RECEIVED BY DTIC MAR 20 1970

MASTER

IS-2242

Mathematics and Computers
(UC-32) TID-4500

UNITED STATES ATOMIC ENERGY COMMISSION
Research and Development Report

AUTOMATIC FILE INTEGRITY
IN COMPUTER SYSTEMS

by

John R. Jordan

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or

B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

January 1970

DISTRIBUTION OF THIS DOCUMENT IS UNLIMITED

Ames Laboratory
at
Iowa State University of Science and Technology
Robert S. Hansen, Director
Contract W-7405 eng-82

feq

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency Thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

DISCLAIMER

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

IS-2242

This report is distributed according to the category Mathematics and Computers (UC-32) as listed in TID-4500.

LEGAL NOTICE

This report was prepared as an account of Government sponsored work. Neither the United States, nor the Commission, nor any person acting on behalf of the Commission:

- A. Makes any warranty or representation, expressed or implied, with respect to the accuracy, completeness, or usefulness of the information contained in this report, or that the use of any information, apparatus, method, or process disclosed in this report may not infringe privately owned rights; or
- B. Assumes any liabilities with respect to the use of, or for damages resulting from the use of any information, apparatus, method, or process disclosed in this report.

As used in the above, "person acting on behalf of the Commission" includes any employee or contractor of the Commission, or employee of such contractor, to the extent that such employee or contractor of the Commission, or employee of such contractor prepares, disseminates, or provides access to, any information pursuant to his employment or contract with the Commission, or his employment with such contractor.

Printed in the United States of America
Available from

Clearinghouse for Federal Scientific and Technical Information
National Bureau of Standards, U. S. Department of Commerce
Springfield, Virginia 22151

Price: Printed Copy \$3.00; Microfiche \$0.65

IS-2242

AUTOMATIC FILE INTEGRITY
IN COMPUTER SYSTEMS

by

John R. Jordan

ABSTRACT

This paper describes a process which automatically regulates the systematic rotation of magnetic tapes used in computer systems. This process ensures that correct generations of data are associated with the proper volumes and automatically provides for back-ups in case of errors.

When a computer system involving updating of magnetic tape files becomes productional, there remain the problems of (1) keeping records of what generations of data are on what volumes, (2) systematic rotation of volumes to ensure back-up in case of errors and (3) communicating to operating personnel what volumes are to be used for a given run of the system. With this much human involvement, the likelihood of errors occurring is high. This paper describes a process by which the above mentioned tasks are either done automatically or the need to do them is eliminated.

The Ames APREN (Automatic Profile Enriching) SDI (Selective Dissemination of Information) System is an example of a system which has these problems. Approximately 6,000 bibliographic entries (author, title and source of documents) are received weekly on magnetic tape from the Institute for Scientific Information, Philadelphia, Pennsylvania (Fig. 1). These documents are matched against users' interest profiles causing the documents judged most likely to be of interest to a given user to result in notifications. The words contained in the notifications are retained until the user responds to his notifications. Then each word is entered into his profile with either a positive or negative weight depending on how the document in which it was found was judged by the user. If a word is already in the profile, its weight is adjusted accordingly. In this way the profile "learns" to give the user more good documents and fewer irrelevant ones.

Thus there are two files -- profile and selected document words which must be updated. Due to the time lag caused by the users' interaction with the system these files cannot be kept on a public disk and must be on magnetic tape. Since these files are constantly being updated this problem of tape rotation is a non-trivial one.

The age old partial solution to this problem is as follows. A block of tapes is assigned to a given function - say the profile. A record is made of the numbers of these tapes. The first copy is put on perhaps SDI002. When a run of the system is made this file is updated with some transactions and the updated volume is SDI003. Although every effort has been made to ensure the updating program has no errors, they sometimes occur. The program can edit usually only for the type of data or the presence of certain quantities in transactions. It cannot, in general, edit for accuracy or correctness of the magnitude of these quantities. Therefore, the old master file (SDI002 in this case) is retained for a period of time. This time is necessary to allow for human checking of results, etc. Only when the person or persons responsible for the file are sure the new file is correct, is the old copy freed to be written on again. This is repeated until the block of tapes is exhausted. At this time the first one is used again. This process is illustrated in Fig. 2. It is as though the names of the tapes were written on a wheel. There are pointers to the slots with the names of the old (current) and new (next) masters. After a run of the system the wheel is rotated one notch causing the new (updated) master to become old (current), the old master

to move out of the picture and an unused (or used but free) volume to be ready for the next run.

This process has been used for several years with moderate success. The problem with it is that it requires a person to "turn the wheel" and to keep track of all the tapes to ensure that one which will be written on is in fact free. The only thing wrong with this is that it doesn't have to be done at all. By using a COBOL program in conjunction with an IBM utility program⁽¹⁾ all this can be done automatically. The IBM Operating System/360 provides a feature whereby data sets may be cataloged.⁽²⁾ That is, a data set may be put on a given volume or tape reel, the name or number of the reel is entered in the catalog under a symbolic name for the data set. Then any reference to the data set is made by referring to the symbolic name. The system looks in the catalog to see where the data set is actually located (i. e. on what volume). This then acts as a pointer to the correct volume. Thus when it is necessary to change volumes it is only necessary to change the pointer to access the correct volume.

Data sets may be cataloged and uncataloged simply by supplying a control card for each action to the IBM utility program IEHPROGM. Now all that is necessary to complete this task is a process which will automatically turn the wheel and supply IEHPROGM with the appropriate information to uncatalog the old volumes and catalog new ones.

The wheel may be represented in table form (Fig. 3). The top two positions of this table may be thought of as containing the names of the old (current) and new (updated) volumes just as the wheel had pointers.

In order to turn the wheel, the top most element is removed, all the others are pushed up one position and the top one placed in the bottom spot. This table is kept on a magnetic disk. It is read in by a program, altered in the above fashion and the new table is written over the old one. In so doing the program can form the appropriate control cards needed for IEHPROGM to uncatalog the old volumes and catalog the new ones. These control cards are then written out on a temporary disk data set for a following IEHPROGM to read and execute.

In addition to supplying this information to the system, the program can print out a summary of the status of tapes and their assignments. This record may be filed or used to inform operating personnel what tapes will be needed for the next run of the system.

This package of the tape rotating program and the utility program can then be included in the job stream with the other programs in the system. Then if for some reason the job terminates abnormally, they will be skipped and the tapes will not be rotated.

Thus the three problems mentioned at the beginning no longer exist. It is no longer necessary to keep a record of what generations of data are on what volumes because the system remembers. Systematic rotation is built in to ensure back-up in case of errors and the summary sheet may be given to operating personnel so they may have the necessary tapes on hand when the system runs.

References:

1. IBM System/360 Operating System Utilities Form C28-6586-6.
2. IBM System/360 Operating System Job Control Language Form C28-6539-5.

FIG.1. THE AMES APREN SDI SYSTEM

FIG. 2 TAPE LABEL WHEEL

FIG. 3 TAPE LABEL TABLES