第 35 卷 第 1 期 计 算 机 工 程 2009 年 1 月
Vol.35 No.1 Computer Engineering January 2009
·软件技术与数据库· 文章编号：1000—3428(2009)01—0087—03 文献标识码：A 中图分类号：N945

改进的基于目录的 Cache 一致性协议

王 铮，张 毅
(重庆大学计算机学院，重庆 400044)

摘 要：介绍几种典型目录一致性协议并分析它们的优缺点。在综合全映射目录和有限目录优点的基础上，通过在存储器层上增加一个存储器高速缓存(Cache)层的方式，提出并讨论一种改进后的 Cache 一致性协议。该协议相对有限目录存储开销增加不多的情况下，提高了系统性能和可扩展性。

关键词：高速缓存；一致性；可扩展性；存储器层次；多处理机系统

Improved Directory-based Cache-Coherent Protocol

WANG Zheng, ZHANG Yi
(College of Computer Science, Chongqing University, Chongqing 400044)

【Abstract】 This paper introduces some typical directory organization schemes and analyzes their advantages and disadvantages. By full using advantages of full-map directory and limited directory, an improved directory-based Cache-coherent protocol is presented and discussed here, which adds a Cache level upon the memory level. It limites the space cost increase, while the system performance and system scalability have a remarkable enhancement.

【Key words】 Cache; coherence; scalability; memory hierarchy; multiprocessor systems

分布式共享存储(DSM)多处理机中的 Cache 一致性问题是困扰计算机设计者的一个重要问题。因为它不只显著影响多处理机系统的性能，而且直接决定着系统的正确性。而经典的 Cache 一致性协议要么依赖于总线监听(监听协议，snoopy protoco1)，要么占用过多的存储空间或者过于复杂的算法(目录法，directory scheme)。本文在综合了全映射(full-map)目录和有限(limited)目录优点的基础上，提出了一种主要应用在分布式共享存储(DSM)多处理机上的使用两级目录的 Cache 一致性协议。

1 经典Cache一致性协议

目前常用的 Cache 一致性协议根据其系统跟踪共享数据状态的不同，主要分为 2 类协议：监听一致性协议(snoopy coherency protocol)和基于目录的一致性协议(directory-based protocols)。

1.1 监听一致性协议

监听一致性协议是需要由总线或环提供的广播机制。其主要思想是不断地监听总线上处理器和存储器模块间的高速缓存操作事件。其下又根据对失效数据的处理不同分为写-更新协议(write-update)和写-无效协议(write-invalidate)。具体而言，写更新协议的实现方法是当某个处理机在更新本地高速缓存的同时，将更新后的数据块发送给其他所有相关的高速缓存，并用新的数据覆盖原来的数据。而写-无效协议仅使所有其他高速缓存中的相应数据拷贝失效。显然，写-更新协议迫使在所有时刻保持高速缓存的数据拷贝皆有效，但这需要耗费大量的总线周期来更新所有的高速缓存和主存中的高速缓存行，所以代价很大。写-更新协议和写-无效协议都必须基于对总线广播的监听。因为总线扩展能力有限，所以当多处理机系统规模较大时，总线很可能成为系统瓶颈。同时监听一致性协议在不支持总线监听的多处理机互联网络拓扑，如网格型和超立方体型等用于多计算机消息传递的网络中无法使用。因而监听一致性协议主要使用于小规模多处理机系统。由于本协议基于目录一致性协议，因此对监听一致性协议不作过多讨论。

1.2 基于目录的一致性协议

基于目录一致性协议的基本思想就是用目录的形式记录所有高速缓存行和共享数据的位置和状态。因而当处理器对某一缓存行进行操作时，便可根据相应的目录项得知该如何进行一致性操作。这种高速缓存位置被称为高速缓存目录(cache directory)。Tang提出了高速缓存一致性控制的第一种集中式目录方案[1]。其主要思想是使用一个集中式目录来记录所有的高速缓存条件，包括当前信息和所有高速缓存行的状态。集中式目录只适用于小规模的多处理机系统中，如清华大学研发的MP860 层次式并行超级计算机[2]。此后Censier和Feautrier提出分布式目录方案[3]。每个存储器模块维护各自单独的目录，目录中记录着各个存储块的高速缓存行的状态和当前信息。状态信息是本地的，但当前信息指明哪些高速缓存才有该存储器块中某高速缓存行的拷贝。分布式目录非常适用于分布式存储器层次结构的多处理机系统，如斯坦福的Dash多处理机[4]。

根据高速缓存目录结构的不同，可分为 3 种不同类型的目录协议：全映射(full-map)目录，有限(limited)目录和链式(chained)目录。现分别对其作具体分析。

