

The ACCESS® Procedure from a Batch Perspective

George Matthews
The University of Georgia

ABSTRACT

As an avid user of proc dbf in the SAS® System for PCs, I was somewhat dismayed, when I migrated to SAS® for Windows, that this procedure would no longer be available. Although SAS® Institute has since made this procedure available to licensed users of the SAS/ACCESS product, this paper takes a look at the benefits of utilizing the SAS/ACCESS product to access PC File Formats in batch mode. The techniques for defining the access and view descriptors, for PC Files created with the following Windows® products: Microsoft Excel®, Paradox®, and Borland Dbase®, will be examined. Also, the flexibility of the descriptor files in comparison to the previous methods of accessing PC Files will be discussed.

INTRODUCTION

Since the earliest release of SAS for PCs, SAS Institute has provided a means for retrieving data in PC File formats. The dbf procedure was provided, as means of accessing dbf files, with the first production version of the SAS System for PCs (6.03). It continues to be utilized as part of the current version of SAS for PCs 6.04. With the 1993 release of SAS for Windows, a separate product, called SAS/ACCESS, was developed as a means of accessing data in PC File Formats. The SAS/ACCESS product provides an interface to PC File Formats like dbf. The release of the 6.10 platform for Windows provides additional enhancements to the PC File Interface. Some of these enhancements are as follows: ODBC support, WK1/WK3 support, and Line Mode Access to PC File Formats. The benefits of utilizing SAS/ACCESS batch command language to access PC Files will be examined in this paper. The execution of PROC ACCESS in batch mode was made available by SAS Institute with maintenance release TS407 of the SAS for Windows 6.08 product.

DESCRIPTION OF THE DATA

Microsoft Excel was used to transfer a spreadsheet file into a .dbf file (survey.dbf). The dbf file consisted of

23 fields and 250 rows. This data was taken from a survey. Eight columns were utilized to describe the demographic information of the participants. These variables are listed in Table #1. Five questions or fields were utilized to measure how much emphasis the participants placed on material wealth. These fields are listed in Table #2. Five questions or fields were utilized to measure participants involvement with religious and social organizations. These fields are listed in Table #3. Five questions or fields were utilized to measure participants opinions about American-made products. These fields are listed in Table #4.

Table 1:Demographics

<u>Fields</u>	<u>Default Type</u>
id	numeric
age	numeric
sex	numeric
race	numeric
relig	numeric
marital	numeric
educ	numeric
born	numeric

Table 2:Material Wealth

<u>Fields</u>	<u>Default Type</u>
q2a	numeric
q2b	numeric
q2c	numeric
q2d	numeric
q2e	numeric

Table 3:Religious Involvement

<u>Fields</u>	<u>Default Type</u>
q3a	numeric
q3b	numeric
q3c	numeric
q3d	numeric
q3e	numeric

Table 4: American-Made Products

<u>Fields</u>	<u>Default Type</u>
q4a	numeric
q4b	numeric
q4c	numeric
q4d	numeric
q4e	numeric

ACCESS METHODS

The SAS System for Windows interacts with a dbf file using SAS/ACCESS descriptor files. There are two types of descriptor files:

- access descriptors
- view descriptors

An *access descriptor* describes the data that is in the dbf file and has a member type of ACCESS. The information includes:

- filename
- field or column names
- field or column types
- default sas variable names
- default sas formats

The access file can be used as a master to create several view descriptors.

A *view descriptor* defines the data that was described by the access descriptor. A view descriptor can define all or just a subset of the data that is in the dbf file. A view descriptor is a SAS data set with a member type of VIEW.

This section will describe the following steps:

- How to create access descriptor named DBFALIB.SURVEY
- How to create view descriptor named DBFVLIB.USASURV

The quickest and sometimes most efficient way of creating the access and view descriptors is in a single PROC ACCESS execution. The following example

provides the SAS code that was utilized to create the descriptor for the data described in the previous section.

```
libname dbfalib 'c:\saswin\saslib\';
libname dbfvlib 'c:\saswin\sasview\';

proc access dbms=dbf;
    create dbfalib.survey.access;
    path = "c:\excel\survey";
    assign=yes;
    create dbfvlib.usasurv.view;
    select all;
run;
```

An explanation of some of the statements that can be utilized are as follows:

libname

utilized to establish the librefs with their respective SAS data libraries.

proc access

invokes the access procedure.

dbms=

specifies the interface engine to be utilized (dbf,wkn,etc.).

create

identifies the access descriptor or view descriptor to be created.

path=

identifies the complete path and filename of the file that contains the data.

assign=

indicates whether or not the variable names and formats can be changed. A value of yes indicates that they can be modified. A value of no indicates that they cannot be modified.

select

selects some or all of the columns to be included in the view descriptor. It is required if the view is to be created.

run

processes the last view descriptor.

Some additional statements that can be utilized as part of the PROC ACCESS are as follows:

rename

renames the SAS variables that were created from the field or column names.

format

assign SAS formats to the SAS variables.

list all

lists the variables, columns, and formats in the SAS log.

subset where

provides where clause processing.

The PROC ACCESS statements can be executed from the program editor or in batch mode.

THE FLEXIBILITY OF DESCRIPTOR FILES

The flexibility of the descriptor files, in comparison to previous methods of accessing PC Files, is obvious when we focus on the power of the view descriptor. The view descriptor can be utilized in any SAS procedure just as you would utilize other SAS data sets. For example, the following code could be used to print the data that was described by the descriptor files in the previous section.

```
proc print data=dbfvlb.usasurv;  
title '1995 Survey Data';  
run;
```

The data is not retrieved until it is needed by the SAS procedure. The flexibility of the descriptor files is further demonstrated by the fact that you can change the variable type. Also, several view descriptors can be created with a single execution of the access procedure. If utilized as described by this paper, the descriptor files provides the SAS programmer with a flexible and efficient tool.

CONCLUSION

The SAS/ACCESS method offers a lot of flexibility as demonstrated in this paper. The command language demonstrated in the previous section was utilized, in a similar manner, to access databases created by Borland Dbase and Paradox. Since SAS/ACCESS is available for other platforms, the techniques employed may be useful for reading other types of files as well.

ACKNOWLEDGEMENT

The author wishes to thank Jeannie McElhannon for word processing support and the PC Interface Support Group at SAS Institute for technical support. The author thanks Joe Kelley, Greg Nelson, and Barbie Matthews for comments, suggestions, and support throughout the development of this paper. The author can be contacted at:

The University of Georgia
UCNS
Computer Services Annex
Athens, Ga. 30602-1911
(706)542-5359
gmatthew@uga.cc.uga.edu

REFERENCES

1. Getting Started with SAS/ACCESS Software, Version 6, First Edition, SAS Institute.
2. SAS/ACCESS Interface to PC File Formats, Usage and Reference, Version 6, First Edition, SAS Institute.

SAS is a registered trademark of SAS Institute, Inc., in the USA and other countries.

Dbase is a registered trademark of Borland, Inc.

Paradox is a registered trademark of Borland, Inc.

Microsoft Excel is a registered trademark of Microsoft, Inc.

Windows is a registered trademark of Microsoft, Inc.