# Operating System Implications of Solid-State Mobile Computers

Ramón Cáceres, Fred Douglis, Kai Li †, and Brian Marsh

Matsushita Information Technology Laboratory
2 Research Way, Princeton, NJ 08540
{ramon,douglis,marsh}@research.panasonic.com

† Department of Computer Science, Princeton University
35 Olden Street, Princeton, NJ 08544
li@cs.princeton.edu

**Abstract**

Trends in storage technology indicate that future notebook, palmtop, and smaller mobile computers will contain battery-backed DRAM as primary storage and direct-mapped flash memory as secondary storage, but no disk. All storage will offer uniform, random-access read times through a single-level 64-bit address space. This paper explores the operating system implications of this storage organization. The system should exploit the benefits of having all data reside in fast memory. It can do away with much of the data duplication and related data movement that take place in conventional organizations. The system also needs to hide the limitations of flash memory: write access times higher than read access times, the need to erase memory before rewriting it, and a limited number of write cycles in the lifetime of the device. It needs to limit write traffic to flash memory and avoid writing repeatedly to the same area of flash memory. These steps will increase performance, improve space utilization, and prolong the life of flash memory.

## 1 Introduction

Rapid advances in solid-state storage technology and the requirements of mobile computing will bring about new storage organizations with significant implications for operating systems. Current workstations and laptop computers use fast semiconductor memory as volatile primary storage, and a much larger amount of slower magnetic disk as stable secondary storage. The prevailing 10:1 ratio in cost between memory and disk has prescribed the large differences in capacity between primary and secondary storage, while steady improvements in disk technology have kept at bay challenges by many other secondary storage technologies.

However, several factors are combining to change the design of storage hierarchies. First, mobile computers are emerging as an important class of system. They demand components

that are not only cost-effective, but also small, light-weight, low-power and robust. Solid-state components are superior or comparable to magnetic disks under these additional measures [4, 5, 6, 7, 13]. Second, semiconductor memories will come to match small-diameter disks in cost and density. The megabytes per dollar and megabytes per cubic inch of these memories are improving faster than those of disks [10]. Third, a new type of non-volatile semiconductor memory, *flash* memory, offers the stability of disk, read access times comparable to those of DRAM, and lower power consumption than either [9].

As a result, future notebook, palmtop, and smaller computers will use DRAM (Dynamic Random Access Memory) as primary storage and flash memory as secondary storage, but no disk. All data will reside in a single-level 64-bit address space. All storage will offer uniform, random-access read times. Primary storage will be highly stable, since battery-backed DRAM is ubiquitous in mobile computers. Secondary storage will be used primarily to provide added stability for long-lived data, not necessarily to expand storage capacity, since primary and secondary storage costs will be comparable. These advantages are balanced by the drawbacks of flash memory devices: write access times higher than read access times, the need to erase blocks of memory before they can be rewritten, and a limited number of erase/write cycles in the lifetime of the device.

This paper explores the operating system implications of using battery-backed DRAM and direct-mapped flash memory in lieu of magnetic disks on small mobile computers. Section 2 compares the performance, cost, size, weight, and power consumption characteristics of DRAM, flash memory, and disk. Section 3 describes how the file system, the virtual memory system, and the physical storage manager can make effective use of the new storage organization. Section 4 discusses the issue of how to apportion the total storage capacity of a mobile computer between DRAM and flash memory. Section 5 concludes the paper.

## 2 Storage Organizations for Small Mobile Computers

The leading storage technologies for mobile computers are DRAM, small-diameter magnetic disks, and flash memory. DRAM and disks are well-known. Flash memory was more recently introduced and warrants further description. It is a semiconductor integrated circuit technology that provides random byte-level access to storage, holds its contents when power is removed, and requires that it be erased in a separate step before it is rewritten. Flash memory is available as discrete components in up to 4-megabit densities, and packaged as cards in up to 20-megabyte configurations. Current devices offer read access times in the 100-nanosecond per byte range and write times in the 10-microsecond per byte range. They impose a minimum erase sector in the 512-byte range, and endure a guaranteed 100,000 erase cycles per area. These devices cost in the 50-dollar per megabyte range and consume power in the tens of milliwatts per megabyte when in use.

