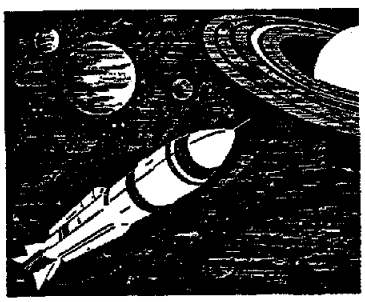
Prepare to Make the Jump to Hiperspace

Curtis A. Smith, *Defense Contract Audit Agency, La Mirada, Ca.*

ABSTRACT

A long time ago, on a mainframe far, far away, temporary SAS data libraries were always residing on disk space. While this works, using disk space requires I/O reads and writes. I/O is slow and usually costs the user mainframe charge-back costs. To eliminate I/O when using temporary SAS data sets, make the jump to Hiperspace. With the advent of the MVS/ESA operating system, memory is available as High Performance Space (Hiperspace) that SAS can use to store temporary SAS data sets. When this is done, SAS does not have to expend I/O to use the temporary SAS data sets. The result - less I/O and shorter processing time. Sound good? The author will show how to use and optimize Hiperspaces under MVS.

INTRODUCTION



Hiperspaces are special areas of expanded memory that can be used to hold temporary SAS data sets. If your organization is running MVS/ESA and has implemented Hiperspaces, you can place the SAS WORK library and your own temporary SAS data libraries into Hiperspaces. Using Hiperspaces under the right data processing conditions you can greatly reduce I/O EXCPs. Doing so can reduce mainframe charge-back costs and will reduce run time for your jobs. Hiperspaces are very attractive because they are simple to implement and can have an enormous positive impact on your SAS processing and charge-back costs. While MVS/ESA Hiperspaces will not cause your SAS jobs to process at light speed, you will notice a positive difference.

IMPLEMENTING A HIPERSPACE

You can place the SAS WORK library and your own temporary disk libraries into Hiperspaces using the HSWORK option. This option can be placed in the SAS configuration file or passed as a parameter when you invoke SAS. Possibly, at your site the SAS configuration file has Hiperspace set to NOHSWORK, disabling Hiperspaces. If so, have your site administrator change the parameter to HSWORK. Alternatively, you can set up your own configuration file, using it rather than the site configuration file. When running SAS in batch mode, you can also pass the HSWORK option to SAS as a parameter in the JCL on the EXEC card as follows:

```
//SAS     EXEC PGM=SASHOST,TIME=299,
//        PARM=('OPLIST FULLSTIMER HSWORK')
```

This is the method I prefer. If you are running SAS interactively, pass the parameter as follows, from the READY prompt.

```
SAS options('hswork')'
```

When wanting to place your own temporary data libraries into Hiperspace, use the HIP (or, HIPERSPACE) option on the SAS LIBNAME statement. For example:

```
LIBNAME TEMP "LONG.MVS.FILENAME.TEMPDISK"
        DISP=(NEW) HIP;
```

SAS will always assign a block size of 4096 and a unit of 3380 regardless of what DCB parameters you specify. So, you don't need to specify the unit or the block size. SAS will create the Hiperspace based upon the space parameters defined in the options discussed below. You should not need to specify any parameters in the LIBNAME statement. At my location, however, I seem to be required to specify the DISP=(NEW) parameter. When I don't do so SAS seems to be unable to allocate the new, temporary Hiperspace library. You can read a thorough and technical discussion of Hiperspaces in Michael A. Raithel's book "Tuning SAS Applications in the MVS Environment," which is available through the SAS Institute.

OPTIONAL PARAMETERS

There are three options that you can use to affect the number and size of Hiperspaces: HSMAXSPC=, HSLXTNTS=, and HSMAXPGS=.

The HSMAXSPC= option sets the maximum number of Hiperspaces SAS can use. The default value is 50.

The HSLXTNTS= option sets the size of each Hiperspace in pages. The default is 1500 which is 1,644,000 bytes, which is not very big. I routinely process files much larger than this.

