The SYSTEM 2000® Interface with the SAS® System: When to Use It to Store Your SAS Data  
Carl F. Carpenter, SAS Institute Inc., Austin, TX

ABSTRACT

SYSTEM 2000® software offers users of the SAS® System a variety of options to improve processing performance, DASD utilization, and user access of their SAS data. A tutorial approach is used to show how SAS Institute's SYSTEM 2000 Data Management Software can be used as a storage vehicle for SAS data to give the SAS user all of these enhancements and more.

INTRODUCTION

The purpose of this paper is to familiarize you with the enhancements available through the use of the SYSTEM 2000 interfaces to the SAS System and to give you some general measurement criteria to help determine whether SYSTEM 2000 software should be used as your SAS data storage medium. Typical SAS System usage is reviewed and correlated with what SYSTEM 2000 Data Management Software can provide. The SYSTEM 2000 Interfaces to the SAS System are discussed briefly to provide you with an overview of how data residing in SYSTEM 2000 data structures can be accessed by specific SAS procedures as well as how data can be transported easily between SAS data sets and SYSTEM 2000 data structures.

The UNISYS® version of SYSTEM 2000 software has special interfaces to the SAS System running on a PC. The IBM® version of SYSTEM 2000 software can be accessed from a mainframe SAS session as well as from a PC-based SAS session. Hereafter, all references in this paper to computers include IBM series computers only.

THE SAS SYSTEM AS A DATA MANAGEMENT TOOL

The SAS System is well known for its ability to handle all forms of computerized data. The DATA step can read any data format that is available on computer and can create SAS data sets or end-user reports from this input. Users have come to rely on this power and flexibility to help them complete all forms of job tasks to the extent that the SAS System is no longer used exclusively as a statistical modeling tool.

The underlying method of retrieving SAS data is sequential. Whether they are large or small, most SAS data sets must be read from beginning to end. Sequential access methods are very efficient when a majority of the data in the data set is actually needed by the process. As the amount of data needed by a particular application decreases, the benefits afforded by access methods other than sequential become much more important. A common rule of thumb is that less than 50% of your data is needed by a process, then a sequential access method would likely be less efficient than another access methodology. The 50% rule occurs most often when individuals are attempting to share common data structures in which any one user requires only a small portion of the total data.

Many SAS users avoid accessing unneeded data by keeping many smaller data sets that are related to particular reporting or processing requirements. These customers reap the benefits of the power and flexibility of the SAS System as well as efficient data access for these applications. However, this approach can be enhanced. Consider a large database where over 100,000 observations exist. Each observation contains expense information and the database covers ten years. If the data are broken down by areas of responsibility such as department, data set merging and, most likely, a separate SORT procedure step before the merge are required each time users want to combine data from multiple departments. Conversely, if the data are kept in a single, large data set, then each time a department report is required, every record within the SAS data set must be read in order to create an appropriate subset of data.

The process just described is referred to as preprocessing of SAS data sets. The SAS System without the SYSTEM 2000 interfaces provides many user-friendly tools that enable data sharing; the addition of SYSTEM 2000 software provides a cost effective means of substantially reducing preprocessing of SAS data sets.

BENEFITS OF USING THE SYSTEM 2000 INTERFACES TO THE SAS SYSTEM

SYSTEM 2000 provides a variety of options that can further enhance the data processing methods used by the SAS System. Specifically, SYSTEM 2000 software can provide improved performance of Input/Output (I/O), reduce DASD requirements, and enhance access to data. In addition, SYSTEM 2000 software provides extensive security options, data integrity features, an English-like query language, and a complete Multi-User facility.

Improved I/O Performance

In the area of I/O performance, SYSTEM 2000 software can provide significant improvements to the SAS System's sequential processing methods. SYSTEM 2000 software can access data either sequentially or directly, based on whether the request is KEYed or NON-KEYed. KEYed requests make use of SYSTEM 2000 software's inverted tables. The inverted tables are used to keep track of the location of the data within the database depending on the values of items within a record. By using an administrative tool provided by SYSTEM 2000 software, referred to as the DEFINE processor, a user has the ability to identify the KEYed attributes of each item within a database. The user can take advantage of these KEYs during the data selection phase in the WHERE process. Selecting data based on KEYed items rather than NON-KEYed items can reduce greatly the I/O requirements of the data selection process. With the use of SYSTEM 2000 software, data processing improvements are realized in most data sets containing more than 10,000 observations. With 50,000 observations, performance improvements are substantial, particularly for SAS/FSP® users taking full advantage of the direct interface between the SAS System and SYSTEM 2000 software.

