (PtObE) a las caches C1 y C3, para invalidar las copias del bloque. El CM espera las respuestas del CC1 y del CC3. Las respuestas llegan a la par y la respuesta del CC3 se almacena en la CR. Una vez el CM ha recibido las respuestas, a las peticiones de invalidación, responde al CC2. El estado estable del bloque, al finalizar la transacción, es M en la cache C2 e I en las caches C1 y C3.

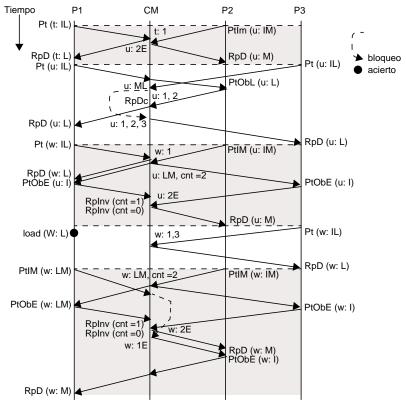
El cuarto par de accesos a memoria requiere sólo una transacción. El primer acceso a memoria efectuado por P1 es un acierto en cache. El segundo acceso a memoria, efectuado por P3, accede al mismo bloque y es un fallo. Al finalizar la transacción, el estado del bloque que contiene la variable w es L en las caches C1 y C3.

El último grupo de accesos a memoria son dos escrituras. Ninguna de las caches accedidas tiene el bloque en exclusividad. La cache C2 no tiene copia del bloque y la cache C1 tiene una copia en el estado L. La petición del CC2 requiere que el CM emita una petición de observación de escritura al CC1 y al CC3, para que invaliden el bloque. El CM espera las respuestas del CC1 y del CC3. Finalmente, el CM emite una respuesta al CC2. Durante este lapso de tiempo la petición del CC1 ha estado almacenada en la CP. Al procesar el CM la petición de CC1, emite un mensaje de petición de observación de escritura al CC2, el cual suministra el bloque al CM. Para finalizar la transacción, el CM suministra el bloque a CC1. Al finalizar las dos transacciones, el VP del bloque en el directorio indica que la cache C1 tiene copia del bloque en exclusividad.



Pregunta 2: Represente mediante un diagrama temporal simplificado la secuencia de accesos previa.

Respuesta: El bloqueo del procesado de una petición en el CM se indica mediante una línea curva a trazos.



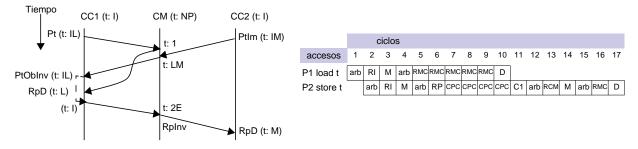
Observemos el bloqueo del procesado de peticiones en el CM en el segundo y último grupo de peticiones.

Recepción de respuestas de los CC en el CM. Peticiones y respuestas del CM se encaminan por redes distintas

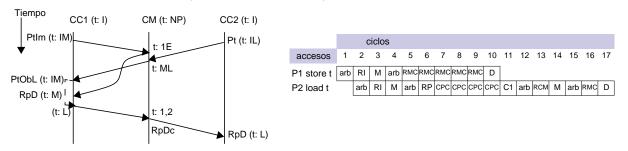
Utilice los ordenes de procesado de accesos a memoria en el CM mostrados en la Figura 8.32. Suponga que la respuesta del CM al CC1 se retrasa del orden de 5 ciclos en la RMC.

Pregunta 1: Muestre un diagrama temporal y un diagrama temporal simplificado para cada uno de los órdenes de procesado en el CM.

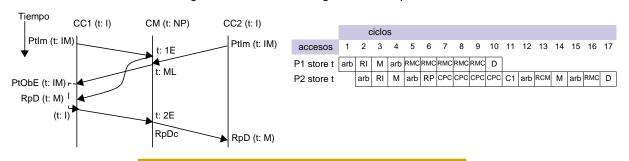
Respuesta: En la figura se muestra el diagrama correspondiente al orden I.



En la figura se muestra el diagrama correspondiente al orden J.



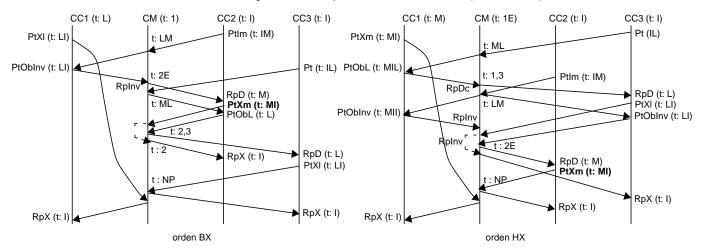
En la figura se muestra el diagrama correspondiente al orden K.



Utilice los ordenes de procesado de accesos a memoria en el CM BX (1ª columna) y HX (1ª columna) mostrados en la Figura 8.33.

Pregunta 2: Muestre un diagrama temporal simplificado para cada uno de ellos. Debe observarse el procesado de la petición de P1 estando el bloque en el directorio en el estado NP.

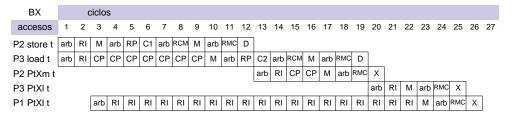
Respuesta: En la parte izquierda de la figura se muestra el diagrama correspondiente al orden BX (1ª columna). En la parte derecha de la figura se muestra el diagrama correspondiente al orden HX (1ª columna).



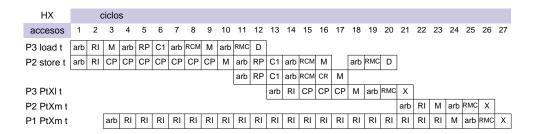
Observemos, que en un CC, una expulsión no se representa hasta que se ha recibido el bloque de la petición previa al mismo bloque.

Pregunta 3: Muestre un diagrama temporal para cada uno de los órdenes de procesado en el CM. Debe observarse el procesado de la petición de P1 estando el bloque en el directorio en el estado NP. Para representar un retardo en la transmisión de un mensaje replique en ciclos consecutivos el acrónimo de la red correspondiente.

Respuesta: En la siguiente figura se muestra el diagrama correspondiente al orden BX (1ª columna).



En la siguiente figura se muestra el diagrama correspondiente al orden HX (1ª columna).



En el multiprocesador se ejecuta la siguiente secuencia de accesos a memoria.

accesos
4. P1 load w
4. P3 load w
5. P2 store w
5. P1 store w

Pregunta 4: Represente mediante un diagrama temporal la secuencia de accesos previa. Suponga que la respuesta del CM al CC2 en el quinto grupo de accesos a memoria experimenta un retardo total de 5 ciclos en la RMC.

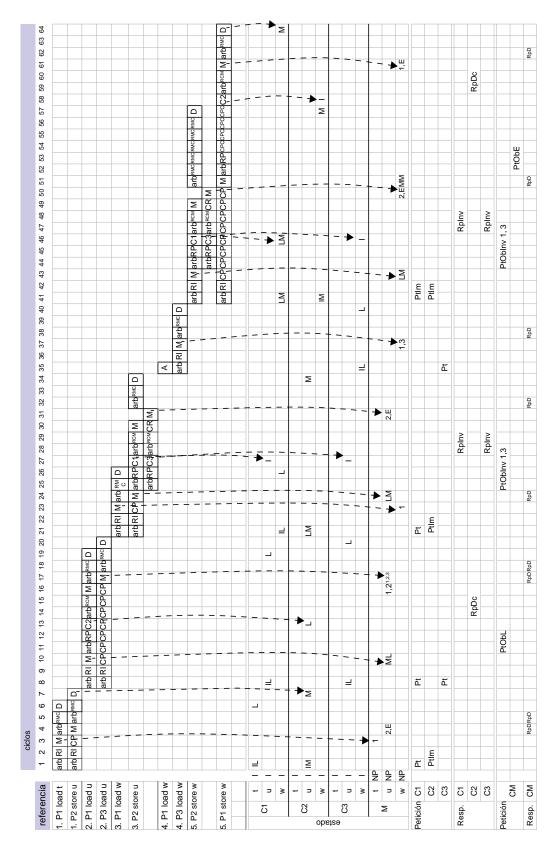
Respuesta: El primer par de accesos a memoria son debidos a un fallo de lectura y a un fallo de escritura. El estado transitorio de los bloques al emitir las peticiones es IL en la cache C1 (Pt) e IM en la cache C2 (PtIm). En las dos transacciones el bloque lo suministra memoria. El estados de los bloques al finalizar las transacciones es L en la cache C1 y M en la cache C2. En el directorio ha sido activado el BE del bloque que contiene la variable u. La petición del CC2 ha estado almacenada en la CP. El CM sólo procesa una petición en un instante determinado.

