The EMAS Archiving Program

A. S. Wight

Department of Computer Science, University of Edinburgh, The King's Buildings, Mayfield Road, Edinburgh EH9 3JZ Scotland,

The Edinburgh Multi-Access System (EMAS) has been described (Whitfield and Wight, 1973). The service now (March 1974) supports up to 55 simultaneous users from an accredited population of 500. File storage is provided on a 700M byte disc-file.

Magnetic tape is used to provide backup and archive facilities for the file system. These facilities have been improved as use of the system expanded. Details of the use of tape as another level in the storage hierarchy are given and also included are outline plans to cope with two linked systems each with 700M bytes of disc-file and an ICL 4-75 processor.

(Received June 1974)

1. Introduction: statement of problem

The Edinburgh Multi-Access System (EMAS) and associated disc-based file system have been described (Whitfield and Wight 1973; Rees, 1975). As with all such systems there are problems of loss of information from the disc and pressure on disc space as users' files expand. This paper describes how EMAS uses magnetic tape to attempt to solve these backup and archive problems.

The literature contains a number of excellent expositions of the problem. Wilkes (1972) and Watson (1968) describe the problem in general. Two detailed descriptions of particular cases are Fraser (1972) and Considine and Weiss (1969). Wilkes distinguishes user-support and database systems. EMAS provides user-support facilities and although not explicitly providing for databases handles files up to 4 M bytes. Contrary to the recommended approaches of the above, backup and archive facilities in EMAS were not designed into the file system. We describe how with this approach we have adjusted to changing system performance and user needs without having to tamper with the file system, which was also in a state of flux. We now propose to experiment with a new design which should bring major improvements and as always the user sees the system improving all the time.

Since mid 1971 the number of accredited users has grown from 50 to 500. As described by Rees a user and all his files are assigned to one quadrant of the file system. Now (March 1974) there are three file system quadrants in use. Each quadrant has up to 80 per cent of its 40,000 pages (each of 4,096 bytes) allocated to user files. A quadrant caters for around 200 users. A user may have up to 120 files on disc depending on the size of his file index. This also dictates an upper limit on the number of disc pages his files occupy. Most users work within limits of 60 files and 1,600 pages. There is no global disc allocation control except the archiving described below. Files may be protected or unprotected. The default mode is unprotected. If they are protected then the backup system keeps copies on magnetic tape. The magnetic tape facilities which have been used are four 120 K bytes/sec nine track tape decks recording at 800 bpi on 2,400′ tapes. A full tape in the EMAS format holds around 4,000 pages. A separate backup and archive service is organised for each file system quadrant. However the dumping programs can be run to deal with:

(*a*) the file system
(*b*) a file system quadrant
(*c*) a user's files.

It is obviously convenient for the recovery program to handle one more level, i.e. a group of files.

The backup system has evolved through the following stages:

(*a*) dump all files daily
(*b*) dump all protected files daily
(*c*) dump protected and changed files daily, and all protected files weekly.

The archive system serves a number of needs:

(*a*) supplies cheap, secure storage
(*b*) allows users to have more files than their disc index will allow
(*c*) holds files which have been deleted from the disc because they have been unused for some time.

This helps to keep the allocated disc space balanced with the demand for more file space.

The archive system evolved through the following stages:

(*a*) dump unused, protected files
(*b*) destroy unused, unprotected files
(*c*) dump a file on demand (up to a week later in practice).

The next stages are to deal with the housekeeping of archived material as it expands and provide a service closer to backup and archive on demand, i.e. greater security but without overloading the system. Note that the total of backup material is much more stable unless another file system quadrant is brought into use.

There is a RESTORE command to allow users to retrieve files from archive tapes. This facility does not apply to backup tapes. The use of this command has been monitored to see the effect of a weekly archive dump and how often old archive material is used.

2. Users view

The EMAS file system and standard user subsystem have been described by Rees (1975) and Millard *et al* (1975). This section describes the effect of the backup and archive systems on what the user sees. Files on EMAS may be protected by having copies made on magnetic tape. The default condition is unprotected. If a user wishes a file to be protected he issues the command

CHERISH (file).
HAZARD (file)

restores the unprotected state. This means no more dumps will be made. However in keeping with the current dumping philosophy no attempt is made to record the fact in a backup dump so backup copies may still exist and the latest may reappear on the disc-file after loss of the current version. This may happen until all copies are destroyed as tapes are reused.

