Filesystem Backup Performance at the University of Utah Mathematics Department

Nelson H. F. Beebe
Center for Scientific Computing
University of Utah
Department of Mathematics, 322 INSCC
155 S 1400 E RM 233
Salt Lake City, UT 84112-0090
USA

Email: `beebe@math.utah.edu`, `beebe@acm.org`,
`beebe@computer.org`, `beebe@ieee.org` (Internet)
WWW URL: `http://www.math.utah.edu/~beebe`
Telephone: +1 801 581 5254
FAX: +1 801 585 1640, +1 801 581 4148

11 October 2001
Version 3.0

1 Introduction

This document contains graphical results of filesystem backup performance from logs collected daily for about 1400 days from Spring 1998 to Fall 2001.

Our site is a large and heterogeneous one, with more than 200 UNIX (Compaq/DEC, GNU/Linux, SGI, and Sun) and Macintosh systems. There are no Microsoft Windows systems, and consequently, we do not have that backup headache.

Because this document reports real data on real systems doing real work (teaching, research, administration, and software development) in an academic environment, the experiences reported here may be of interest to other sites who are engaged in planning, or justifying, backup strategies.

Over the last fourteen years, our backups have moved from nine-track tape, to QIC tape, to 4mm DAT, to 8mm Exabyte, to DLT (digital linear tape). Networked file servers have grown in power from early 25MHz Motorola 68020 systems to much faster systems, such as quad-processor 400 MHz Sun UltraSPARC Enterprise 5500 and dual-processor 600 MHz Intel Pentium III servers.

Since 1994, backups have been managed by the freely-available amanda (*Advanced Maryland Automatic Network Disk Archiver*) software. amanda directs the simultaneous backup of (usually) one filesystem per server: backup savesets are compressed on each server, and then transferred across the network to the local backup machine, where they are stored on holding disks. As soon as each saveset is complete, amanda adds it to a queue of files to be sent to the output tape. Ideally, the tape drive should operate in streaming mode for the entire transfer, but server load sometimes precludes this. Savesets are deleted from disk as soon as they have been successfully transferred to tape.

All dumps are written to tape with software compression (GNU gzip). Tape drive hardware compression is never used, because it does not permit the final data size to be estimated in advance, and because hardware-compressed tapes might not be interchangeable between drives from different vendors.

Each nightly backup run is a combination of incremental and full saves; the average at our site for 1999–2001 is about ten DLT 7000 tapes for a complete backup cycle. This corresponds to about 262GB from 26 filesystems, or 150GB after compression on holding disks.

2 Data compression

Data compression sacrifices CPU cycles (either on the backup server, or in the tape drive) to conserve tape capacity and reduce tape write time. At current prices of about US$60 per DLT 7000 tape, one year's supply of backup tapes represents an investment of about US$22,000, an amount comparable to our twenty-tape robot system with dual DLT 7000 tape drives. Ideally, we would like to keep backup tapes indefinitely, but financially, that has not been feasible, so for several years, we have provided a one-year window. Files lost before the window are not recoverable. Since faculty are sometimes absent for a sabbatical year, this window really should be larger, so in 2001, we increased it by six weeks.

In Figure 1, and all following ones, the thick continuous line is a cumulative average, and each plotted point represents one daily backup statistic.

As expected, evolution of hardware technologies for file servers, networks, and tape systems has not had much effect on compression ratios.

