What's Up with the Storage Hierarchy?

Philippe Bonnet
IT University of Copenhagen
Rued Langaard Vej 7
2300 Copenhagen, Denmark
phbo@itu.dk

ABSTRACT

Ten years ago, Jim Gray observed that flash was about to replace magnetic disks. He also predicted that the need for low latency would make main memory databases commonplace. Most of his predictions have proven accurate. Today, who can make predictions about the future of the storage hierarchy? Both main memory and storage systems are undergoing profound transformations. First, their design goals are increasingly complex (reconfigurable infrastructure at low latency, high resource utilization and stable energy footprint). Second, the status quo is not an option due to the shortcomings of existing solutions (memory bandwidth gap, inefficiency of generic memory/storage controllers). Third, new technologies are emerging (hybrid memories, non-volatile memories still under non-disclosure agreements, near-data processing in memory and storage). The impact of these transformations on the storage hierarchy is unclear. Yet, they raise interesting research questions.

CCS Concepts

• CCS→ Hardware→ Communication hardware, interfaces and storage→ **External storage**• CCS→ Information systems→ Data management systems→ Database management system engines→ **DBMS engine architectures**

Keywords

Storage Hierarchy; Solid State Drives; Flash; Non Volatile Memory; Offload Engines.

1. INTRODUCTION

Jim Gray's Predictions. At CIDR 2007, Jim Gray gave a Gong Show presentation entitled *"Tape is Dead. Disk is Tape. Flash is Disk. RAM Locality is King"* [5]. In 10 memorable slides, he shared his insights on the upcoming evolution of the storage hierarchy: magnetic disks as cold-storage archives, flash-based solid state drive as secondary storage of choice, and affordable RAM holding entire databases. Many of Jim's predictions have proven accurate. Commercial grade NAND flash-based SSDs of 2TB cost under 700$ (Jim predicted that by 2012, a 1TB flash would cost 400$). Also, 1TB of RAM costs about 5500$, so the ratio between NAND flash and RAM is around 10:1, as Jim predicted. In terms of performance, SSDs reach 100s of K IOPS, as Jim predicted. While the death of tape is debatable, main-memory databases have become commonplace.

Today, new architectures and technologies are proposed for both memory and storage systems in order to match increasingly complex design goals. In this talk, I will briefly review these transformations and the research questions they raise.

2. DESIGN GOALS

Legacy. The traditional storage hierarchy is a pyramid of layers representing memory and storage components attached to a host equipped with compute cores (cache, RAM, secondary storage and tertiary storage). Each layer is orders of magnitude faster and more expensive than the next. The storage hierarchy has been relevant for building balanced systems where cores, memory and storage performance are aligned. The goal is to avoid diverging performance trends across layers, thus making it possible to scale up systems without changes to legacy software.

Scale-out Workloads. Today, large-scale data systems including main-memory databases, key-value stores or map-reduce frameworks qualify as scale-out workloads [2]. They are designed for main-memory in a shared nothing cluster, and result in significant data movement, and thus high energy footprint. On a cluster, the storage hierarchy is stretched across two dimensions (memory/disk, local/rack/cluster) and offers a range of possible trade-offs in terms of latency, bandwidth and capacity [3].

The quest for lower latency, high resource utilization, energy efficiency and reduced cost is pushing the cloud service providers that run large-scale data systems to assemble clusters of rack-scale computers directly from custom ODM[1] components. Conserving legacy software is no longer a requirement. In fact, cloud service providers constantly adapt their infrastructure to the changing needs of their customers. This has two consequences on the storage hierarchy: (i) *disaggregation*: hardware resources should be provisioned and accessed at (server/)rack/cluster scale, regardless of their initial packaging (with a push towards shared memory and shared disks[2]), and (ii) *software-defined infrastructure*: software should control how hardware resources are provisioned and accessed throughout the storage hierarchy.

