

Paper 193-25

SAS ® USER SUPPORT AND THE INTRANET: WEB PAGES, IN-HOUSE DOCUMENTATION, AND SAMPLE LIBRARY

Bruce Gilson, Federal Reserve Board
Scott Hoenig, Federal Reserve Board

INTRODUCTION

The SAS ® system is used at the Federal Reserve Board on the MVS ®, UNIX, and PC platforms for tasks ranging from data processing applications to econometric research. The Board's in-house SAS consulting group, comprised of one full-time and two part-time consultants, provides SAS support at the Board, including installation, testing, and cutover of SAS software; technical consulting via telephone hotline, e-mail, and office visits; and manual distribution. In 1998, over 800 users executed the SAS system on Board computers, and 1250 requests for technical assistance and 350 requests for documentation and SAS training information were received.

In the last few years, the Board's intranet, FedWeb, has become an important component of the Board's operating environment. This paper describes how FedWeb has been used to improve SAS support at the Board. The following topics are discussed.

1. The SAS consulting web pages on FedWeb.
2. In-house SAS documentation on FedWeb.
3. In-house SAS sample library on FedWeb.
4. SAS Tips articles in the *Information Exchange*.

Also discussed are some issues that were considered as the web pages were developed, such as the following.

1. Should large web documents be stored in one HTML file or multiple HTML files? What type of printed table of contents, if any, should be generated for a web document displayed without page numbers in a browser?
2. How should online sample programs be formatted? How much HTML, if any, should they contain?

SAS CONSULTING PAGES ON FEDWEB

The SAS consulting web pages were first added to FedWeb in January 1996. The SAS consulting home page, shown in **Figure 1**, contains links to the other SAS consulting web pages, some of which are discussed elsewhere in the paper.

IN-HOUSE SAS DOCUMENTS ON FEDWEB

The documents on FedWeb include the following.

1. SAS user's guides that document site-specific information, such as the SAS software products licensed on each platform, how to invoke the SAS system using in-house scripts or programs, how to obtain assistance, and the macros in the Board's in-house SAS macro libraries.
2. Papers by Board staff that were presented at previous SAS conferences or at in-house SAS users group meetings.

How the documents were written

1. Most new in-house SAS documents are being written as web pages using HTML.
2. Some documents were written with Word or WordPerfect. Master copies of these documents are maintained in Word or WordPerfect, and converted to PDF files for web display with Acrobat when they are updated.
3. Master copies of a few old, large documents written with IBM script were converted to HTML files. The most time-efficient way to accomplish this task was to write Perl scripts to generate nearly usable HTML files, and fine tune the files by hand.

Design considerations for "web documents"

In this paper, larger documents written with HTML are called "web documents," as distinguished from documents containing one or two screens of information. Some design considerations for web documents are as follows.

1. Simple, text-based content is preferred to fancier graphical content.
2. Web documents are displayed with two frames, as follows.
 - A. The left-most 25% of the browser contains a scrollable table of contents frame with links to each

chapter or section of the document.

B. The frame in the rest of the browser displays the document.

3. An entire web document is stored in a single file. An alternate approach is to store each chapter or section of a web document in a separate file. Some advantages and disadvantages of each approach are as follows.

A. A single file allows users to scroll through one contiguous document. Separate files requires users to select a link whenever they want to see another chapter or section.

B. A single file makes it easier for users to print an entire document. However, separate files makes it easier for users to bookmark or print a chapter or section.

C. A single file allows users to search for text in an entire document using their browser's "Find in page" or "Find in frame" command (the Board's internal search engine is not yet in production).

D. A single file is easier for developers to update, using common editing tools such as "Find and replace" or "Replace all." This is especially true when the same change is necessary in multiple locations.

E. A single file takes longer to load into a browser. The largest document currently on the SAS consulting web pages, 314,000 bytes (307Kbytes), loads without delay. Documents that are very large or contain many graphical images could load noticeably slower.

4. Updating a web document in one location is much easier than updating a widely distributed hard copy document. However, frequent updates can confuse users. Including a publication date and a "Changes to this document" section in web documents avoids confusion by informing users what changes were made and when they occurred.

5. Since web documents do not have page numbers, detailed chapter and section numbers were added when hard copy documents were converted to web documents. Users receiving technical consulting assistance who were previously told, "see example 2 on page 37," can now be told, "see example 2 in section 4.3.2." Users can find section 4.3.2 any of the following ways.

