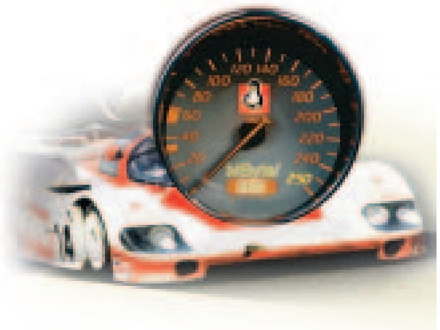
Data back up on the network

BETTER SAFE THAN SORRY

ALBERT FLÜGEL, OLIVER KLUGE



Even the most reliable hard disk will give up the ghost one day. Only by making regular backups can you protect your data against the worst case scenario. The following overview sheds some light on the various strategies.

Effective data backup makes sensible data management necessary, especially in view of the explosive growth in the size of files. Modern data back up is much more than just copying data onto a tape cassette. It concerns not only the selection of backup software and hardware, but also the configuration of the data server and the behaviour of the user.

Most users would rather not need worry about technology themselves; they take the attitude that the administrator should handle those kinds of problems on their own. But disk space fills up sooner rather than later. Co-operative users can contribute to clarity by tidying up, compressing and packing.

Long term archiving

Offline storage of data should be distinguished from a backup. When it comes to back up, the usual assumption is that only data that is fairly recent (say, three months old) needs to be restored. However, this does not apply to archiving. Candidates for archiving are those files containing data that is not currently needed but will (or could) become important again at a later date. Archives – possibly existing on several media as clones – should be part of your standard repertoire when it comes to managing data on a computer network. This function does not necessarily have to be performed by the backup system in use. A *tar* or *cpio* on several redundant tapes with labels is usually enough.

Hierarchical storage management

A very exciting technique, well worth discussing, is HSM, such as the one employed by Veritas or SAM-FS from LSC. This involves copying data at a predefined interval from the hard disk onto slower and cheaper media. This step is referred to as 'archiving'.

Data which is not accessed for a certain length of time is removed from the online medium (or 'released'). The file system entry is retained, but the data blocks disappear.

Later accesses to data which is no longer on the hard disk lead to staging. This means that the data is retrieved, and made accessible, completely automatically from the slower media, which can of course take some time. HSM cannot be realised by the file system without support, as the procedures described are intended to be invisible to the user processes and/ or their system calls.

HSM in its simplest form does not replace back up. If the data has gone from the online medium, it now only exists in a single copy. If the tape on which it is stored then fails, it will have to be restored from somewhere else. HSM systems therefore offer the option of producing several copies at once.

Backup scope

The results of your daily work must be backed up. But certainly, the backup capacities should not necessarily be stuffed full of things which are still available on CD or the Internet. This includes such things as operating systems. But a computer environment will not normally be equipped with unaltered system installations. Adaptations to the respective requirements are always necessary.

The software should, apart from the back up of all data, also be able to handle the *incremental* mode. This means that only the files that have changed since the last back up are written into the backup.

Often, additional back ups of the type Level-N are also possible with N = 1, 2, etc. With a Level-N back up, all data which has changed since the last back up with the same level is backed up. As you might imagine, a total back up is Level 0 and an incremental is Level *infinite*.

With a Level-3 back up everything new since the last Level-3, Level-4, or an incremental back up is backed up. The software *Afbackup* by this author evaluates it differently, so that one can be open to higher levels.

Typically, a complete back up is done at the weekend with incremental back ups every night. Another option is complete back ups every first weekend in the month and a Level-1 back up on the other weekends with incremental back ups each night. The longer it is since a complete back up, the longer restoring is likely to take. In principle, no backing up should be done while lots of people are working, since this represents a considerable load for the computers and the network concerned.

It may be desirable to back up several computers at the same time. But if the data is being sent to just one single backup server or a tape drive, the backup software must support this type of operation. If several computers have a lot of data, a parallel start, especially of incremental back ups, is very useful. Another example is that of many backup clients, which back up on a central server via slow lines, but in this case you only benefit from parallelisation, when the flow rates can add up on the server.

Tapes do break now and then. So quite a few administrators tend to configure the use of a new tape for each full back up. In this way, there is always a complete backup available, which was done not more than two full back ups ago, even if a single tape does fail. Multiple backups of the same data on various media can benefit the user, apart from the higher level of redundancy.

