

## A USER-FRIENDLY DISK MANAGEMENT SYSTEM

Tom Hoffman and Linda Tanis, Lederle Labs

### ABSTRACT

A disk management system has been developed that automatically backs up SAS data libraries and partitioned data sets to tape. These OS data sets are cataloged as members of generation data groups. Only SAS data libraries that have been modified since the previous backup are copied to tape. This feature not only provides the user distinct backups, but reduces both tape space and the number of entries on the system catalog. Partitioned data sets are not copied to tape unless they have been accessed since the previous backup.

All OS data sets and their attributes are included in a SAS data library (DMSSASDL). This library contains separate SAS data sets for SAS data libraries, partitioned data sets, and other OS sets. Inactive SAS data libraries and partitioned data sets are deleted from disk by the disk management system, but maintained in the DMSSASDL. This data library, in addition to the attributes contained on the disk's volume table of contents (VTOC), includes for each SAS data library and partitioned data set, a unique numeric code, creator identification, last modification date, and the dates of the three most recent backups.

Users easily can access a tape backup after a simple modification in the data set name (DSNAME). The system also includes a keyword-driven facility for producing special reports and managing data sets.

### INTRODUCTON

Three-hundred-ninety million bytes of on-line disk space for storing clinical data should be sufficient space for a company submitting one new drug application per year. For example, a recent SAS data library required only 13 million bytes to store the safety data on 2000 patients, many on study for over two years. However, disk space is finite and when released to the SAS user community, with no restrictions and no management system, the end result is too predictable. Many jobs abend because of insufficient space either to add new data sets or update existing ones.

This was the situation at Lederle four months ago. The proposed solution was to increase disk storage capacity. The standard reports that were available for listing information contained on the disk's volume table of contents (VTOC) did not include last usage date. However when the VTOC option of the SAS INFILE statement was used to read the last usage date from the volume table of contents, a simple calculation showed that over one-third of the 335 SAS data libraries had not been accessed during the past 90 days. The solution to our disk storage problem was not more space but more management.

In order for a disk management system to meet our needs, the following objectives had to be

satisfied:

automatic The user should not be required to back up or delete data sets.

selective Only recently modified data sets should be backed up. Distinct multiple backups should be available.

easily retrievable backups Restoring data sets to disk must be simple for all users. Tape backups should be usable without restoring to disk.

tracking Data sets deleted from disk should be maintained as part of the system. Backup dates should be available.

reporting capabilities Status reports on disk usage and special reports on subsets of data sets should be easily obtainable. It should be possible to separate SAS data libraries from other OS data sets.

It was clear that in order to meet these objectives it would be simpler to develop a SAS system then to modify an existing disk management system. This SAS system is described in the following sections.

### USER-FRIENDLINESS

The automatic backing up and cataloging of data sets to tape is the most important user-friendly feature of the system. There is no need for users to back up their own data sets or to track data sets deleted from disk. Users easily can access tape backups of a SAS data library or partitioned data set after a simple modification in the DSNAME (e.g., one would use PR.T.SDA.FEN.SAFETY(0) when accessing the most recent tape backup of PR.SDA.FEN.SAFETY.

The system includes additional user options for printing reports and managing data sets. These keyword-driven options require only one JCL statement (EXEC SASDMS).

A description of these options follow:

#### Printing Reports

- |         |   |
|---------|---|
| PSTATUS | Prints a three part one page report summarizing disk usage.   |
| PSAS    | Prints listing of SAS data libraries and their attributes, including backup dates. The report contains data libraries currently on disk as well as those deleted from disk. |
| PPDS    | Prints listing of partitioned data sets and their attributes, including backup dates. The report contains data sets currently   |

on disk as well as those deleted from disk.

POTHER Prints a listing of other data sets (neither SAS nor PDS) and their attributes. This report only contains data sets on disk.

A MACRO named SUBSET may be included before the print requests in order to print a subset of the data sets. A KEEP statement can be used to restrict the variables printed.

#### Managing Data Sets

Each of the following keywords must be followed by one or more data set names (DSNAME), their numeric identification (CID), or a mixture of DSNAME's and CID's.