El segundo par de accesos a memoria son fallos de lectura. El CM procesa en primer lugar el mensaje emitido por el CC1. El procesado del mensaje requiere que el CM solicite el bloque a la cache C2, mediante una petición de observación de lectura (PtObL). El segundo fallo hace referencia al mismo bloque (P3). Durante el lapso de tiempo transcurrido hasta finalizar la transacción del CC1, la petición del CC3 ha estado almacenada en la CP. Como memoria ha sido actualizada en la transacción previa, el CM suministra directamente el bloque. Al finalizar las dos transacciones el VP del bloque identifica que las caches C1, C2 y C3 tienen copia del bloque.

El tercer grupo de accesos a memoria requiere dos transacciones. La primera petición es de lectura (Pt) y el bloque lo suministra la memoria. La segunda petición es de lectura con intención de modificación (PtIm). El CM emite peticiones de invalidación del bloque (PtObInv) a las caches C1 y C3. El CM, antes de responder al CC2, espera las respuestas del CC1 y CC3. El estado estable del bloque, al finalizar la transacción, es M en la cache C2 e I en las caches C1 y C3.

El cuarto par de accesos a memoria requiere sólo una transacción. El primer acceso a memoria es un acierto en cache. El segundo acceso a memoria, accede al mismo bloque, y es un fallo. Al finalizar la transacción, el estado del bloque que contiene la variable w es L en las caches C1 y C3.

El último grupo de accesos a memoria son dos escrituras. Ninguna de las caches accedidas tiene el bloque en exclusividad. La cache C2 no tiene copia del bloque y la cache C1 tiene una copia en el estado L. La petición de CC2 requiere que el CM emita una petición de invalidación al CC1 y al CC3. El CM espera las respuestas del CC1 y CC3 antes de responder al CC2. Esta respuesta al CC2 experimenta un retardo de 5 ciclos en la RMC. El CM, al procesar la petición del CC1, emite un mensaje de petición de observación de escritura al CC2. Esta petición no se procesa en el CC1, ya que no se dispone del bloque. La petición está almacenada en la CPC. Una vez el CC1 dispone del bloque respode al CM y este a su vez al CC1. Finalmente el CM suministra el bloque a CC1. Al finalizar las dos transacciones, el VP del bloque en el directorio indica que la cache C1 tiene copia del bloque en exclusividad.



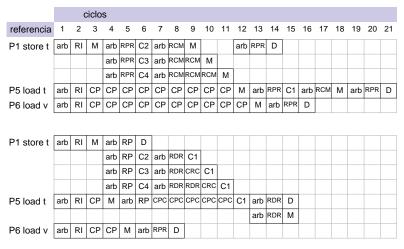
Recepción de respuestas de los CC en el CC solicitante

Utilice la siguiente secuencia de accesos concurrentes a memoria para comparar, el protocolo donde las respuestas se recolectan en el CC solicitante con el segundo protocolo descrito en este capítulo. El bloque que contiene la variable t está almacenado en las caches C2, C3 y C4.

accesos
P1 store t
P5 load t
P6 load v

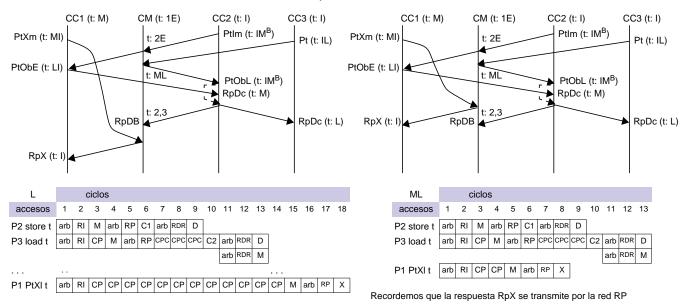
Pregunta 1: Utilice un diagrama temporal para cada protocolo con el objetivo de efectuar la comparación.

Respuesta: En la siguiente figura se muestra la comparación. Notemos la serialización que introduce el segundo protocolo de este capítulo en el procesado de peticiones en el CM (parte superior de la figura). Ello es debido a que el suministro del bloque al CC solicitante se efectúa siempre desde el CM.

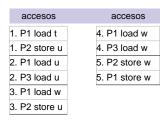


Pregunta 2: Utilice el orden de procesado G en el CM de una secuencia de accesos para mostrar, mediante sendos diagrama temporales, el cruce de peticiones en el estado ML y en el estado L (Figura 8.59).

Respuesta: En la parte izquierda de la figura se muestra la inferencia del cruce de peticiones en el estado L. En la parte derecha de la figura se muestra la inferencia del cruce de peticiones en el estado ML.



En el multiprocesador se ejecuta la siguiente secuencia de accesos a memoria.



Pregunta 3: Represente mediante un diagrama temporal la secuencia de accesos previa. Suponga que la respuesta del CM al CC2 en el quinto grupo de accesos a memoria experimenta un retardo total de 5 ciclos en la RP.

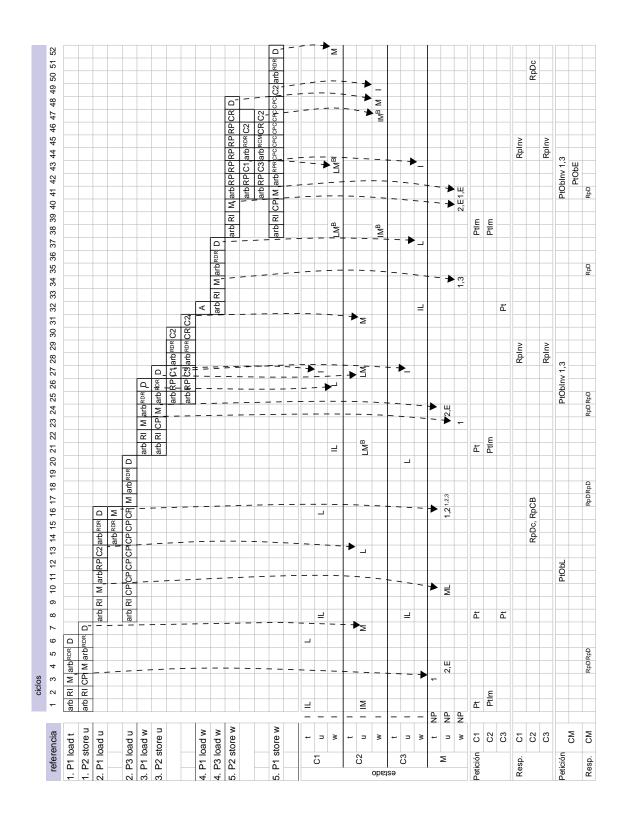
Respuesta: El primer par de accesos a memoria son debidos a un fallo de lectura y a un fallo de escritura. El estado transitorio de los bloques al emitir las peticiones es IL en la cache C1 (Pt) e IM en la cache C2 (PtIm). La petición del CC2 se almacena en la CP. El CM sólo procesa una petición en un instante dado. En las dos transacciones el bloque lo suministra memoria. El estados de los bloques al finalizar las transacciones es L en la cache C1 y M en la cache C2. En el directorio ha sido activado el BE del bloque que contiene la variable

El segundo par de accesos a memoria son fallos de lectura. El CM procesa en primer lugar el mensaje emitido por CC1. El procesado del mensaje requiere que el CM solicite el bloque a la cache C2, mediante una petición de observación de lectura (PtObL). El CC2 envia sendas respuestas al CC1 y al CM. El segundo fallo hace referencia al mismo bloque. Como memoria ha sido actualizada en la transacción previa, el CM suministra directamente el bloque. Al finalizar las dos transacciones el VP del bloque identifica que las caches C1, C2 y C3 tienen copia del bloque.

El tercer grupo de accesos a memoria requiere dos transacciones. La primera petición es de lectura (Pt) y el bloque lo suministra la memoria. La segunda petición es de lectura con intención de modificación (PtIm). El CM emite peticiones de invalidación del bloque (PtObInv) a las caches C1 y C3. Los CC correspondientes responden al CC2. El estado estable del bloque, al finalizar la transacción, es M en la cache C2 e I en las caches C1 y C3.

El cuarto par de accesos a memoria requiere sólo una transacción. El primer acceso a memoria es un acierto en cache. El segundo acceso a memoria (P3), accede al mismo bloque, pero es un fallo. Al finalizar la transacción, el estado del bloque que contiene la variable w es L en las caches C1 y C3.

El último grupo de accesos a memoria son dos escrituras. Ninguna de las caches accedidas tiene el bloque en exclusividad. La cache C2 no tiene copia del bloque y la cache C1 tiene una copia en el estado L. La petición de CC2 requiere que el CM emita una petición de invalidación al CC1 y al CC3. El CM también responde al CC2. Esta respuesta experimenta un retardo en la RP. Los CC de las caches C1 y C3 responden al CC2. La respuesta del CC3 se espera en la CR. También se espera en la CR la respuesta del CM. Recordemos que un CC sólo procesa un mensaje en un instante dado. El CM, al procesar la petición del CC1, emite un mensaje de petición de observación de escritura al CC2. El CC2 no dispone del bloque y la petición se queda encolada en la CPC. Cuando el CC2 dispone del bloque responde al CC1. Al finalizar las dos transacciones, el VP del bloque en el directorio indica que la cache C1 tiene copia del bloque en exclusividad.



