

DEVELOPING AN ENHANCED FACILITY FOR INSTALLING SAS* SOFTWARE
FOR THE MICROCOMPUTER UNDER MS DOS AND PC DOS

Kevin L. Kiser, University of Wyoming

Abstract

Installing the SAS system on a microcomputer is an eleven step process. The Microcomputer Resource Center (MRC) at the University of Wyoming (UW) has developed an enhanced facility eliminating most of the work for the end-installer. This paper presents the key considerations in installing SAS software under PC DOS, the assumptions governing development of UW's installation facility, SAS Institute's SASLOAD batch file, and a discussion of the influence of MS DOS upon installation.

Introduction

The microcomputer version of the SAS system is released by SAS Institute in the familiar form of packages. There are several optional packages in addition to the basic software package. Each package is released as a set of diskettes created using an archiving command (BACKUP) under PC DOS. After some system preparation, the archival restoration process dominates installation. The process is straightforward but relatively lengthy and cumbersome. Manual installation is acceptable for a few workstations; for the projected installation of 100 sites, an installation facility was deemed important.

UW signed an educational discount purchase agreement with Zenith Data Systems (ZDS) in January, 1986. ZDS personal computers are IBM compatible but operate under MS DOS. MS DOS and PC DOS archive files in slightly different formats. Consequently, the two versions of RESTORE read files slightly differently. Restoring the archived files of PC DOS while operating under MS DOS yields eccentric results. The issue of archival format becomes important.

In June, 1986, UW signed a 500 site SAS system license with a provision allowing distribution of the SAS system for microcomputers on campus and to the seven community colleges throughout the state. UW would violate copyright laws if it distributes PC DOS incorporated as part of an installation facility. Requiring remote sites to purchase PC DOS for SAS system installation is an unattractive proposition. The issue of archival format assumes greater importance.

These factors prompted UW to develop an installation facility for SAS software,

with particular emphasis on installation under MS DOS. This paper will briefly touch upon the SAS system and its installation process, present an overview of the UW installation facility, and discuss considerations about MS DOS.

The SAS System and Installation

The SAS system for microcomputers currently consists of the base SAS software version 6.02 and the three optional packages of SAS/STAT* software (v6.02), SAS/IML* software (v6.02), and the SAS/RTERM* software (v1.24). These packages are released in PC DOS archival form on 5 1/4 inch floppy diskettes. Each package, with the exception of SAS/RTERM, comes with a diskette of sample procedures (figure 1.).

UW licensed only the first two of these packages: base SAS software and the SAS/STAT software. The software is delivered as 21 diskettes: one INSTALL diskette containing SASLOAD.BAT, twelve diskettes for the base SAS software, six diskettes for the SAS/STAT software, and one sample library diskette for each of these products [1]. Manual installation of the SAS system for microcomputers is an eleven step process involving the creation and/or modification of special system files, removing the old copy of the SAS system (if present), creating directories, restoring the new copy of software, and testing the new installation.

Correct SAS software operation depends upon a particular host environment. Whether the operating system is PC DOS or MS DOS, two special system files execute and initialize the system during boot-up. These files are CONFIG.SYS and AUTOEXEC.BAT; both must exist and contain at least the minimal information for configuration. CONFIG.SYS directs the system to allocate buffer size for I/O, and to indicate the number of open files that can be accessed by certain system calls (figure 2.) [2]. The system default values work well for most applications and the CONFIG.SYS file is often nonexistent before SAS software installation. Creation of the file is necessary if it does not exist.

As system initialization continues, AUTOEXEC.BAT executes system commands before completing boot-up and returning a prompt. There are many uses for this type of an automatically executed file: to display menus, messages, and other

special notices; establish paths to various directories; begin the execution of other jobs or software packages, such as print spoolers and word processors; or any other executable system command. SAS software will reside in a special directory and the PATH command executed in AUTOEXEC.BAT informs the system of SAS software location.

The SAS system also requires the correct date and time. The DATE and TIME commands are often included here. Clock/calendar cards free the user of the necessity to establish the correct date and time (figure 3a.). Otherwise, accurate chronology need exist before SAS software execution. Typically AUTOEXEC.BAT exists and installation requires modification of the existing file (figure 3b.). Note the absence of the DATE and TIME commands in (3b.). If a clock/calendar card is installed on the host machine, the commands are redundant and can be removed from the file.

