

An Alternative Technique for Analyzing System Paging Workloads

H. Pat Artis
Morino Associates
Vienna, VA 22180

Abstract: The presentation of paging data to management in an easily understood form has been a persistent problem. This paper examines a graphical technique for presenting paging data activity as a relative fraction of the system's total I/O activity. This technique will also be used to comment on the benefits of paging into Expanded Storage on the new IBM 3090 series of processors.

1.0 Introduction

Paging is a complex activity that can consume a significant portion of a system's overall capacity. As such, paging is a significant factor in system performance that must be included in any management reporting scheme. Historically, most analysts have chosen to report paging rate as an indicator of system paging activity.

The use of paging rate as a management reporting indicator makes the implicit assumption that the reader of the report is aware of some range of paging values that are *ideal* for the reported system. In environments that are comprised of multiple disparate systems, the readers of the performance reports are required to remember numerous *rules of thumb* for interpreting each system's paging data. Using such a reporting scheme places a burden (i.e., remembering the ideal values) on the recipients of the reports. Moreover, even if the readers remember ideal values for each system, the use of paging rate as an indicator tends to hide the true cost of paging from the recipients of the reports. The values used to define these rules of thumb are also subject to some question. With the variety of paging devices, memory sizes, processor sizes, and channel rates available to the designers of today's configurations, the selection of such ideal values is not a trivial decision.

In this paper, we will examine a graphical approach to reporting paging data. The key to this approach is the comparison of the data transferred to support paging relative to the total I/O activity of the system. The principal advantages of this technique are:

- the cost of paging is graphically presented as a fraction of the system's total DASD I/O throughput,
- the diminishing return of system throughput as the paging rate increases can be clearly depicted,
- the dependence on the recipients of the reports remembering the ideal paging rate values to evaluate the report are eliminated.

This approach is based on paging values provided in the Resource Management Facility (RMF) RMF type paging activity records (record type 71) and the new device connect time values provided by MVS/XA in the RMF device activity records (record type 74). In this paper, we will discuss a SAS implementation of this approach using data from Morino Associates MVS Integrated Control System (MICS).

2.0 What are the Costs of Paging?

Although the concept of paging in an MVS system is relatively straight forward, a brief review of the functions performed by paging in an MVS/XA environment will assist in understanding the true costs of the system paging workload. Briefly, pages are moved in blocks to support group requests (i.e., swapping, VIO, etc.) and as individual pages, to steal pages from inactive address

spaces or to obtain pages to service page faults (i.e., the program waits on the system to retrieve a page that is not in memory) for active programs and tasks. Paging consumes two system resources: CPU and I/O. Since only a few thousand instructions are required to service a page fault (i.e., schedule an I/O to obtain the desired page), the CPU impact of even large paging rates is not overwhelming for large systems [1, 2]. However, the cumulative effect of transferring hundreds of 4096 byte pages per second can easily represent a large percentage of the system's I/O throughput.

In fact, there are significant compounding effects introduced by paging. A high paging rate is indicative of the fact that numerous programs and tasks are waiting on paging I/Os to be completed. Since waiting programs and tasks cannot issue user I/O requests, the user I/O rate (on a per address space basis) tends to decline as the paging rate increases. This effect tends to maximize the percentage of the system's I/O throughput represented by paging since page faults tend to moderate the rate at which user I/Os are issued.

Thus, one very effective way to present the true cost of paging is to show the percentage of the system's DASD I/O throughput consumed as a function of the paging rate. Such a representation demonstrates both the cost of paging (in terms of I/O) and the diminishing nature of throughput returns as the paging rate increases.

3.0 Calculating and Reporting DASD I/O Throughput

To evaluate percentage of DASD I/O throughput that is represented by paging, two values must be calculated. They are 1) the paging data transfer rate and 2) the total data transfer rate for all DASD channels and devices. The calculations required to develop each of these measurements are discussed in the following sections. The reader should note that the technique discussed is only valid for MVS/XA systems. The variables required to perform the calculations are not available in MVS/370 systems.

3.1 Calculating the Paging Data Transfer Rate

To calculate the data transfer rate (in megabytes per second) for paging, the page counts for all of the types of paging measured by the system must be accumulated and then multiplied by the page size (i.e., 4096 bytes) for a selected interval. In MICS, the following variables from the MICS MVS Paging Activity (SCPPAG) File are used to calculate the total paging rate for all types of paging activity:

PAGPIN	non-VIO non-swap page-ins
PAGPOUT	non-VIO non-swap page-outs
PAGSIN	swap-in pages
PAGSOT	swap-out pages
PAGVIN	VIO page-ins
PAGVOT	VIO page-outs
DURATION	recording interval time