;RESTORE Restore to disk.  
;DELETE Delete from disk.  
;REMOVE Delete from disk and remove from the disk management system.  
;CANCEL Cancel a previous DELETE or REMOVE request.  
;ADDSpace Recreate a SAS data library with additional space. (This option is automatic for data sets with less than five extents remaining).  
;PCONTENT Execute PROC CONTENTS and print the first ten observations of each data set in a SAS data library, either on disk or tape.  
;ADDBADGE Add creator identification to a data set. (Badge numbers are paired with either DSNAME's or CID's).  
;LBACKUP List all backup information (dates, tape volumes, and file numbers).  
;RUNDMS Execute one or more of the above requests. Must be included as the last statement in the program.

A sample user request program follows:

```
// EXEC SASDMS
PSTATUS
MACRO SUBSET IF DSNAME='PR.SDA'; %
PSAS
;REMOVE
PR.SDA.LGT.DISKMGT
;DELETE
397
;RESTORE
259 PR.SDA.FEN.DAT
;ADDSpace
755
;PCONTENT
259 PR.FEN.SAFETY
;RUNDMS
```

The output resulting from this job is shown in Table 1 and Table 2. Table 1 contains the output from the PSTATUS request. The last two parameters on the disk usage summary indicate whether there is a need to compress the disk pack. 'PERCENT < 5 TRACKS' is the number of free tracks (expressed as a percent of total disk space) unavailable unless the requested secondary allocation is less than five tracks. The final parameter is the percent of data sets requiring more than one location. The right hand report in Table 1 is a concise summary of free space extents. It gives the number of locations as a function of size as well as the number of tracks in these locations.

The graph at the bottom of Table 1 plots percent free space across time. If disk space is properly managed, percent free space should be a constant, assuming no change in usage. The data plotted in the example illustrates the need for a disk management system to automatically delete inactive data sets. This feature was not implemented initially in the hope that users would delete their own data sets.

The output from the PSAS request is shown at the top of Table 2. The STATUS variable highlights the status of each data set. Values of STATUS equal to NEW, MOD, ACC, and DEL indicate which data sets were newly created, modified, accessed, or deleted during the prior week. STATUS equals OLD if the data set has not been accessed for 60 days. These data sets will be deleted by the system on Friday unless accessed by the user.

The final listing in Table 2 summarizes the data management requests. Note that the RESTORE, ADDSPACE, and PCONTENT requests each submit a second job using the internal reader. The name of the submitted job is shown in the COMMENT column.

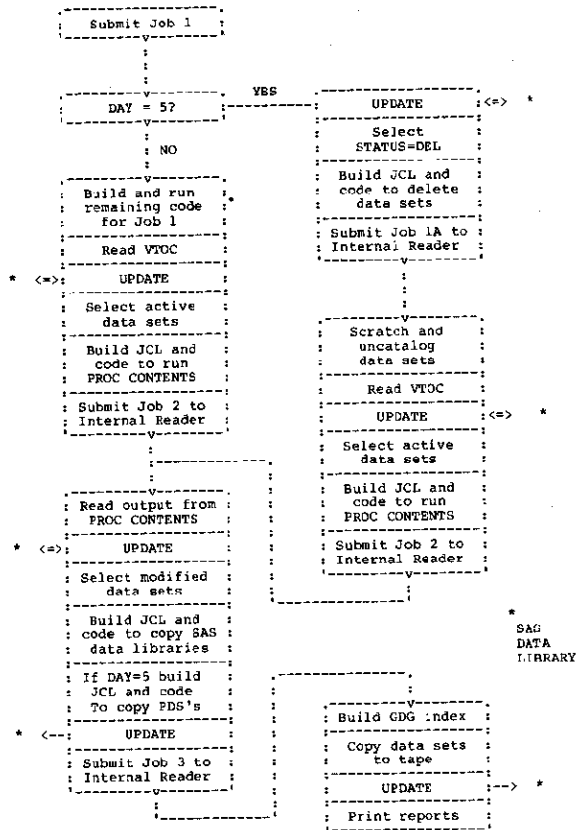
#### METHODOLOGY

The SAS data library (DMSSASDL) that contains information on disk and tape data sets, also controls the disk management system. The JCL and program code that the system generates depend on the values of the following variables in the DMSSASDL:

TAPE	volume name of current tape
LABEL	last used file number on the tape
TTLTRKS	total tracks copied to the tape
BACKUP	date and time of last backup
LCID	last assigned CID number
COUNT	the number of times each data set has been opened
NB	number of tape backups for each data set
STATUS	activity status of each data set

A flow chart for the system is shown on the next page.

## System Flow Chart



Each work day one job (Job 1) is submitted in an overnight class. The action of this job depends upon the day of the week.