EJERCICIOS

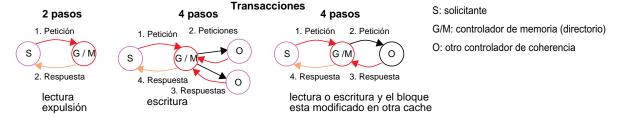
Descripción de un protocolo de directorio MLI denominado A

Suponga un multiprocesador donde las caches privadas son de mapeo directo y utilizan escritura retardada. El multiprocesador utiliza un directorio para mantener la coherencia y el protocolo de coherencia es de invalidación (MLI).

Las caches privadas de los procesadores son bloqueantes. En un fallo de cache o en una solicitud de exclusividad se suspende la interpretación de instrucciones y se reanuda al finalizar la transacción.

El directorio utiliza un vector de presencia y un bit de exclusividad por bloque. El vector de presencia (VP) es un vector de bits, con tantos bits como procesadores y cada bit está asociado a un procesador. El bit de exclusividad (BE) se utiliza para indicar que sólo existe una copia del bloque en una cache privada, la cual está identificada en el vector de presencia.

Las secuencias de mensajes de las transacciones son las siguientes:



Las peticiones de procesador y los mensajes utilizados en la transacciones para mantener la coherencia son:

Procesador	Controlador	de cache (CC)	Controlac	lor de memoria (CM)	
Peticiones	Peticiones del CC al CM	Respuestas del CM al CC	Peticiones del CM a los CC	Respuestas del CC al CM	Acciones
LPr : lectura	Pt : petición de bloque	RpD: respuesta con el bloque a una petición Pt o PtIm	PtObE: petición de observación de escritura, inducida por una petición PtIm	RpDc: respuesta con el boque a una petición PtObL o PtObE y el estado del bloque en cache es M	Actualización del directorio
EPr: escritura	Ptlm: petición de bloque con intención de modificarlo	RpX: respuesta de confirmación a una petición PtXm o PtXI	PtObL: petición de observación de lectura, inducida por una petición Pt y el estado del bloque en el directorio es M	RpInv: Respuesta a una petición PtObE y el estado del bloque en cache es L	Dev: actualización de memoria
	PtXm: petición de expulsión de un bloque en estado M				
	PtXI: petición (notificación) de expulsión de un bloque en estado L				

El controlador de cache también efectúa acciones de reemplazo cuando es necesario (CcRe). En una acción de reemplazo se distingue la acción de notificación al directorio, ya que éste es preciso y si es el caso, una actualización de memoria con el bloque expulsado, si éste ha sido modificado durante su estancia en la cache. En una petición PtXm se actualiza el directorio y memoria, mientras que en una petición PtXI sólo se actualiza el directorio.

Cuando el servicio de un acceso a memoria requiere un reemplazo, éste se efectúa antes de gestionar el acceso a memoria que produce la acción de reemplazo.

La red de comunicación entre los CM y los CC se utiliza para transmitir peticiones y respuestas. Esta red sólo mantiene orden punto a punto entre un emisor y un receptor.

Las fases de cada uno de los mensajes son:

		cic	los
mensajes	1	2	3
Pt, PtIm, PtXm, PXI	arb	RI	М
RpD, RpX	arb	RPR	Dóλ
PtObE, PtObL	arb	RPR	Сх
RpDc, RpInv	arb	RCM	М

arb: arbitraje en la red correspondiente (RI, RPR, RCM)
RI: red de peticiones desde los CC a los CM
RPR: red de respuestas desde los CM a los CC
RPR: red de peticiones desde los CM a los CC
RCM: red de respuestas desde los CC a los CM

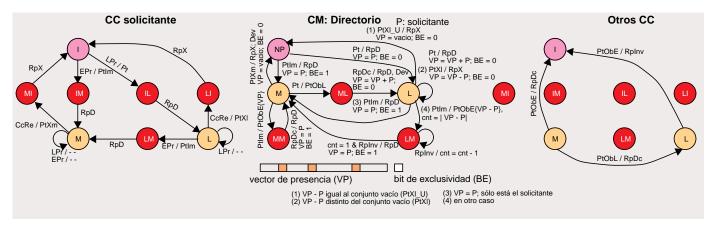
M: memoria (directorio)
D: dato (RpD)
X: confirmación (RpX)
Cx: cache, siendo x el ordinal

Suponemos que un CM genera los mensajes de petición y respuesta en paralelo. Cuando no hay conflictos en la red, los mensajes se propagan en paralelo.

Cuando varios mensajes llegan en el mismo instante a un destinatario se encolan en paralelo en la cola correspondiente. Un destinatario sólo procesa un mensaje en un instante dado.

En un CC se utilizan estados transitorios (IL, LM, IM, LI, MI) mientras se espera la respuesta de una transaccción iniciada previamente. En el CM se utilizan estados transitorios (ML, MM) mientra se esperan las respuestas de los CC involucrados en la acción de coherencia.

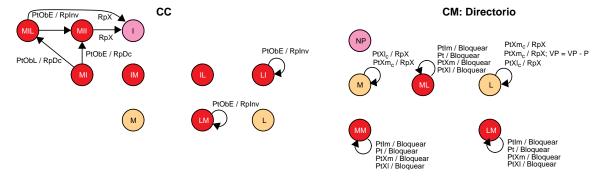
En los siguientes diagramas de estados se muestran todas las transiciones entre estados, ya sean estables o transitorios, de un bloque en cache y en el directorio. En el protocolo que se describe, el bit de exclusividad del directorio se activa cuando una cache solicita el bloque para actualizarlo.



Al tener en cuenta la concurrencia de peticiones de los CC hay que considerar los posibles cruces de peticiones en el CM y en los CC. En el CM sólo se procesan peticiones a bloques en estados estables. Una petición en la cabeza de la cola de peticiones (CP) que accede a un bloque en un estado transitorio, determina un bloqueo del análisis de esta petición y las que le siguen en la CP. La cola de respuestas en un CM se etiqueta como CR. En la CR se pueden encolar varios mensajes en el mismo instante de tiempo. Ahora bien, en el CM se procesan en serie.

En una cola de mensajes, el bloqueo del procesado por parte del autómata correspondiente, la espera por riesgo estructural, o cualquier otra causa que indique que no se procesa el mensaje se representa mediante el acrónimo de la cola.

En la siguiente figura se muestra la gestión de peticiones en los cruces. En el diagrama de estados de un bloque en el directorio, se utiliza el subíndice c para identificar una petición del CC que se ha cruzado con una petición del CM y la petición del CC se analiza en un estado estable. Notemos que en un cruce no se actualiza la memoria al procesar la petición PtXm.



Descripción de un protocolo de directorio MLI denominado B

Mensajes. Los mensajes que se han añadido o modificado respecto al protocolo A se muestran en la siguiente tabla.

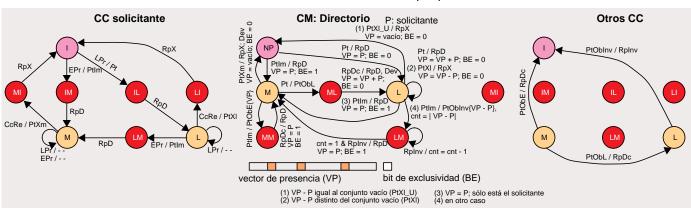
Controlador de memoria	Controlador de coherencia	Comentario
Mensajes de petición a CC y acciones	Respuestas	
PtOblnv: petición de invalidación op dirección Id	RpInV: respuesta a una petición de invalidación	El CC que recibe una petición PtOblnv en el estado L invalida el bloque y responde con RpInv.
PtObE: petición de observación de escritura op dirección Id	RpDc: respuesta con el bloque Id bloque	El CC que recibe la petición PtObE tiene el bloque en exclusividad. Emite una respuesta con el bloque y la acción de invalidación está implícita en el tipo de respuesta. Esta respuesta también se utiliza si la petición del CM es PtObL. En este caso no se invalida el bloque al responder.

