Can Large Disk Built-in Caches Really Improve System Performance?

[Extended Abstract] *

Yingwu Zhu and Yiming Hu
Department of Electrical & Computer Engineering and Computer Science
University of Cincinnati
Cincinnati, OH 45221-0030

{zhuy, yhu}@ececs.uc.edu

ABSTRACT

Via detailed file system and disk system simulation, we examine the impact of disk built-in caches on the system performance. Our results indicate that the current trend of using large built-in caches is unnecessary and a waste of money and power for most users. Disk manufacturers could use much smaller built-in caches to reduce the cost as well as power-consumption, without affecting performance.

1. INTRODUCTION

Almost all modern hard disk drives have an internal buffer, or *built-in cache*, which acts both as a speed-matching buffer (between the bus and the disk media) and as a disk block cache. In the last couple of years, hard disk manufacturers have dramatically increased the size of disk built-in caches in their products. Today, a 2 MB built-in cache is common on retail low-end IDE/ATA drives, and some SCSI drives are now available with 16 MB. There are possibly two main reasons for this dramatic increase in built-in cache sizes. The first is that memory prices have dropped precipitously over the last few years. The second is pertinent to marketing: hard disk consumers have a perception that doubling or quadrupling the size of the built-in cache will have a great impact on the system performance.

While large built-in caches may show their usefulness under some artificial disk benchmarks, there are no published research results, as far as we know, to demonstrate that they can really improve the performance of real systems. In fact, we believe that large disk built-in caches will not significantly benefit the overall system performance. The reason is that *all modern operating systems already use large I/O buffer caches to cache reads and writes*. Nowadays, the OS can use an I/O buffer cache ranging from tens of MBs to several GBs. With such a large OS buffer cache, the effectiveness of disk built-in cache is unclear.

*A full version of this paper is available as [4]

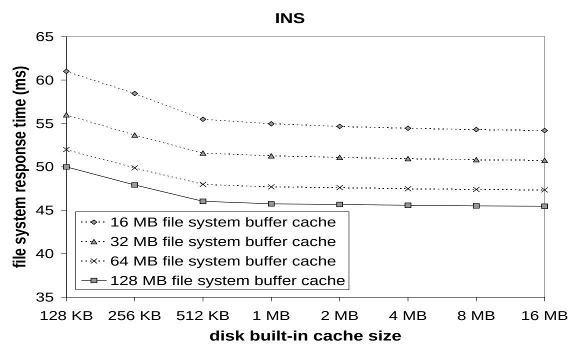

Figure 1: File system response time with various cache sizes. Readahead at the disk is enabled. Write caching at the disk is disabled.

This paper uses detailed file system and disk system simulation to explore the effect of the built-in cache on the system performance, especially when the file system buffer cache gets larger and larger.

2. SIMULATION RESULTS AND PERFORMANCE ANALYSES

We used five real-world workloads in our simulation, which are **INS**, **RES**, **WEB**, **HARP** and **SITAR** [1][2]. Due to space constraints, we show only a few graphs to illustrate our results. Omitted figures and tables show that our conclusions hold for all real-world workloads used in our experiments.

2.1 Impacts of Sizes of File System Caches and Disk Built-in Caches

Figure 1 shows that, with a file system buffer cache size of 16 MB or more, *the built-in cache has very little impact on the file system response time when the disk built-in cache size is larger than 512 KB*. We observed similar results for other four real-world workloads as well. When the file system buffer cache is 16 MB, a system using a 16 MB disk built-in cache is only 1.1–4.1% faster than a system with a 512 KB disk built-in cache. When the file system buffer cache is 128 MB, the former is only 0.3–1.4% faster than the latter.

We also observed that, when the file system buffer cache increases from 16 MB to 128 MB, the disk built-in cache miss rate increases

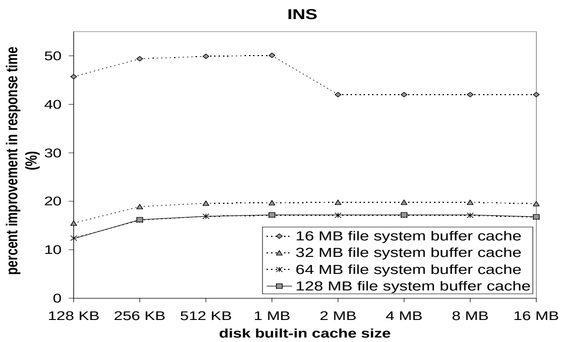

Figure 2: Percent improvement of enabling readahead on response time with different file system buffer cache sizes and disk built-in cache sizes. Write caching at the disk is disabled.

