Caché (informática)

De Wikipedia, la enciclopedia libre

En informática, la **caché** es la memoria de acceso rápido de una computadora, que guarda temporalmente los datos recientemente procesados (información).[1]

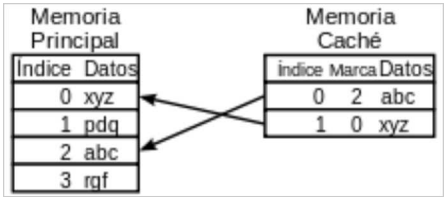

Diagrama de una memoria caché de CPU.

La memoria caché es un búfer especial de memoria que poseen las computadoras, que funciona de manera similar a la memoria principal, pero es de menor tamaño y de acceso más rápido. Es usada por el microprocesador para reducir el tiempo de acceso a datos ubicados en la memoria principal que se utilizan con más frecuencia.

La caché es una memoria que se sitúa entre la unidad central de procesamiento (*CPU*) y la memoria de acceso aleatorio (*RAM*) para acelerar el intercambio de datos.

Cuando se accede por primera vez a un dato, se hace una copia en la caché; los accesos siguientes se realizan a dicha copia, haciendo que sea menor el tiempo de acceso medio al dato. Cuando el microprocesador necesita leer o escribir en una ubicación en memoria principal, primero verifica si una copia de los datos está en la caché; si es así, el microprocesador de inmediato lee o escribe en la memoria caché, que es mucho más rápido que de la lectura o la escritura a la memoria principal.[2]

Índice

Etimología

La palabra procede de la voz inglesa *cache* (/kæʃ/; «escondite secreto para guardar mercancías, habitualmente de contrabando») y esta a su vez de la francesa *cache*, (/kaʃ/; «escondrijo o escondite»). A menudo, en español se escribe con tilde sobre la «e» del mismo modo como el que se venía escribiendo con anterioridad al neologismo la palabra «caché» («distinción o elegancia» o «cotización de un artista»), proveniente de un étimo también francés, pero totalmente distinto: *cachet*, (/ka'ʃɛ/; «sello» o «salario»).

La Real Academia Española, en el Diccionario de la lengua española sólo reconoce la palabra con tilde,[1] aunque en la literatura especializada en Arquitectura de computadoras (por ejemplo, las traducciones de libros de los autores Andrew S. Tanenbaum, John L. Hennessy y David A. Patterson) se emplea siempre la palabra sin tilde por ser anglosajona y debería escribirse en cursiva (*cache*).

RAM caché y caché de disco

La unidad caché es un sistema especial de almacenamiento de alta velocidad. Puede ser tanto un área reservada de la memoria principal como un dispositivo de almacenamiento de alta velocidad independiente.

Hay dos tipos de caché frecuentemente usados en computadoras personales: memoria caché y caché de disco.

Una **memoria caché**, a veces llamada "RAM caché", es una parte de RAM estática (*SRAM*) de alta velocidad, más rápida que la RAM dinámica (*DRAM*) usada como memoria principal. La memoria caché es efectiva dado que los programas acceden una y otra vez a los mismos datos o instrucciones. Guardando esta información en SRAM, la computadora evita acceder a la lenta DRAM.

Cuando se encuentra un dato en la caché, se dice que se ha producido un acierto, siendo un caché juzgado por su tasa de aciertos (*hit rate*). Los sistemas de memoria caché usan una tecnología conocida por caché inteligente en la cual el sistema puede reconocer cierto tipo de datos usados frecuentemente. Las estrategias para determinar qué información debe ser puesta en la caché constituyen uno de los problemas más interesantes en la ciencia de las computadoras. Algunas memorias caché están construidas en la arquitectura de los microprocesadores. Por ejemplo, el microprocesador Pentium II: tiene 32 KiB de caché de primer nivel (*level 1* o *L1*) repartida en 16 KiB para datos y 16 KiB para instrucciones; la caché de segundo nivel (*level 2* o *L2*) es de 512 KiB y trabaja a mitad de la frecuencia del microprocesador. La caché L1 está en el núcleo del microprocesador, y la L2 está en una tarjeta de circuito impreso junto a éste.

La **caché de disco** trabaja sobre los mismos principios que la memoria caché, pero en lugar de usar SRAM de alta velocidad, usa la convencional memoria principal. Los datos más recientes del disco duro a los que se ha accedido (así como los sectores adyacentes) se almacenan en un búfer de memoria. Cuando el programa necesita acceder a datos del disco, lo primero que comprueba es la caché de disco para ver si los datos ya están ahí. La caché de disco puede mejorar notablemente el rendimiento de las aplicaciones, dado que acceder a un byte de datos en RAM puede ser miles de veces más rápido que acceder a un byte del disco duro.