Recall the previous example of a database with 100,000 observations, and consider that DEPARTMENT is an Item within an observation in that example. If this data set had 1,000 departments and the department is coded into an 8-byte field, it is almost certain that any single department's records will be identified with several I/Os by SYSTEM 2000 software. Assuming that a department would have at most 1000 records, it can be concluded that SYSTEM 2000 software would be able to subset any single department within about 1100 I/O operations instead of the 100,000 required by the SAS System.
Note that SAS users can arrange their data into sequences such that I/O would terminate upon finding a department within a collating sequence beyond the sequence of the requested department. However, an average of all users accessing the data would require 5 times the access, and that would only work for one item, such as DEPARTMENT, within the database without additional processing.

Reduced DASD Requirements

Another area where SYSTEM 2000 software can enhance resource utilization is DASD (disk space) usage. Again, as the data set increases in size, the benefits obtained are magnified, but on data sets with as few as 8,000 observations, the savings are measurable. Even with Indexes, SYSTEM 2000 data files actually take less space than a SAS data set containing the same information. All of this is accomplished by reducing data redundancy and using the space a data file has more efficiently through the use of variable length fields.

Within the SAS System, character data are defined as fixed length fields of data within an observation, and every observation must carry the maximum field length for possible data values without regard to actual usage. In a file that contains many descriptions, such as DEPARTMENT NAME, EMPLOYEE NAME, COMPANY NAME, or any such commonly occurring data, it is quite common to allocate 30 to 40 bytes of data for each field, even though the average number of significant characters in any field within any observation can be as small as 13. With SYSTEM 2000 software, you can specify the length to be the average, 13 characters, and values greater than 13 characters are stored in variable length overflow tables. Efficient pointers to this data make the retrieval process for the overflowing values almost as fast as if they were in the same actual records. DASD savings can be substantial as illustrated in Example 1.

<table>
<thead>
<tr>
<th>field name</th>
<th>Average Val. Size</th>
<th>SAS # Cylinders</th>
<th>Size</th>
<th>Size</th>
<th>Tracks</th>
<th>Tracks</th>
</tr>
</thead>
<tbody>
<tr>
<td>City</td>
<td>8</td>
<td>30</td>
<td>5</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Last Name</td>
<td>13</td>
<td>60</td>
<td>9</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Comments</td>
<td>27</td>
<td>36</td>
<td>83</td>
<td>19</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Address</td>
<td>15</td>
<td>25</td>
<td>6</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Process Name</td>
<td>10</td>
<td>50</td>
<td>11</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>131</td>
<td>135</td>
<td>75</td>
<td>36</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Net savings of 45 tracks/ 3 3180 cylinders

Example 1 10,000 Observation Data Set

Eliminating data redundancy is another way SYSTEM 2000 software reduces space usage. The hierarchical structure of SYSTEM 2000 databases enables the user to take advantage of naturally occurring hierarchies within the data. In the 100,000 observation database with 1,000 departments, if departmental data occupied 100 bytes, 9,000,000 bytes could be saved by using SYSTEM 2000 software as the storage vehicle, yet the user can view the data as if it were a part of each observation. Example 2 shows how these two benefits translate into savings for an organization from two actual applications that SAS Institute has converted from SAS data sets to SYSTEM 2000 data files.

<table>
<thead>
<tr>
<th>Application</th>
<th>SAS Data Set Cylinders</th>
<th>DASD Requirement Cylinders</th>
<th>Annual Savings</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application 1</td>
<td>100</td>
<td>60</td>
<td>$6,450</td>
</tr>
<tr>
<td>Application 2</td>
<td>550</td>
<td>150</td>
<td>$18,600</td>
</tr>
<tr>
<td>Total</td>
<td>670</td>
<td>190</td>
<td>$25,050</td>
</tr>
</tbody>
</table>

Note: 1 3180 cylinders costs $800.00/year.

Example 2 Savings from Using SYSTEM 2000 Software

Enhanced Access to User Data

Users of the SYSTEM 2000 interface to the SAS System can avoid preprocessing of their SAS data. This can be accomplished efficiently through the SAS interface procedures. More importantly, users have shared access to their data.

All of the SAS procedures to which SYSTEM 2000 software has interfaces have new commands that offer increased functions, such as a WHERE processor to return the data requested based on a single or multiple conditions that the data values must satisfy. Use of WHERE processing enables the user to subset data prior to processing, thus eliminating preprocessing needs.