Redes. Un CM utiliza redes distintas para transmitir las peticiones y respuestas a los CC. La red de peticiones de los CM a los CC mantiene el orden entre un emisor y un receptor. Las fases de cada uno de los mensajes son:

		cic	los
mensajes	1	2	3
Pt, Ptlm, PtXm, PXI	arb	RI	М
RpD, RpX	arb	RMC	DóX
PtObE, PtObL	arb	RP	Сх
RpDc, RpInv	arb	RCM	М

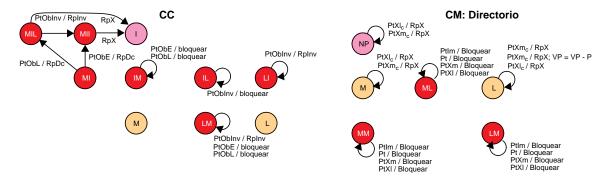
arb: arbitraje en la red correspondiente (RI, RP, RCM, RMC)	
RI: red de peticiones desde los CC a los CM	M: memoria (directorio)
RMC: red de respuestas desde los CM a los CC	D: dato (RpD)
RP: red de peticiones desde los CM a los CC	X: confirmación (RpX)
RCM: red de respuestas desde los CC a los CM	Cx: cache, siendo x el ordinal

En los siguientes diagramas de estados se muestran todas las transiciones entre estados, ya sean estables o transitorios, de un bloque en cache y en el directorio. En el protocolo que se describe, el bit de exclusividad del directorio se activa cuando una cache solicita el bloque para actualizarlo.



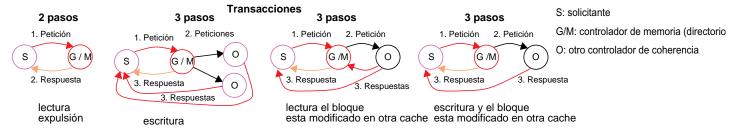
La cola de peticiones en un CC se etiqueta como CPC. En la CPC se pueden encolar varios mensajes en el mismo instante de tiempo. Ahora bien, en el CC se procesan en serie.

En la siguiente figura se muestra la gestión de peticiones en los cruces. En el diagrama de estados de un bloque en el directorio, se utiliza el subíndice c para identificar una petición del CC que se ha cruzado con una petición del CM y la petición del CC se analiza en un estado estable. Notemos que en un cruce no se actualiza la memoria al procesar la petición PtXm.



Descripción de un protocolo de directorio MLI denominado C

Las secuencias de mensajes de las transacciones son los siguientes:



Mensajes. Los mensajes que se han añadido o modificado respecto al protocolo A se muestran en las siguientes tablas.

Controlador de coherencia (CC)	Controlador de memoria (CM) Comentario
Mensajes de petición al CM	Mensaje de respuesta	
Pt: petición de bloque op dirección Id	RpD, NR: respuesta con el bloque y el número de respuestas que debe esperar Id bloque NR	e Se lee el bloque de la memoria, se actualiza el directorio, se suministra el bloque y se indica el número de respuestas que el CC solicitante debe recibir (NR).
Controlador de memoria	Controlador de coherencia	Comentario
Mensajes de petición a los CC y acciones	Respuestas	
PtObE: petición de observación de escritura op dirección Id DEST	RpDc: respuesta con el bloque e invalidación Id bloque DEST	El CC emite una respuesta al solicitante (DEST) con el bloque e invalida el bloque.
PtObL: petición de observación de lectura op dirección Id DEST	RpCB: respuesta con el bloque al CM Id bloque RpDc: respuesta con el bloque al CC Id bloque DEST	El CC emite una respuesta al solicitante (DEST) con el bloque y cambia el estado del bloque para indicar que no hay exclusividad. El CC suministra el bloque al CM.
PtOblnv: petición de invalidación op dirección Id DEST	RpInv: respuesta a una petición de invalidación Id DEST	El CC que recibe la petición invalida el bloque y responde al solicitante (DEST).

Redes. El CM utiliza redes distintas para transmitir las peticiones y respuestas a los CC. Los CC y los CM utilizan la misma red para transmitir las respuestas a los CC y CM. La red de peticiones de los CM a los CC mentiene el orden entre un emisor y un receptor. Las fases de cada uno de los mensajes son:

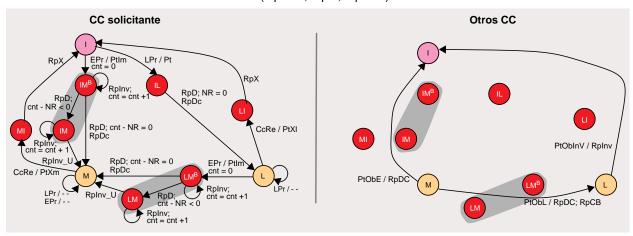
		cic	los
mensajes	1	2	3
Pt, Ptlm, PtXm, PXI	arb	RI	М
RpD	arb	RDR	DóX
PtObE, PtObL, RpX	arb	RP	Сх
RpDc, RpInv	arb	RDR	М

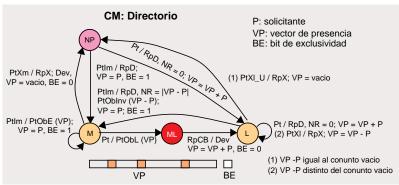
arb: arbitraje en la red correspondiente (RI, RP, RDR)
RI: red de peticiones desde los CC a los CM
RDR: red de respuestas desde los CM a los CC
RP: red de peticiones desde los CM a los CC
RDR: red de respuestas desde los CC a los CM

M: memoria (directorio)
D: dato (RpD)
X: confirmación (RpX)
Cx: cache, siendo x el ordinal

La respuesta de un CM a una petición de expulsión se transmite por la red RP.

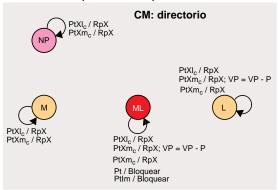
En los siguientes diagramas de estados se muestran todas las transiciones entre estados, ya sean estables o transitorios, de un bloque en cache y en el directorio. En el protocolo que se describe, el bit de exclusividad del directorio se activa cuando una cache solicita el bloque para actualizarlo. La recepción de dos tipos de respuestas, sin orden, en los CC requiere utilizar un reconocedor de secuencia: (RpInv*, RpD, RpInv*).

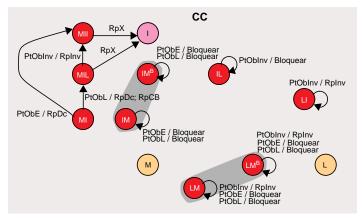




La cola de respuestas en un CC se etiqueta como CRC. En la CRC se pueden encolar varios mensajes en el mismo ciclo. Ahora bien, en el CC se procesan en serie.

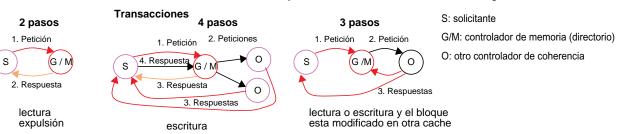
En la siguiente figura se muestra la gestión de peticiones en los cruces. En el diagrama de estados de un bloque en el directorio, se utiliza el subíndice c para identificar una petición del CC que se ha cruzado con una petición del CM y la petición del CC se analiza en un estado estable. Notemos que en un cruce no se actualiza la memoria al procesar la petición PtXm.





Descripción de un protocolo de directorio MLI denominado D

Las secuencias de mensajes de las transacciones son los siguientes:



Mensajes. Los mensajes que se han añadido o modificado respecto al protocolo A se muestran en las siguientes tablas.

Controlador de coherencia (CC) Controlador de memoria (CM)

Controlador do Controlada (CO)	oonaronaaor ao momona (om	, comontano	
Mensajes de petición al CM	Mensaje de respuesta		
Pt: petición de bloque op dirección Id	RpD, NR: respuesta con el bloque y el número de respuestas que debe esperar Id bloque NR	e Se lee el bloque de la memoria, se actualiza el directorio, se suministra el bloque y se indica el número de respuestas que el CC solicitante debe recibir (NR).	
Controlador de memoria	Controlador de coherencia	Comentario	
Mensajes de petición a los CC y acciones	Respuestas		
PtObE: petición de observación de escritura op dirección Id DEST	RpDc: respuesta con el bloque e invalidación Id bloque DEST	El CC emite una respuesta al solicitante (DEST) con el bloque e invalida el bloque.	
PtObL: petición de observación de lectura op dirección Id DEST	RpCB: respuesta con el bloque al CM Id bloque RpDc: respuesta con el bloque al CC Id bloque DEST	El CC emite una respuesta al solicitante (DEST) con el bloque y cambia el estado del bloque para indicar que no hay exclusividad. El CC suministra el bloque al CM.	
PtOblnv: petición de invalidación op dirección Id DEST	RpInv: respuesta a una petición de invalidación ld DEST	El CC que recibe la petición invalida el bloque y responde al solicitante (DEST).	
	RpCF: notificación al CM de que los CC participantes en la acción de coherencia la han observado Id bloque	El CC que tenía el bloque en exclusividad notifica al CM que ha emitido la respuesta al solicitante. El CC solicitante notifica al CM que ha recibido todas las respuestas	