1.2.1 全映射目录

全映射目录的基本思想是该目录包含全局范围内共享的所有高速缓存行的信息。即存储模块的每一个高速缓存行对

作者简介：王 铮(1953－)，男，副教授、博士，主研方向：分布式操作系统；张 毅，硕士

收稿日期：2008-10-29 **E-mail：**zagyi81@yahoo.com.cn

应一个目录项，每个目录项包含 N 个指针，N 是指处理器的个数，这些指针通过位向量标识。位向量的每一位与一个处理器相对应，用于指出该处理器局部 Cache 中有无该高速缓存行的拷贝。

假定全映射目录系统有P个处理器节点和N个存储器节点，每个存储器节点拥有M bit，每个高速缓存行含有B bit，即每个存储器节点包含$C=M\div B$个高速缓存行。其全映射目录结构如图 1 所示。假定处理器数P与存储节点数N同数量级，即$O(P)=O(N)$。则整个目录表占用的存储空间是$P\times (M\div B)\times N= O(P)^2$ 位(忽略状态标志位)，可见它与处理器个数呈平方倍增长。带入假定值，P=64, N=64, M=2^{28}, B=2^9，则全映射目录表的存储开销(MemoryOverhead)为 12.5%。因此，全映射目录对于含处理器数量太多的多机系统不适用。但是如果处理器个数适中，则它是一个十分有效的协议。例如，斯坦福的DASH多机系统由 64 个处理器组成，4 个为 1 组，共 16 组，组内的Cache一致性采用侦听策略，组间Cache一致性采用全映射目录机制。整个目录表占有存储器资源的 13.3%。

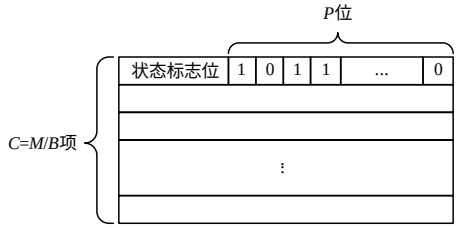

图 1　全映射目录

从表面上看来，为了减少目录表的存储开销，可以增加高速缓存行的大小，但是这样做同时也会使假共享现象增加，从而大量增加维护 Cache 一致性操作和通信量，可能严重降低系统性能。

1.2.2　有限目录

有限目录的基本思想是每个目录项只使用一定数量的指针，而不管系统的大小如何。具体而言，每个目录项使用 Q 个指针指示 Q 个拥有该高速缓存行拷贝的处理器节点，当大于 Q 个处理器节点共享此拷贝(目录项溢出)时，则要么向所有处理器节点广播作废命令(广播式有限指针 DIRiB)；要么按特定算法选取一指针作废其指向的相应拷贝，腾出一个指针空间存放新申请的指针(非广播式有限指针 DIRiNB)；要么使用可以一个指针对应多个处理器的复合指针(超集法 DiriX)。

现假定系统条件不变，只是每个目录项使用Q个指针，则每个指针需要 lb$P$ 位。其目录结构如图 2 所示。整个目录表占用的存储空间是 $(Q\times \mathrm{lb}p)\times\frac{M}{B}\times N = O(P\mathrm{lb}P)$ 位。占用空间随着处理器个数呈 Plb$P$ 倍增长，比全映射目录的缩放性好。带入假定值，P=64, N=64, M=2^{28}, B=2^9, Q=8，则有限目录表的存储开销为 9.4%。

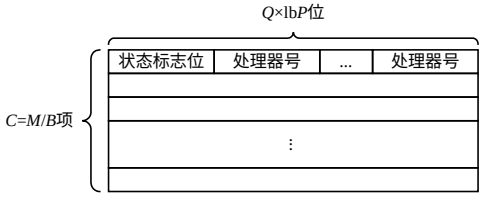

图 2　有限目录

虽然，有限目录存在目录项溢出的问题。实际上，在共享读操作所占比率不大于 70%时，在 2 次写操作之间共享某一给定存储块的处理器节点数一般不超过 5 个[5]。因而当存储器共享比较少时，有限目录是比较经济高效的。

1.2.3　链式目录

链式目录基本思想是通过将目录信息分布到多个小规模的本地目录中来模拟全映射机制。想要得到所有存储器的共享情况，必须搜索整个高速缓存目录链表。链表的结构可以是单向链表(如Stanford的分布式目录协议[6])和双向链表(如IEEE的标准P1596-SCI(Scalable Coherence Interface)协议)。下面主要讨论获得广泛支持的SCI协议。

SCI 共享链是一个双向链表。存储器中的目录项指向共享链的首地址。目录结构如图 3 所示，其中通过每条链路的双箭头表示前、后向指针。即共享链的每一个节点都有一个前驱指针和一个后续指针。SCI 共享链支持 3 个基本操作：插入，删除和简化。插入操作用于一个新的处理器节点要求共享数据时；删除操作用于共享链中一个目录项要作其他用途时；简化操作用于写作废操作时，除了最新写操作的项以外，删除共享链表中的所有项。