The Institute has provided a batch file to assist in erasing the old copy of the SAS system and installing the base SAS software. The Institute's INSTALL DISKETTE contains the installation file SASLOAD.BAT and two associated text files. The SAS/STAT software package installs with a single RESTORE command. Presumably the other packages are archived and restored similarly.

SASLOAD is built around the idea of restoring a fresh copy of the SAS system to a clean SAS directory. A clean directory prevents the accumulation of different versions of files, especially with possible file name changes between versions. This necessitates either discarding the entire old directory and creating a new one, or cleaning up the current SAS directory prior to restoring files. The Institute reuses the same few subdirectories for each SAS software product. Deleting all subdirectory files for the base SAS software intentionally deletes all files for the other SAS software packages. This directory structure of the SAS system allows the Institute to delete the entire contents of the SAS directory in a few commands rather than by extensive individual file deletion. Wholesale discarding of all files quickly and easily empties the directory but such file erasure can be lethal if performed indiscriminately. Prior to erasing the SAS subdirectories, the user should be warned about accidental loss of files. If the user has stored files in these subdirectories, by accident or by design, they will be lost during execution of this erasure method. Informing the user constitutes the first portion of SASLOAD, pausing twice to

give the user time to abort before user files would be lost.

There are two optional directories that can be created at installation time. One is the sample libraries directory and the other is a directory for personal work. The first directory might be a single directory for all SAS software sample libraries or several different directories, one for each software package. The personal directory should be any directory external to the SAS directories. This prevents accidental erasure of any of the SAS system files, either directly or by saving another file of identical name.

The Institute verifies accurate restoration and correct operation using benchmark comparisons. A test program is SUBMITTED to the SAS system for each package installed and the results are saved. The output and log files of each test program are compared against known counterparts in the SASTEST subdirectory. The COMPARE command of PC DOS returns a 0 ERRORS message if the two files (for each package) have identical counterparts in SASTEST, signifying program transfer was accurate and that execution was correct.

SOME CONSIDERATIONS

User files must remain intact. It is the primary concern behind any SAS system installation at UW. This includes while creating, modifying, and/or erasing files. It is not acceptable to merely copy a generic CONFIG.SYS file to the receiving microcomputer or to simply delete all files in a subdirectory; either process can destroy information critical to the user. This extensive file creating, altering, copying and deleting--along with directory creation and removal--increases the chance of file loss. Safeguards must be designed into the facility to allow for the comparisons of differing versions of files and to protect against indiscriminate file erasures.

Another danger to erasing user files arises from knowingly or unknowingly storing files in the SAS directory. Executing the SAS system from within a SAS subdirectory results in the FILE statement saving files in that subdirectory by default. For example, executing the SAS system from within the SASHELP subdirectory will save files in this subdirectory when there is not a specified path name. The Institute considers this highly unlikely of a user, either to calling SAS from within the SAS directory or to saving files in that location. As a result, the Institute designed SASLOAD.BAT to perform whole-

sale directory deletion, deciding user files actually being stored in a SAS subdirectory was of low probability [4]. UW disagrees, having had a number of unfortunate experiences with inexperienced microcomputer users. Selective (individual) file erasing avoids the loss of user files possibly residing in a SAS subdirectory. This is preferable for preserving the integrity of user files.

Microsoft Corporation created the DOS operating system and OEMs the code to others. ZDS has their version of DOS, commonly known as MS DOS, as opposed to the PC DOS of IBM. But this MS DOS differs from Compaq MS DOS. Compatibility problems have existed in the computer field from the first days of computing so it is no surprise to see IBM PC DOS and MS DOS compatibility as a contested issue in the microcomputer domain. Any type of an enhanced facility must squarely address this issue. Viability in porting cannot exist without accounting for implementation differences within the DOS family. A particularly obstructing example of incompatibility arises from the archiving format of PC DOS and MS DOS. This difference constituted the second consideration UW explored.

Incompatibility extends to within ZDS MS DOS itself. PC_GARNR.BAT and MS_GARNR.BAT copy and erase DOS command files. Correct performance of their task is dependent upon exercising accurate knowledge about the implemented DOS architecture. If differences exist between perceptions encoded in the enhanced facility software and the resident DOS software of the installation site, then accurate installation is degraded. Surprisingly, numerous differences exist within the same versions of MS DOS as released by ZDS and must be accounted for during software development. For ZDS MS DOS version 3.10, UW has found six distinct versions of IO.SYS (counterpart to IBMBIO.COM). Along with the successive IO.SYS releases, the architecture of MS DOS is evolving. Commands have been removed, such as CIPHER.EXE. Commands have been added, such as SYSCLR.EXE. Commands have been renamed, such as FORMAT.COM, formerly accessed as FORMAT.EXE. This same renaming holds true for COMP, LABEL, MORE, SYS and others. ZDS MS DOS appears to be moving toward the naming convention of PC DOS, but continuing to extend the scope of operating system activity.