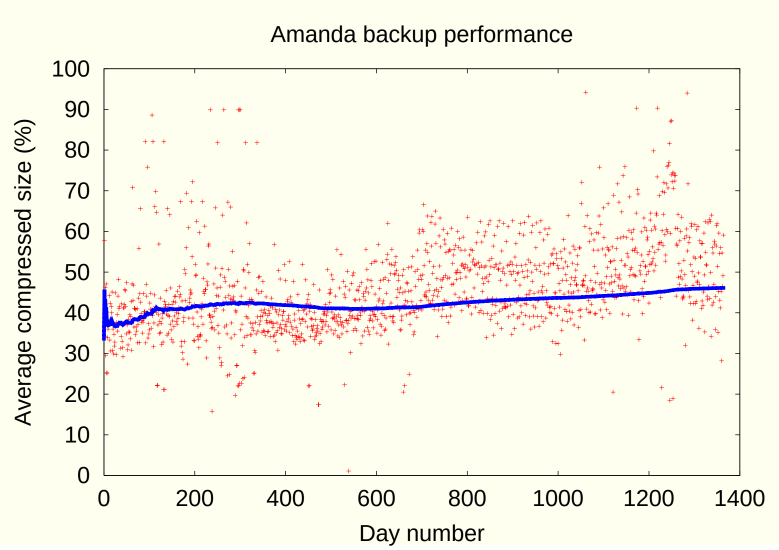

Figure 1: Software (`gzip`) compression efficiency of backup savesets.

The worst-case numbers in Figure 1 are highly relevant: they must not exceed tape capacity, since the backup software is not capable of splitting savesets across tape volumes.

Curiously, two decades ago, when nine-track tapes were common, backup systems routinely did this. Sadly, in the UNIX world, such support is rare, perhaps because some types of tape drive technologies have been incapable of reliable signalling of, and recovery from, end-of-volume conditions.

3 Data size

The plots in Figure 2 show how much data is backed up at our site, and how much tape space is used. The growth reflects increasing system size, rather than technology changes. As noted in the last paragraph of Section 1, the true filesystem size is about ten times the numbers on the vertical axes of these plots.

The upgrade, at about day 350, from 8mm Exabyte to DLT 7000 tape is dramatically evident in both these plots: DLT tapes have higher capacity.