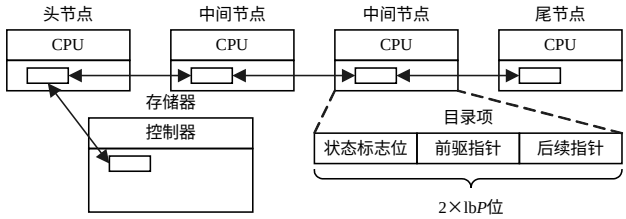

图 3　SCI 链式目录

如前所述，共享某一给定的存储块的处理器节点数一般不超过 5 个。故假定共享链表的平均共享节点数为中间值 2.5 个。加上存储器中的目录项，则一个高速缓存行一共需要 3.5 个目录项。而每个目录项需要 2×lb$P$ 位。整个目录链占用的存储空间是 $(3.5\times 2\times \mathrm{lb}P)\times\frac{M}{B}\times N = O(P\mathrm{lb}P)$ 位。占用空间随着处理器个数呈 Plb$P$ 倍增长，在比全映射表降低了存储开销的同时，对给定的高速缓存行又能够提供与全映射表相同精确的共享信息。这方面，它优于受目录项溢出限制的有限目录。带入假定值，P=64, N=64, M=2^{28}, B=2^9，则有限目录表的存储开销为 8.8%。与有限目录基本相同，再次验证了前面的结论。

但是，链式目录也有缺点：

(1)需要为每一个高速缓存行维护一条链，不但增加硬件复杂性而且减慢了访问时间；

(2)写作废操作只能采用从链表头到尾的串行操作，无法像前 2 个协议一样并行进行。

2　一种改进的目录 Cache 一致性协议

通过比较以上 3 种不同类型的目录协议，可以得出这些协议都有各自的优缺点。全映射(full-map)目录占用过大的存储开销；有限(limited)目录会受目录项溢出的限制；链式(chained)目录的时间有效性较低。现通过将有限目录与全映射目录结合在一起的方式，进而提出了使用两级目录的 Cache 一致性协议。

2.1　系统体系结构

此协议结合了有限目录与全映射目录，在使用了存储器 Cache 的基础上实现了一种双协议的两级目录存储结构。系统体系结构如图 4 所示。替换算法高速缓存行读请求部分如