In addition, sorting also can be accomplished through the SAS interface procedures. Syntactically, sorting is accomplished through the use of an ORDERBY statement. This ensures that the data are returned to the user in the desired sequence, eliminating a separate sorting procedure and, many times, eliminating the space required for a sorted copy of the data.

Finally, the most significant access enhancement available to the SAS user is the ability to share data in either browse or update mode. SYSTEM 2000 software provides various levels of data locking. Data locking is the ability of the software to control access to data by several concurrent users. With SYSTEM 2000 software's data locking capabilities, the user always has the latest copy of an observation, regardless of how many users are attempting to access or update the observation.

The Multi-User feature of SYSTEM 2000 software is required to implement shared data access. Multi-User is a control program that is executed in an address space that is separate from the user's TSO or batch address space. Special interaddress space communication vehicles are used that enable the user region and the Multi-User control program to interchange requests and data. With Multi-User, users can share large amounts of data, thus significantly reducing the cost of preprocessing SAS data.

Increased Data Integrity

Data integrity becomes important when real-time concurrent updating of shared data is implemented. One of the most critical issues regarding data integrity is the ability to recover from various types of outages and user errors. SYSTEM 2000 software enables a user to use two methods of recovery, either separately or in tandem.

A transaction log—referred to as a KEEPFILE, can be invoked that keeps track of every change made to a database. In order to initiate this processing, a database must be SAVED in a special SYSTEM 2000 format. SYSTEM 2000 software provides a full set of commands to SAVE, RESTORE, and APPLY updates to databases.
To invoke recovery using this recovery vehicle, a database must be restored and then the updates contained in the keepfile transaction log are applied to the restored database. With this method of recovery, an accidental update to the system can sometimes be nullified. This form of recovery has the least impact on real-time updates but requires greater resources to accomplish the restoration.

A second method of recovery is a real-time recovery method referred to as Rollback. Rollback protects a user from system outages and other similar disasters; it cannot be used to nullify accidental updates as could be done using the keepfile transaction log. Although this method increases the overhead associated with updates, the ability to recover from system outages without user intervention is often worth the expense. In addition, the System 2000 interface to SAS via the FSEdit procedure requires Rollback to be activated for a database if the user wants to guarantee that partial updates do not occur.

Data Security Provided by System 2000 Software

Security and administration of data become critical issues when many users are trying to access the same data. System 2000 software enables the user to restrict access to data at an item level. System 2000 software provides for many user-defined passwords. Each password can define various combinations of read and write access authorities for each group of items related to the password.

Consider a database containing five items named ITEM1 through ITEM5. A password could be defined giving read access to ITEM1, ITEMS2, and ITEM4. Not only would a user who had access to this password be able to retrieve only these three items, but System 2000 software would refuse to acknowledge the existence of ITEM2 and ITEM5, giving the user the appearance of having a separate set of data.

System 2000 Interfaces to the SAS System

Currently, there are five SAS procedures that can access System 2000 data files: they are the FSEdit, FSbrowse, S2k, S2kload, and Quest procedures. PROC QUEST is a procedure that enables the SAS user to access System 2000 databases with System 2000's English-like native language. PROC FSEdit and PROC FSbrowse are only available if SAS/FS/4 software is installed.

PROC QUEST is not discussed at length in this paper because it does not relate to the use of existing SAS data sets. However, it is important to note that the administrative functions and data integrity features of System 2000 software are available via the SAS System only through this procedure.

In addition to the SAS procedures mentioned above, System 2000 software offers users the ability to access databases with COBOL, FORTRAN, PL/1, and IBM ALC (Assembler Language Coding) programming languages. The programming language interfaces are not discussed in this paper but are mentioned for readers who may have a requirement for such access.

Creating System 2000 Databases Using SAS Procedures

Two procedures are available to create System 2000 databases, PROC S2kload and PROC FSEdit. These procedures enable a user to define the relationships that exist between items within a SAS data set and the corresponding items defined in a System 2000 database.

Using a predefined database, PROC S2kload, and a SAS data set, a user can move data easily from SAS data sets to System 2000 data structures. Example 3 is the panel that allows a user to define the relationships between a SAS data set and System 2000 data structures.

Example 3 PROC S2kload Selection Panel

PROC S2kload allows the user to define the hierarchies, if any, that should be used when loading the database. If such hierarchies are defined, the SAS data must be sorted into an ascending sequence that corresponds to the hierarchy.