On Monday thru Thursday Job 1 next reads the volume table of contents (VTOC) for three disk packs. The VTOC count variable is compared to the value of COUNT in the DMSSASDL to determine which data sets have been accessed since the previous backup. Since the value of COUNT in the DMSSASDL is incremented any time that the disk management system opens a data set, only user activity is monitored. After updating the DMSSASDL (retaining the old last use date for data sets not accessed by the user), the job writes to the internal reader (Job 2) the JCL and code to run PROC CONTENTS on active SAS data libraries.

On Friday Job 1 updates the STATUS variable in the DMSSASDL and selects all data sets scheduled for deletion either because of inactivity or a user request. The JCL and code to delete these data sets (using the IBM utility IEHPRGM) is written to the internal reader (Job 1A). The SAS code usually executed in Job 1 is also written to the internal reader as part of Job 1A.

Job 2 reads the output from PROC CONTENTS and calculates the most recent modification date for SAS data sets. This date is compared to the last backup date (BACKUP) in the DMSSASDL to determine

the subset of active data sets that were modified. Job 3 is submitted by writing to the internal reader the JCL and program code to copy and catalog data sets on tape. On Friday Job 3 also includes the JCL and code to copy and catalog partitioned data sets to tape.

Job 3 first builds generation data group (GDG) indexes for data sets not previously backed up using the IBM utility IEHPRGM. PROC COPY is used to copy SAS data libraries to tape and PROC PDSCOPY is used to copy partitioned data sets to tape. Many data sets can be written to the same tape since the current tape statistics are kept in the DMSSASDL. When a new tape is mounted, Job 3 obtains its volume name from PROC TAPELABEL's output and updates the DMSSASDL. Although the three most recent backups are cataloged and easily accessible, any backup is available. All backup dates, tape volumes and tape file numbers for each CID are stored in the DMSSASDL.

All program code used in the system is written in version 79.6 of SAS. The system runs in OS batch.

## PLANNED ENHANCEMENTS

### Adding Control

The next addition will give the system more power to control data sets added to disk. Four areas of control are being considered:

1. Enforcing naming conventions (DSNAME, SAS data set names, and SAS variable names).
2. Requiring the addition of creator ID.
3. Checking for double-precision storage of numeric data.
4. Automatically releasing unused space.

### Reading PDS Directories

The last modification date for each member of a PDS is written to the directory if TSO/SPF is used to edit the member. Currently the system backs up partitioned data sets weekly. When this feature is added to the system, SAS programs and MACRO libraries stored as members of partitioned data sets will be backed up daily, if modified.

### Archiving Data Sets

Although archiving data sets was not one of the objectives of the system, it is possible to 'freeze' a SAS data library or partitioned data set in the current system. This can be done by first copying the data set to disk using a naming convention that denotes an archived data set. This new data set is then deleted from disk after the system has copied it to tape. This procedure will be keyword-driven in the future.

## PORTABILITY

This disk management system is easily adaptable to other installations that allow the use of the SAS VTOC option and the internal reader.

However, two of the concepts used in developing the system may be more important than the system itself to other SAS users:

1. using information in a SAS data library to control a system, and
2. writing programs that write programs.

Of course the %INCLUDE statement in SAS and the internal reader are essential to implement these concepts. The %INCLUDE statement allows the same program to generate and execute SAS source code. The internal reader allows one program to generate JCL and program code to be submitted as a second job.

For example, using these concepts, a system could be developed that would automatically load data into a SAS data library from case record forms. The essential information for writing the load program would be contained in separate SAS data library controlling the system. This control library would require updating each time a new case record form is designed.

