A Menu-driven CPE Reporting System (CPE Starter Set)
R.A.Russell, SAS Institute GmbH, Heidelberg, W.Germany

1. INTRODUCTION

This paper describes an application, written using the Base SAS(*) system along with SAS/AF(*) software and SAS/GRAPH(*) software. The application attempts to provide a reasonable starting point for users who wish to use the SAS System for evaluation of Computer Performance data, specifically evaluation of SMF and RMF data produced by IBM's MVS operating system.

The application (normally called the CPE Starter Set) does not attempt to teach the user concepts of Computer Performance Evaluation (CPE) but is directed more at the user who understands what indicators may be important for the system in question and allows him to produce a variety of reports without having to get involved in coding SAS statements. A secondary goal is to provide example code which the user can modify to suit his own needs. It is expected that after a few sessions using the CPE Starter Set the user will begin to modify the application and to steal techniques from the application to integrate into his own systems.

2. TOOLS USED AND DATA SOURCES

The CPE Starter Set is an interactive application which uses full-screen panels and menus to accept user directives. Because the system used for the report generation and to store the data is the SAS System, SAS/AF Software is used as the full-screen manager. Some more advanced features of SAS/AF software are used in the CPE Starter Set and these are described in more detail in Section 5.

The data source for the application is the Performance Data Base (PDB) created by Merrill's Expanded Guide (MXG). This PDB is created from SMF and RMF data which is read by a standard job (BUILDPDB). The BUILDPDB job is described in Ref 1. and can be treated as separate from this application. The PDB contains a number of datasets, these are:

- RMFINTRV - Resource utilisation and workload
- JOBS - TSO,STC and Job statistics
- STEPS - Step level statistics
- IPLS - One record for each IPL of the system

The CPE Starter Set allows for analysis of each of these datasets.

It would be a fairly simple matter to adapt the CPE Starter Set to accept data from a source other than the MXG PDB. The only dependencies are in a few places where explicit SAS dataset names are expected.

3. SOME DESIGN CONSIDERATIONS

A major problem encountered when conducting performance analysis is the large number of different indicators available. In the RMFINTRV dataset of the PDB there are some 118 different variables. The 8 characters of the SAS variable name allow only for cryptic acronyms which means that some other method of identifying the variables is necessary. MXG has used SAS labels to provide a short description of the meaning of the variable name and the standard SAS/AF software DIR command can be used to give a neat overview of the variables available for analysis at a particular time and their descriptions. An example of this feature in action is in Section 4.

The second area of concern is the type of reports which the system should produce. Most installations would like to make tailored reports specifically set up with indicators of interest represented in a form
which is useful for those who will work with the report. Because the range of such reports is extremely wide an application such as the CPE Starter Set can only address a subset of desired reports which are more or less common to all installations involved in Computer Performance.

The reports generated by the system can be split into two broad groups:

a) Time Based Reports

These reports deal with the change in a variable over time. Two forms of time based reports are possible - Bar chart form and Table form. Regardless of the form chosen there are two subdivisions of time based reporting, DAILY and WEEKLY.

DAILY reports have one report per day for each day in the period selected for analysis. In each report the day is broken into hourly slots and one bar (in the Bar chart form) or one table entry (in the Table form) is presented for each hour. An example of a DAILY report is in Figure 1.

WEEKLY reports have one report per week for each week in the period selected for analysis and within each report there are slots for each day. An example of a WEEKLY report is in Figure 2.

b) Period Reports

These reports deal with complete analysis periods. In this category it is possible to create the following reports.

Plots of one variable against another. This report would be used to investigate how variables are related to one another. An example is in Figure 3.

Tables based on classification variables such as JOBNAME or SHIFT. These reports typically have counts associated with each level of the classification. An example is in Figure 4.
4. AN EXAMPLE SESSION

The following is an example of a session using the CPE Starter Set. A number of reports are produced using some data from August 1986.

We begin the session by calling a CLIST. As well as allocating the required files and calling the SAS System the CLIST also sets up some MACROS which allow for easy switching between the SAS/AF and the SAS Display Manager System. The relevant part of the CLIST is to be found in Section 5.