3. STORAGE TRENDS

Main Memory Modules. The cost-per-bit of DRAM has maintained its exponential decrease over the years[3]. However, DRAM latency per core has not improved, while capacity and bandwidth per core have worsened [7]. This is a traditional performance gap in the storage hierarchy. A problem is that memory controllers have been designed (by OEMs) to optimize cost-per-bit at the detriment of latency. Another problem is that DRAM's fabrication process is not expected to keep on scaling down. New technologies are developed to address these shortcomings: e.g., 3D stacking[4] improves memory bandwidth (at higher cost and reduced energy efficiency), and hybrid memories improve capacity (e.g., Diablo's Memory1 based on flash-backed RAM provides 256GB DIMM module at a fraction of the storage price and of the energy consumption of DRAM: *Flash is RAM*!).

But what is the point of improving memory modules if it results in more data movement and higher energy footprint? A solution to this problem is to introduce application software within a package that combines memory module and compute cores. The idea is to apply near-data processing ideas, initially developed for storage

[1] Original Design Manufacturers (ODM) sell customized components as opposed to Original Equipment Manufactureres (OEM) that sell pre-packaged systems (assembled from ODM components).

[2] DSSD (from EMC) is an example of a shared disk architecture at rack-level.

[3] http://www.jcmit.com/MemDiskPrice-xl95.xls

[4] 3D stacking technologies (TSV) connected to computing cores through a serial (HMC) or parallel (HBM) interface.

systems [2], to the memory system (e.g., Altera Stratix M10). Designing such modules raises a set of interesting problems, discussed by Babak Falsafi in a recent IEEE Computer Society op-ed[5]. Using such modules as offload engine also raises interesting issues for data system design.

Another point, made by Omar Mutlu [7], is that memory interferences between cores are largely uncontrolled. These interferences result in unpredictable performance and potential exploits. He advocates the design of hardware mechanisms, within the memory system, that enable system software to enforce QoS policies. An interesting question is how data system could leverage such QoS policies.

Solid State Drives. SSDs are composed of a compute core (running the storage controller) directly attached to NVM chips. NVMe has emerged as the storage protocol of choice for SSDs directly attached via PCIe, or remotely via a fabric interconnect. The advent of flash chips has resulted in orders of magnitude performance improvements for Solid State Drives (SSDs), both in terms of throughput and latency. As a result, it has been necessary to revisit system software to streamline the data path. The quest for predictable performance, high resource utilization, and reconfigurable infrastructures has led to open-channel SSDs, equipped with a minimal storage controller that exposes SSD resources directly to the host[6], as opposed to hiding them behind a proprietary Flash Translation Layer (designed by an OEM).

The Second Coming of Active Disks. SSDs are ideal vehicles for near-data processing with the objective to (i) minimize data movement and (ii) support software-defined reconfigurations. Seshadri et al. proposed Willow [9], that allows applications to drive an SSD by installing custom software on small processors running within the SSD. Also, Lee et al. [5] explored a new system architecture, BlueDBM, which systematically pushes processing to FPGA-based SSDs. An interesting development for those of us who are not hardware designers, is that platforms are now available to experiment with SSD programming: (1) EMC and NXP have designed the DFC card (also called iSSD) equipped with an ARMV8 processor directly attached to NVRAM chips via four channels; (2) GS Madhusudan's group at IIT Chennai has designed a FPGA-based storage controller, that is publically available, in the context of the Lightstor project[7]. This is an opportunity for the data management community to revisit active disks concepts, and to finally realize the vision of trusted storage.

Persistent Memories. Persistent memories are byte addressable non-volatile memories, directly accessible from the processor, just as RAM. Different classes of technologies are emerging: resistive RAM (Hynix, SanDisk, Crossbar, Nantero), ST-MRAM (IBM, Samsung, Everspin), and PCM (IBM, HGST, Micron/Intel)[8]. Much remains unknown about the actual chips, but it is expected that PCM and RRAM will provide high capacity with write speed in the 100s nsec, while ST-MRAM will provide high endurance at write speed in the 10s nsec. Today, packaged ST-MRAM costs 5*1E-3 KB/$[9], which is approximately 100x the storage price of RAM in the early 80s. However, without insider knowledge, it is difficult to predict how NVMs will be priced when they hit the market in 2017 or 2018.