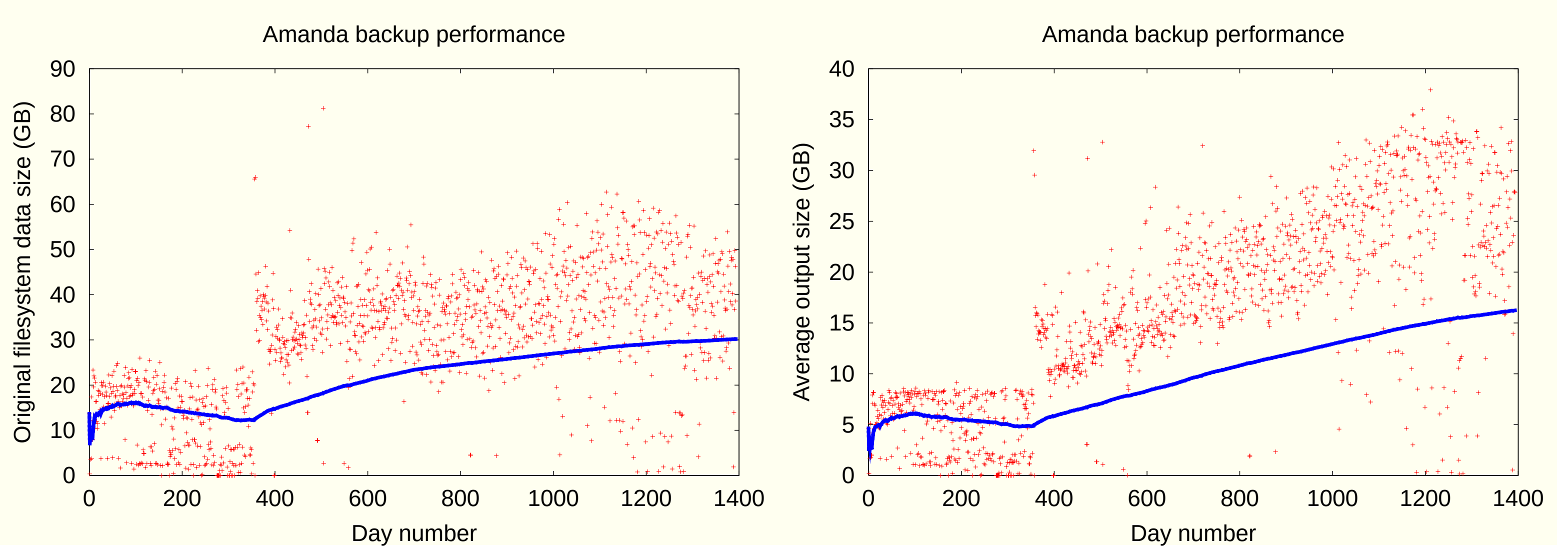

Figure 2: Filesystem size (GB), and backup saveset size (GB) with software (`gzip`) compression.

4 Data rates

The tape units installed in January 1999 are DLT 7000 systems, with a nominal capacity of 35GB/tape, and a 5 MB/s (17.6GB/hr) data rate. They replaced an 8mm Exabyte system, which was increasingly loaded and failure prone: we wore out about one 8mm drive a year, and had to retire several tapes each month because of data errors.

The plots in Figure 3 show the data rates for remote-server-to-backup-machine-holding-disk, and backup-machine-holding-disk-to-tape operations.

The first of these plots has a lower day count than the others, but spans the same interval: the backup logs sometimes lacked dump rate data, so those days were simply skipped.

In the first plot, the significant increase in dump rates marks the installation of a large Sun Enterprise 5500 file server with dual RAID filesystems, and the gradual migration of filesystems from older servers to it.

The second plot is perhaps the most interesting of all: it shows a *nine-fold increase* in tape writing speed when we moved from 8mm Exabyte to DLT 7000 tapes, while still using the four Sun SPARC 20/512 fileservers, and then, when the Sun Enterprise 5500 was installed, a further 10% increase, and a significant reduction in the variation.