The first screen to appear is below:

```
Welcome to 'e CPE Starter Set
Select Option and press ENTER
1. General information
2. Enter Or CHange Session Parameters
3. Vertical Bar Chart over Time
4. Plot one Variable against Another
5. Tabular Time based Reports
6. Other Tabular Reports
7. Go to SAS Display Manager
8. Submit to SAS Display Manager (Batchjob)
9. Submit the 'Standard Reports' (Batchjob)
10. Dump all generated SAS code to an external file
   Press END to return
   Exit the CPE Starter Set
```

Pressing PF3 returns us to the Master Menu where we are now ready to make the first report. Selecting option 3 takes us to the Vertical Bar Chart over Time section.

Select Option 3

Select one option and press ENTER
1. General information
2. Enter or Change Session Parameters
3. Vertical Bar Chart over Time
4. Plot one Variable against Another
5. Tabular Time based Reports
6. Other Tabular Reports
7. Go to SAS Display Manager
8. Submit the 'Standard Reports' (Batchjob)
9. Submit the 'Standard Reports' (Batchjob)
10. Dump all generated SAS code to an external file
   Press END to return
   Exit the CPE Starter Set

We can choose which type of data in the PDB is to be used for the report, in this case RMF data about resource utilisation.

Command:

```
USER from RMFINFO, Jobs or STEPS?
```

We are now taken to a screen where we can actually create the report. We need to fill in the date and the SYSID from the session parameter screen which are displayed on this screen and can also be altered. First we must determine the name of the variable we are to use. We use the DIR command provided by SAS/AF software.

Command:

```
dir pdb.mfintrv
```

On selecting option 2 we are taken to a fill-in-the-blank screen where we can enter parameters which will control the look of our session. We need to fill in the dataset name where the PDB is stored and the date range and SYSID in which we are interested. These dates and SYSID may be changed later in the session without returning to this screen. Control over the destination of the output is also possible, either to graphics devices, printers or to sequential dataset. The GOPTIONS area at the bottom of the screen can be used to direct output to any device supported by SAS/GRAPH software.

We can choose which type of data in the PDB is to be used for the report, in this case RMF data about resource utilisation.

Command:

```
SAS from RMFINFO, Jobs or STEPS?
```

On selecting option 2 we are taken to a fill-in-the-blank screen where we can enter parameters which will control the look of our session. We need to fill in the dataset name where the PDB is stored and the date range and SYSID in which we are interested. These dates and SYSID may be changed later in the session without returning to this screen. Control over the destination of the output is also possible, either to graphics devices, printers or to sequential dataset. The GOPTIONS area at the bottom of the screen can be used to direct output to any device supported by SAS/GRAPH software.

Pressing PF3 returns us to the Master Menu where we are now ready to make the first report. Selecting option 3 takes us to the Vertical Bar Chart over Time section.

Select Option 3

Select one option and press ENTER
1. General information
2. Enter or Change Session Parameters
3. Vertical Bar Chart over Time
4. Plot one Variable against Another
5. Tabular Time based Reports
6. Other Tabular Reports
7. Go to SAS Display Manager
8. Submit the 'Standard Reports' (Batchjob)
9. Submit the 'Standard Reports' (Batchjob)
10. Dump all generated SAS code to an external file
   Press END to return
   Exit the CPE Starter Set

We can choose which type of data in the PDB is to be used for the report, in this case RMF data about resource utilisation.

Command:

```
USER from RMFINFO, Jobs or STEPS?
```

We are now taken to a screen where we can actually create the report. We need to fill in the date and the SYSID from the session parameter screen which are displayed on this screen and can also be altered. First we must determine the name of the variable we are to use. We use the DIR command provided by SAS/AF software.

Command:

```
dir pdb.mfintrv
```

On selecting option 2 we are taken to a fill-in-the-blank screen where we can enter parameters which will control the look of our session. We need to fill in the dataset name where the PDB is stored and the date range and SYSID in which we are interested. These dates and SYSID may be changed later in the session without returning to this screen. Control over the destination of the output is also possible, either to graphics devices, printers or to sequential dataset. The GOPTIONS area at the bottom of the screen can be used to direct output to any device supported by SAS/GRAPH software.

We can choose which type of data in the PDB is to be used for the report, in this case RMF data about resource utilisation.

