A CMS-based, Multi-User Planning System for Clinical Data using SAS/AF® and SAS/FSP®.

Thomas E. Schneider, The Upjohn Company
Mary Catherine Krug, The Upjohn Company

Overview

This paper describes a multi-user planning/tracking system that is used to project manpower estimates several years into the future. The system projects estimates across clinical projects and arranges manpower estimates by type of activity or by type of employee (i.e. statistician, programmer, etc.). By entering the start date of a clinical study along with a few other key variables, the system, based upon the model for that study, will automatically generate dates for all key activities in that study.

Several reports are produced by the system, including bar charts. When the report option is selected a template of the report or chart to be produced is displayed. The user can easily design the structure of the report by selecting the desired variables within the template.

The system was developed using Version 5.18 SAS/AF® and SAS/FSP® under the CMS environment. Several simple CMS EXEC’s are called from within the SAS/AF® system to provide multi-user access with data integrity, thus allowing multiple users to add, modify, or delete data on a central SAS dataset. Report output from the system is produced on an IBM® 3820 laser printer and graphical output is produced on an HP 7550 plotter.

Interesting Features

The system includes the ability to view the SAS Log and SAS Listing file of the interactive program while it is running. This is to assist in debugging during the test phase.

The system does not use SAS/SHARE® to provide multi-user access to the data base but rather uses an exec to do checking and linking to the minidisk where the data resides.

These features will be discussed in greater detail throughout the paper.

Description of Data

The planning/tracking data is currently stored in two SAS data sets. The first data set contains one observation (record) per protocol or clinical study. Each record contains information pertinent to that clinical study, such as the name of the study, therapeutic area, drug name, start date of study, description of the study, drug indication, etc. The second data set contains multiple observations associated with each clinical study. Each record contains an activity associated with the protocol along with the start date, stop date, duration, and manpower requirements for various manpower types associated with that activity. Some examples of activities are the design of the data management plan, Case Report Form Design and Production, Quality Control, Table Production, etc.

The Main Menu

The main menu of the system (see Figure 1) allows the user to select one of several sub-menus. Each of these sub-menus are described in more detail below. The main menu also allows the user to get general help about the system and to access CMS subset mode without having to completely exit out of the system.
Add/Update/Delete/Browse Data

Through this sub-menu the system allows users to add, update, delete or browse the data. Each time data is accessed an exec is called from within SAS to link to the minidisk where the data resides. This exec checks to see if someone is linked to the minidisk. If so, it will try 4 more times waiting 3 seconds in between each try. If access is approved, additional security and integrity checking is done depending upon the type of access that is required and depending upon certain assumptions we have made about users accessing the data before access to the data is allowed. All linking and checking is done transparent to the user.

By choosing the Update option the user will be allowed to make changes to data already existing on the data base. When this option is selected, the program first calls an exec to see if anyone is linked to the main disk containing the data base. If successful, it will check to see if someone else is currently updating the data (i.e. is making changes to a copy in their own area) by looking at an access status file that contains information about who is currently accessing the data. If no one else is updating, it will allow the user to update by placing their ID, type of access, etc. in the access status file and by making a copy of the data on temporary disk space which they can now freely update. The data is updated using PROC FSEDIT. When the user is finished updating, the data is merged back with the original data. While the user is updating the data (actually a copy of the data), no one else is allowed to update the data base.

Browsing data is done in the same way as updating data except that FSBROWSE is used to browse the copy of the data, and that the data is not merged with the original data after browsing. Also, there is no need to make a check for concurrent update.

For Adding data, it is assumed that no two people will be adding a project with exactly the same key values at exactly the same time. Information about the project is keyed into a temporary data set containing one observation through an FSEDIT screen. After exiting the FSEDIT screen, this new observation is then merged (after linking) to the original data.

To Delete a project, a temporary dataset containing only identifying fields about projects is created and displayed through FSEDIT. The user then selects which projects to delete by placing a 'D' in the appropriate place. After exiting FSEDIT, this temporary data set is then merged with the original data set, keeping only the observations that were not marked delete.

The Reporting Options

The reporting menu (see Figure 3) allows the
user to select 3 different reports.