We compared the performance, cost, size, and power consumption of currently available DRAM, flash memory, and disk products. We chose a DRAM product from NEC that features 3.3-volt operation and a special low-power self-refresh mode [7]. We considered flash memory products from Intel and SunDisk. The SunDisk product is intended to replace hard drives

and is optimized for both read and write performance [13]. The Intel product is intended to support memory-mapped access, and has much faster read times but slower write times [6]. Finally, we considered a 1.3-inch KittyHawk disk drive from Hewlett-Packard [5] and a 2.5-inch drive from Fujitsu [4]. To summarize their differences, DRAM is faster than flash memory but somewhat costlier, while disk is slower than flash memory but considerably cheaper. Furthermore, flash memory has lower power consumption than either DRAM or disk.

Extrapolating our findings using established technology trends yields important insights. We note that the cost of DRAM will match the cost of disks. The megabytes per dollar of DRAM increases by 40% a year, compared to 25% for disk [10]. Thus, while a 20-megabyte DRAM package currently costs ten times more than a 20-megabyte disk drive, these prices will become comparable. Similarly, the density of DRAM will shortly exceed that of disk. The megabytes per cubic inch of DRAM also increase by 40% a year, compared to 25% for disk [10]. The NEC DRAM already provides 15 megabytes per cubic inch compared to the 19 megabytes per cubic inch provided by the KittyHawk. These trends suggest that DRAM will come to represent a larger percentage of a system's storage capacity than it currently does.

However, DRAM cannot by itself provide stable storage. Battery-backed DRAM is ubiquitous in mobile computers and affords more stability than in conventional configurations. Nevertheless, the contents of DRAM will not survive a battery failure. Such failures will be relatively common in mobile computers: batteries can be depleted by other devices, and computers can be dropped. Non-volatile storage that survives power losses is essential.

Again applying technology trends to current products, we conclude that flash memory will replace disks for stable storage on small mobile computers. The densities of both the SunDisk and Intel products are already within 20% of the density of the KittyHawk drive. Although their densities are only half that of the Fujitsu drive, manufacturers expect flash memory densities to match and follow the increases in DRAM densities [6]. As the cost of flash memories also drops along with that of DRAM, the advantage offered by small disks like the KittyHawk will amount to at best a few dollars per drive. Some estimates predict that, for 40-Megabyte configurations, the cost per megabyte of flash memory will match that of magnetic disks by the year 1996 [6]. In addition to the cost and density considerations, flash memory offers significant power savings over disk drives, thus prolonging battery life. Finally, flash memory is more robust than disk drives – it has no moving parts.

These combined factors will bring about small mobile computers that contain only semiconductor memory but no disk. Such solid-state computers already exist. Small personal information managers like the Sharp Wizard and the Casio Boss are early examples. Storage for these machines typically consists of less than one megabyte of semiconductor memory. Larger-capacity notebook computers have also appeared with only battery-backed DRAM and flash memory. For instance, the Hewlett-Packard OmniBook is available with a 10-megabyte flash memory card as its only source of secondary storage. New personal digital assistants such as the Apple Newton MessagePad and the Casio/Tandy Zoomer constitute additional examples of solid-state machines. We envision that an increasing number of mobile computers, both specialized and general-purpose, will follow this trend.

# 3 Operating System Implications

A hierarchy of DRAM as primary storage and flash memory as secondary storage has important implications for the operating system. The challenge is to exploit the benefits of this organization while hiding its drawbacks.

The most notable benefit is uniform, random-access read times across primary and secondary storage. Combined with the advent of 64-bit address spaces, the resulting single-level store allows all application programs and their data to be memory-resident along with the operating system. Since everything resides in fast memory, the system can do away with much of the data duplication and related data movement that takes place in conventional storage organizations. These steps will improve performance and space utilization.

The disadvantages of the new organization come from the erase and write characteristics of flash memory. Flash memory must be erased in sectors before it can be rewritten, and there is a limit on the maximum number of erase/write cycles per sector. Furthermore, write access times are two orders of magnitude higher than read access times. The system needs to limit write traffic to flash memory and avoid writing repeatedly to the same area of flash memory. These steps will reduce latency and prolong the life of flash memory.