The HSMAXPGS= option sets the maximum size of all Hiperspaces. This value is typically equal to the value of HSMAXSPC= times the value of HSLXTNTS=. The default is 75000.

You can experiment with the size of the Hiperspaces by determining the size in bytes of the SAS data sets that you need to process in the WORK and/or other temporary SAS data libraries. I process some really big SAS data sets, but do not use many temporary libraries simultaneously. So, I keep the HSMAXSPC= value low, but greatly increase the size of HSLXTNTS=. For example:

```
OPTIONS HSMAXSPC=5 HSLXTNTS=15000 HSMAXPGS=75000;
LIBNAME TEMP "LONG.MVS.FILENAME.TEMPDISK"
        DISP=(NEW) HIP;
```

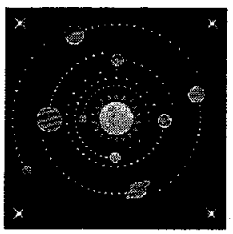
WHAT'S THE DIFFERENCE?

I thought you might want to know some real benchmark statistics comparing SAS jobs using Hiperspaces to SAS jobs not using Hiperspaces. So, I ran a few comparative jobs. Where you get the most benefit in reduced I/O EXCPs and reduced time is when you read data into the temporary work space in a Hiperspace and then process that data over and over again, or do tasks such as a SORT or SUMMARY procedure. If you simply read data into the Hiperspace, perform minor calculations, and then write the data back out you may not realize any savings.

To benchmark using Hiperspaces, I ran a SORT procedure and then a SUMMARY procedure with a BY statement and also a SUMMARY procedure with a CLASS statement against an unsorted SAS data set. I did this against two source SAS data sets, one with 33,781 observations and the other with 133,162 observations. In each case, I used four character variables in my BY and CLASS statements. I then ran both using a Hiperspace and not using a Hiperspace. The table at the end of this paper summarizes my

results.

Here we see that the elapsed time dropped noticeable with our SORT procedure (which sorted source data from permanent storage) as did our EXCPs. But once we processed that sorted data sitting in Hiperspace with our SUMMARY procedure using the BY statement our elapsed time and EXCPs dropped dramatically. When we used the SUMMARY procedure with our unsorted source data from permanent storage using the CLASS statement our elapsed time was reduced when processing the smaller file but increased when processing the larger file. But, in both instances our EXCPs still dropped significantly. In my tests I found the reduction to CPU time to be much less dramatic than the reduction to elapsed time.



If you would like a more detailed benchmark analysis, I recommend Fred Forst's paper Get Hiper About Hiperspaces, presented at SUGI 24.

CONCLUSION

Hiperspaces are one of those SAS features that are simple to use and will likely have a dramatic impact on your performance. There is probably no reason not to implement Hiperspaces for WORK and other temporary libraries at a site that has MVS/ESA and expanded memory.

REFERENCES

"Tuning SAS Applications in the MVS Environment", Michael A. Raithel, Chapter 11

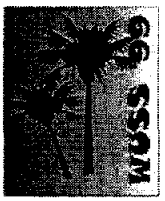
Get Hiper About Hiperspaces, Fred Forst, SUGI 24 paper

SAS Technical Report P-218, Changes and Enhancement to the SAS System for the MVS Environment, Pages 9-11 and 53-56

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

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

Questions: e-mail the author at casmith@mindspring.com



Hiperspace Benchmark Results

Procedure	No. Obs	No Hiperspace Elapsed Time (sec)	Hiperspace Elapsed Time (sec)	Percent Reduction	No Hiperspace EXCP	Hiperspace EXCP	Percent Reduction
SORT	33,781	2.80	1.28	54.29	122	83	31.97
SUMMARY w/ BY	33,781	2.17	.61	71.89	52	8	84.62
SUMMARY w/ CLASS	33,781	2.37	1.96	17.3	38	28	26.32
SORT	133,162	11.29	6.20	45.08	324	191	41.05
SUMMARY w/ BY	133,162	4.44	2.52	43.24	162	8	95.06
SUMMARY w/ CLASS	133,162	8.43	9.43	-11.86	109	86	21.1