**Figure 3: The Report Menu**

<table>
<thead>
<tr>
<th>REPORT MENU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Help</td>
</tr>
<tr>
<td>2. Enter CMS Subset Mode</td>
</tr>
<tr>
<td>3. Return to Main Menu</td>
</tr>
<tr>
<td>4. Print Last Report Generated</td>
</tr>
<tr>
<td>5. Display Last Report Generated</td>
</tr>
<tr>
<td>6. Subset Data for Reporting</td>
</tr>
<tr>
<td>7. FTE Summary Report</td>
</tr>
<tr>
<td>8. List Project ID Data</td>
</tr>
<tr>
<td>9. List Project Activity Detail</td>
</tr>
</tbody>
</table>

When a report is selected, the user can enter the information by which they want to print and the information they wish to display. The program then executes and the report appears to the user using PROC FSLIST against the CMS file containing the report.

**Figure 4: FTE Report Template Screen**

FTE SUMMARY REPORT DESIGN TEMPLATE

Enter YEAR, QTR, MON, ACTLABEL, or FTELABEL for rows or columns. Leave Row2 or Column2 blank to return back to menu.

Page by: &BYVARA
(DATABASE PROGRAM AREA PRODUCT PROTOCOL INDICAT QUALI-8 DESCRIPT STATDIR CDGR)

Row1: &ROW1
Row2: &ROW2

| Column1: &COL1 |
| Column2: &COL2 |

Date Window From: &BDATE to &EDATE

PF3=End

After exiting from FSLIST to view the report the user is brought back to the reporting menu and is given the option to print the report just generated. Under this option the user is given a choice of printers to which to print the report. One of the reports that can be selected is the FTE Summary report. When the FTE Summary report option is selected, the user is shown a template (see Figure 4) of the report to be produced. (Since PROC TABULATE is used to produce the report, the template shown is a 2-dimensional Tabulate grid.) The user can then select up to two variables for the row and two variables for the column. For example, the user could select for the column dimension YEAR and MONTH with month nested within year. The user can also enter dates in which to "window" the data. After the user has selected at least one option for each of the dimensions the report is run (see Figure 5).
Graphics Options

In addition to text reports, the system will produce three graphs of planning/tracking data. The first of these is a vertical bar chart representation of data that is shown in a tabular FTE report. The user can select the page by variable, the horizontal and vertical axes variables, and the sub-group variable. As in the reporting option the user can select a date window in order to subset the data within a specified time frame.

Figure 6: The Plot Menu

PLOT MENU

1. Help
2. Enter CMS Subset Mode
3. Return to Main Menu
4. Print Last Plot Generated
5. Display Last Plot Generated
6. Vertical Bar Chart of FTE's
7. Gantt Chart of Activities

To produce them, the Actual Start and Actual Finish variables are not used, and the same variable is used for the early and late start options and for the early and late finish options. Following are the actual options used in the GANIT procedure:

\[ \text{ES} = \text{STRTDATE} \quad \text{LS} = \text{STRTDATE} \]
\[ \text{EF} = \text{FNSHDATE} \quad \text{LF} = \text{FNSHDATE} \]

These options will produce a chart that contains one simple line for each activity without the additional early and late information. The first Gantt chart shows one protocol per chart and displays each activity associated with that protocol. The second Gantt chart displays multiple protocols on one chart but is limited to displaying only one activity for each protocol.

The second and third graphs are Gantt charts of activities for selected protocols. PROC GANTT from SAS/OR® is used to produce these charts (See Figure 8 for an example).
Figure 7: Sample Vbar Chart of FTE's

Vertical Bar Chart of Full Time Equivalents (FTE's)

Figure 8: Sample Gantt Chart

Gantt Chart of Clinic Portion of Clinical Study

972
Database/Program Design

A total of 3 minidisks are used to store data, programs, screens, execs, etc. for the entire planning/tracking system (see Figure 9). The 192 disk contains data for the system. Currently the data is normalized into two SAS data sets. The 192 disk is accessed in update mode. When the 192 disk is accessed, a copy of the data is made for the user and then the disk is released. A user updating data on 192 disk Data is “joined” together by using the MERGE statement.

The 193 disk contains the production SAS/AF programs, SAS/FSP screens, execs, and other associated programs. The production planning/tracking system is comprised of the 192 and 193 disks. The 194 disk is essentially a duplicate of the 193 disk and contains the test code of the 193 disk. This design allows modifications to be made to the 194 disk while users continue to use the system. After modification is made to the 194 disk, the entire 194 disk is copied to the 193 disk thereby updating the current production version with the test version. Only the 193 disk is accessed by the user and since it contains only screens and programs it only needs to be accessed in read mode.