If information is lost from the disc file then the user may have lost unprotected files. For a protected file the restored copy may be up to 24 hours out of date. There is no automatic way a user can request a file from the backup tapes.

When the user finds files missing from his file index then the archive system has been at work. If an unprotected file has been unused for four periods then it is destroyed. In practice a period is a week. Similarly for a protected file, but a copy will have been made on two magnetic tapes. This also applies to files for which a user has requested archiving with the command

ARCHIVE (file).

To combat or cope with this situation the user is given two more commands.

FINDFILE (file)

allows him to enquire about his archive material (whether requested or automatic) and

RESTORE (file)

puts a copy of a file back on the disc and adds the name to the user's file index. The output from FINDFILE can be directed to a file so that the user can manipulate it and display it in forms other than the chronological ordering supplied. A user can also ask the administration to write a private tape.

In the case of files which have been *permitted* (Rees, 1975) to other users then these 'permissions' are dumped to tape with the file. A file restored from a backup tape has the 'permissions' restored. This is not done for an archive file nor does the fact that a file is permitted prevent destruction or archiving.

3. Backup

Each file system quadrant on EMAS is backed up independently. The backup program is run in an executive process (Whitfield *et al*, 1973) under operator control. This dumping is done with users running but overnight when the load is light. Any files in use which are open for writing are ignored. On a daily basis those files created or altered since the previous day's dump are copied to magnetic tape and the 'written-to' flag in the ARCH byte (Fig. 1) in the file index entry reset to zero. In addition once a week a dump is made of all protected files. If this quadrant is lost then all protected files can be put back by reading the daily dump tapes back to and including the most recent weekly dump in reverse chronological order. Copies of files other than the most recent are ignored. As a further precaution a number of these weekly cycles are kept. Fresh tapes are written each day. Tapes are not mounted with a write ring while they contain valuable information. After a dump the tapes are read as a further check. If daily tapes cannot be read then a complete weekly dump of all protected files is made. A list of what is on a tape is produced and the files on tape can be completely identified by reading the tape. No other records are kept nor are the file indices dumped. They record only what is on the disc and have no usefulness on tape until we decide to record all changes. When restoring from tape rebuilding the index is simple. No record is kept of a user destroying files so recovery may see 'dead' items reappearing on the disc. In the same way (see Section 2) permissions which have been revoked may be set up again if the permitted file is recovered from a backup tape. In the next more flexible versions of this system these situations will be improved.

4. Archive

The archive system is run in exactly the same way as the backup system but using more bits of the ARCH byte to drive it. Ideally it is done immediately before a weekly backup dump to prevent ARCHIVE material reappearing in a recovery situation. If the archive bit for a file is set then the file is copied to tape. The usage information on which the other archive actions are based is generated as follows. The rightmost bit of four is set when a file is accessed. Once a week, or whatever 'period' is chosen, these four bits are shifted left. So if a file is not used for four complete periods these four bits will be 'O'. The archive system destroys unprotected files with this pattern. Cherished files with the same pattern are copied to tape. Once the file has been copied to two tapes, a record added to the index of archive material and a line-printer index of the newly dumped material produced then the disc copy of the file is destroyed. Each archive run starts with fresh tapes. Material from previous weeks is not put at risk by mounting the tapes with write rings again. This obviously results in wasted tape space, especially now as we move to 1,600 bpi tapes. However the flexibility demanded for other reasons (see Section 7) and attempting to satisfy requests for archive on demand means that this problem must be solved.

The archive index is a file owned by the MANAGR process. The FINDFILE and RESTORE commands access this index on behalf of a user to list any entries required and to find their tape addresses. RESTORE sends a request to the VOLUMES process to have the appropriate tape mounted, the file read if the one found at the tape address has the same identification as the requested item and the name added to the requesting user's file index.

It has turned out without any 'tuning' that this system leaves each file system quadrant in a balanced state, i.e. each week the space created by archiving is sufficient to hold the files RESTORED and created.

5. Implementation

The programs for backup and archive run as part of the privileged executive processes MANAGR and ENGINR. So two file system quadrants can be dealt with simultaneously if enough tape decks are available. For the period covered by this report we have had four 9-track, 800 bpi, 120 K bytes/sec decks. See Section 7 for the effects of new hardware.

The following data is used:

1. List of users
2. Each user's file index
3. File belonging to MANAGR which contains an index to the archived material for this quadrant
4. Date and time supplied by system
5. Tape identifiers typed in by operators.