Command:

```
SAS from RMFINFO, Jobs or STEPS?
```

On selecting option 2 we are taken to a fill-in-the-blank screen where we can enter parameters which will control the look of our session. We need to fill in the dataset name where the PDB is stored and the date range and SYSID in which we are interested. These dates and SYSID may be changed later in the session without returning to this screen. Control over the destination of the output is also possible, either to graphics devices, printers or to sequential dataset. The GOPTIONS area at the bottom of the screen can be used to direct output to any device supported by SAS/GRAPH software.

Pressing PF3 returns us to the Master Menu where we are now ready to make the first report. Selecting option 3 takes us to the Vertical Bar Chart over Time section.

Select Option 3

Select one option and press ENTER
1. General information
2. Enter or Change Session Parameters
3. Vertical Bar Chart over Time
4. Plot one Variable against Another
5. Tabular Time based Reports
6. Other Tabular Reports
7. Go to SAS Display Manager
8. Submit the 'Standard Reports' (Batchjob)
9. Submit the 'Standard Reports' (Batchjob)
10. Dump all generated SAS code to an external file
   Press END to return
   Exit the CPE Starter Set

We can choose which type of data in the PDB is to be used for the report, in this case RMF data about resource utilisation.

Command:

```
USER from RMFINFO, Jobs or STEPS?
```

We are now taken to a screen where we can actually create the report. We need to fill in the date and the SYSID from the session parameter screen which are displayed on this screen and can also be altered. First we must determine the name of the variable we are to use. We use the DIR command provided by SAS/AF software.

Command:

```
dir pdb.mfintrv
```

On selecting option 2 we are taken to a fill-in-the-blank screen where we can enter parameters which will control the look of our session. We need to fill in the dataset name where the PDB is stored and the date range and SYSID in which we are interested. These dates and SYSID may be changed later in the session without returning to this screen. Control over the destination of the output is also possible, either to graphics devices, printers or to sequential dataset. The GOPTIONS area at the bottom of the screen can be used to direct output to any device supported by SAS/GRAPH software.

We can choose which type of data in the PDB is to be used for the report, in this case RMF data about resource utilisation.

Command:

```
SAS from RMFINFO, Jobs or STEPS?
```

On selecting option 2 we are taken to a fill-in-the-blank screen where we can enter parameters which will control the look of our session. We need to fill in the dataset name where the PDB is stored and the date range and SYSID in which we are interested. These dates and SYSID may be changed later in the session without returning to this screen. Control over the destination of the output is also possible, either to graphics devices, printers or to sequential dataset. The GOPTIONS area at the bottom of the screen can be used to direct output to any device supported by SAS/GRAPH software.

The DIR command gives us a list of all the variables in the dataset concerned, along with their types and, where present, their labels. MKG provides labels for all variables and thus simplifies the task of remembering which indicator is represented by which variable.
Let us select PCTCPU8V for this report. We also request a DAILY report over the MEAN value of PCTCPU8V.

where we choose RMF data again, and can now select variables, time ranges and so on :-

which shows us how busy the CPU was for each hour on the 13 August 1986.

The next report we will make is a table with similar information to the report above. After returning to the Master Menu we select option 5 :-
The next report is an example of a non-time based report. We will select option 6.

Select Option 6 Press END to return.

Once more, we can select the type of data for the report. This time we will select JOBS data which contains information about each job which ran in the system.

To obtain a tabular report about jobs in the system we fill in the category variable as JOB (the Jobname) along with the names of the variables we wish to see reported, EXCPTOTL (total EXCP's issued by the job) and CPUCTSTM (the TCB CPU time in seconds). Again DIR is available if we need help with variable names.

The following is a section out of the report which results:

The next report is an example of a non-time based report. We will select option 6.

Select Option 6 Press END to return.

Select one of: a) JOB/TSO/STC resource, response and billing b) Combined RMF resource, workload, paging and so on c) General Information d) Vertical Bar Chart over Time e) Plot one Variable against Another f) Report on Time based Reports g) Report on Non-time based Reports h) Report on SAS Inputs (Terminate) i) Submit the "Standard Reports" (Return) j) Go to SAS Display Manager k) Report on All generated SAS Code to an external file l) Exit the CPE Starter Set

Once more, we can select the type of data for the report. This time we will select JOBS data which contains information about each job which ran in the system.