The original intent of the SAS INSTALL facility ordered batch file execution through a compiled BASIC program. SASLOAD.BAT would install the base SAS software; two batch files written by the

author at the UW MRC would install the SAS/STAT software and the sample libraries (figure 4.). SASLOAD was incorporated into UW's installation facility. Later, concerns arose about the integrity of user files. Individual file deletion within SAS subdirectories was felt to better safeguard the user and address the major concern of installation: installing the SAS system without disturbing user files. With SASLOAD.BAT as proprietary software preventing modification, an early version incorporating a virtual disk was stymied. UW abandoned SASLOAD.BAT in favor of creating a batch file to perform a similar role. A batch file template evolved as direct result of displacing SASLOAD.BAT for base SAS software installation, making it possible to duplicate the same code structure for installing other SAS software packages with minimal change. A template eases the concern about modifying the installation software to accept new packages as they are released by the Institute. The consistency enforced by templates also enhances and simplifies documentation (figure 5.).

Installation Facility Overview

The UW installation facility is divided into two diskettes: installation activity and documentation. The number of files and the sizable documentation prevent the use of a single diskette. SAS INSTALL is a single diskette to fully install SAS software and includes a single menu to access diskette activity, installation activity, and activity to copy or print the diskette. A second diskette--SAS DOCUMNT--fully documents the SAS INSTALL diskette and includes a similar menu to access diskette activity, all documentation of the SAS INSTALL diskette, the source code for the two BASIC programs, two batch files for compiling source code, and activity to copy or print the diskette. Controlling batch file execution order with a compiled BASIC program was retained in the final version of the SAS INSTALL facility. Additional batch files replace the functions of SASLOAD.BAT (figure 6.).

The installation facility begins from a boot-up with the SAS INSTALL diskette in drive A. CONFIG.SYS establishes a VDISK as drive D. AUTOEXEC.BAT copies most of the SAS INSTALL directories and files to the virtual disk. This decreases the installation time. AUTOEXEC.BAT then executes DISKMENU.EXE to control installation activity. The user is offered a choice of installing the SAS software, copying diskettes, or copying the documentation. When installation is selected, INSTALL.EXE is activated. The user is asked which brand of DOS is used

on the installation machine: PC or MS. PC_GARNR.BAT or MS_GARNR.BAT act upon this information to copy transient commands from the root directory to a subdirectory. It was anticipated that half of the installation sites would belong to inexperienced microcomputer users and such users would either copy the transient (PC or MS) DOS files to the root directory or would not copy the commands from diskette to the microcomputer fixed disk. Yet, efficiency increases by locating all DOS transient commands within a separate directory and creating a path to this directory through AUTOEXEC.BAT. If this has not been done by the user, it is done by the installation software (PC_GARNR.BAT or MS_GARNR.BAT) prior to installing the SAS system. The garnering activity can be bypassed if a user desires to maintain the transient commands in the root directory of their machine.

INSTALL.EXE checks for the existence of the files CONFIG.SYS and AUTOEXEC.BAT, required for correct SAS software operation. Creating these files without checking for a current user version could cause a loss of commands, thus INSTALL.EXE checks for the existence of the file (in the root of drive C) using FILEXIST.BAT and displays its contents if the file exists. It is not wise to alter or create these files without alerting the user. INSTALL.EXE will repeat the same process for the files CONFIG.DOC and AUTOEXEC.DOC. These files document any modifications and/or file creations. They also document for the user the commands and parameters necessary for SAS software operation. INSTALL.EXE inquires of the user whether to append, create, ignore, or replace the old (users) file with the new (SAS) version. The existence of these two document files is checked before copying, though it is unlikely that a user would have such filenames.

These four files may need to be edited. For example, an installed clock/calendar card does not need the DATE and TIME commands in the AUTOEXEC.BAT file, or there might be duplicate DATE and TIME commands if the SAS software version of the file is appended to the current user's version. INSTALL.EXE issues a reminder that editing may be needed. Earlier versions invoked PC_WRITE, a public domain word processor. Editing of all four files could be done within PC_WRITE, and exiting PC_WRITE returned the installer to INSTALL.EXE. PC_WRITE was removed from the SAS INSTALL diskette in the later versions for space management reasons.