If one configures just one full back up on disks (in the case of backups on tape stored for a long time), the current files can be restored more quickly from the hard disks with security at the same time.

Getting it taped

Considering current prices of hard disks it's worth thinking about storing data on hard drives. The throughputs achievable, even with slow disks, are higher than with the usual tape technologies. Nevertheless, there is a considerable price difference in favour of tapes: A DLT with approximately 35GB capacity without compression costs about £50, and you would not get a disk of that capacity for the same money. Tape changes are also easy to automate.

Tape technologies

Before buying a tape drive or a changer, there are some tough choices to make. Many technologies try to court the buyer. The main ones are presented below. The choice of one of the technologies described should primarily be made on the basis of the amount of data to be backed up, rather than on the price of the drives and tapes.

Quarter inch cartridges

QIC now plays a very small role in systems management, but in private use there are still plenty to be found, because the drives are especially cheap and offer acceptable capacities for home users.

QIC makes serpentine linear recordings, the tape is drawn at high speed past the head, and as soon as the tape comes to the end, the head is lifted and the whole tape is run through again. So with an economical system you get both capacity and speed. But when buying such drives, make sure they can cope with read-after-write for the sake of data security.

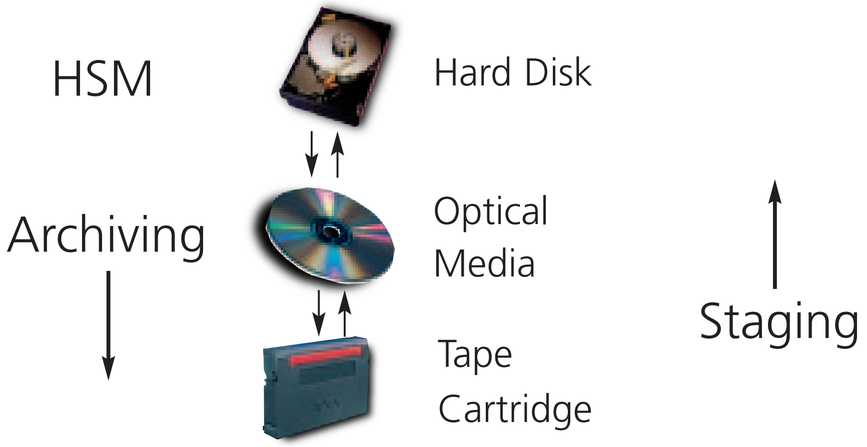


Figure 1: Overview of hierarchical storage management

Exabyte

Derived from the technology of Video-8, the tapes were seen as susceptible to wear and tear due to the narrow tape guides, loose head contact and the resulting strain. In newer products these problems are supposed to have been corrected, but there are others. If you insert a tape to be read into a drive with a different construction, the reading does not always work. This happens even with drives from the same manufacturer.

This is nothing to do with the typical problem of correct adjustment of block sizes, which often crops up in newsgroups. If a drive does fail there should thus be a matching replacement within reach. Exabyte is now achieving capacities of up to 60GB per tape (uncompressed).

DAT

The capacity specifications for DAT are usually worked out using unrealistically high compression rates. In practice, it is only the uncompressed value that is relevant. Because of the comparatively large wastage, as a result of bad spots on the tape (drops), the real quantity achieved is normally less. Plus, to make things harder, DAT drives to the DDS-3 standard usually recognise and report contamination of the head too late.

A phenomenon which is occurring increasingly often is that the markings on the tape are overlooked in a fast search. This can lead to data not being found or in the worst case, parts of the tape being overwritten unnoticed. From DDS-3 on, the head is cleaned automatically in the drive during relatively frequent use.

But this should not result in any excessive wear. It is strongly recommended that you keep to the cleaning intervals with DAT advised by the manufacturer in the accompanying documentation (but not exceed them). DAT can theoretically handle, with DDS-4, 20GB uncompressed.