A. Select the link to section 4.3.2 in the table of contents frame (if there is a table of contents frame).

B. Scroll through the document to locate section 4.3.2.

C. Use the "Find in page" or "Find in frame" command.

6. Printed copies of web documents were generated. For documents with a table of contents frame, the document and the table of contents were both printed, but it was unclear how to handle page numbers, given the following.

A. Page numbers are not shown if a web document is viewed in a browser, but are generated if a document is printed from within a browser. Page numbers can change any time a document is updated and reprinted.

B. A table of contents frame has links to a document's chapters and sections but no page numbers.

Initially, printed tables of contents with page numbers were generated as follows: copy a table of contents frame to a text file, remove all HTML tags, and manually add page numbers from the printed copy of the web document. In the future, tables of contents will just be printed from within a browser, and page numbers will not be added, for the following reasons.

A. Users have made almost no requests for printed copies of web documents.

B. The page numbers need to be manually updated every time the document is updated.

Example of a web document

Figure 2 shows a small section of the document, "The SAS System for UNIX at the Federal Reserve Board."

IN-HOUSE SAS SAMPLE LIBRARY ON FEDWEB

An in-house SAS sample program library was added to FedWeb in October 1999. The library initially included 88 programs, and more programs will be added in the future.

Most of the programs were originally written to assist SAS users at the Board. Some programs were written to show a user how to accomplish a task, and others were written to debug a user error.

Sample program design considerations

Users are expected to employ the programs in one of the following ways.

1. To learn by example how to accomplish a specific task, they can view a program online or copy it to a file or a SAS session and execute it.

2. To accomplish a small piece of a larger programming

task, they can copy a program into their application and change it as necessary to conform to their application.

Given the expected usage, the programs were designed as follows.

1. The programs are stored as text files with no HTML tags, as described in the next section.
2. Most of the programs are small and simple, and perform a clearly defined task.
3. To make the programs self-contained, data used in a program is created in the program. To make the programs easy to incorporate into an application, data is created at the top of the program, before the rest of the code.
4. Some programs (regressions, frequency counts, etc.) require certain types of data values and a meaningful quantity of data. In other cases, the programs use as little data as possible, and the following conventions are used.

A. The number of observations and variables is similar but not equal. For example, three observations/two variables or two observations/three variables is used instead of three observations/three variables.

B. Data values are unique. For data with three observations/two variables or two observations/three variables, the values 1-6 are frequently used.

These conventions make it easier to follow the flow of data values as they are transformed in a program.

5. Most of the programs are independent of the platform and release of SAS software. Programs specific to certain platforms or releases of SAS software contain documentation at the top noting their limitations.

6. Users can browse the programs from one of the following web pages.

A. An alphabetical list of programs displayed in a two column HTML table. The left column contains alphabetical links to the programs, and the right column contains program summaries. See **Figure 3**.

B. A list of programs ordered by topic (program task) and displayed with two frames, as shown in **Figure 4**.

(1) The frame in the left-most 25% of the browser contains links to the topics.

(2) The frame in the rest of the browser contains a two column HTML table. The left column contains links to the programs ordered by topic. The right column contains program summaries.

HTML tags and the sample programs

The sample programs needed to be readable, with comments, indentation for loops, and a consistent style. The following alternatives were considered when deciding how much HTML, if any, to include in the programs.

1. Interleaving HTML tags in the code makes the programs look good in a browser, but users must remove the tags to execute the programs or insert them into their applications.

2. Maintaining one copy of the programs with HTML tags for display and one without HTML tags for users to employ requires a duplication of effort to maintain.

3. Making the programs text files (files with no HTML tags and a .txt extension) allows users to right-click on a link to a program, select Save Link As (or their browser's equivalent), and save the program to a file they can execute or include in a SAS program without modification.

4. If text files are bookmarked, the titles of the files are blank in Bookmark menus, because text files do not contain an HTML TITLE tag. If clear bookmarks are important, another approach is to make the programs HTML files with only two lines of HTML, as follows.

at the top (abc.sas is the name of the program):

```
<hml><head><title>abc.sas</title></head><pre>
```

at the bottom:

```
</pre></body></html>
```

To make the programs simple and easy to use, they were set up as text files with a .txt extension and no HTML tags. **Figure 5** shows a sample program, putnosp.txt.

SAS TIPS ARTICLES

SAS Tips articles have been published in the Board's computing newsletter, the *Information Exchange*, since March 1990. The *Information Exchange* is published bi-monthly (monthly until October 1998), and switched from a hard copy newsletter to a webzine in January 1994.