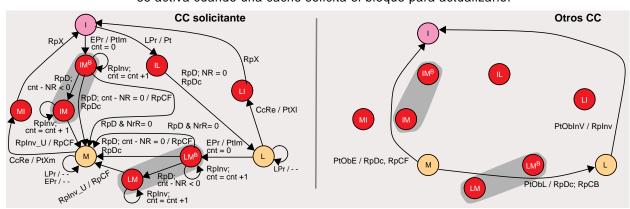
Comentario

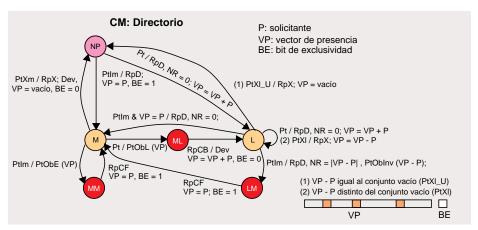
Redes. El CM utiliza redes distintas para transmitir las peticiones y respuestas a los CC. Los CC y los CM utilizan la misma red para transmitir las respuestas a los CC y CM. Ninguna de las redes mantiene el orden de los mensajes transmitidos entre dos nodos. Las fases de cada uno de los mensajes son:

		cic	los
mensajes	1	2	3
Pt, PtIm, PtXm, PXI	arb	RI	М
RpD	arb	RDR	DóX
PtObE, PtObL, RpX	arb	RP	Сх
RpDc, RpInv	arb	RDR	М

M: memoria (directorio)
D: dato (RpD)
X: confirmación (RpX)
Cx: cache, siendo x el ordinal

En los siguientes diagramas de estados se muestran todas las transiciones entre estados, ya sean estables o transitorios, de un bloque en cache y en el directorio. En el protocolo que se describe, el bit de exclusividad del directorio se activa cuando una cache solicita el bloque para actualizarlo.

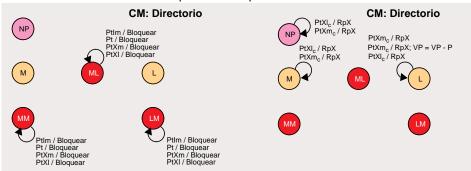


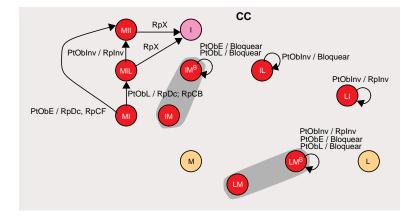


Dada una cola, se pueden encolar varios mensajes en el mismo ciclo. Ahora bien, se procesan en serie.

En una cola de mensajes, el bloqueo del procesado por parte del autómata correspondiente, la espera por riesgo estructural, o cualquier otra causa que indique que no se procesa el mensaje se representa mediante el acrónimo de la cola.

En la siguiente figura se muestra la gestión de peticiones en los cruces. En el diagrama de estados de un bloque en el directorio, se utiliza el subíndice c para identificar una petición del CC que se ha cruzado con una petición del CM y la petición del CC se analiza en un estado estable. Notemos que en un cruce no se actualiza la memoria al procesar la petición PtXm.





Ejercicio

8.1

Un multiprocesador dispone de una red entre los CM y los CC por la que se transmiten peticiones y respuestas. Esta red mantiene el orden entre un emisor y un receptor. El CM es el encargado de recolectar las respuestas de los CC participantes en una acción de coherencia. Después de ello, el CM emite la respuesta al solicitante. El procolo de coherencia utilizado es el denominado A.

En una expulsión silenciosa, el directorio no recibe notificación de la expulsión de un bloque. Esta característica determina que un controlador de coherencia pueda recibir una petición de observación de escritura (PtObE) de un bloque que no tiene almacenado en cache. Esto es, el directorio es impreciso.

Un ingeniero utiliza las siguientes secuencias de accesos a memoria concurrentes para efectuar el análisis de transiciones entre estados cuando se utiliza expulsión silenciosa. En las tres secuencias hay tres grupos de accesos. En ocasiones, en un grupo sólo hay un acceso a memoria.

	Α	В	С	
	accesos	accesos	accesos	
P	1 load t	P1 load t	P1 load t	Las variables t y u están contenidas en blo-
P	1 load u	P1 load u	P1 load u	que distintos.
P	2 store t	P2 store t	P2 store t	Estos bloques se almacenan en el mismo
		P1 load t	P1 store t	contenedor de cache.

Ahora bien, para simplificar el análisis supondremos que la cache C1 ha expulsado el bloque que contiene la variable t de forma silenciosa. En consecuencia el CC1 está identificado en el VP. Posteriormente se producen las siguientes secuencias de accesos a memoria. El CM las procesa en el orden especificado.

Α	В	С	
accesos	accesos	accesos	Estado
P2 store t	P2 store t	P2 store t	C1, C2: I
	P1 load t	P1 store t	directorio: 1 (L)

Pregunta 1: Para cada una de las secuencias de acceso muestre un diagrama temporal simplificado. En los casos en que no exista, en la descripción del protocolo, una transición para un par (petición, estado) determine las acciones que deben efectuarse.

Pregunta 2: Muestre en el diagrama de estados de un CC, correspondiente al agente procesador, la transición entre estados en una acción de reemplazo. Muestre en el diagrama de estados de un directorio preciso las transiciones que no se utilizan cuando el directorio es impreciso. Muestre también en un

CC, agente observador, los estados en los cuales se pueden recibir peticiones que son debidas a que el directorio es impreciso. Indique la transición entre estados y la respuesta.

Ejercicio 8.2

Un multiprocesador dispone de una red entre los CM y los CC para transmitir las peticiones y de otra red para transmitir las respuestas. Estas redes mantienen el orden entre un emisor y un receptor. El CM es el encargado de recolectar las respuestas de los CC participantes en la acción de coherencia. Después de ello, el CM emite la respuesta al solicitante. El protocolo de coherencia utilizado es el denominado B.

Un ingeniero decide utilizar expulsión silenciosa. Para efectuar el análisis de las modificaciones en el protocolo de coherencia, el ingeniero utiliza las siguientes secuencia de acceso a memoria concurrente.

Α	В	С	
accesos	accesos	accesos	Estado
P2 store t	P2 store t	P2 store t	C1, C2: I
	P1 load t	P1 store t	directorio: 1 (L)

La cache C1 ha expulsado el bloque que contiene la variable t de forma silenciosa antes de procesar la secuencia previa. En consecuencia el CC1 está identificado en el VP. Las secuencias de accesos se procesan en el CM en el orden especificado.

Pregunta 1: Para la secuencia de accesos C muestre un diagrama temporal simplificado. En los casos en que no exista, en la descripción del protocolo, una transición para un par (petición, estado) determine las acciones que deben efectuarse.

Pregunta 2: Para la secuencia de accesos B muestre un diagrama temporal simplificado. Justifique de forma razonada que se produce una situación de abrazo mortal.

En estas condiciones hay que diseñar un mecanismo que detecte la posibilidad de abrazo mortal y actuar para que no se produzca. Un ingeniero diseña el siguiente mecanismo, teniendo en cuenta que el CM puede detectar la situación analizando el VP.

Detección: Cuando el CM analiza la cabeza de la CP y determina que la petición es Pt y que ésta accede a un bloque en el estado transitorio LM, analiza el VP. Si el CC, que ha emitido la petición, está en el VP el CM infiere un abrazo mortal.

Actuación: El CM responde al CC (RpD), pero no elimina al CC del VP. El CC bloquea el procesado de una petición PtOblnv en el estado IL.

Dada la secuencia de accesos B, podemos decir que el CM modifica el orden de procesado de las peticiones de CC1 y CC2. Había empezado procesando en el orden Ptlm, Pt y al detectar la situación procesa Pt, Ptlm.

Pregunta 3: Para la secuencia de accesos B muestre un diagrama temporal simplificado. Utilice el mecanismo descrito para soslayar el abrazo mortal. Refine el mecanismo para que funcione en cualquier caso.

Pregunta 4: Muestre en el diagrama de estados de un CC, correspondiente al agente procesador, la transición entre estados en una acción de reemplazo. Muestre en el diagrama de estados de un directorio preciso las transiciones que no se utilizan cuando el directorio es impreciso. Muestre también en un CC, agente observador, los estados en los cuales se pueden recibir peticiones que son debidas a que el directorio es impreciso. Indique la transición entre estados y la respuesta. Muestre las transiciones adicionales en el diagrama de estados del directorio.

Otro ingeniero decide modificar de forma más radical el protocolo de coherencia del que parte, el cual es el protocolo con un directorio preciso.

Una petición PtOblnv del CM es respondida siempre por un CC cuando el bloque está en el estado IL. Por otro lado, si un CC ha procesado una petición PtOblnv, con el bloque en el estado IL, el CC descarta la respuesta del CM a su petición Pt. Posteriormente vuelve a emitir la petición Pt.

