

From Micro- to Macro-processing: A Generic Data Management Model

Mehmet Balman and Tevfik Kosar

Center for Computation & Technology, and Computer Science Department
Louisiana State University, Baton Rouge, LA 70803, USA
{balman,kosar}@cct.lsu.edu

Rapid progress in distributed computing have provided collaborative studies and essential compute power for science, and made data administration a critical challenge in this area. Scientific applications have become more data intensive; moreover, data management happens to be more demanding than computational requirements in terms of needed resources. There have been many recent studies investigating new approaches for data management and data transfer in distributed systems. On the other hand, data management has been one of the crucial problems in every stage of computer engineering. Accessing data in a transparent and efficient manner is also a major issue both in operating system design and in microprocessor architecture. Data management and data access approaches in operating system level and processor design can be applicable to distributed systems.

We analyze data access and data management issues in different layers of computer systems starting from microprocessor level, extending to operating system level, and then, investigating data handling in distributed systems. We define characteristics of each layer in terms of similarities and differences in storage operations. We study problems and challenges in data access and data storage, and we investigate applied techniques in small and large scale systems. Studying and analyzing in different scales will enable us to better understand the data management problem in distributed systems.

In distributed systems, different service layers have been defined and overall load is shared between separate components. Therefore, there are different software and hardware elements serving for specific purposes to maintain the overall system. Basic components of a distributed system are computing and storage resources which are reachable to each other over network connections. Specialized data servers and data storage sites have been formed due to storage requirements of today's applications. Necessitated input data is either transferred and stored in a temporary space or accessed remotely over the network. Due to the nature of interconnects between distributed elements there are many factors affecting performance such as latency and bandwidth. This leads to staging and caching services in order to make data easily and quickly accessible.

In recent past, microprocessors have been one of the fastest growing technologies. In the early models, accessing data was one of the major problems as it is still a crucial issue today; such that, data retrieval from memory to execution unit is slow due to the speed of memory and also the latency in the bus between memory and CPU. In order to overcome the speed gap between arithmetic operations and memory access operations, many techniques have been applied. Scheduling the execution of instructions starts with out-of-order execution technique used in early x86 designs and extends to dynamic scheduling algorithms in which operations are buffered in the issue phase before they are sent to execution units. Moreover, pre-fetch registers, multi-level caches, complex branch prediction algorithms are some of the methodologies used in processor design. Execution stages are split into execution units and also into multiple pipelines in order to enhance performance using pipelining and superscalar design. Load-store operations are separated from other execution units

and I/O instructions are scheduled and executed by specialized units in microprocessors.

Besides, many approaches have been used in operating system level to access data in an efficient manner. DMA, I/O scheduling, file systems, disk management, simultaneous resource sharing, and low-level parallelism such as disk striping are some common examples. The I/O bottleneck has also been studied in parallel computer systems and many interesting techniques have been proposed and applied successfully. Data handling has been dealt with separate components and special action has been required throughout the evolution of computer systems.

We first start with microprocessors, the basic component of computer systems, and study data management in processor architecture. We focus on problems on accessing and storing data that have been encountered in the early phases of processor design, and also analyze how those issues have been resolved to enable today's fast and efficient microprocessors. Then, we study disk scheduling and I/O scheduling techniques in operating systems for single or multiprocessor environments. We mainly concentrate on mapping traditional operating system concepts to widely distributed environments in terms of scheduling and I/O management. Since small and large scale systems have similar problems, we are planning to utilize approaches in computer and operating system kernel architecture to come up with a broader perspective in which we can extend known methodologies to be used in distributed systems.

REFERENCES

- [1] Balman, M. and T. Kosar, T. *Data Scheduling for Large Scale Distributed Applications, Proceedings of the 5th ICEIS Doctoral Consortium*, In conjunction with the International Conference on Enterprise Information Systems (ICEIS'07), Funchal, Madeira-Portugal, June 2007.
- [2] Kosar, T. and Livny, M. *Stork: Making Data Placement a First Class Citizen in the Grid*. In In Proceedings of the 24th Int. Conference on Distributed Computing Systems, Tokyo, Japan, March 2004.
- [3] Hey, T. and Trefethen, A. *The Data Deluge: An e-science perspective. Grid Computing: Making the Global Infrastructure a Reality*. Chichester, UK: John Wiley & Sons, Ltd., pages 809-824. 2004
- [4] Kosar, T. *A new paradigm in data intensive computing: Stork and the data-aware schedulers*. In Challenges of Large Applications in Distributed Environments. HPDC 2006.
- [5] Laure, E. *The EU Datagrid Setting the Basis for Production Grids*. Journal of Grid Computing, Springer, 2(4). 2004.
- [6] J. M. Stokes. *Inside the Machine - An Illustrated Introduction to Microprocessors and Computer Architecture*. No Starch Press, December 2006, 320 pp.
- [7] O. Wechsler. *Inside Intel Core Microarchitecture - Setting New standards for Energy Efficient Performance*, 2006.
- [8] M. J. Flynn. *Basic Issues in Microprocessor Architecture*. J.Syst. Archit., 45(12-13):939-948, 1999.
- [9] J. L. Hennessy and N. P. Jouppi. *Computer technology and Architecture: An evolving interaction*. Computer, 24(9), 1991.
- [10] Ultra Sparc III Cu Processor - Overview, Sun Microsystems, 2005.