Additional Options

Other options that are incorporated into the system are the ability to enter into CMS subset mode from several menus and the ability to view the SAS log while actually running the planning program.

When the user selects the option to enter CMS subset mode, the program executes the following statement:

```sql
CMS;
```

The user is then brought into CMS subset mode where many (but not all) CMS commands can be used. The current configuration of the SAS job running during this time is kept so that the user can return instantly to the planning system. To return to the system the user types the word RETURN.

Having an option to look at the SAS log while running the system has been very helpful in assessing and debugging the system during its development. The CMS filedef statement

```sql
CMS FILEDEF 17 DISK PLAN SASLOG A;
```

is used to associate the data definition number 17 with the saslog. This is invoked once at the very beginning of the program. While the SAS/AF program is running it appends log information to the log file. The user can select the option to look at this log file while the program is running from within the additional options menu. The option simply uses the statement

```sql
PROC FSLIST UNIT=17;
```

and FSLIST displays the current log file.

Figure 9: Mini-disk Set-up

<table>
<thead>
<tr>
<th>Disk</th>
<th>Data</th>
<th>Production</th>
</tr>
</thead>
<tbody>
<tr>
<td>192</td>
<td>Programs,</td>
<td>Production</td>
</tr>
<tr>
<td></td>
<td>Screens,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Execs</td>
<td></td>
</tr>
<tr>
<td>193</td>
<td></td>
<td></td>
</tr>
<tr>
<td>194</td>
<td>Programs,</td>
<td>Test</td>
</tr>
<tr>
<td></td>
<td>Screens,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Execs</td>
<td></td>
</tr>
</tbody>
</table>

Converting to Version 6.06

This planning system was written in version 5.18 of SAS. It is obvious that release 6.06
of SAS would provide much greater enhancement and functionality for the system. Although we have not converted to version 6, we have begun experimenting with converting by running PROC VSTOV6 against the programs and screens, and we have some initial ideas about how we will convert.

Converting the programs and screens using PROC VSTOV6 went very smoothly. We found, however, that the program code had to be compiled after the conversion had been done. We started to do this by hand for each program module until we discovered that there is a compile option when you run PROC BUILD. By running PROC BUILD with the compile option all of the program modules can be compiled at once. Of course, messages produced by the compilation for each module will need to be checked individually to insure that they were compiled correctly.

In running the version 6 programs we encountered two problems. The first problem occurred in the resolution of macro variables using the %STR function. The statement

```
%LET &libname=%STR(PLAN.PRGMS);
```

was not resolving correctly in version 6 even though it worked in version 5. We were told by SAS Institute to remove the %STR function to make it work.

Another problem we encountered is that for our hardware/software configuration (running IRMALAN mainframe emulation software on a LAN) we had to use the FSDEVICE=IRMA option when invoking SAS on the mainframe. Using this option narrowed the screen availability for FSEDIT screens by 2 columns. This means that screens we had developed that had taken up the entire available number of columns in version 5 did not have enough space in version 6. These screens would need to be redesigned.

Problems Encountered

One problem that has been encountered is having enough work space available for users to run the program. Because CMS does not dynamically allocate space for users, work files can exceed the amount of space available on their personal minidisks while the program is running. To get around this, the program currently allocates a fixed amount of space prior to entering the main menu. There are two problems with this solution. The first is that the amount of space is still fixed which means if you are unable to accurately estimate the amount of space needed either more space may be needed or a great deal of space can be wasted. The second problem is that the amount of up front time it takes to format this space (which seems to increase geometrically with respect to the number of cylinders requested) is less than acceptable.

Another problem we have encountered is the desire to make what has been termed a 'band-aid' chart. (See Figure 10. This chart was appropriately named by a medical person because of the band-aid appearance of the bars on the chart.) The band-aid chart is similar to a Gantt chart except that it has three activities (or possibly more) for each horizontal line. We have not found an easy way to do it with PROC GANTT and we are currently looking into PROC GCHART with horizontal bars.
Conclusion

In conclusion, we have found that the SAS System has many excellent tools for building a user-friendly system to do planning and tracking of data. This planning/tracking system allows users to view information about manpower and clinical studies and to get a perspective on resources needed in the future. We have just scratched the surface in beginning to use version 6.06 of the SAS System which would allow us to produce a system with much greater functionality and efficiency.