Below we discuss the effects of the new storage organization on the file system, the virtual memory system, and a physical storage manager shared between the two.

## 3.1 Changes to the File System

An important result of having all storage directly accessible to the processor will be a memory-resident file system. In such a system, many traditional policies and mechanisms do not apply. For example, there is no need to cluster related data, since the latency of seek operations is not a consideration. The complexity of multiple levels of indirect blocks may also be eliminated. Finally, traditional file system caches are unnecessary because all data and metadata always reside in fast storage.

The use of flash memory for stable storage has further implications for the file system. Files that are read but not often written can be left in stable storage without loss of performance. In systems that present a memory-mapped file interface to application programs, files in flash memory can be mapped directly into the address spaces of interested processes without having to make a copy in primary storage. These techniques save both the storage needed for duplicate copies and the time needed to perform the copies.

Copy-on-write techniques can be used to postpone the complications brought on by the erase/write behavior of flash memory until application-level writes actually take place. Thus, long-lived data can be left in flash memory until it is written. When a write operation occurs, the affected block can be copied to DRAM, where it is left in a write buffer until it is eventually saved to stable storage. Newly created file data is first written to DRAM and eventually saved to stable storage.

It is important to note that in mobile computers all of main memory is battery-backed. The primary batteries in these systems discharge gradually and predictably. They can preserve the contents of main memory in an otherwise idle system for many days. A second set

of small lithium batteries often provide a backup power source for use when the primary batteries drain completely, or while a fresh set of primary batteries is swapped for a depleted set. These backup batteries can preserve the contents of main memory in an otherwise idle system for many hours. With appropriate care to ensure that an untimely crash is unlikely to corrupt data [1, 2], DRAM can safely hold file system data for much longer than in conventional configurations.

### 3.2 Changes to the Virtual Memory System

When primary storage prices are comparable to secondary storage prices, virtual memory will be used primarily to provide protection across multiple address spaces, rather than to expand capacity as in many current systems. DRAM will constitute a larger percentage of a system's total storage capacity than it currently does. This development will improve performance by reducing the need to page or swap processes between primary and secondary storage.

In addition, a system with flash memory can obtain significant performance gains over a system with magnetic disks by processing data directly from stable storage. For example, programs residing in flash memory can be executed in place without loss of performance [15]. There is no need to load their code segment into primary storage before execution, again saving both the storage needed for duplicate copies and the time needed to perform the copies. The data and stack segments can be allocated in primary storage as usual. This technique is already in use, for example in the Hewlett-Packard OmniBook, where bundled software is shipped in removable memory cards and executed in place [12].

### 3.3 The Physical Storage Manager

The storage manager will be responsible for migrating data between DRAM and flash memory to keep data that is frequently written in DRAM, and data that is mostly read in flash memory. It can buffer written data in DRAM before eventually flushing it to flash memory. This technique can keep the rate of writes into flash memory manageably low because a large percentage of write operations are to short-lived files or to file blocks that are soon overwritten [3, 8]. Trace-driven simulations of networked workstations have shown that as little as one megabyte of battery-backed RAM can reduce write traffic by 40 to 50% [1].

In order to maintain fast read access to programs and other data in secondary storage during the slow erase/write cycles of flash memory, it may prove necessary to partition flash memory into two or more banks. One bank would hold read-mostly data, such as application programs, while others would be used for data that is more frequently written. File systems would be spread across flash memory banks appropriately, just as file systems are balanced across disks today.

Finally, in order to evenly balance the write load throughout flash memory, the storage manager can use garbage collection techniques like those used in log-structured file systems [11] and some programming language environments [14]. The storage manager can also maintain a list of free flash memory sectors and a list of free DRAM pages, allocating them to the file and virtual memory systems as needed.

## 4 Sizing DRAM and Flash Memory

A mobile computer with a limited cost, weight, or volume budget can hold only a limited amount of storage. Today, one may have to choose between 12 megabytes of DRAM, 20 megabytes of flash memory, or 120 megabytes of magnetic disk for the same cost. At some point DRAM and flash memory are likely to attain costs and densities comparable to each other and better than disks. At that point, the question arises: How should a system apportion its storage capacity between the two technologies? Should the ratio between DRAM and flash memory capacities be 1:1, or something else?