Additional information on the system can be obtained by contacting

Thomas R. Hoffman  
Lederle Labs  
Pearl River, NY 10965

TABLE 1  
CLINICAL INFORMATION DISK MANAGEMENT SYSTEM

STATUS REPORT: 14JAN83:01:18

SUMMARY OF DISK USAGE					FREQUENCY DISTRIBUTION ON FREE SPACE					
	MEDIC1	MEDIC5	MEDIC6	TOTAL	EXTENT SIZE	MEDIC1	MEDIC5	MEDIC6	TOTAL	TRACKS
NUMBER OF OS DATA SETS	127	128	168	423	< 5	7	47	69	123	287
NUMBER OF SAS DATA LIBRARIES	20	93	133	246	5-9	15	10	20	45	347
ALLOCATED SPACE (TRACKS)	5561	7430	12038	25029	10-19	4	0	27	31	456
PERCENT ALLOCATED TO SAS	26	80	78	67	20-39	14	0	4	18	417
FREE SPACE (TRACKS)	2095	226	3294	5615	40-59	2	0	7	9	144
PERCENT FREE SPACE	27	3	21	18	60-79	0	0	2	2	830
PERCENT < 5 TRACKS	0.2	1.6	1.0	1.0	80-99	1	0	8	9	617
PERCENT EXTENDED DATA SETS	13	39	47	34	112-149	1	0	4	5	
					TOTAL	44	57	141	242	
					FOUR	161	7	157		
					LARGEST	198	8	157		
						414	16	240		
						458	17	285		