El CM no efectúa ninguna acción de detección.

Pregunta 5: Muestre un diagrama de transiciones entre estados en el observador del CC que tenga en cuenta la descripción efectuada.

Pregunta 6: Para la secuencia de acceso B muestre un diagrama temporal simplificado. Utilice el último mecanismo descrito para soslayar el abrazo mortal.

Pregunta 7: Justifique si se puede producir inanición.

Para efectuar una comparación con el protocolo con directorio preciso el ingeniero utiliza la siguiente secuencia de accesos a memoria y orden de procesado en el CM.

D	
accesos	Estado
P1 load t	C1, C2: I
P2 store t	directorio: 1 (L)

Pregunta 8: Para la secuencia de acceso D muestre un diagrama temporal simplificado. Utilice el último mecanismo descrito para soslayar el abrazo mortal. Suponga que la respuesta del CM a la petición del CC1 llega retrasada respecto a la petición del CM inducida por la petición de CC2. Efectúe el mismo análisis cuando el directorio es preciso. En este caso, en el VP no está incluido el CC1 al empezar la secuencia de accesos.

Pregunta 9: ¿Es factible que el CC, en lugar de descartar la respuesta del CM y volver a efectuar la petición, ejecute la instrucción load que ha inducido la transacción Pt y posteriormente invalide el bloque?. Justifique la respuesta. Razone el efecto del mecanismo en el rendimiento.

Otro diseñador decide modificar el protocolo de la siguiente forma. Cuando el CM procesa una petición Pt en la cabeza de la CP y el bloque está en el estado transitorio LM y además, el CC está identificado en el VP, el CM contabiliza la petición Pt como una respuesta del CC, que ha emitido la petición Pt, a la petición PtOblnv. Entonces, decrementa el contador de respuestas de invalidación.

Cuando el valor del contador es cero el CM responde al CC que ha establecido el estado transitorio LM del bloque en el directorio. Posteriormente, el CM inicia el procesado de la petición Pt y responde al CC que la ha emitido. Finalmente, antes de dar por finalizada esta última transacción, el CM espera la respuesta RpInv del CC que ha emitido la petición Pt. Esta respuesta es descarta por el CM.

Pregunta 10: Para la secuencia de acceso B muestre un diagrama temporal simplificado. Utilice el mecanismo que se acaba de describir para soslayar el abrazo mortal. Refine el mecanismo para que funcione correctamente en cualquier caso.

Pregunta 11: Muestre un diagrama de transiciones entre estados en el directorio que tenga en cuenta la descripción efectuada.

Ejercicio

8.3

Un multiprocesador dispone de una red entre los CM y los CC para transmitir las peticiones y de otra red para transmitir las respuestas desde los CC y CM a los CC y CM. Estas redes mantienen el orden entre un emisor y un receptor. El CC solicitante es el encargado de recolectar las respuestas de los CC participantes. El protocolo de coherencia utilizado es el denominado C.

Notemos que en este protocolo de coherencia el CM no conoce cuando finaliza una transación de exclusividad (transición de los estados L o M al estado M). Por otro lado, recordemos que la respuesta RpX se transmite por la red de peticiones entre los CM y los CC para eliminar un cruce de peticiones (que la respuesta RpX se adelante a una petición previa del CM).

Un ingeniero decide utilizar expulsión silenciosa. Para efectuar el análisis de las modificaciones en el protocolo de coherencia utiliza las siguientes secuencias de acceso a memoria concurrente.

Α	В	С	
accesos	accesos	accesos	Estado
P2 store t	P2 store t	P2 store t	C1, C2: I
	P1 load t	P1 store t	directorio: 1 (L)

La cache C1 ha expulsado el bloque que contiene la variable t de forma silenciosa antes de procesar la secuencia previa. En consecuencia el CC1 está identificado en el VP. Las secuencias de accesos se procesan en el CM en el orden especificado.

Pregunta 1: Para la secuencia de acceso C muestre un diagrama temporal simplificado. En los casos en que no exista, en la descripción del protocolo, una transición para un par (petición, estado) determine las acciones que deben efectuarse.

Pregunta 2: Para la secuencia de acceso B muestre un diagrama temporal simplificado. Justifique de forma razonada que se produce una situación de abrazo mortal.

En estas condiciones hay que diseñar un mecanismo que detecte la posibilidad de abrazo mortal y actuar para que no se produzca. Un ingeniero diseña el siguiente mecanismo, teniendo en cuenta que el CM puede detectar la situación analizando el VP.

Detección: En el CC que ha efectuado la petición de exclusividad (CC2) se puede detectar la situación analizando el VP. Para ello el CM transmite el contenido de VP junto con el bloque en la respuesta RpD, NR. Entonces, si el CC que emite la petición Pt está includido en el VP, el CC2, al recibir la

petición PtObL, infiere que el CC1 ha expulsado de forma silenciosa el bloque y hay una situación de abrazo mortal. Recordemos que la petición PtObL se transmite al CC destinatario de la respuesta.

Actuación: Cuando el CC analiza la cabeza de la CPC y es una petición PtObL que referencia un bloque en el estado transitorio IM, analiza el VP. Si el CC, que ha inducido esta petición del CM, está en el VP, el CC infiere un abrazo mortal y responde al CC cuando dispone del bloque.

Pregunta 3: Para la secuencia de acceso B muestre un diagrama temporal simplificado. Utilice el mecanismo descrito para soslayar el abrazo mortal. Refine el mecanismo para que funcione correctamente en cualquier caso.

Pregunta 4: Muestre en el diagrama de estados de un CC, correspondiente al agente procesador, la transición entre estados en una acción de reemplazo. Muestre en el diagrama de estados de un directorio preciso las transiciones que no se utilizan cuando el directorio es impreciso. Muestre también en el agente observador de un CC los estados en los cuales se pueden recibir peticiones que son debidas exclusivamente a que el directorio es impreciso. Indique la transición entre estados y la respuesta.

Otro ingeniero decide modificar de forma más radical el protocolo de coherencia del que parte, el cual es el protocolo con un directorio preciso.

Una petición PtOblnv del CM es respondida siempre por un CC cuando el bloque está en el estado IL. Por otro lado, si un CC ha procesado una petición PtOblnv, con el bloque en el estado IL, el CC descarta la respuesta del CM a su petición Pt. Posteriormente vuelve a emitir la petición Pt.

El CM no efectúa ninguna acción de detección.

Pregunta 5: Para la secuencia de acceso B muestre un diagrama temporal simplificado. Utilice el último mecanismo descrito para soslayar el abrazo mortal.

Pregunta 6: Justifique si se puede producir inanición.

Un ingeniero novel utiliza la siguiente secuencia de accesos a memoria y orden de procesado en el CM para intentar afinar el mecanismo descrito.

D	
accesos	Estado
P1 load t	C1, C2: I
P2 store t	directorio: 1 (L)

Pregunta 7: Para la secuencia de acceso D muestre un diagrama temporal simplificado. Utilice el último mecanismo descrito para soslayar el abrazo mortal. Suponga que la respuesta del CM a la petición del CC1 llega retrasada respecto a la petición del CM inducida por la petición de CC2. Compare este diagrama con el diagrama de la pregunta previa, donde el orden de procesado de las peticiones en el CM está intercambiado. Justifique si un CC, a partir del tipo de respuestas recibidas, puede inferir el orden en el cual el CM ha procesado su petición y mantener la coherencia.

Pregunta 8: Muestre un diagrama de transiciones entre estados en el observador del CC que tenga en cuenta la descripción efectuada.

Ejercicio

8.4

La red que se utiliza en el multiprocesador es genérica y el protocolo MLI utilizado es el denominado D. Utilice las siguiente secuencias de accesos a memoria para identificar cruces de peticiones.

Pregunta 1: Para las secuencias de accesos a memoria que se muestran en la siguiente tabla, extraidas de la Figura 7.30, muestre diagramas temporales simplificados. Explique la forma de inferir el cruce y las acciones que se toman para mantener la coherencia.

Orden en el CM		
Α	В	
P2 store t	P2 store t	
P1 PtXI t	P3 load t	
	P1 PtXI t	

Pregunta 2: Para las secuencias de accesos a memoria que se muestran en la siguiente tabla, extraidas de la Figura 7.32, muestre diagramas temporales simplificados. Explique la forma de inferir el cruce y las acciones que se toman para mantener la coherencia.

Orden de procesado en el CM					
E	F	G	Н		
P2 store t	P2 load t	P2 store t	P2 load t		
P1 PtXm t	P1 PtXm t	P3 load t	P3 store t		
		P1 PtXm t	P1 PtXm t		

Pregunta 3: Para las secuencias de accesos a memoria que se muestran en la siguiente tabla, extraidas de la Figura 8.32, muestre diagramas temporales simplificados. Explique la forma de inferir el cruce y las acciones que se toman para mantener la coherencia.