LOADBASE, LOADSTAT, and LOADSMPL are batch files derived directly from the batch file template. The template

provides a consistent framework for installing additional SAS software. Each of these batch files clears the screen, prints a header, prints simple directions for inserting diskettes, then either restores or copies the SAS software to the installation site. PRSNLDIR.DIR creates a separate directory from which to call the SAS software. CONFIG.SAS and AUTOEXEC.SAS are copied into this directory. The SAS software should always be called from this directory or another directory that is NOT from the SAS directory. This is because the SAS system saves files into the default directory, unless paths are specified as part of the file names. Also, the directories SASWORK and SASUSER are created by the SAS software if they do not currently exist in the default directory. Thus, it is easy for the SAS directory and subdirectories to become cluttered with files and/or directories that are not readily visible to the user. Installation activity as controlled by INSTALL.EXE ceases. Control is returned to DISKMENU.EXE for a choice of other diskette activity.

MS DOS Considerations

The SAS system is archived for restoration under PC DOS. Because of their educational contract with ZDS, UW departments operate equally under MS DOS. ZDS microcomputer acquisition and usage is expected to increase. Therefore, a major goal of the installation facility was to devise a successful method to unarchive the PC DOS format of software on ZDS machines. Early experimentation demonstrated diskettes archived under PC DOS were definitely of a different character than diskettes archived under ZDS MS DOS. As many as nine PC DOS archived diskettes can be successfully restored under ZDS MS DOS. For the tenth diskette, an error message is generated:

Disks out of sequence.
Insert volume 10 and
press any key to continue.

Pressing any key repeats the message. The presence of diskette 10 is never recognized and restoration must cease. The Institute is attempting to solve the archiving incompatibility problems between the two operating systems by avoiding the use of either version of the RESTORE command. A future release of the installation procedure, beyond the upcoming 6.03 release, will use a restore facility written by the Institute [3]. This will not resolve problems that arise because of differing operating systems. A solution is to release a second version of SAS software for microcomputers specifically tailored for MS DOS machines.

The SAS software was restored under PC DOS at the UW MRC and then archived under MS DOS to allow restoration at installation sites to occur under MS DOS. This was an acceptable approach to UW. However, it necessitates the maintenance of two sets of archived software: one set under PC DOS and another set under MS DOS, as UW performs installations for its community on both IBM machines and compatibles under MS DOS.

Once the SAS system is successfully restored under PC DOS and the machine re-booted under MS DOS, BACKUP and RESTORE under MS DOS are successful. UW has been able to release diskettes to the community colleges archived under MS DOS. This simplifies the legal situation of remote site installers.

SAS software installs correctly under MS DOS. The output and log files of the test programs are compared against known counterparts in the SASTEST subdirectory. The COMPARE command of PC DOS returns a 0 ERRORS message indicating the two files have identical counterparts in SASTEST and software installation was accurate. When the machine is re-booted under MS DOS and the testing procedure repeated, the files have counterparts of a different size, though the results appear identical. The Institute does not support the operation of its software under MS DOS [4], yet the software appears to execute successfully. Without Institute support, it is distinctly possible that ultimately SAS software will fail under MS DOS. The incompatible nature of the microcomputer domain was expressed earlier, particularly within the family of MS DOS. It is possible that during the evolution of MS DOS, the Institute may currently or in the future may access commands available in PC DOS but not available under ZDS MS DOS. At the time of this writing, SAS software continues to operate without failure.

Conclusion

SAS software for microcomputers is designed, archived, and supported to operate solely within the influence of PC DOS. By necessity, UW investigated the possibility of operating the software under the control of MS DOS. An installation facility was developed to install the software under either system. The facility has successfully and conveniently installed SAS software at UW. In addition, the SAS software continues to work well within the ZDS MS DOS environment.

Acknowledgements

Appreciation is extended to Alice Freeman, the MRC, and the Division of Computing Services of the University of Wyoming for their endless patience and support.

References

- [1] The SAS SYSTEM under PC DOS, Release 6.02 Installation Instructions, SAS Institute, Inc. Box 8000, Cary, N.C., 27511-8000, 1985, pp.10-11.
- [2] Microsoft MS-DOS Version 3 manual, Microsoft Corporation, 1984, pp.9--21,23.
- [3] PC Technical Support Group at SAS Institute, conversation with the author, August 26 & 27, 1986.
- [4] Memo from SAS Institute Marketing Support Department, March 26, 1986 and March 18, 1985.