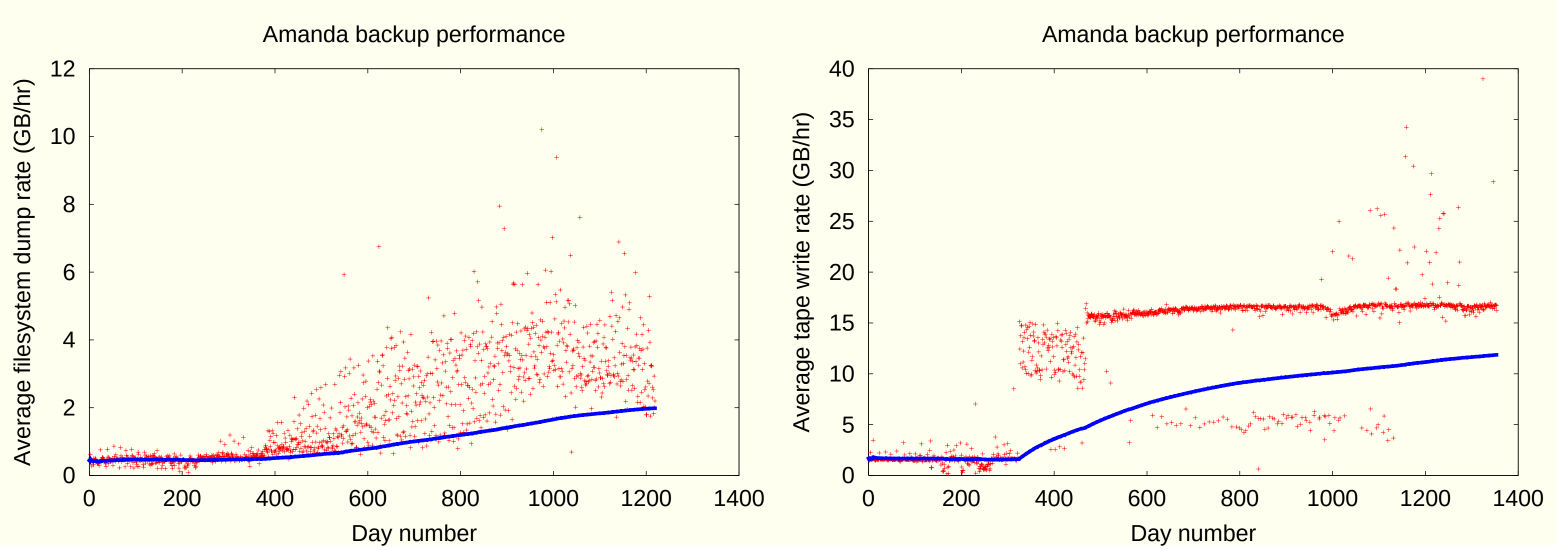

Figure 3: Disk filesystem dump rate (GB/hr), and tape write rate (GB/hr).

The older backup server often was unable to keep the tape drive streaming, so tape write performance suffered. With the new backup server, and DLT tapes, we consistently write about *15GB/hr to tape*, which is 85% of the DLT 7000 capacity.

5 Saveset counts and tape usage

The plots in Figure 4 show how many backup savesets are written, and what fraction of the tape is used.

The fall in the number of filesystems beginning about day 500 reflects the upgrade from four Sun SPARC 20/512 fileservers to a single Sun Enterprise 5500 with dual RAID filesystems and larger filesystem partitions.

The abrupt fall in the percent of tape used beginning about day 350 marks the switch from 8mm Exabyte tapes to the larger capacity DLT 7000 tapes. When the percent utilization exceeds 100%, this simply means that multiple tapes must be written. Since the nightly backup only wrote one tape, any data remaining on the holding disk had to be manually flushed to an additional tape by a systems person: we are certainly glad to be rid of that tedious task!