The SAS Tips articles answer questions from users, offer tips on problem resolution, and highlight changes and enhancements to SAS software.

Selecting the SAS Tips Articles in the *Information Exchange* link on the SAS consulting home page displays an article index, as shown in **Figure 6**. **Figure 7** shows a recent article.

SAS CONSULTING WEB PAGES: USAGE

Since the SAS consulting web pages were added to FedWeb in January 1996, the SAS consulting home page has averaged about 100 hits per month, and the other pages have averaged 15-30 hits per month.

Prior to events such as SAS Board Users Group meetings and software cutovers, a link to one of the SAS consulting web pages is added to the weekly schedule on *Inside the Board*, the FedWeb page that appears on most Board PCs when Netscape is invoked. The number of hits to the SAS consulting web pages increases during those weeks.

Board SAS consultants employ the SAS consulting web pages when they provide consulting assistance. When applicable, users are directed to a sample program, SAS Tips article, or in-house SAS document (most often "The SAS System for UNIX at the Federal Reserve Board") that answers their question. Besides being an effective way to assist users, this also familiarizes users with the web pages.

FUTURE PLANS

The Board's conversion plan for converting from Version 6 to Version 8 will be added to the SAS consulting web pages. New pages will include a conversion timeline and a changes and enhancements guide. As new information becomes available or new problems are identified (which can happen frequently during a conversion), the ease with which web documents can be updated will be very helpful.

Other pages might be added based on user requests or needs determined by the Board's SAS consulting group.

CONCLUSION

This paper described how the Federal Reserve Board's intranet, FedWeb, has been used to improve SAS support at the Board. It described the SAS consulting web pages, in-house SAS documentation, in-house SAS sample library, and SAS Tips articles added to FedWeb. The authors of this paper hope that other sites can benefit from this information.

For more information, contact

Bruce Gilsen
Federal Reserve Board, Mail Stop 171
Washington, DC 20551
phone: 202-452-2494
e-mail: m1bfg00@frb.gov

Scott Hoenig
Federal Reserve Board, Mail Stop 171
Washington, DC 20551
phone: 202-452-3516
e-mail: scott.hoenig@frb.gov

REFERENCES

Gilsen, Bruce, and Shankman, Bert (1992), "Improving SAS System Support (Version 6) at a Large Site," in the Proceedings of the Seventeenth Annual SAS Users Group International Conference, 17, 1459-1468.

ACKNOWLEDGMENTS

The following people contributed extensively to the development of this paper: Steve Feder, Donna Hill, and Steve Taubman at the Federal Reserve Board. Their support is greatly appreciated.

TRADEMARK INFORMATION

SAS, SAS/GRAPH, and SAS/TUTOR software are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. MVS is a registered trademark or trademark of International Business Machines Corporation. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.



- ① Includes testing and cutover information for new SAS software releases.
- ② News messages initially distributed to Board UNIX SAS users by e-mail.
- ③ Board training options (classes, videos, SAS/TUTOR® software, and CBTs), and a link to an internal training form.
- ④ UNIX SAS error notices from SAS Institute.
- ⑤ Brief summaries and links to Board-written SAS documents.
- ⑥ "SAS Tips" articles in the Board's bi-monthly computing newsletter, *The Information Exchange*.
- ⑦ See "In-House SAS Sample Library on FedWeb" section.
- ⑧ Information about the Board's in-house SAS users group.
- ⑨ Links to the home pages of national, regional, and local SAS users groups, reports from recent conferences, and future conference dates.
- ⑩ Pictures, phone numbers, and e-mail addresses for the Board's SAS consultants.
- ⑪ Hotline number and e-mail address.
- ⑫ The globe icon is a suggested way on FedWeb to indicate a link to an external site.