It is likely that, in the next few years, these emerging Non-Volatile memories will complement rather than replace RAM and flash. They will be packaged as memory modules (possibly in the context of hybrid memories) accessed via the memory controller, or as SSDs accessed via the I/O or network interface controller. They will thus add persistence on the memory channel, and decrease SSD latency by orders of magnitude.

4. SYSTEM DESIGN

Storage Hierarchy 2025. So can we make predictions about how the storage hierarchy will look like in a few years? Most of the transformations described in this paper are disruptive (QoS, Near-data processing, persistent memories). There are no public data that we can extrapolate from in order to predict the storage price of persistent memories or the sales volume of programmable SSDs. The question is whether we will still talk of a storage hierarchy then, or whether we will consider various combinations of memory, storage (as well as processor and interconnect) subsystems that fit the requirements of scale-out workloads to meet diverse cost, energy, latency or bandwidth goals.

Research Issues. If we accept the premise that a system should be custom-built from a range of memory and storage components in order to support a scale-out workload, then there is no single definition of a balanced system. Does it mean that anything goes and that we can design data systems based on any assumption in terms of memory or storage? No. It looks to me like we have several reasonable options: We can (1) assume that near-data processing is a viable option for system design and explore how it impacts data systems; (2) assume that memory and storage provide Quality of Service mechanisms and design data systems that leverage these guarantees; (3) assume that persistent memories are available through memory or storage controllers and study how to leverage them in the context of programmable memory or storage modules (e.g., [1][8]).

[5] https://www.computer.org/csdl/mags/mi/2016/01/mmi2016010006.pdf

[6] https://www.phoronix.com/scan.php?page=news_item&px=Linux-4.4-LightNVM

[7] http://www.lightstor.org/

[8] No detail has been disclosed about Micron's and Intel's X-point. It might fall into a category of its own, but is classified as PCM by Yole Dev. Storage industry leaders such as Samsung, IBM or Western Digital are concurrently developing several technologies.

[9] A 2MB ST-MRAM chip from Everspin costs 40.9 K$ at Digikey.com.

REFERENCES

[1] Arulraj, Joy, Andrew Pavlo, and Subramanya R. Dulloor. "Let's talk about storage & recovery methods for non-volatile memory database systems." *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data*. ACM, 2015.

[2] Balasubramonian, Rajeev, et al. "Near-data processing: Insights from a MICRO-46 workshop." *IEEE Micro* 34.4 (2014): 36-42.

[3] Dean, Jeff. "Designs, lessons and advice from building large distributed systems." *Keynote from LADIS* (2009): 1.

[4] Ferdman, Michael, et al. "Clearing the clouds: a study of emerging scale-out workloads on modern hardware." *ACM SIGPLAN Notices*. Vol. 47. No. 4. ACM, 2012.

[5] Gray, Jim. "Tape is dead, disk is tape, flash is disk, RAM locality is king." *Gong Show Presentation at CIDR* (2007): 231-242.

[6] Lee, Sungjin, et al. "Application-managed flash." *14th USENIX Conference on File and Storage Technologies (FAST 16)*. 2016.

[7] Mutlu, Onur. "Memory scaling: A systems architecture perspective." *2013 5th IEEE International Memory Workshop*. IEEE, 2013.

[8] Roberts, David, et al. *Is storage hierarchy dead? co-located compute-storage nvram-based architectures for data-centric workloads*. Technical Report HPL-2010-119, HP Labs, 2010.

[9] Seshadri, Sudharsan, et al. "Willow: a user-programmable SSD." *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI 14)*. 2014.