Orden de procesado en el CM				
I	J	K		
P1 load t	P1 store t	P1 store t		
P2 store t	P2 load t	P2 store t		

Pregunta 4: Para las secuencias de accesos a memoria que se muestran en la siguiente tabla, extraidas de la Figura 8.33, muestre diagramas temporales simplificados. Explique la forma de inferir el cruce y las acciones que se toman para mantener la coherencia.

Orden de procesado en el CM				
AX	EX			
P2 store t	P2 store t			
P2 PtXm t	P2 PtXm t			
P1 PtXI t	P1 PtXm t			

Ejercicio

8.5

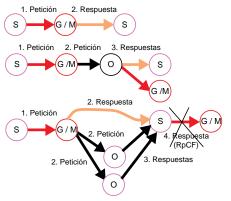
En el protocolo MLI denominado D, el cual se utiliza en una red genérica, se dispone de mensajes de notificación (RpCF) para ordenar las peticiones y respuestas del CM a un CC.

En un directorio preciso, un CC que recibe una petición PtOblnv del CM para un bloque no recibirá posteriormente ninguna petición más, a menos que solicite otra vez el bloque.

Por otro lado, un CC recibe una respuesta RpX si ha emitido una petición PtXI de un bloque en el estado L. También recibe una respuesta si ha emitido una petición PtIm, que referencia a un bloque en el estado L o I. El protocolo D garantiza que estas respuestas, si se produce un cruce tardío, se reciben después del procesado en el CC de una petición PtObInv. El CM no procesa la petición PtXI o PtIm hasta que recibe la respuesta RpCF del CC cuya petición ha inducido la emisión de la petición PtObInv.

Una vez el CM procesa una petición PtXI o PtIm (no se produce un cruce tardío), el CC que las ha emitido no recibe una petición PtObInv. En el directorio el CC no está identificado en el VP (PtXI) asociado al bloque, o el estado del bloque en el directorio es M.

La petición PtXI y respuesta asociada son para que el directorio sea preciso. Por otro lado, que el CM espere la respuesta RpCF de un CC, el cual la emite después de recolectar las respuestas RpInv, representa una serialización en el procesado de transacciones en el CM, aunque garantiza que la respuesta RpX no llega al CC, que ha emitido PtXI, antes que una petición PtObInv.



Esta serialización también elimina cruces de peticiones tempranos en los estados IM y LM.

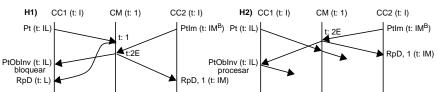
En estas condiciones, la utilización de un directorio impreciso permite eliminar la serialización descrita en el CM y reduce el tráfico de peticiones de los CC (PtXI) y respuestas de los CM (RpX). Por otro lado, puede incrementar el tráfico de peticiones del CM (PtOblnv). El coste es gestionar cruces adicionales en un CC.

A este protocolo MLI lo denominamos DL.

Pregunta 1: Para el protocolo DL, muestre los diagramas de transiciones entre estados de un bloque en el CC y el CM.

Suponga los siguientes dos accesos a memoria (parte izquierda de la siguiente figura). Utilizando el protocolo DL, en la parte de la derecha se muestran diagrama temporales donde se producen cruces de mensajes (peticiones, respuestas).





Del análisis de los diagramas observamos que en el estado IL es necesario efectuar dos respuestas incompatibles (un oráculo nos indica el orden de procesado en el CM): a) bloquear (el CM ordena 1º Pt), b) procesar (el CM ordena 1º Ptlm). Una forma de solventarlo es responder y utilizar un estado transitorio (ILI). Este estado se utiliza para recordar que entre la petición Pt y la recepción de la respuesta (RpD, RpDc) ha sido procesada una petición PtOblnv. Cuando éste sea el caso, después de recibir el bloque se ejecuta la instrucción load y se descarta (invalida) el bloque.

Pregunta 2: Utilice el protocolo DL. Muestre un diagrama de transiciones entre estados de un bloque en el CC donde se tenga en cuenta la descripción que acaba de efectuarse.

Para efectuar un análisis de las implicaciones de utilizar expulsión silenciosa se utilizan las siguientes secuencias de accesos a memoria.

Orden	Orden de procesado en el CM			Estado del bloque en las
Α		В	С	caches y directorio
P2 store	e t	P2 store t	P2 store t	C1, C2: I
		P1 load t	P1 store t	directorio: 1 (L)

Pregunta 3: Utilice el protocolo DL. Para las secuencias de accesos a memoria relacionadas previamente muestre diagramas temporales simplificados. Analice los cruces en el CC1 y en el CC2. Explique la forma de inferir el cruce y las acciones que se toman para mantener la coherencia.

Suponga que en el multiprocesador se ejecutan los siguientes dos hilos.

H1	H2	H1	H2	Estado del bloque en las caches y directorio
A = 3.21	while (A == 0) { }	P1 store A	P2 load aviso	A = B = 0; C1, C2: A, B; estado I
aviso = 1	T = A	P1 store aviso	P2 load A	directorio: A: NP, aviso: 2 (L)

Nos centraremos en las instrucciones que acceden a la variable aviso y tendremos en cuenta dos entrelazados.

Entrelazado 1	Entrelazado 2	Comentarios
P2 load aviso	P1 store aviso	En el "entrelazado 1" P2 load aviso se ordena antes que P1 store aviso.
P1 store aviso	P2 load aviso	En el "entrelazado 2" P2 load aviso se ordena después que P1 store aviso
P2 load aviso	P2 load aviso	

Pregunta 4: Muestre que es necesario serializar en un CC los accesos a una posición de memoria para un funcionamiento correcto (protocolo DL). Esto es hay que inferir el orden que ha determinado el CM. Para ello utilice los entrelazados previos.

Pregunta 5: Utilice el protocolo DL. Proponga secuencias de dos accesos a memoria donde se observen los cruces de peticiones en los estado LM^B y LM. Muestre diagramas temporales simplificados. Explique la forma de inferir el cruce y las acciones que se toman para mantener la coherencia.

Pregunta 6: Utilice el protocolo DL. Para las secuencias de accesos a memoria relacionadas seguidamente muestre diagramas temporales simplificados, donde se observen cruces tempranos.

	procesado el CM	Estado del bloque en las caches y directorio
F	G	
P2 store t	P2 store t	C1, C2, C3: I; directorio: 1 (L)
P3 load t	P2 PtXm t	
P1 store t	P1 store t	

Pregunta 7: Utilice el protocolo DL. Para las secuencias de accesos a memoria relacionadas seguidamente muestre diagramas temporales simplificados, donde se observen cruces tempranos.

	Orden de procesado en el CM		Estado del bloque en las caches y directorio
Н	I		Dos casos para H e I
P2 store t	P2 store t		1) C1, C2: I; C3: L; directorio: 3 (L)
P1 load t	P1 store t		2) C1: I; C2, C3: L; directorio: 2 ,3 (L)

Pregunta 8: Utilice el protocolo DL. Para la secuencia de accesos a memoria mostrada seguidamente muestre diagramas temporales simplificados, donde se observen cruces tardíos.

Orden de el CM	Estado del bloque en las caches y directorio
II	
P1 load t	C1, C2: I; C3: M; directorio: 3E (M)
P2 store t	

El cruce de una petición PtObE o PtObL con una petición PtXm en el CM es una oportunidad para reducir el trafico en la red. Por otro lado, se reduce la serialización en el CM.

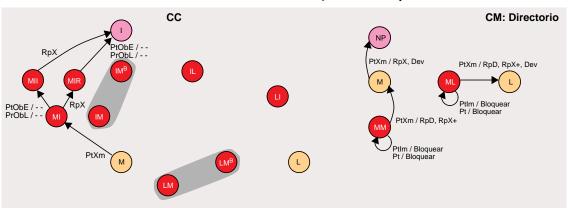
El mecanismo que diseñan los ingenieros es el siguiente:

- Un CM al procesar una petición PtXm en el estado M emite una respuesta al CC (RpX) y actualiza memoria.
- Un CC cuando emite una petición PtXm establece como estado del bloque MI. Un CC no responde a peticiones PtObE o PtObL cuando está en el estado MI. Además, para recordarlo, se utilizan estados transitorios.

- Un CM al procesar una petición PtXm (se transmite por la red de respuestas de los CC a los CM) utiliza esta petición como respuesta, si el bloque está en un estado transitorio en el directorio. También emite una respuesta al CC que ha emitido la petición PtXm, indicando que la ha utilizado como respuesta (RpX+).
- Cuando un CM utiliza una petición PtXm como respuesta emite una respuesta (RpD) al CC que ha establecido el estado transitorio del bloque en el directorio. Además, si es el caso, actualiza memoria.
- Un CC que ha emitido una petición PtXm, cuando recibe una respuesta RpX+ pasa el bloque al estado I, una vez ha recibido una petición PtObE o PtObL (antes o después).