Figure 1. SAS Consulting home page

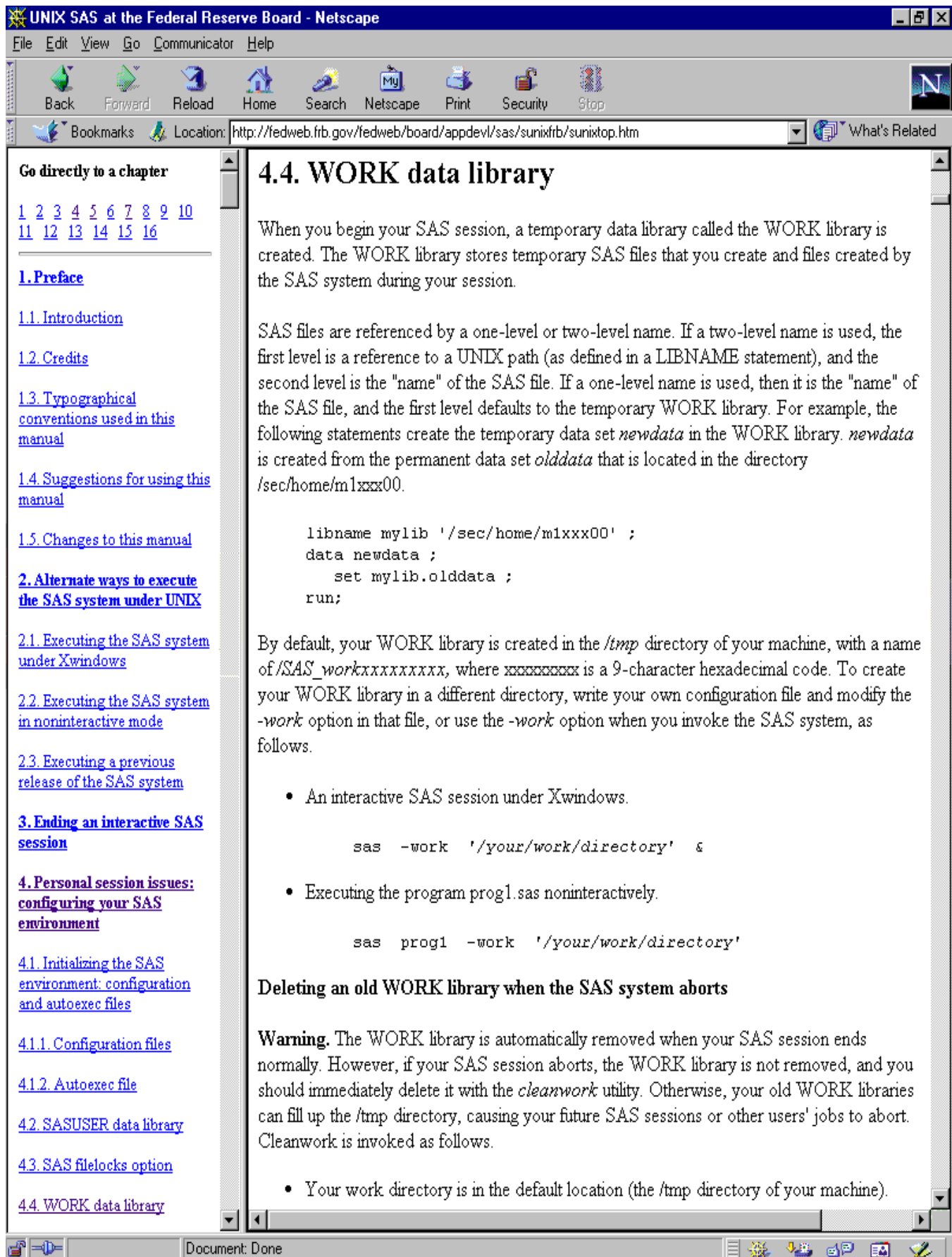


Figure 2. SAS System for UNIX at the Federal Reserve Board

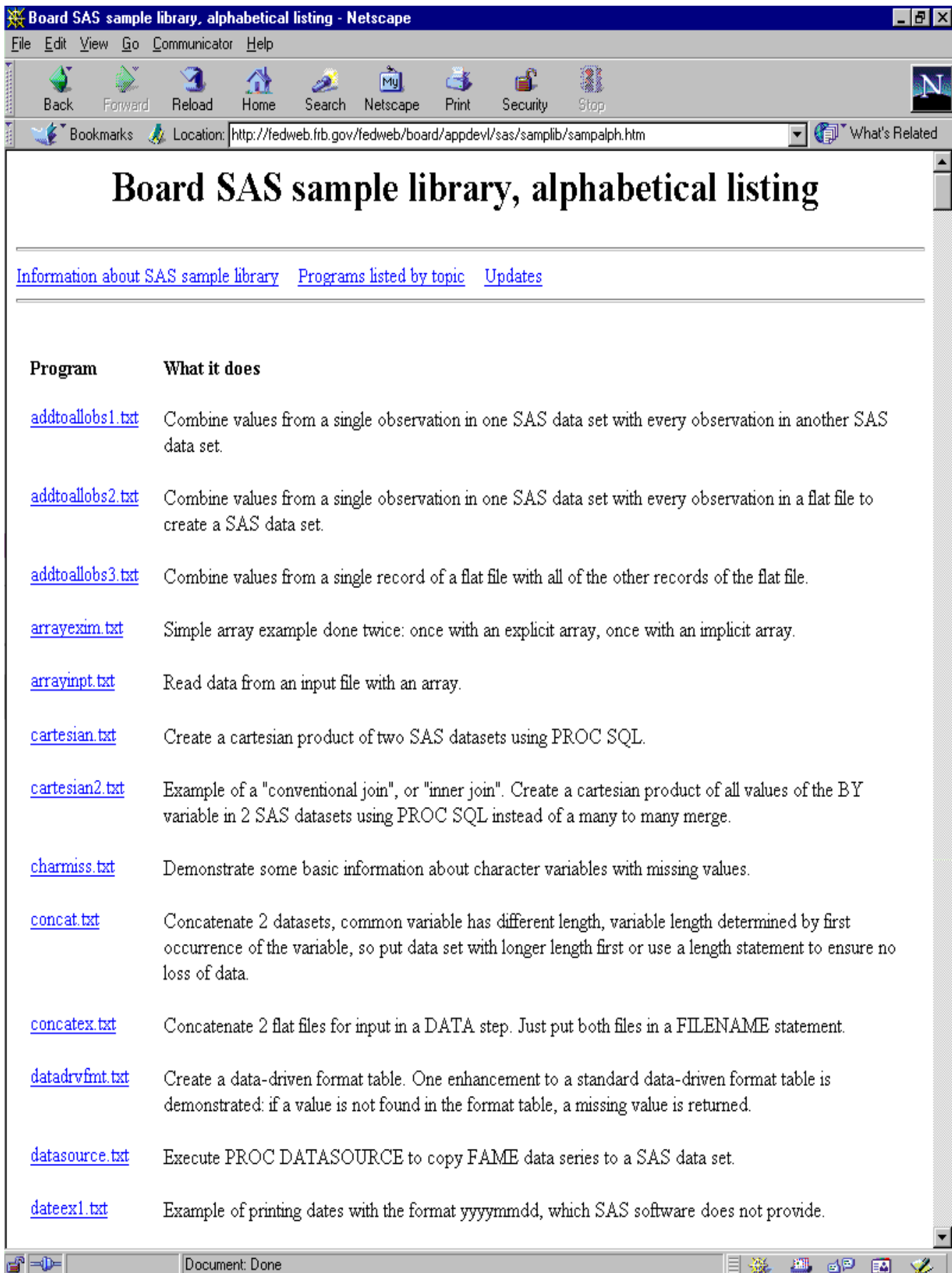


Figure 3. Sample library listed alphabetically

Board SAS sample library, listed by topic - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Stop

Bookmarks Location: <http://fedweb.frb.gov/fedweb/board/appdevl/sas/samplib/sampltop.htm> What's Related

1. Arrays
 2. Cartesian products
 3. Character variables and text
 4. Combining data (sorting, merging, concatenating, etc.)
 5. Converting variable type (character/numeric)
 6. Date handling
 7. FAME and SAS
 8. Formats and informats
 9. Graph
 10. Input (reading input into SAS)
 11. Lags and leads
 12. Macros
 13. Missing values
 14. Operating system (host) related
 15. Options
 16. Output
 17. Sorting
 18. SQL
 19. Statistics and econometrics
 20. Subsetting observations or variables
 21. Timeseries data
 22. Web output
 23. Where statement
 24. Version 8 sample programs

Board SAS sample library, listed by topic

[Information about sample program library](#) [Programs listed alphabetically](#) [Updates](#)

1. Arrays

[arrayexam.txt](#) Simple array example done twice: once with an explicit array, once with an implicit array.

[arrayinpt.txt](#) Read data from an input file with an array.

2. Cartesian products

[cartesian.txt](#) Create a cartesian product of two SAS datasets using PROC SQL.

[cartesian2.txt](#) Example of a "conventional join", or "inner join". Create a cartesian product of all values of the BY variable in 2 SAS datasets using PROC SQL instead of a many to many merge.

3. Character variables and text

[charmss.txt](#) Demonstrate some basic information about character variables with missing values.

[dateex2.txt](#) Examples of converting numeric and character values to SAS date values.

