ARC: A SELF-TUNING, LOW OVERHEAD REPLACEMENT CACHE

Nimrod Megiddo and Dharmendra S. Modha
IBM Almaden Research Center

Summarized by Preethy Vaidyanathan

Dharmendra S. Modha discussed how to manage cache or what page to replace to maximize hit-ratio. The cache replacement strategy presented was with respect to demand paging.

Two popular techniques, Least Recently Used (LRU), and Least Frequently Used (LFU), are algorithms that have long been used for cache replacements. LRU captures locality of reference; LFU, the frequency of reference. Modha presented a new scheme, ARC, that captures both these characteristics by maintaining two self-consistent lists. The first lists the pages seen only once and the second lists pages seen at least twice.

In ARC a sliding window of the size of the cache is used to determine what page to replace. The sliding window overlaps the two lists and the percentage of overlap dynamically varies depending on the workload. This implementation has low computational overhead and is tested with a wide range of trace data. ARC consistently outperforms LRU and has similar performance to an offline replacement algorithm that is optimally tuned for the workload.

In the Q&A Modha clarified that the sliding window starts initially with the midpoints in the two lists and the sliding movement is sensitive to the request.

The source code is available at http://almaden.ibm.com/cs/people/dmodha/