Digital linear tape

DLT has been developed for high densities, low mechanical wear and high recording speeds. There

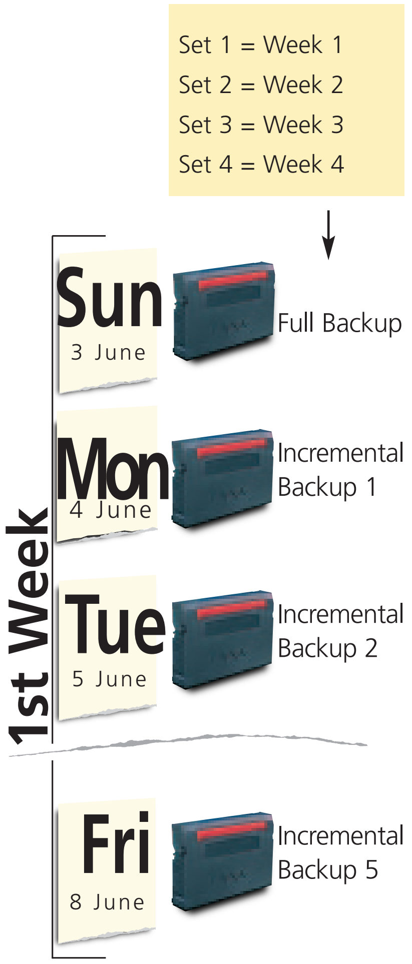


Figure 2: Backup strategies

can be problems from time to time with the tape getting out of line; the second spool is in the tape drive. The start of the tape can come out in sympathy, as the result of which it becomes unusable. The start of the tape contains information managed by the drive. If this cannot be evaluated, the drive will not even accept the tape when it is inserted. For this reason, in the AIT technology from Sony a writeable chip has been built into the cassettes.

The way to data back up

One simple and effective variant is to connect the drive directly to the respective file server. This also means there is no load imposed on the network and there are no security worries with respect to data being overheard. But if the server goes up in smoke the tapes cannot be saved. This problem can be mitigated somewhat by regularly taking them out and storing them elsewhere. The crucial question is what periods without backing up are acceptable.

If you want to guarantee security even if the building collapses, online data and backup must be geographically separated. The administrator achieves this by means of back ups over the network or the use of a suitable bus technology between computer and drive.

The commonest way is to back up via a network to a computer which then acts as backup server. If this is not to impose a heavy load on the network via which the computers of the users are being operated, you could consider the option of an additional network connection between the two computers. If security is an important aspect, all the typical problems in network services are relevant.

Correct access rights

Can the backup data only be read and written by those authorised to do so? Are there back doors into the system resulting from the architecture? Can bugs (such as buffer-overflows) lead to unauthorised access? In any case, the permissions of the devices must be tested in */dev*.

Normally everyone has writing permissions on tapes; even big-name backup products work like this or give no instructions in the documentation. If it is not possible to limit permissions here without the backup software refusing to work, you must consider barring the backup server to any login by normal, potentially malicious users. Of course, this consideration does not apply only to backing up via the network.

Storage area network back ups

Another option has been becoming fashionable for some time: Backup in a storage area network. SAN means that there is not only a connection between a computer and mass memories as on a SCSI bus, but that several computers with several mass memory systems – possibly also via several redundant paths – are networked. In this way, fast connections from all connected devices can be used as alternatives, similar to the communication between computers in a LAN.

Backup devices (usually jukeboxes) can be connected to a SAN. The back up of the data then runs, not via the file server, but direct from the online mass memory to the backup system. In this data transfer, neither the file server computer nor the network outside the SAN is put under strain.

But since most data is backed up from a file system, the controlling software is given an additional task: neither the mass memory nor the backup device know the file system structure. This information is in the exclusive possession of the file system driver in the server operating system. If a file is backed up, the mass memory is informed which blocks it should send to the backup device. The restore function is more time-consuming: The hardware components involved in such installations are in the rack format and the software is expensive. There is no way that someone who wishes to invest in such a solution will be able to avoid working out his own individual strategy.

Here is a brief sketch of one other variant: There are devices (for example Celerra from EMC^2, Server from Network Appliance or devices from Transtec), which combine mass memory and logic in one housing, so that on the network they appear as a pure fileserver (network attached storage). They typically offer no other services and nor can one log on. If their back up does not run via the file service in the network, then there is still the option of connecting drives and changers directly to these devices. Backup software on the devices themselves and control software on a computer in the network (NetApp NFS-Server and Veritas NetBackup) then enable the back up.

When backing up via NFS-Mounts at least one read-only-root-export must be available at the time of back up, as otherwise read-protected data is not backed up. When restoring, root must even be able to write via NFS. Since a forged UDP packet with the sender of the NFS client is all it takes to manipulate data on the NFS server, this is a potential security risk.

