

High-Speed NFS

Diploma Thesis

March 1, 1999

Ivo Sele

Department of Computer Science

Institute of Computer Systems

Prof. Thomas Stricker

Supervisor:

Felix Rauch

Contents

1	Introduction	7
1.1	Motivation	7
1.2	Performance Gap	7
1.3	Problem Description	8
1.4	Outline	8
2	Overview of the NFS Protocol	9
3	NFS in Linux	11
3.1	Client Side	11
3.2	Server Side.....	12
3.2.1	Universal NFS Daemon.....	12
3.2.2	Kernel NFS Daemon	13
3.3	Interaction with other Parts of the Kernel.....	14
4	Networking Software Overhead	17
4.1	Profiling Library	17
4.1.1	Time Stamp Counter.....	17
4.1.2	Performance Monitoring Event Counters.....	17
4.1.3	Profiling Functions	17
4.1.4	Measuring Idle Time.....	19
4.2	Measured Operations.....	19
4.2.1	Network Interface Driver.....	19
4.2.2	IP Layer	20
4.2.3	UDP Layer.....	21
4.2.4	RPC Layer	21
4.2.5	VFS Layer.....	21
4.3	Profiling Overhead.....	21
4.4	Benchmarking Programs	22

5	Measurements	23
5.1	Experiments	23
5.2	Factors and Levels	23
5.3	Fractional Factorial Design.....	25
5.4	System Configuration.....	25
5.4.1	Hardware.....	25
5.4.2	Software.....	26
6	Results	27
6.1	Influence of Different Factors on Performance.....	27
6.1.1	Throughput	27
6.1.2	Latency	31
6.2	Operation Times	34
7	Timing Issues.....	43
8	Suggested Improvements.....	45
9	Conclusions	47
10	Acknowledgements	48
11	Bibliography	49
12	Appendix	52
12.1	Original Problem Description.....	52
12.2	Profiling Library Source.....	55
12.3	Experimental Design	59
12.4	Linux 2.1.127 Memory Management Patch.....	60

List of Figures

Figure 1: Kernel layers involved in NFS transfers.....	15
Figure 2: Read throughput for various configurations.....	28
Figure 3: Read latency for various configurations.....	32
Figure 4: Accumulated client side operation times for experiment 4.....	36
Figure 5: Accumulated server side operation times for experiment 4.....	36
Figure 6: Accumulated client side operation times for experiment 24.....	37
Figure 7: Accumulated server side operation times for experiment 24.....	37
Figure 8: Accumulated client side operation times for experiment 17.....	38
Figure 9: Accumulated server side operation times for experiment 17.....	38
Figure 10: Accumulated client side operation times for experiment 9.....	39
Figure 11: Accumulated server side operation times for experiment 9.....	39
Figure 12: Accumulated client side operation times for experiment 10.....	40
Figure 13: Accumulated server side operation times for experiment 10.....	40
Figure 14: Accumulated client side operation times for experiment 8.....	41
Figure 15: Accumulated server side operation times for experiment 8.....	41
Figure 16: Accumulated client side operation times for experiment A1.....	42
Figure 17: Accumulated server side operation times for experiment A1.....	42

List of Tables

Table 1: Influence of factors on throughput	27
Table 2: Influence of other factors on throughput (excl. block size).....	29
Table 3: Influence of combined effects on throughput (excl. block size)	30
Table 4: Influence of factors on latency (incl. disk reads on server).....	31
Table 5: Influence of factors on latency (excl. disk reads on server)	33
Table 6: Influence of combined effects on latency	34

For many applications, especially when large amounts of data are retrieved from a server and only simple operations are performed on the data, high NFS-throughput is crucial for achieving good performance. Nowadays, high-speed networks are readily available and affordable. Standard PCs and operating systems are far from being able to fully take advantage of very fast networks like Gigabit-Ethernet though. In this thesis, we will concentrate on network transfer of data. The transfer of data to and from the server's storage devices can also be a serious bottleneck, but this can at least partially be circumvented by using disk arrays (RAID) and/or large memory caches.

1.2 Performance Gap

DRAM cycle time is improving very slowly, decreasing by about one-third in 10 years [21]. Bandwidth per chip increases as the latency decreases. Additionally changes to the DRAM interface have also improved the bandwidth. Processor speed improves by about a factor of 1.5 per year or a factor of 60 per decade [22]. Caches are widely used to help bridge the growing gap between CPU and DRAM performance. The advances in communication systems have been even more impressive: In the 1970s data communication links offered a bandwidth of 50 kbps. Twenty years later system optical communications offered bandwidths of 1 Gbps [20], a gain of approximately a factor of 130 per decade. There is no reason to expect that network speeds will not continue to increase rapidly in the future.

Indeed the bandwidth of modern networks is increasing far faster than memory bandwidth, copying of data in memory after receiving it from or before sending it over a network is a very expensive operation. When the operating systems which are widely used today were designed, networks were slow and thus the efficiency of the networking software was not of great importance. Frequently data is copied several times between different layers of the networking software. With current high-speed networks the situation has changed drastically. Nowadays the trend goes towards single and zero-copy implementations in order to alleviate this bottleneck.

Network protocols develop affected by the rapid improvement in networking technology. A protocol that was designed for slow links with high error rates will usually