图 5 所示。

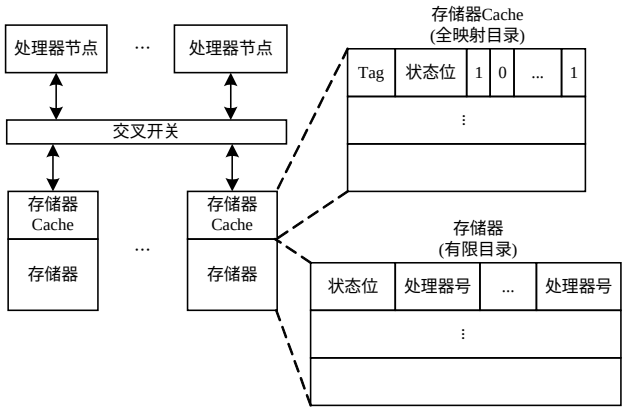

图 4　一种改进的目录 Cahce 一致性协议

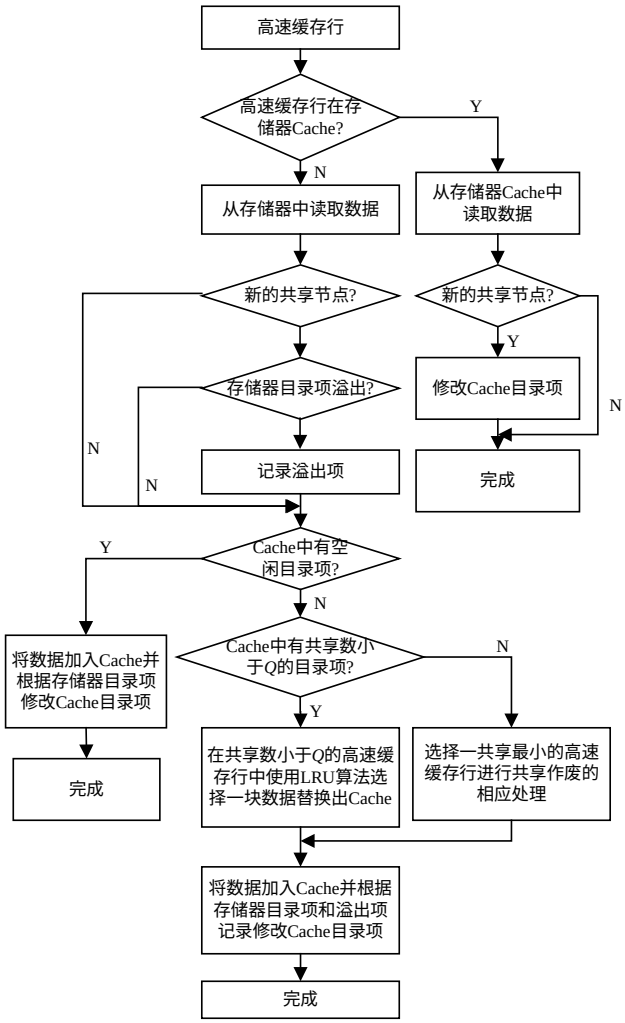

图 5　高速缓存行读请求

在存储器层使用有限目录，由于有限目录单个目录项占用空间小，因此能够使需要大量目录项的存储器层节省不少存储空间。同时，在存储器 Cache 层使用全映射目录，Cache 容量有限，所以即使单个目录项占用空间大，总共占用的空间也不会多。在存储器层和存储器 Cache 层之间的替换算法使用一种共享数加权的伪最近最少使用(LRU)算法来保证，所以共享数目超过存储器层有限目录指针数的高速缓存行都能保证在存储器 Cache 中。假定存储器层的每个目录项使用 Q 个指针，则替换时只对共享数小于 Q 的高速缓存行使用 LRU 替换算法进行替换。只有当存储器 Cache 中所有高速缓存行的共享数皆大于 Q 时，才将一共享数最小的高速缓存行替换出存储器 Cache 并且进行相应作废处理。而这种目录项溢出的情况能够通过适当设置存储器 Cache 的大小来避免。

2.2　性能评估

在占用的存储空间方面，假定系统条件不变，存储器层使用有限目录，每个存储器目录项使用Q个指针，每个指针 $\mathrm{lb}P$ 位，所以每个目录项占 $Q\times\mathrm{lb}P$ 位。而存储器Cache层使用全映射目录，每个目录项占P位，总共C个目录项。整个目录系统占用的存储空间是 $(Q\times\mathrm{lb}P)\times\frac{M}{B}\times N+P\times C\times N$ 位。带入假定值，P=64, N=64, $M=2^{28}$, $B=2^{9}$, $C=2^{16}$，则改进后的目录表的存储开销为 10.9%。由此可见，改进后的协议存储开销与有限目录基本相当，比全映射有显著降低。同时在系统可扩展性方面，当处理器和存储器节点皆扩大至 256 时，全映射存储开销猛增至 50.0%，有限目录增至 12.5%，改进后的协议增至 18.7%。可见此时改进后的协议仍保持有一定的可扩展性。

在系统的性能方面。根据任务的粗细粒度和应用的时间局限性以及读写操作所占比例，合理设置存储器 Cache 和高速缓存行的大小并且使用适合的替换算法来保证存储器 Cache 获得较高的命中率。当存储器层的某一目录项发生溢出时，目录项信息将会复制到存储器 Cache 层中。而根据共享数加权的替换算法，此项目录信息将一直保持在存储器 Cache 中，直到其共享节点数小于存储器层的有限目录项指针数 Q 时，才可以被替换出存储器 Cache。因此，不但解决了有限目录的目录项溢出问题，而且使用频率最高的数据及其目录项一直处于使用全映射目录的存储器 Cache 层中，因而这种两级式存储结构的访问速度可以与其第一级存储器即存储器 Cache 速度相当。

3　结束语

基于目录一致性协议发展到今天已有几十年的历史了，分析前面几种典型的目录一致性协议的目的是为了了解不同协议的技术特点和长短处。本文在结合了全映射目录和有限目录的基础上，使用 Cache 实现了一种改进的目录一致性协议。并对其存储空间开销和性能进行了定量和定性分析。此协议不但有不错的性能，而且空间开销不大，有良好的可扩展性。

参考文献

[1] Tang C K. Cache Design in the Tightly Coupled Multiprocessor System[C]//Proc. of AFIPS National Computer Conf.. New York, USA: [s. n.], 1976: 749-752.

[2] 张　瀛, 黄　巍, 马群生. MP860 层次式并行超级计算机的设计和实现[J]. 计算机学报, 1998, 21(增刊): 230-232.

[3] Censier L, Feautrier P. A New Solution to Coherence Problems in Multicache Systems[J]. IEEE Trans. on Computer, 1978, 27(12): 1112-1118.

[4] Leisersonl D E. The DASH Prototype: Logic and Performance[J]. IEEE Trans. on Parallel and Distributed Systems, 1993, 4(1): 41-61.

[5] 邓让钰, 谢伦国. 多处理机系统中共享数据分布形式[J]. 计算机工程与科学, 1998, 20(1): 66-69.

[6] Thaper M, Delagi B. Stanford Distributed-directory Protocol[J]. Computer, 1990, 23(6): 78-79.