DataMesh parallel storage servers

John Wilkes, Chia Chao, Robert English, David Jacobson, Bart Sears, Carl Staelin, Alex Stepanov

Concurrent Systems Project, Hewlett-Packard Laboratories, Palo Alto, CA
wilkes@hplabs.hpl.hp.com

DataMesh is a research project investigating fast mass storage systems [HPL–DSD–89–44]. Our emphasis is on high performance I/O, while maintaining high availability, scalability, and ease of use. Our initial target is file servers for groups of high-performance workstations, although our approach is applicable to several different problems.

Hardware architecture

The DataMesh hardware architecture is that of an array of disk nodes, with each disk having an embedded 20 MIPS single-chip processor and 8-32MB of RAM. The nodes are linked by a fast, reliable small-area network, and programmed so that the ensemble appears as a single storage server. The first hardware prototype is operational now (software is still being developed); we hope to have the second, full-performance prototype about a year from now.

Software architecture

DataMesh is basically a software project: our goal is to exploit the hardware architecture at our disposal to develop fast, parallel storage servers. To that end, we are pushing hard on a few key ideas:

- *parallelism*: using up to a few hundred disks to achieve both high bandwidth and low latency
- *divide and conquer*: using specialized algorithms and policies for different types of workload
- *adaptive policies*: automatic workload recognition, and selection of policies to meet its needs
- *global resource management*, e.g. of the cache RAM available in the DataMesh nodes

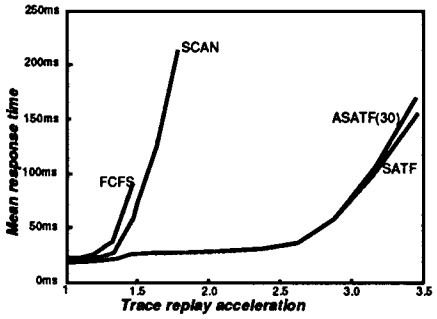
The layered architecture we have developed corresponds roughly to the phases through which the DataMesh project itself will pass:

	Function	*Interface*	*Layer*
1	smart disks	SCSI	virtual devices
2	file server	AFS/Novell	chunk vector
3	database	(O)SQL	access methods
4	programmable	??	(all)

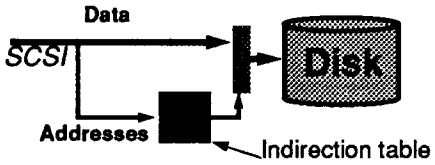
DataMesh phase 1 results

The SOSP presentation touched briefly on three ideas we have developed and explored as part of the "smart disks" phase of the DataMesh project:

- 2D disk scheduling [Seltzer90b, HPL–CSP–91–7]. Seek times on modern disks are much smaller than they used to be by comparison to rotational latencies. This suggests that existing algorithms that emphasize cylinder ordering for requests are going to be sub-optimal, and this is indeed the case:



- Loge indirect disk [English92]. Reserving a small amount (3–5%) of the disk as "free blocks", and directing a write to the next rotationally-available free block (thereby freeing up the slot where the old copy was stored), means that it is possible to develop a device that appears externally just like a pure SCSI disk drive, but which produces very much better write latency.



- Disk shuffling [HPL–CSP–91–30]. Loge is much better at writes but slightly worse at reads than a regular disk. But by using the observed access pattern to the stored data, and rearranging the data to optimize future accesses of the same type, it is possible to reclaim the lost performance.

References

[English92] Robert English and Alex Stepanov, "Loge: a self-organizing disk controller". Accepted for presentation at Winter USENIX, Jan 1992.

[HPL–CSP–91–7] David M. Jacobson and John Wilkes. "Disk scheduling algorithms based on rotational position." HP Labs Technical Report, 24 Feb. 1991.

[HPL–CSP–91–30] Chris Ruemmler and John Wilkes. "Disk shuffling." HP Labs Technical Report, Oct 1991.

[Seltzer90b] Margo Seltzer, Peter Chen, and John Ousterhout. Disk scheduling revisited. *Proc. Winter 1990 USENIX Conference* pp 313–23, Jan 1990.

[HPL–DSD–89–44] John Wilkes. "DataMesh—scope and objectives: a commentary." HP Labs technical report, July 1989.