A la modificación del protocolo DL que se acaba de describir la denominamos DLM.

Para el protocolo DLM los ingenieros diseñan los siguientes diagramas de transiciones entre estados de un bloque en el CC y el CM. .



Para comprobar los diagramas de transiciones entre estados utilizan las siguientes secuencias de accesos a memoria.

Orden de procesado en el CM				Estado del bloque en las
AA	BB	CC		caches y directorio
P2 store t	P2 store t	P2 load t		C1: M; C2, C3: I
P2 PtXm t	P3 load t	P3 load t		directorio: 1E (M)
P2 PtXm t		P2 PtXm t		

Pregunta 9: Muestre, utilizando las secuencias previas, que los diagramas de transiciones entre estado previos no son correctos. En concreto, en el CM no se serializan adecuadamente los accesos a la misma posición de memoria. Para ello tenga en cuenta que se pueden producir cruces entre mensajes.

Para efectuar un análisis de la reducción de tráfico en las redes y de la reducción de serialización en el CM, aportada por el protocolo DLM, utilizaremos las siguientes secuencias de accesos a memoria.

	Orden de	Estado del bloque en las			
J	K L M			N	caches y directorio
P2 store t	P2 store t	P2 load t	P2 load t	P2 load t	C1: M; C2, C3: I
P1 PtXm t	P3 store t	P1 PtXm t	P3 store t	P3 load t	
P3 store t	P1 PtXm t	P3 store t	P1 PtXm t	P1 PtXm t	directorio: 1E (M)

Pregunta 10: Suponga que sólo se producen cruces en el CC1 y en el CM. Los mensajes del CM al CC1 llegan en el orden de emisión desde el CM. Muestre sendos diagrama temporales para el protocolo D y para el protocolo DLM.

Pregunta 11: Un CM utiliza el mensaje PtXm como una respuesta, cuando el estado del bloque en el directorio es transitorio. Entonces, un ingeniero se plantea que el CM no responda a mensajes PtXm. Justifique que no es una opción.

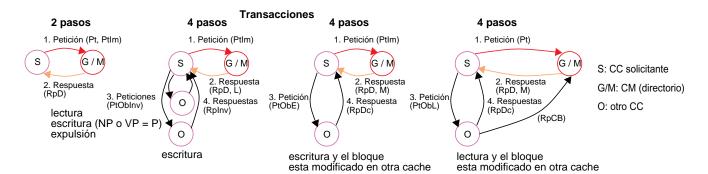
Pregunta 12: Justifique la necesidad de que el CM, al responder al CC que ha emitido un mensaje PtXm, indique si ha utilizado el mensaje como respuesta. Para ello analice cruces de mensajes al procesarse la siguiente secuencia de accesos a memoria.

Ord	en de proc	esado en el	Estado del bloque en las	
0	O P Q		R	caches y directorio
P2 load t	P2 load t P2 store t P2 load t		P2 store t	C1: M; C2: I
P1 PtXm t	P1 PtXm t	P1 PtXm t	P1 PtXm t	directorio: 1E (M)
P1 load t	P1 load t P1 store t		P1 store t	

Ejercicio 8.6

El protocolo de directorio que se describe utiliza una comunicación peticiónrespuesta estricta. Esto es, una petición al procesarse en un directorio genera una respuesta.

En estas condiciones, cuando es necesaria la participación de terceros en una acción de coherencia, el CM responde a un CC con la identidad de los participantes en la acción de coherencia. Entonces, el CC solicitante se encarga de efectuar las peticiones oportunas a los CC participantes y recolectar sus respuestas. En la siguiente figura se muestra el flujo de mensajes en las distintas situaciones.



El directorio es preciso. Cuando es necesaria la participación de terceros, en una acción de coherencia, son necesarios 4 pasos.

Mensajes. Los mensajes que se han añadido o modificado respecto al protocolo D se muestran en las siguientes tablas.

Controlador de coherencia (CC)	Controlador de m	emoria (CM)	Comentario	
Mensajes de petición al CM	Mensaje de re	spuesta		
Ptlm: petición de bloque op dirección Id	RpD, L: respuesta co la lista de nodos que del bloque en el esta Id bloque	tienen copia	Se lee el bloque de la memoria, se actualiza el directorio, se suministra el bloque y se indican los nodos que tienen copia del bloque (L).	
	identificador del nod	RpD, M: respuesta con el identificador del nodo que tiene el bloque en el estado M Id bloque IdM		rectorio, se o que tiene el do M.
	Controlador de coherencia		nentario	
F	Respuestas			
RpCB: respu	esta con el bloque al CM bloque	El CC sumini: CM.	stra el bloque al	

Se utiliza una red para transmitir peticiones (RP) y otra red para transmitir respuestas (RR) Las redes de interconexión no mantienen ningún orden entre los mensajes emitidos desde un nodo emisor a un nodo receptor.

Pregunta 1: Partiendo de la descripción previa, diseñe los diagramas de transiciones entre estados de un bloque en un CC y en un CM. No tenga en cuenta posibles cruces de peticiones. No tenga en cuenta expulsiones de bloques.

Suponga la siguiente secuencia de accesos a memoria.

Accesos	Estado del bloque en las
Α	caches y directorio
P1 store t	C1, C2, C3: L;
P2 store t	directorio: 1, 2, 3 (L)
P3 load t	
P1 load t	

Pregunta 2: Para la secuencia previa de accesos a memoria muestre un diagrama temporal simplificado. Un acceso a memoria no se empieza ha procesar hasta que el acceso previo ha finalizado completamente.

Una transacción de coherencia se implementa mediante una secuencia de mensajes entre nodos. Entonces, mientras se está efectuando una transacción otro nodo puede iniciar otra transacción al mismo bloque. Por tanto, es necesario una serialización de las transacciones para mantener la coherencia.

Para constatar el problema un ingeniero utiliza las siguientes secuencias de accesos a memoria. En una petición PtXm o PtXI, el CC efectúa una transición a un estado transitorio (MI, LI) esperando la respuesta (RpX). Desde el estado MI se suministra el bloque si se recibe una petición. Desde el estado LI se responde RpInv.

Orden de procesado en el CM			CM		Comentarios
В	С	D E			
P2 load t	P2 store t	P3 store t	P3 store t		B: la respuesta a PtXm llega antes que la petición
P1 PtXm t	P1 PtXI t	P2 store t	P2 load t		C: la respuesta a PtXI llega antes que la petición
					D: C1, C2, C3 : L. Directorio: 1, 2, 3 (L)
					E: C1, C3 : L. C2: I. Directorio: 1, 3 (L)

Otras secuencias de acceso a memoria son:

Orden de procesado en el CM		Comentarios	
D1	D2		
P3 store t	P3 store t	DD: C1, C2, C3 : L. C4: I. Directorio: 1, 2	2, 3 (L)
P2 store t	P2 store t	E: C1, C3 : L. C4: I. Directorio: 1, 3 (L)	
P4 store t P4 load t			

Pregunta 3: Muestre diagramas temporales simplificados para las secuencias previas de accesos a memoria. Constate la necesidad de serialización, teniendo en cuenta desorden en los mensajes. Analice una serialización en el directorio.

Una alternativa para solventar el problema descrito, es que un CC no pueda recibir peticiones a un bloque, si el CM ha procesado su petición. La forma de gestionarlo es que el CM reciba una notificación de que la transacción previa al bloque ha finalizado. Esto es, cuando un CC ha recibido respuesta a todas las peticiones que ha emitido a otros CC, emite un mensaje de notificación al CM (RpCF). También, si se permite cierto solapamiento, el CC que tiene el bloque en estado M puede enviar la notificación, después de servir la petición.

Pregunta 4: Partiendo de la descripción previa, añada los mensajes necesarios al protocolo y diseñe los diagramas de flujo de mensajes. En el caso de existir alternativas tengalas en cuenta. No tenga en cuenta expulsiones de bloques.

Pregunta 5: Compare, desde el punto de vista de serialización en el procesado de transacciones al mismo bloque, el protocolo de este ejercicio con el protocolo D.

Pregunta 6: Partiendo de la descripción previa, añada los mensajes necesarios al protocolo y diseñe los diagramas de transiciones entre estados de un bloque en un CC y en un CM. En el caso de existir alternativas proponga diagramas para cada una de ellas. No tenga en cuenta expulsiones de bloques.

Pregunta 7: Para la secuencia de accesos a memoria previa a la segunda pregunta muestre un diagrama temporal simplificado. Todos los accesos son concurrentes excepto el último que debe ser causal. El orden en el cual se procesan en el CM es el orden de la secuencia de accesos.

Pregunta 8: Diseñe las transiciones entre estados en un CC y en un CM para tener en cuenta la expulsión de bloques. El directorio es preciso.