Starting about day 1100 (Fall 2000), the graph shows several in-

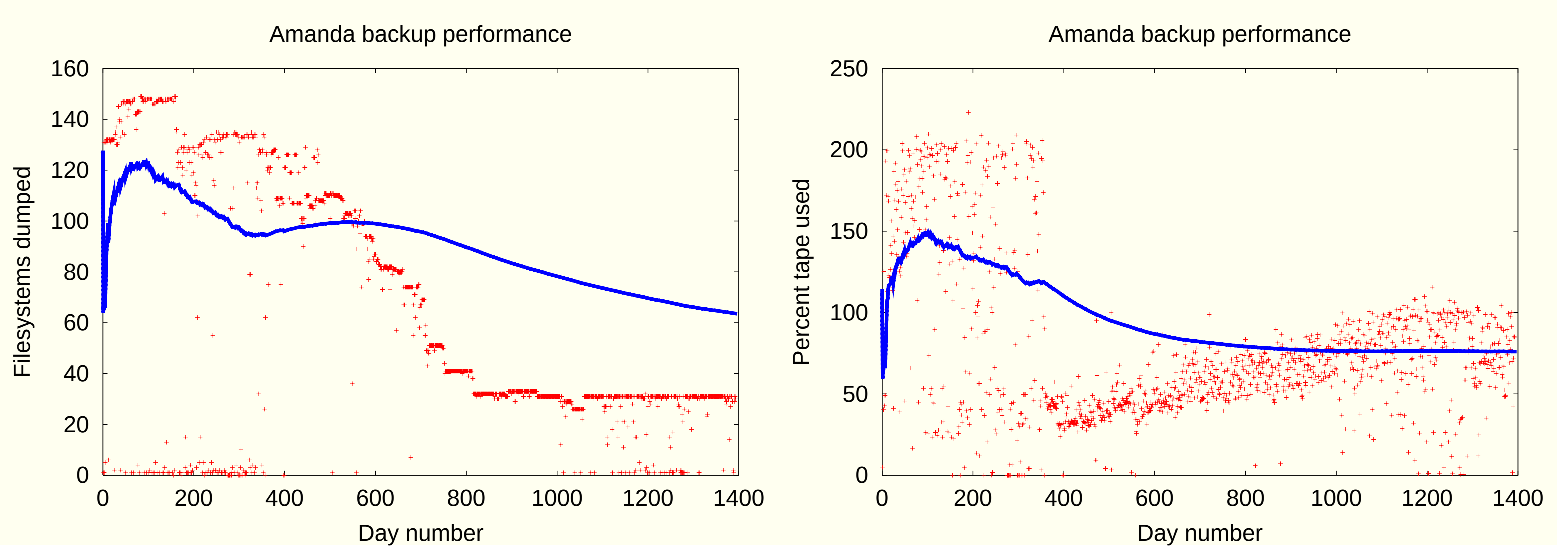

Figure 4: Filesystems dumped, and percentage of tape used.

stances where more than a single tape was required, and a general upward trend in tape utilization. Evidently, we have reached a point where either two tapes will have to be written daily (which undesirable doubles our already-substantial media costs), or where we have to stretch the backup cycle beyond ten days, or ultimately, we have to move to a higher-capacity tape technology. The only zero-cost option here is increasing the backup cycle period.

6 Dump time

By definition, a daily dump must complete within one day. The final plot, in Figure 5, shows that that goal was sometimes not reached, with a worst case in the final weeks of our 8mm Exabyte system of two and a half days. When a file server has to spend the entire day backing up files, all users who need files and other services from it are severely impacted.

From the plot in Figure 5, the upgrade to the Sun Enterprise 5500 has largely removed the overload problem. The fifteen- to twenty-hour peaks from about day 400 onward are attributed to two older Sun SPARC 20/512 servers (one with 18GB, 9GB, 2GB, and 1GB disks, and the other with 9GB, 9GB (not backed up), and 2GB disks), plus an SGI Origin 200 with 24GB, 18GB, and 2GB disks.

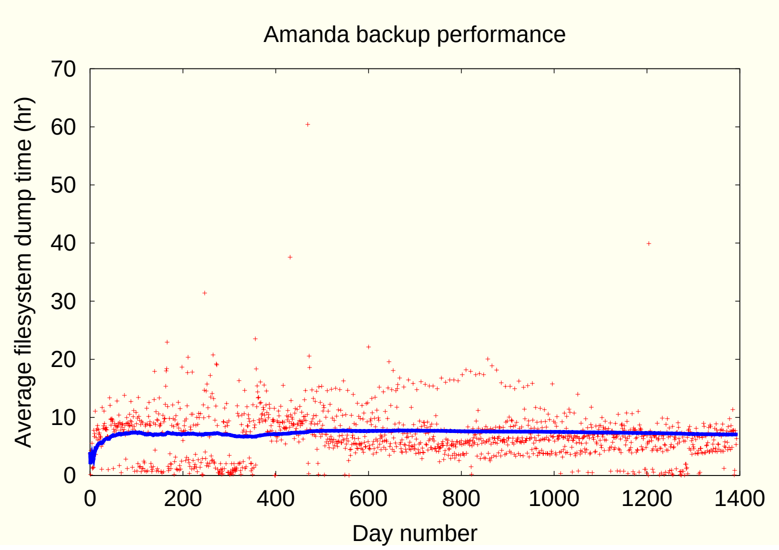

Figure 5: Wall clock dump time (hr).

On day 1393 [12-Oct-2001], we replaced the SGI `gzip` with one recompiled with the native `c89` compiler using optimization options that gave the fastest executable from more than a score of option choices, to try to reduce the backup time. The new executable is about 40% faster than the old one compiled with `gcc`. A similar experiment on Sun Solaris produced a new `gzip` that is 6% faster than previously.

The Sun Enterprise 5500 has four CPUs, two 100 Mb/s Ethernet interfaces, and dual RAID filesystems (UltraSCSI and FibreChannel), so even when it is busy writing data to tape, it still has plenty of CPU, network, and filesystem capacity to provide prompt file services for our users.