The answer depends on the workload. DRAM has the advantage of better write performance and relatively unlimited endurance, but flash memory uses less power and must ultimately be the repository for long-lived data. If one could be certain that the writable working set of all running applications would never exceed some threshold, one could configure enough DRAM to buffer these writes and keep the remaining data in flash memory. Unfortunately, it is not possible to predict access patterns for a general-purpose computer. It seems likely that the ratio of DRAM to flash memory capacities will be higher than the current ratio of DRAM to disk capacities. However, the exact proportions will be based on assumptions about access patterns in an attempt to provide good performance, low power consumption, and a sufficiently large repository for permanent data.

## 5 Conclusions

Rapid cost and density improvements in semiconductor memory technology, together with the stringent size, weight, power and robustness requirements of mobile computers, will bring about new storage organizations. The storage hierarchy for notebook, palmtop and smaller computers will consist of battery-backed DRAM for primary storage and direct-mapped flash memory for secondary storage, but no disk. The expected workload of these systems will determine the relative sizes of DRAM and flash memory. The resulting organization will have the following characteristics: uniform, random-access read times across primary and secondary storage; higher write times to secondary storage; the need to erase blocks of secondary storage before they can be rewritten; and a limited number of erase/write cycles in the lifetime of secondary storage. The operating system needs to exploit the advantages of this organization while hiding its limitations. For example, the file system can be entirely memory-resident; read-only data can be accessed directly from flash memory; and a DRAM buffer can reduce write traffic to flash memory. These steps will increase performance, improve space utilization, and prolong the life of flash memory.

## References

[1] Mary Baker, Satoshi Asami, Etienne Deprit, John Ousterhout, and Margo Seltzer. Non-volatile memory for fast, reliable file systems. In *Proceedings of the Fifth International*

*Conference on Architectural Support for Programming Languages and Operating Systems*, pages 10–22, Boston, MA, October 1992. ACM.

[2] Mary Baker and Mark Sullivan. The recovery box: Using fast recovery to provide high availability. In *Proceedings of the USENIX 1992 Summer Conference*, pages 31–44, San Antonio, TX, June 1992.

[3] Mary Baker et al. Measurements of a distributed file system. In *Proceedings of the 13th Symposium on Operating System Principles*, pages 198–212, Pacific Grove, CA, October 1991. ACM.

[4] Fujitsu. *M263x Data Sheet*, 1993.

[5] Hewlett-Packard. *Kittyhawk HP C3013A/C3014A Personal Storage Modules Technical Reference Manual*, March 1993. HP Part No. 5961-4343.

[6] Intel. *Memory Products*, 1993.

[7] NEC. *Memory Products Data Book, Volume 1: DRAMS, DRAM Modules, Video RAMS*, 1993.

[8] John Ousterhout et al. A trace-driven analysis of the Unix 4.2 BSD file system. In *Proceedings of the 10th Symposium on Operating System Principles*, pages 15–24, Orcas Island, WA, December 1985. ACM.

[9] Richard D. Pashley and Stefan K. Lai. Flash memories: the best of two worlds. *IEEE Spectrum*, December 1989.

[10] David A. Patterson and John L. Hennessy. *Computer Architecture: a Quantitative Approach.* Morgan Kaufmann, 1990.

[11] Mendel Rosenblum and John Ousterhout. The design and implementation of a log-structured file system. *ACM Transactions on Computer Systems*, 10(1):26–52, February 1992. Also appears in Proceedings of the 13th Symposium on Operating Systems Principles, October 1991.

[12] Andrew Seybold. The Hewlett-Packard OmniBook 300. *Outlook on Mobile Computing*, June 1993.

[13] SunDisk. *SunDisk SDI OEM Manual*, 1993.

[14] David Ungar. Generation scavenging: A non-disruptive high performance storage reclamation algorithm. In *Proceedings of the Software Engineering Symposium on Practical Software Development Environments*, pages 157–167, Pittsburgh, PA, April 1984.

[15] Don Verneer. eXecute-In-Place. *Memory Card Magazine*, March/April 1991.