Apart from the tapes written, output is the updated archive indices, teletype monitoring of the running program and a line printer record of the files destroyed and written to tape.

A file is written to a tape as a CHAPTER. This is the standard EMAS tape format. A CHAPTER is an 80-byte header and a number of pages. A page block is actually 4,120 bytes (4,096 data +24 identifier). The tape is addressed as chapter and page within chapter. The backup and archive programs put in an extra page of information as the first of the file. This contains as much identifying information as possible and the list of permissions if there are any. The average file written to tape is eight pages. A 2,400′ tape holds up to 4,000 pages. The maximum size of a file is 1,024 pages. We do not split a file across tapes. Separate tape sequences are maintained for each file system quadrant and we do not add to tapes at the next dump so the average tape is around half full. See Section 7 for changes under way.

The, thankfully very rare, job of replacing a complete file system quadrant is done by reading the backup tapes in reverse chronological order up to and including the most recent weekly dump. If an individual file is required from a dump tape the material will be read from the tape position derived from the line printer records. All the standard programs will only hand over a tape file to its owner as recorded in the extra page with the file on tape.

S	A	U4	U3	U2	U1	W	C

Meaning for each flag if set.

C	protected
W	file connected in 'write' mode
U1	file connected i.e. used in current period
U2, U3, U4	usage over previous three periods
A	request for archiving via ARCHIVE command
S	spare

Fig. 1 ARCH byte of file index entry

For archive recovery the FINDFILE and RESTORE routines are part of the user subsystem (Millard, 1975) and use the information stored in the MANAGR files. These files are connected in shared mode in the user's memory. If a user were to detect this he could gain access to other users' archive records. Strictly this is a breach of security, and the information should be handled behind the system interface as represented by DIRECTOR (Rees, 1975). There are separate mechanisms to dump the supervisor and MANAGR files.

6. Operational experience

The backup and archive system described above has been in operation for 16 months. It was preceded by a much simpler one and will be followed by a more comprehensive and integrated one. This section describes how the system has coped with the demands made on it.

Users protected one half of their files. With an active population around 500 holding 70,000 pages spread over three file system quadrants this generated a weekly checkpoint dump of 35,000 pages (140 M byte). The daily dump of new and changed material on the five working days (a weekend service was not a regular feature) was around 5,000 pages (20 M byte). In practice this material has seldom been used. Tapes are recycled after a few weeks. No copies are removed from the building.

The archive system has been generating tapes for 16 months. None of this has been discarded, although users can mark files no longer required. On-line directories are kept. The material extends to 900 M byte and the directories occupy 1·4 M byte (roughly equivalent to the on-line file store of three users).

An average weekly run of the archive program destroys 2,500 pages of unused, unprotected material and transfers to tape 5,000 to 7,000 pages. About 80 per cent of this is unused and protected. The remaining 20 per cent has been requested by ARCHIVE commands during the past week. This figure is small and tends to be dominated by one user in any week. The result is to free up to 10,000 pages of disc space to cope with new and extended files over the next week.

The RESTORE command was monitored for a three month period to find what use was made of the archive material.

Number of users issuing requests	287
Average number of requests/user	14
Average number of requests/day	60
Average size of file restored	12 pages
Average time between archiving and restoring (i.e. 90 to 100 days since last used)	64 days
70 per cent of requests referred to files dumped in the previous month	
40 requests were for files over one year old	
20 separate tapes were required each day	
Average time from RESTORE command to file available	5 minutes

This is the aspect that users like most. It is convenient to have the system doing file housekeeping for one and to be able to retrieve migrated items quickly. Obviously some users write programs to access all their files and ensure they remain 'in use'.

As a sidelight on the control that archiving applies, when some users were recently transferred to the second 4-75 and initially archiving was not done one file system quadrant (150 M byte) was full within a month.

7. Historical development and planned improvements

EMAS as planned by the EMAP team (Whitfield *et al*, 1973) was to have an elaborate backup and archive system under the control of processes activated whenever action was required. This would have provided full checkpoint and incremental facilities. However the first working file system was simple and the backup and archive was restarted to develop in parallel with this and a user service.

The following were the development stages:

1. Copy the half disc-file in use to the free half
2. Dump daily for each user his listed protected material to his own tapes
3. Dump daily for each file system every protected file
4. Change the frequency of 3 to weekly and add a daily dump of created and changed files. This is the present system.