To obtain a tabular report about jobs in the system we fill in the category variable as JOB (the Jobname) along with the names of the variables we wish to see reported, EXCPTOTL (total EXCP's issued by the job) and CPUCTSTM (the TCB CPU time in seconds). Again DIR is available if we need help with variable names.

The following is a section out of the report which results:

The following is a section out of the report which results:

5. SOME TECHNIQUES USED IN THE CPE STARTER SET

As well as some of the basic techniques available in SAS/AF software, the CPE Starter Set uses some more advanced facilities. Amongst these used are:

a) Option 8 allows the user to drop into display manager. Any code which the user enters and submits while in the display manager can be submitted to the SAS System and will be executed. This is useful for functions such as supporting non-standard graphics devices where many GOPTIONS must be entered or making subsets of data by hand.

The option 8 simply invokes another member, called SASDM.PROGRAM. SASDM.PROGRAM is nothing more than a RUN; statement with no PROC DISPLAY to indicate that another member is to be called after termination. Thus we simply drop into display manager.

b) If the user issues option 8 then he will also require a means by which he can return to the CPE Starter Set easily. The user can obviously issue the PROC DISPLAY to restart the system by hand but a more easily used method is to have a user written command (let's call it RESTART) to automatically invoke PROC DISPLAY. This can be easily achieved by including in the CLIST used to call the application some statements similar to those below.

c) In option 2 the user may define certain parameters such as date range and SYSID which are to be in force for the rest of the session. On each report screen the user is given the possibility of altering these parameters for future reports without having to leave the report screen. This feature makes use of Associated MACRO variables in SAS/AF software. A MACRO variable may retain its value across multiple
SAS steps. SAS/AF software lets us assign a MACRO variable from which the value of a SAS/AF screen variable will be filled in. By assigning MACRO variables in option 2 and displaying them in subsequent report screens we can carry over the parameters we require.

d) The CPE Starter Set makes extensive use of list validation possibilities in SAS/AF software. For most fields where the user can enter a number of selections eg. choosing a statistic in a report screen the application uses an XXX.LIST entry to define a list of permissible values. For fields where a variable from a dataset must be entered the application uses a *PDB.XXX entry to cause SAS/AF software to validate that the variable chosen exists in the dataset. In this way we try to ensure that the code generated by the application and passed to the SAS System is as correct as possible thereby minimising the SAS System errors with which the user must cope.

6. CONCLUSION

The CPE Starter Set as presented in this paper is a first attempt at a generalised CPE reporting system which can be used to get started in the use of the SAS System for CPE. It is hoped and expected that users will create their own extensions to the application as they gain experience in the SAS System. SAS Institute in Europe also hopes to provide more facilities in the CPE Starter Set. Already planned are:-

- "Top Ten" or "Hit List" reports.
- More possibilities to run reports in batch.
- More standard reports about "important" system indicators.

If you have any suggestions for new areas which could be covered in the CPE Starter Set please contact your local SAS Institute office or write to SAS Institute GmbH, Postfach 105307, Heidelberg, W.Germany

Reference:

Merrill, H.W. "Barry", 1984
Merrill's Expanded Guide to Computer Performance Evaluation using the SAS System,
SAS Institute Inc.

Extract from the CLIST used to call the CPE Starter Set. This part sets up the user defined command RESTART.

OPENFILE SASEXEC OUTPUT
SET &SASEXEC = &STR(SUBMIT '%MACRO RESTART/CMD;';)
PUTFILE SASEXEC
SET &SASEXEC = +
&STR(SUBMIT 'PROC D C=CPE.STARTER.MASTER.MENU;RUN;';)
PUTFILE SASEXEC
SET &SASEXEC = &STR(SUBMIT '%MEND ;RUN;';)
PUTFILE SASEXEC
SET &SASEXEC = +
&STR(SUBMIT 'PROC D C=CPE.STARTER.MASTER.MENU; RUN; ';

* SAS, SAS/GRAFH and SAS/AF are registered trademarks of SAS Institute Inc., Cary, NC, USA

241