Handling the media

If the quantity of data to be backed up at one go is greater than the capacity of a tape, you ought to acquire a changer (stacker, jukebox or a tape library). The simplest stackers for example have a drive and six compartments or slots for tapes that the robot can then change. Large jukeboxes have a hundred slots and six or more drives. Frequently there are several loadports or loadbays as well. They considerably alleviate the work of the administrator when assembling the device.

With respect to the backup software, on security grounds, one should find out from the manufacturer whether the changer is supported. Usually however, changers implement at least one subset of a standard protocol, with which the hardware can be driven via

Info

Website on afbackup incl. general HOWTO & FAQ:
http://www.afbackup.org
ftp://www.vic.com/af
Yellow Guide at Transtec:
http://www.transtec.co.uk -> Guide -> Mass Storage
General details on back up and storage:
http://www.backupcentral.com
Overviews of backup software for Linux:
http://linux.tucows.com/conhtml/adm_backup.html and http://www.linux.org/apps/all/Administration/Backup.html
Legato homepage (Product »Networker«):
http://www.legato.com
Info on Budtool (now also belongs to Legato): http://www-ftp.legato.com/Products/html/budtool.html
Info on BRU:
http://www.estinc.com/bruinfo.php
Veritas homepage:
http://www.veritas.com/uk/products/
Websites on ADSM from IBM:
http://www.storage.ibm.com/storage/software/adsm/adsmhome.htm
EMC^2: http://www.emc2-uk.co.uk/products/networking/
Networking appliance:
http://www.netapp.com/

■

the SCSI bus. In terms of software, it is also possible to use the programs available in source code *mtx* or *stc* (for Solaris).

Software

Freely available packets such as Amanda, Burt or Afbackup by the authors are just as interesting as commercial software.

In principle, when it comes to choosing software, the same rules apply as with all other products. Anyone who believes what a manufacturer says without having verified the facts in a test is taking a risk. You should always conduct a test installation, in which you should test with marginal conditions which are as realistic as possible. The problems that really hurt only come to the fore in conditions of higher complexity, using combinations of features or in connection with other components.

So far it has been tacitly assumed that certain functionalities will be present: These assumptions include the facts that a ‘verify’ (such as comparison of the content of the backup with the file system) is possible, or that when archiving (when the tape is subsequently read) such a comparison takes place. But this is not necessarily the case.

One fairly expensive backup and archiving product reads the tape following a back up and sends the data over the network to the computer from which the data originates. A comparison with the file system is not done, though. The evaluation of how important any advantage or disadvantage of a product is for the respective purpose is a decisive factor. If security features are important, one should not shrink from using *strace*, *tcp-dump*, *lsof*, *truss*, *snoop*, or other tools on the respective system.

Also, the permissions with which the software is installed must be tested. For example, if there is no Set-UID-bit in programs which users can start (this does not necessarily have to be Set-UID on root) and if the shared version of the *Libc* is used (test with *ldd*), then internally implemented access restrictions are pretty certain to be worthless. These are really easy to get round by redefining functions such as *get-uid* with the aid of the environment variable LD_PRELOAD.

Potential index problems

One typical quality of most products can turn out to be an Achilles’ heel: So that one can target specific data to restore and the systems administer an online index. This stores the entire structure of the backed up directory trees. With the appropriate program, users or administrators can navigate in the backed up data as in a file browser and make a selection for restoring.

If the same restrictions on rights are to be as effective here as when working in the file system, the rights, owners and ACLs ought to be in safe storage. Plus, information has to be managed, such as date of back up, storage location of the data and the flag, as to whether the file system entry can be restored.

Basically, a file system without datablocks is constructed here, but with additional information. The more entries there are in the file system to be backed up, the bigger the index. If there are only empty files or Symlinks in the original directory, this online index, which is also in a file system, cannot take up any less space than the original data. It is also subject to consistency requirements, like a file system.

This means that if a process which manipulates the index expires uncontrolled, the index can be inconsistent. Then it has to be tested and repaired. Thus run a type of *fsck*. In this case it has to be restored or else you will lose the option of navigating in the backup. Safe storage of the flags for selection in the index is a problem. This can mean that one cannot run parallel restores on the same client, although this is supposed to be theoretically possible. A selection in the restore front-end leads to another, previously made, selection being cancelled. This is seen by the fact that parts of the first restore are not restored, as the flags have been deleted in the meantime. ■