The average paging rate is defined as the sum of the page-in and page-out operations of all types divided by the duration of the recording interval. To avoid misleading average values caused by summarizing long periods of data, the author recommends that either the DETAIL or DAYS timespan in the MICS data base be selected for these calculations. Although the DETAIL timespan offers a slightly smaller granularity, most installations only retain a single cycle of the DETAIL timespan making the DAYS timespan preferable because of the number of observations available. A sample of the SAS source code necessary to calculate the paging data transfer rate (in megabytes per second) and create a SAS dataset called PAGE is shown below:

```
DATA PAGE (KEEP = SYSID DAY HOUR PAGES
           PAGEBYTE);
  SET _SCPD. SCPPAG01
      _SCPD. SCPPAGnn;
  PAGES= (SUM (PAGPIN, PAGPOUT, PAGESIN, PAGESOT,
             PAGVIN, PAGVOT) /DURATION) ;
  PAGEBYTE = PAGES*0.0039063;
PROC SORT; BY SYSID DAY HOUR;
```

Where nn is the number of cycles of the MVS Paging Activity (SCPPAG) File maintained in the DAYS timespan by the installation. The reader might wish to note that the constant 0.0039063 is the quotient of dividing the number of bytes in a page, 4096, by one megabyte, i.e., 1048576 bytes.

3.2 Calculating the DASD Data Transfer Rate

There are two alternatives for calculating the total DASD data transfer rate for the system. The first alternative is based on the channel utilization for those channels that support DASD devices. Simply described, one can estimate the total data transferred by multiplying the channel transfer rate (e.g., 1.8, 2.0, or 3.0 megabytes per second depending on the type of channel) by the percent busy for each of the DASD channels. Although this technique is the most simplistic, it can result in misleadingly high values if all of the devices supported by the channel are not able to transfer data at the channel's rated speed. For that reason, a somewhat more complex second alternative is recommended based on calculating the product of the individual device connect times multiplied by the devices specific data transfer rate, in megabytes per second.

To implement the second alternative, the following variables from the MICS Hardware Device Activity (HARDVA) File are used to calculate the total DASD transfer rate for the system:

DVACNN	total device connect time
DEVTYPE	device type
DURATION	recording interval time

To estimate the data transferred by a device, the total connect time for the device must be multiplied by the rated transfer speed for the device. Table 1 provides a list of the transfer rates for common IBM device types [3].

Device Type	MB/Sec
3330	0.806
2305	1.500
3340	0.885
3344	0.885
3350	1.198
3310	1.031
3370	1.859
3375	1.859
3380	3.000

Table 1. Device I/O Transfer Rates

The reader should note that this table may have to be expanded to include certain third party devices that may have different transfer rates from the IBM device types which they replace or IBM devices like 3350s that are located behind cache control units.

A sample of the SAS source code necessary to calculate the total DASD data transfer rate (in megabytes per second) and create a SAS dataset called DATA is shown below:

```
DATA CHAN (KEEP = SYSID DAY HOUR XFER);
  SET _HARD. HARDVA01
      _HARD. HARDVAnn;
  XFER=.;
  IF SUBSTR (DEVTYPE, 1, 4)='3330'
    THEN XFER=DVACNN*0.806;
  ELSE IF SUBSTR (DEVTYPE, 1, 4)='3350'
    THEN XFER=DVACNN*1.198;
  ELSE IF SUBSTR (DEVTYPE, 1, 4)='2305'
    THEN XFER=DVACNN*1.500;
  ELSE IF SUBSTR (DEVTYPE, 1, 4)='3380'
    THEN XFER=DVACNN*3.000;
  IF XFER=. THEN DELETE;
  XFER=XFER/DURATION;
PROC SORT; BY SYSID DAY HOUR;
DATA CHAN (KEEP= SYSID DAY HOUR TOTBYTE);
  SET CHAN; BY SYSID DAY HOUR;
  RETAIN TOTBYTE 0;
  IF FIRST.HOUR THEN TOTBYTE=0;
  TOTBYTE+XFER;
  IF LAST.HOUR THEN OUTPUT;
```

Where nn is the number of cycles of the Hardware Device Activity (HARDVA) File maintained in the DAYS timespan by the installation. The reader should note that this value must agree with the number of cycles specified for the MVS Paging Activity (SCPPAG) File in the prior section.

3.3 Plotting DASD I/O Throughput

The final step in analyzing the impact of paging on DASD I/O throughput is the creation of plots that depict the relative amount of DASD I/O throughput being consumed by paging as a function of the paging rate of the system. To accomplish this objective, the CHAN and the PAGE SAS files generated in the prior two sections must be merged to produce an input file for the SAS PLOT or GPLOT procedures. Sample SAS source code for merging the files and producing the plots is shown below:

```

DATA ALL (KEEP=SYSID DAY HOUR PAGES PAGEBYTE
TOTBYTE) ;
MERGE PAGE CHAN;
BY SYSID DAY HOUR;
PROC PLOT;
PLOT TOTBYTE*PAGES='T' PAGEBYTE*PAGES='P'
/ OVERLAY; BY SYSID;

```

Of course, the reader may wish to add titles and axis labels to this plot specification to customize the report to their specific environment.