Composición interna

Los datos en la memoria caché se alojan en distintos niveles según la frecuencia de uso que tengan, estos niveles son los siguientes:

Memoria caché nivel 1 (*Caché L1*)

También llamada memoria interna, se encuentra en el núcleo del microprocesador. Es utilizada para acceder a datos importantes y de uso frecuente, es el nivel en el que el tiempo de respuesta es menor. Su capacidad es de hasta 128 kb. Este nivel se divide en dos:

- **Nivel 1 *Data Cache***: Se encarga de almacenar datos usados frecuentemente y cuando sea necesario volver a utilizarlos, accede a ellos en muy poco tiempo, por lo que se agilizan los procesos.
- **Nivel 1 *Instruction Cache***: Se encarga de almacenar instrucciones usadas frecuentemente y cuando sea necesario volver a utilizarlas, inmediatamente las recupera, por lo que se agilizan los procesos.

Memoria caché nivel 2 (*Caché L2*)

Se encarga de almacenar datos de uso frecuente. Es más lenta que la caché L1, pero más rápida que la memoria principal (RAM). Se encuentra en el procesador, mas no es su núcleo. Genera una copia del nivel 1. Su capacidad es de hasta 1 Mb.

- **Caché Exclusivo**: Los datos solicitados se eliminan de la memoria caché L2.
- **Caché Inclusivo**: Los datos solicitados se quedan en la memoria caché L2.

Memoria caché nivel 3 (*Caché L3*)

Esta memoria se encuentra en algunos procesadores modernos y genera una copia a la L2. Es más rápida que la memoria principal (RAM), pero más lenta que L2. En esta memoria se agiliza el acceso a datos e instrucciones que no fueron localizadas en L1 o L2.

Es generalmente de un tamaño mayor y ayuda a que el sistema guarde gran cantidad de información agilizando las tareas del procesador.

Diseño

En el diseño de la memoria caché se deben considerar varios factores que influyen directamente en el rendimiento de la memoria y por lo tanto en su objetivo de aumentar la velocidad de respuesta de la jerarquía de memoria. Estos factores son las **políticas** de ubicación, extracción, reemplazo y escritura.

Política de ubicación

Decide dónde debe colocarse un bloque de memoria principal que entra en la memoria caché. Las más utilizadas son:

- **Directa**: al bloque i-ésimo de memoria principal le corresponde la posición *i módulo n*, donde n es el número de bloques de la memoria caché. Cada bloque de la memoria principal tiene su posición en la caché y siempre en el mismo sitio. Su **inconveniente** es que cada bloque tiene asignada una posición fija en la memoria caché y ante continuas referencias a palabras de dos bloques con la misma localización en caché, hay continuos fallos habiendo sitio libre en la caché.
- **Asociativa**: Los bloques de la memoria principal se alojan en cualquier bloque de la memoria caché, comprobando solamente la etiqueta de todos y cada uno de los bloques para verificar acierto. Su principal **inconveniente** es la cantidad de comparaciones que realiza.
- **Asociativa por conjuntos**: Cada bloque de la memoria principal tiene asignado un conjunto de la caché, pero se puede ubicar en cualquiera de los bloques que pertenecen a dicho conjunto. Ello permite mayor flexibilidad que la correspondencia directa y menor cantidad de comparaciones que la totalmente asociativa.

Política de extracción

La política de extracción determina cuándo y qué bloque de memoria principal hay que traer a memoria caché. Existen dos políticas muy extendidas:

- **Por demanda**: un bloque sólo se trae a memoria caché cuando ha sido referenciado y no se encuentre en memoria caché.
- **Con prebúsqueda**: cuando se referencia el bloque i-ésimo de memoria principal, se trae además el bloque (i+1)-ésimo. Esta política se basa en la propiedad de localidad espacial de los programas.

Política de reemplazo

Determina qué bloque de memoria caché debe abandonarla cuando no existe espacio disponible para un bloque entrante. Básicamente hay cuatro políticas:

- **Aleatoria**: el bloque es reemplazado de forma aleatoria.
- ***FIFO***: se usa el algoritmo *First In First Out* (*FIFO*) (primero en entrar primero en salir) para determinar qué bloque debe abandonar la caché. Este algoritmo generalmente es poco eficiente.
- **Menos recientemente usado** (***LRU***): Sustituye el bloque que hace más tiempo que no se ha usado en la caché, traeremos a caché el bloque en cuestión y lo modificaremos ahí.
- **Menos frecuencias usadas** (***LFU***): Sustituye el bloque que ha experimentado menos referencias.