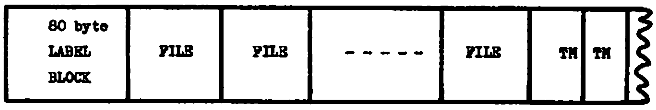
It is now planned to implement backup and archive in the style of the original proposals. The situation will be made more complex by the extended configuration of two 700 M byte disc files, two 4-75 processors and a Front-End Processor. The present system provides two levels of checkpoint dumps. This means a user may wait up to 24 hours for a dump and a week for an archive. This is too long. An improvement would be to dump more frequently and archive requested and unused material on different cycles, e.g. daily and weekly. We have always aimed to keep the load generated by these services low. Certainly the reliability of the disc-file has been a great help.

Apart from providing an incremental service and implementing it with a privileged process activated by user request or the system on an alarm clock basis the major changes are to cut the number of sets of tapes so that those in use are filled and to maintain a dump and archive index for each user. This will be maintained by the backup process and exist as another level of each user's current File Index and be moved with him if he is transferred between file system quadrants for administrative reasons.

This means that a database as opposed to user-support problems must also be solved, i.e. the many changes to the records in the new index must be secure. The backup for these changes is therefore very important. Tape material will still be self-identifying so that on-line indices can be reconstructed.

One final problem is the control and compression of the volume of ARCHIVE material.

In the new situation there are going to be new tape decks and new tape handling software. Instead of four decks and one



TM = Tapemark

FILE = TM, 1 page identifier, TM, file in page blocks

Fig. 2 Tape format

processor there will be four 1,600 bpi 120 K_bytes/sec decks accessible to two processors and supervisor will provide tape-handling primitives rather than a specified format. The dump and archive system will use the simple format shown in Fig. 2. The header will contain all the information previously held in the various identifying blocks. This change provides a convenient time to 'lose' all unwanted archive material. This may point to the cost-effective solution to the problem of archive explosion in general. Organisational and administrative considerations will outweigh any algorithmic results based on charging for space whether by explicit allocation or using expiry dates. Note that although the archive index may continue to grow this is not expensive as there is no automatic search of it if a specified file is not found in the user file index.

Conclusion

The previous sections have described a backup and archive system for a user support environment. It has grown to match an evolving system and user population. Having seen what users need it can now be changed to give an improved service in a more complex system situation. This would appear to be the best way to do things.

The above does not solve the problem in other systems, e.g. small configurations, RJE systems and any genuine database systems. Again an evolving system to match usage will almost certainly be better than one aimed at coping with all possible situations.

Acknowledgements

Many people have contributed to EMAS over its lifetime. The archiving component is no exception and acknowledgement must be made of the many helpful suggestions received. Particular mention must be made of the arch user Professor S. Michaelson and A. McKendrick and R. McLeod for their feedback from operational experience.

References

CONSIDINE, J. P., and WEISS, A. H. (1969). Establishment and maintenance of a storage hierarchy for an on-line data base under TSS/360, *AFIPS Conference Proc.*, Vol. 35, pp. 433-440.

DALEY, R. C., and NEUMANN, P. G. (1965). A General Purpose File System for Secondary Storage, *AFIPS Conference Proc.*, Vol. 27, pp. 213-229.

FRASER, A. G. (1972). File Integrity in a Disc-Based Multi-access System, in *Operating Systems Techniques*, C. A. R. Hoare and R. H. Perrot (Ed.). London: Academic Press Inc. (London) Ltd.

MILLARD, G. E., REES, D. J., and WHITFIELD, H. (To be published). The standard EMAS subsystem, *The Computer Journal*, Vol. 18, No. 3.

REES, D. J. (1975). The EMAS Director, *The Computer Journal*, Vol. 18, No. 2, pp. 122-130.

WATSON, R. W. (1968). *Time-Sharing System Design Concepts*, New York: McGraw-Hill.

WHITFIELD, H. and WIGHT, A. S. (1973). The Edinburgh Multi-Access System, *The Computer Journal*, Vol. 16, No. 4, pp. 331-346.

WILKES, M. V. (1972). *Time Sharing Computer Systems*, Second Edition, Chapter 8. London: Macdonald & Co. Ltd.

WILKES, M. V. (1972). On preserving the integrity of data bases, *The Computer Journal*, Vol. 15, No. 3, pp. 191-194.