4.0 Sample Problem

The graphical analysis technique outlined in the prior section was used to evaluate the paging workload of a twenty-four megabyte 3081-K supporting several hundred TSO users. The graphical report produced by the analysis is shown in Figure 1. This figure depicts the total (T) and paging (P) I/O throughput of the system as a function of paging rate. The data was collected for the system over a period of five days during which paging rates of 10 to more than 500 pages per second were observed for the system.

A number of interesting observations can be drawn from this report. Principal among them are:

- a majority of the observations collected over the five day period correspond to off-peak periods very low paging rates, i.e., 0 to 50 pages per second. For these off-peak periods, the total I/O throughput of the system is independent of the paging rate. These observations correspond to idle periods or off shift maintenance activities like DASD backups.
- during time periods where the systems paging rate would be considered moderate (i.e. 50 to 200 pages per second), approximately 20 to 30 percent of the system's DASD I/O throughput is consumed by paging.
- at high paging rates (i.e., greater than 200 pages per second), paging consumed as much as 45 percent of the DASD I/O throughput for the system being evaluated.

Depending on the nature of a specific installation's workload, similar observations might be made at substantially different paging rates. For example, a large system dedicated to an online CICS workload would likely show substantial variations in DASD I/O throughput at low paging rates due to CICS's pronounced sensitivity to system paging [4,5].

5.0 Remarks

The graphical technique that has been introduced in this paper provides the analyst with a more effective tool than the traditional paging rate value for presenting the cost and impact of system paging. As we have seen in the sample case, the cost of paging in terms of I/O throughput can be substantial at paging rates which many installations may have considered to be quite reasonable.

One of the significant benefits offered by the new 3090 generation of IBM processors, is the redirection of the majority of paging activities away from the I/O subsystem to a large memory buffer called Expanded Storage. The objective of the storage management algorithms in a 3090 system under MVS/XA 2.1.3 is to ensure that a large percentage of all paging requests are satisfied from the memory buffer. Paging into Expanded Memory has two significant benefits. They are 1) the return of channel resources to the system which may be used to service user I/O requests, and 2) a significant reduction in page fault service time which, in turn will significantly increase the performance (i.e., decrease the time spent waiting) of programs and tasks allowing them to drive the system's DASD configuration to previously

unattainable levels of throughput. With large numbers of commercial 3090 installations scheduled for the final quarter of this year, it will be interesting to evaluate their impact on DASD I/O throughput using this technique.

6.0 References

- [1] Felix-Simpson, S. and Friedman, M., CPU Utilization Measurements: Part 1, Morino Software News, Mar/Apr 1985.
- [2] Felix-Simpson, S., CPU Utilization Measurements: Part 2, Morino Software News, May/June 1985.
- [3] Bohl, M., Introduction to IBM Direct Access Devices, SRA, 1981.
- [4] Deitch, M., Analytic Queueing Model for CICS Capacity Planning, IBM Systems Journal, Vol. 21 No. 4, 1983.
- [5] Felix-Simpson, S. and Artis, H.P., Implementing an Analytic Model for CICS Systems, CMG XIV, December 1983.

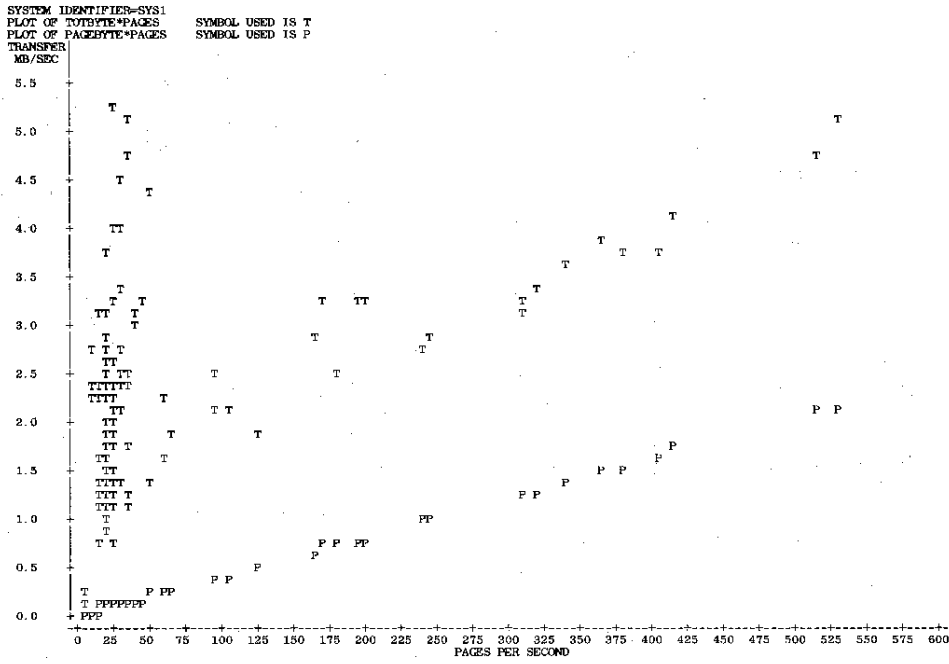


Figure 1. Sample 3081 System Paging Data