Véase también: Algoritmos de reemplazo de páginas

Política de Actualización o Escritura

Determinan el instante en que se actualiza la información en memoria principal cuando se hace una escritura en la memoria caché.

- **Escritura Inmediata:** Se escribe a la vez en Memoria caché y Memoria principal. *Desventaja:* genera cuello de botella.
- **Escritura Aplazada:** Actualiza únicamente la Memoria caché luego de la modificación de sus datos. Cuando el bus de sistema se encuentra libre, actualiza la memoria principal. Esto puede generar que los periféricos lean datos erróneos, pero es poco frecuente.
- **Escritura Obligada:** Actualiza únicamente la Memoria caché luego de la modificación de sus datos. Cuando no hay otra alternativa, actualiza la memoria principal. Esto puede producirse por cualquiera de estas causas:

1. Se accede a la posición de memoria principal modificada en la caché. Antes de permitir la lectura/escritura, debe actualizarse el dato en la memoria principal.
2. Debe eliminarse una línea de la caché, entonces se actualiza la memoria principal (en caso de ser necesario) antes de proceder a la eliminación.

Optimización

Para una optimización en la manera en que se ingresa a la memoria caché y cómo se obtienen datos de ella, se han tomado en cuenta distintas técnicas que ayudarán a que haya menos reincidencia de fallos.

Mejorar el rendimiento.

- **Reducir fallos en la caché (miss rate).**
- **Reducir penalizaciones por fallo (miss penalti).**
- **Reducir el tiempo de acceso en caso de acierto (hit time).**

Reducción de fallos

Tipos de fallos

Existen 3 tipos de fallos en una memoria caché:

- **Forzosos (Compulsory):** En el primer acceso a un bloque éste no se encuentra en la caché (fallos de arranque en frío o de primera referencia).
- **Capacidad (Capacity):** La caché no puede contener todos los bloques necesarios durante la ejecución de un programa.
- **Conflicto (Conflict):** Diferentes bloques deben ir necesariamente al mismo conjunto o línea cuando la estrategia es asociativa por conjuntos o de correspondencia directa (fallos de colisión).

Técnicas para reducir fallos

Existen diversas técnicas para reducir esos fallos en la caché, algunas son:

- **Incrementar el tamaño del bloque.** *Ventajas:* Se reducen los fallos forzosos como sugiere el principio de localidad espacial. *Inconvenientes:* Aumentan los fallos por conflicto al reducirse el número de bloques de la caché y los fallos de capacidad si la caché es pequeña. La penalización por fallo aumenta al incrementarse el tiempo de transferencia del bloque.

- **Incremento de la asociatividad.** *Ventajas:* Se reducen los fallos por conflicto. *Inconveniente:* Aumenta el tiempo de acceso medio al incrementarse el tiempo de acierto (multiplexión). También aumenta el coste debidos a los comparadores

- **Caché víctima.** Consiste en añadir una pequeña caché totalmente asociativa (1-5 bloques) para almacenar bloques descartados por fallos de capacidad o conflicto. En caso de fallo, antes de acceder a la memoria principal se accede a esta caché. Si el bloque buscado se encuentra en ella se intercambian los bloques de ambas cachés.

- **Optimización del compilador.** El compilador re-ordena el código de manera que por la forma en cómo se hacen los accesos se reducen los fallos de caché.

Véase también

- Unidad central de procesamiento
- Arquitectura de von Neumann
- Caché web
- Caché de disco

Referencias

1. «caché (http://lema.rae.es/drae/srv/search?key=cach%C3%A9)», *Diccionario de la lengua española* (22.ª edición), Real Academia Española, 2001, http://lema.rae.es/drae/srv/search?key=cach%C3%A9, consultado el 31 de julio de 2014.
2. Behrouz A. Forouzan, Sophia Chung Fegan (2003). *Foundations of Computer Science: From Data Manipulation to Theory of Computation*. Cengage Learning Editores. ISBN 9789706862853.

Enlaces externos

- Artículo sobre la caché (http://www.zator.com/Hardware/H5_2.htm)

Obtenido de «https://es.wikipedia.org/w/index.php?title=Caché_(informática)&oldid=84737484»

Categoría: Caché