15

PLOT OF PERCENT FREE SPACE VS TIME  
PAST 26 WEEKS

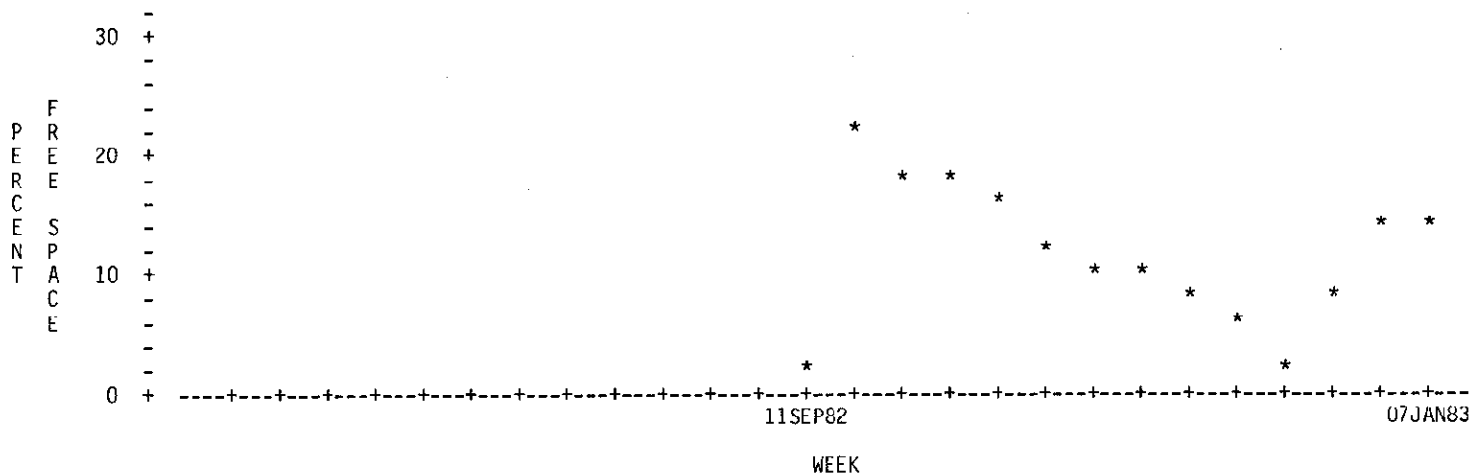


TABLE 2

CLINICAL INFORMATION DISK MANAGEMENT SYSTEM

8:33 FRIDAY, JANUARY 14, 1983 2

DISK MAP--SAS DATA LIBRARIES

DATA SETS WITH STATUS=OLD WILL BE DELETED ON FRIDAY  
UNLESS ACCESSED BY USER

OBS	CID	BADGE	STATUS	DSNAME	VOLSER	CREATED	LASTUSE	LASTMOD	TRACKS	UNUSED	XREM	TRKREM	NB	BACKUP1	BACKUP2	BACKUP3
1	393	4128	MOD	PR.SDA.ALLEN.TEST	MEDIC6	17SEP82	13JAN83	06JAN83	362	0	0	0	7	07JAN83	27DEC82	13DEC82
2	757	4457	D??	PR.SDA.FEN.AGELIFE	MEDIC6	07JAN83	12JAN83	07JAN83	100	4	6	60	1	07JAN83	.	.
3	395	3946		PR.SDA.FEN.DAT		13SEP82	05OCT82	.	382	.	.	.	1	02OCT82	.	.
4	248	3946	ACC	PR.SDA.FEN.SAFETY	MEDIC5	04MAY82	13JAN83	09DEC82	925	19	12	300	6	13DEC82	12NOV82	27OCT82
5	249	3946	MOD	PR.SDA.FEN.SAFETY1	MEDIC6	18MAY82	12JAN83	07JAN83	768	96	12	576	3	07JAN83	12NOV82	11SEP82
6	397	3946		PR.SDA.FEN.SUM		13SEP82	05OCT82	.	135	.	.	.	1	02OCT82	.	.
7	398	3946		PR.SDA.FEN.XLAB		16SEP82	05OCT82	.	476	.	.	.	1	02OCT82	.	.
8	250	4382	DEL	PR.SDA.F2TN2B.FIL.SASDB		10FEB82	03NOV82	.	37	.	.	.	1	11SEP82	.	.
9	251	4382		PR.SDA.F2TN2C.FIL.SASDB		13FEB82	14APR82	.	37	.	.	.	1	11SEP82	.	.
10	252	4382		PR.SDA.F2TN29.FIL.SASDB		20FEB82	14APR82	.	25	.	.	.	1	11SEP82	.	.
11	253	7226	OLD	PR.SDA.LGT.DISKMGT	MEDIC5	04AUG82	06AUG82	.	10	0	15	75	1	11SEP82	.	.
12	755	4382	MOD	PR.SDA.LOX.GER.SJ6320	MEDIC6	21DEC82	12JAN83	04JAN83	74	6	7	70	2	07JAN83	27DEC82	.
13	254	4382		PR.SDA.MLSMNS.EX2.SASDB		18DEC81	02JAN82	.	9	.	.	.	1	11SEP82	.	.
14	255	4382		PR.SDA.M5T022.AL2.SASDB		17DEC81	04JAN82	.	9	.	.	.	1	11SEP82	.	.
15	256	3576		PR.SDA.SJ5783.CUM.SASDB		28JUN82	11SEP82	.	40	.	.	.	1	11SEP82	.	.
16	257	3576		PR.SDA.SJ5783.HUM.SASDB		07JUL82	11SEP82	.	50	.	.	.	1	11SEP82	.	.
17	258	3576		PR.SDA.SJ5783.ZUM.SASDB		06JUL82	11SEP82	.	40	.	.	.	1	11SEP82	.	.
18	259	4854		PR.SDA.TOX.CONTROL.WT		22JUL82	23SEP82	.	285	.	.	.	1	11SEP82	.	.
19	261	3946		PR.SDA.TRH.FENADV	MEDIC6	17APR82	16NOV82	.	228	0	15	750	1	11SEP82	.	.
20	262	3946	MOD	PR.SDA.VTOC	MEDIC6	24AUG82	13JAN83	06JAN83	70	1	10	50	10	07JAN83	27DEC82	13DEC82
21	427	3946	R??	PR.SDA.VTOCOLD	MEDIC6	24AUG82	11JAN83	07JAN83	37	0	15	75	5	07JAN83	27DEC82	13DEC82

16

CLINICAL INFORMATION DISK MANAGEMENT SYSTEM

8:33 FRIDAY, JANUARY 14, 1983 3

SUMMARY OF USER REQUESTS

N	REQUEST	CID	BADGE	STATUS	DSNAME	TYPE	VOLSER	TRACKS	NB	BACKUP1	COMMENT
1	REMOVE	253	7226	R??	PR.SDA.LGT.DISKMGT	SAS	MEDIC5	10	1	11SEP82	WILL BE REMOVED FRIDAY
2	DELETE	397	3946	D??	PR.SDA.FEN.SUM	SAS		135	1	02OCT82	WILL BE DELETED FRIDAY
3	RESTORE	259	4854	RES	PR.SDA.TOX.CONTROL.WT	SAS		285	1	11SEP82	REQUEST COMPLETED - JOB PS#4854 SUBMITTED
4	RESTORE	395	3946	RES	PR.SDA.FEN.DAT	SAS		382	1	02OCT82	REQUEST COMPLETED - JOB PS#3946 SUBMITTED
5	ADDSPACE	755	4382	MOD	PR.SDA.LOX.GER.SJ6320	SAS	MEDIC6	74	2	07JAN83	REQUEST COMPLETED - JOB PS#4382 SUBMITTED
6	PCONTENT	259	4854		PR.SDA.TOX.CONTROL.WT	SAS		285	1	11SEP82	REQUEST COMPLETED - JOB PS#4854 SUBMITTED
7	PCONTENT				PR.FEN.SAFETY						REQUEST FAILED - DSNAME NOT FOUND