PROC FSEdit, a full-screen editor, enables the user to create a new System 2000 database, using all of the screen definition features available within PROC FSEdit. These are the same control features used with SAS data sets plus several others that are specific to System 2000 databases.

Example 4 is the panel displayed when the user specifies NEWDB—BANKTWO when invoking PROC FSEdit.

Example 4 PROC FSEdit Definition Panel without System 2000 Software

Example 5 is the panel displayed when the user specifies NEWDB—BANKTWO when invoking PROC FSEdit.
The major difference between SAS System and SYSTEM 2000 definition panels is the addition of fields required by SYSTEM 2000 software to define the data structures. All of the information required by the SAS System is still present.

Accessing SYSTEM 2000 Databases Using SAS Procedures

PROC FSEDIT and PROC FSXBROWSE enable users to access SYSTEM 2000 databases via the SAS System. They differ only in that PROC FSXBROWSE users cannot modify their data. With these procedures, SAS users have all of the features that are available when accessing SAS data sets, such as custom screen definition, data validating, record scrolling, and so on. However, when editing a SYSTEM 2000 database, users can subset their data with the use of a WHERE clause and sort the resulting subset of data with an ORDERBY statement.

Upon invocation of either of these procedures, the user is asked to select items from the database that are to be displayed by PROC FSEDIT or PROC FSXBROWSE. Furthermore, the user can define the SAS data format for the item if it should differ from the SYSTEM 2000 definition. However, the user cannot define a SAS data item to be numeric when the SYSTEM 2000 definition is character. For example, if the SYSTEM 2000 format is $CHAR20, the SAS data format could be $30 or $CHAR20, and so on.

Example 6 is an example of the panel displayed when accessing a SYSTEM 2000 database with the FSEDIT or FSXBROWSE procedure.

Once the user has completed the selection phase, SYSTEM 2000 manages any existing hierarchies within the definition without further user intervention.

Creating SAS Data Sets from SYSTEM 2000 Databases

Users who choose to manage their data using the SYSTEM 2000 interfaces have many requirements for the data to be in SAS data set formats. To get their data in SAS data set formats, users can invoke PROC S2K or use the OUTPUT command of PROC FSEDIT.

PROC S2K enables the user to select items from the database that are to be a part of a SAS data set. The selection panel displayed to the user is the same panel displayed when the FSEDIT and FSXBROWSE procedures are invoked. Again, each user is able to define SAS data formats that are different than the defined SYSTEM 2000 format.

PROC FSEDIT enables a user to create a SAS data set in an ad hoc fashion. The user invokes PROC FSEDIT, completes the subsetting or sorting that may be required, and enters the command OUTPUT on the command line of the FSEDIT screen. This automatically creates a new SAS data set for use with other SAS analytical tools. The SAS data set can be temporary (WORK.<data-set-name>) or permanent (USER.<data-set-name>). The OUTPUT command uses the SAS definition created when invoking PROC FSEDIT.

CONCLUSION: IS SYSTEM 2000 SOFTWARE THE RIGHT CHOICE?

In using the SAS System, you have made the right choice for the most comprehensive set of data analysis tools available in the world today. As the amount of data that you process grows, SYSTEM 2000 software enables you to continue to use these state-of-the-art tools with the same ease and performance you have come to expect.

SYSTEM 2000 software should be considered as a storage management vehicle for your SAS data if you find that the following statements are becoming more the rule than the exception:

- The data need to be shared by several users.
- More efficiency is required when subsetting and retrieving data.
• The data set is becoming very large, or it has a high percentage of redundant data.
• Data sets must be merged or preprocessed prior to the beginning of an application.
• Updates are becoming increasingly difficult to coordinate.
• The data are highly sensitive and control to the data must be tightly controlled.
• Data lost due to CPU failure or power outage is not acceptable.

The SAS interfaces to SYSTEM 2000 software can provide numerous benefits to the SAS user. SYSTEM 2000 software can significantly reduce the costs associated with maintaining data files. In addition, it offers features that enhance data integrity. Finally, SAS procedures offer enhanced access to SYSTEM 2000 data files.

The following data characteristics can help identify likely candidates for SYSTEM 2000 data structures:
• Data contain a large number of character data.
• There are a large number of observations (10K to 20K).
• There are many groups of commonly occurring data (for example, in a table).
• Data need to be shared by many users.
• Data have security requirements below the data set level.

SAS, SASIFSP, and SYSTEM 2000 are registered trademarks of SAS Institute Inc., Cary, NC, USA.

UNISYS is a trademark of UNISYS Corporation.