NOTE: Because of space limitations, the separate appendices as noted in the conference abstract are not included here. They are available from the author at the address below.

SAS, SAS/GRAPH, SAS/AF, SAS/FSP, SAS/ETS, SAS/OR, SAS/QC and SAS/RTERM are registered trademarks of SAS Institute Inc., Cary, NC, USA. SAS/STAT and SAS/IML are trademarks of SAS Institute Inc., Cary, NC, USA. Zenith DOS, copyright portions by Zenith Data Systems, copyright portions by Microsoft Corporation. MS is a trademark of Microsoft Corporation.

Kevin L. Kiser
Microcomputer Resource Center
Division of Computer Services
P.O. Box 3945, Univ. Station
University of Wyoming
Laramie, Wyoming 82071
(307) 766-4879

PACKAGE		NUMBER OF DISKS	ADDITIONAL SAMPLE DISK?
base SAS	v6.02	12	yes
SAS/STAT	v6.02	6	yes
SAS/IML	v6.02	2	yes
SAS/RTERM	v1.24	1	no
SAS/AF	v6.03	(3M)?	*forthcoming*
SAS/FSP	v6.03	(4M)?	*forthcoming*
SAS/GRAPH	v6.03	(12M)?	*forthcoming*
SAS/ETS	v6.03	(3M)?	*forthcoming*
SAS/OR	v6.03	(2M)?	*forthcoming*
SAS/QC	v6.03	(0.5M)?	*forthcoming*

Figure 1. SAS packages. Parenthesized values are approximate mainframe sizes in megabytes.

COMMAND	REQUIRED SAS VALUE	SYSTEM DEFAULT VALUE
BUFFERS	4 or more	2
FILES	50 or more	8

Figure 2a. The CONFIG.SYS system default values and values required for SAS software.

```

BUFFERS = 15
FILES = 57
FCBS = 57,10
DEVICE = C:\dos\vdisk.sys 330 512 48
BREAK = on

```

Figure 2b. Example of a typical CONFIG.SYS file.

```

PATH C:\;C:\SAS;
DATE
TIME

```

Figure 3a. The AUTOEXEC.BAT commands required for SAS software.

```

ECHO OFF
CLS
PROMPT $PSG
CD \KICK
SK
PATH C:\DOS;C:\LOTUS;C:\DBASE;C:\SAS;
\PARADISE\HIRES
SAVE ZSPOOL 20

```

Figure 3b. Example of an AUTOEXEC.BAT file implementing a terminate and stay resident utility, a screen saver and a print spooler.

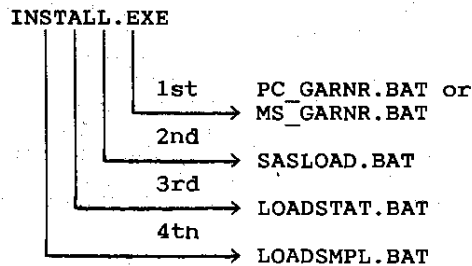


Figure 4. Major batch file order in an earlier version of the facility.

```

ECHO off
cls
ECHO V.      Loading XXXXXXXX software.
ECHO .
ECHO .
ECHO .
ECHO .Inetall XXXXXXXX 6.02 diskettes
using drive A.
ECHO .
ECHO .
ECHO .
PAUSE .....
restore a: c:\sas\*.* /r /s

```

Figure 5. The LOADxxxx.BAT template where xxxx is one of the SAS system packages.

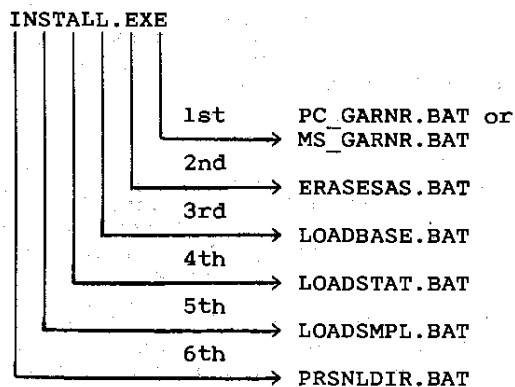


Figure 6. Major batch file order in a later version of the facility.