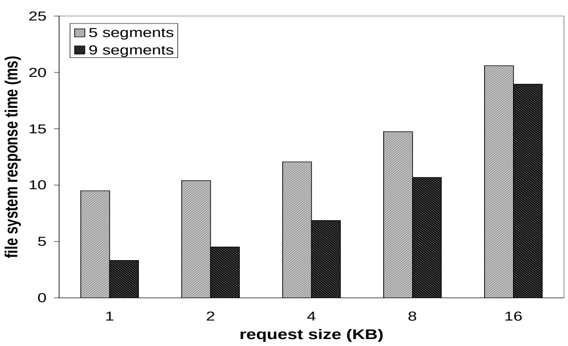

Figure 3: File system response time for sequentially reading nine 64 KB files with 5 readahead cache segments and 9 readahead cache segments. The file system buffer cache is 16 MB and the disk built-in cache size is 2 MB. Write caching at the disk is disabled.

slightly. This suggests that larger file system buffer caches captures more locality, those requests missing the file system buffer cache therefore have poorer locality, resulting in higher built-in cache miss rates.

2.2 Impacts of Readahead Policy

To improve the read cache hit rate for sequential access patterns, modern disk drives prefetch data after read requests. With the file system buffer cache size increasing from 16 MB to 128 MB, the percent improvements of enabling readahead on file system response time decrease dramatically. This is because larger file system buffer caches mean higher cache hit, and can absorb most reads and writes, thus amortizing the negative effect of not performing readahead to some extent. Figure 2 shows that a 512 KB disk built-in cache performing readahead nearly achieves a maximum percent improvement on response time over one without performing readahead, even compared to a larger built-in cache.

The effect of readahead depends on a number of factors, including the number of cache segments, the number of concurrent sequential workloads, and request sizes. Figure 3 compares the file system response time when servicing nine concurrent workloads with 5 cache segments and the response time with 9 cache segments. Obviously, 9 cache segments perform much better than 5 segments in concurrently reading nine files. This suggests that, if there are enough cache segments to service multiple streams of sequential requests, readahead will work more effectively.

2.3 Writing Caching at the Disk

When write caching is enabled at the disk, the write to the disk can be reported as complete as soon as the write is written into the disk built-in cache [3]. Since the disk built-in cache is made of volatile RAM, it is by default write-through to preserve data in case of a power failure.

We designed a new policy — *selective write caching*, wherein metadata requests and user-data requests are treated differently at the disk level. That is, metadata, which uses synchronous writes, will be directly written to the disk media bypassing write caching, while user-data will be serviced with write caching. In some write-intensive workloads, the benefits of selective write caching should be worth the effort of its implementation.

3. CONCLUSIONS

Disk drive manufacturers are putting increasingly larger built-in caches into disk drives. Some disk drives nowadays have a cache of 16 MB or bigger. However, there are few published results to demonstrate that such large built-in caches can really improve overall system performance. While most of the research work has been busy trying to come up with better caching schemes and finding new ways to use caches, our work takes the opposite view.

In this paper, via detailed file system and disk system simulation with realistic configurations driven by both real-world file system workloads and synthetic micro-benchmark workloads, we studied the impact of the disk built-in cache on file system response time when the file system buffer cache becomes larger. With a reasonably-sized file system buffer cache (16 MB or more), there is very little performance benefit of using a built-in cache larger than 512 KB. Our results indicate that the current trend of using large built-in caches is unnecessary and a waste of money and power for most users. Disk manufacturers could use much smaller built-in caches to reduce the cost as well as power-consumption, without affecting performance.

4. ACKNOWLEDGMENTS

This work is supported in part by the National Science Foundation Career Award CCR-9984852, and an Ohio Board of Regents Computer Science Collaboration Grants.

5. REFERENCES

[1] J. Griffioen and R. Appleton. The design, implementation, and evaluation of a predictive caching file system. Technical Report CS-264-96, Department of Computer Science, University of Kentucky, June 1996.

[2] D. Roselli, J. R. Lorch, and T. E. Anderson. A comparison of file system workloads. In *Proceedings of the 2000 Annual Technical USENIX Conference*, pages 41–54, June 2000.

[3] C. Ruemmler and J. Wilkes. An introduction to disk drive modeling. *IEEE Computer*, 27(3):17–28, 1994.

[4] Y. Zhu and Y. Hu. Can large disk built-in caches really improve system performance? Technical Report TR259/03/02ECECS, Department of Electrical & Computer Engineering and Computer Science, University of Cincinnati, March 2002.