[sortchars.txt](#) Sort a SAS data set by the numeric part of a character variable. Just sorting by the original character variable causes (for example) bank101 to be less than bank2, etc. The desired order is bank1, bank2,..., bank999.

[strcount1.txt](#) Count how many times a character string is contained in another character string or character variable. Equivalent to strcount2.txt.

Document: Done

Figure 4. Sample library listed by topic


```

/* putnosp.txt */

/* Demonstrates list-style PUT statements with and */
/* without a space between the fields.          */

/* Bruce Gilson 6/99 */

/* create some sample data */
data one;
  input a b c;
  cards;
1 2 3
4 5 6
;run;

data two;
  set one;

  /* this PUT statement prints the 1st observation as: 1 ,2 ,3 */
  put a "," b "," c;

  /* this PUT statement removes the blank between each field
  and prints the 1st observation as: 1,2,3 */
  put a +(-1) "," b +(-1) "," c;
run;

```

Figure 5. Sample program putnosp.txt

INFOX
GO TO TOPICS

SAS TIPS
See also: [SAS](#)

- [SAS sample program library now on FedWeb.](#) *Nov/Dec 99*
- [A Y2K tip: Clearly displaying date variables copied from a DB2 table.](#) *Sep/Oct 99*
- [Using host commands inside conditional expressions, and replacing patterns in character variables.](#) *Jul/Aug 99*
- [Using KEEP and RENAME in a SET statement.](#) *May/June 99*
- [The in-house SAS Unix manual is now online.](#) *Mar/Apr 99*
- [The SAS Consulting Tips FAQ.](#) *Jan/Feb 99*
- [Creating a GIF or JPG file with SAS/GRAPH software.](#) *Nov/Dec 98*
- [Reading data from a flat file that contains non-display characters in Unix SAS.](#) *Oct 98*
- [Using BY variables in titles.](#) *Sep 98*
- [Copying SAS data sets from one platform to another.](#) *Aug 98*
- [Data values that could generate an error in the Year 2000.](#) *Jul 98*
- [A macro to center text in a DATA NULL step.](#) *Jun 98*
- [A little-known way to use wildcards in variable lists.](#) *May 98*
- [Millenia, a new SAS Year 2000 test tool.](#) *Apr 98*
- [Fuzzy merging: Comparing the BY variable to a range of values.](#) *Mar 98*
- [Sorting by the numeric part of a character variable.](#) *Feb 98*
- [Taming the multidimensional array.](#) *Jan 98*
- [The SAS Consulting Tips FAQ.](#) *Nov/Dec 97*
- [Determining the number of days in a month using SAS date values.](#) *Oct 97*
- [Creating SAS date values from data that does not correspond to a date informat.](#) *Sep 97*
- [Inserting a SAS graph in a WordPerfect for Unix document.](#) *Aug 97*
- [Adding or subtracting years to a SAS date value.](#) *Jul 97*
- [Converting a variable from character to numeric in a DATA step and keeping the](#)

Figure 6. SAS Tips article index

INFOX May/Jun 1999: SAS Tips - Netscape

File Edit View Go Communicator Help

Back Forward Reload Home Search Netscape Print Security Stop

Bookmarks Location: <http://infox.frb.gov/Issu99/mayjun/SASTips.shtml> What's Related

INFOX *May/Jun 1999* << 8 >>

Bruce Gilson
SAS Tips
[About](#)

Using KEEP and RENAME in a SET statement
 By Bruce Gilson, IRM

IN THIS ARTICLE
[KEEP= option](#)
[RENAME= option](#)
[Combining the KEEP= and RENAME= options](#)

Two options that can be used in the SET statement in a SAS data step are the KEEP= and RENAME= options. Since users frequently ask questions about these options, I will discuss them in this column.

In the following DATA step, SAS data set ONE is created, with two variables. Data set ONE will be used in the examples in this column.

```
data one;
  input var1 var2;
  cards;
1 2
3 4
5 6
run;
```

KEEP= option
 The KEEP= option in a SET, MERGE, or UPDATE statement limits the variables that are read from a data set. If the KEEP= option is used, the only variables from the existing data set that are available in the DATA step for processing are the variables specified in the KEEP= option.

The following DATA step illustrates the KEEP= option in a SET statement. Variable VAR1 is available for use in the following DATA step, and is written to data set TWO. Variable VAR2 is not available for use in the following DATA step, and is not written to data set TWO.

```
data two;
  set one (keep= var1);
  /* more SAS DATA step statements that can employ VAR1
```

Document Done

Figure 7. SAS Tips article