HP-UX – SAS Software Configuration Guide

or

Now that I purchased SAS and my HP-UX server, how do I configure it?

Version 2

September 2001

For Updates and Questions Please Contact:
HP/SAS Competency Center, Cary, NC
(919) 677-8000 x2264
hp.ctc@sas.com

The most recent version of this document may be obtained at
http://www.sas.com/partners/directory/hp/whitepap.html

PREFACE:

Tuning a server for any application is not "one size fits all," and this is no different for an HP server running SAS software. The following guidelines were compiled from various sources, both within HP and within SAS Institute Inc. Because things in our industry seem to change daily, this document is meant to be a living document. If you have any comments or inputs, please contact us.

PURPOSE:

This document is intended to be a checklist to use when setting up an HP server to run the SAS System software. Although the intended audience is HP-UX system administrators, there is information that can be useful in optimizing SAS application performance by SAS users and application designers. It can also serve as a reference guide for configuring and tuning HP servers for SAS applications.

WORDING:

All names below that are in **boldface** type are either SAS parameter names (**UPPERCASE**) or HP-UX kernel parameter names (**lowercase**). Commands, command parameters, and other system related names are noted in *Italics*.

REVISION HISTORY:

V 1.0	August 1999	Version 1
V 2.0	September 2001	Version 2

Introduction

Configuring hardware to run any application is not an easy task. It becomes not just a science, but an art as well. The information in this document should be used as a guideline for setting up a system that is to be running the SAS System software. Every environment is different and, therefore, not all of the recommendations may apply to your particular environment. Treat these recommendations as they were intended - as guidelines.

Designing systems to run Decision Support (DSS) and Business Intellegence (BI) applications can be quite different from setting up for more traditional On-Line Transaction Processing (OLTP) systems. Typically, OLTP uses many, small files or buffers to move data to and from disk. In contrast, DSS and BI uses relatively fewer, larger files or buffers when moving data within the server. Making accommodations for the types of data transfers in the server design and layout can significantly improve overall system performance.

PART I: I/O Subsystem

There are three kinds of storage requirements on a computer system: boot and swap, primary or permanent data storage, and temporary or scratch storage. Each of these types has specific characteristics and design considerations for a high performance server design.

Planning the I/O subsystem layout (or the lack of planning) can be the most significant portion of the SAS application's performance characteristics. This section discusses the use of different disk technologies, JBODs (Just a Bunch Of Disks) and Storage System solutions, disk striping, as well as file system creation and placement.

General File System Creation and Placement

- SAS executables should be installed in their own directory (for example, */sas*). Subdirectories would be used for different versions of the SAS System (for example, */sas/sas612* and */sas/sas8*). Sometimes applications such as SAS are placed in the */opt* directory. If this is the case, make sure there is enough space in */opt* for the SAS application as well as other HP and Non-HP applications.
- Permanent SAS data sets (access – read/write, sequential) should also be in their own directory – preferably their own file system, on a separate I/O controller, if possible. This directory is typically very large. The data should be partitioned into logical subdirectories, perhaps building a mount point for each partition of the data store. Large data storage requirements may call for multiple directory/filesystem combinations.

- The SAS Work Area (/*saswork*) should be in its own unique I/O path. By default, *saswork* is created in the */usr/tmp* directory. The SAS config file *-work* parameter should be modified to reside in a separate directory that has been tuned for heavy I/O activity. This directory tends to be heavy read/write, somewhat sequential. This directory should be separated from the SAS executables, SAS data sets, and HP-UX operating system on an entirely separate disk and controller if possible.
- The nature of the SAS configuration options allow for tuning by usage. By tuning the filesystem charactistics of each file system, the overall performance of the server can be optimized.
- Keep file name paths as short as possible. This allows HP-UX to fit more entries into the lookup cache that allows faster file access.
- Depending on how many files are in a particular file system and whether or not these files are being extended, change the density of inodes in the file system from the default of 1 inode per 6144 to 1 inode per 32K of file space.
- Spread file systems utilized by SAS applications out across as many physical disk drives as possible. This may be accomplished many different ways, depending on the performance and application needs.
- Use RAID-0 (independent mode) for better performance. IMPORTANT NOTE: RAID-0 does not provide safeguards against failure.
- Use RAID-1 (disk mirroring) or RAID-5 (disk striping with parity) for data integrity/availability.
- RAID-1+0 should be used for */saswork* and other write intensive directories. Avoid using RAID-5 for these areas since the overhead of data redundancy may cause extra I/O activities.
- For optimum performance, avoid using NFS to access large files, specifically SAS data sets and the /*saswork* area.
- If there is a significant amounts of serial reading, creating SAS data sets with larger blocks should improve overall performance. This will at least fetch as much data as possible for each disk I/O.
- For some decision support applications it may be desirable to presort data in common accessed key order prior to loading. This will achieve a high degree of data locality and hopefully minimize the number of pages, which need to be read.

Disk Hardware Options – JBODs vs. RAID Storage Subsystems

- Configure a maximum of 4 to 6 JBODs per F/W SCSI card for best performance As many as 10 JBOD disks may be configured on a single interface, depending on the I/O bandwidth requirements of the SAS application environment.
- Configure a maximum of 8 to 10 JBODs per Fiber Channel (FC) interface for best performance. As many as 20 JBODs may be configured on a single FC interface, depending on the I/O bandwidth requirements of the SAS application environment.
- The number of JBOD disks to configure on a single interface can vary greatly from site to site depending on the I/O activity. Please contact a HP Technical

Consultant if you have questions regarding the best configuration for your environment.

- Disk storage solutions like the HP XP256 and HP XP512 are configured with significant amounts of cache memory. These subsystems prefetch data into the data cache area for delivery to the requesting system, significantly reducing the latency effects seen in JBOD implementations.
- There is controversy over whether the RAID Storage Subsystems should be striped using HP-UX Logical Volume Manager (LVM) striping or just allow the array to manage the disk itself. For the initial implementation of the data, there often is minimal impact to performance, striped versus not striped. Over time, with lots of update activity, the striped solution has been observed to show better performance. LVM striping is recommended whenever possible, for both JBOD and RAID subsystems. A 64KB stripe size has been found to perform well for all types of data access patterns.
- It is important to understand the external, as well as the internal controller scheme used with RAID arrays. Internally, the disk mechanisms reside on a communications bus. In an XP 256/XP 512 array, this is not the same bus that connects the server to the array. Conventional wisdom indicates that I/O performance is greatly increased when logical volumes are striped across disks and both internal and external controllers. This holds true even for logical volumes that house sequentially accessed structures, such as logs.
- Use or RAID-5 for SAS executables and permanent SAS data files for higher availability.
- Use RAID-0 or RAID1/0 for the SAS workarea for higher performance.

HFS vs. JFS and mounting options

There are two types of file systems that are currently available on HP-UX: the High Performance File System (HFS) and the Journaled File System (JFS). Note: JFS is the default file system shipped with systems running HP-UX 11x.

- If using HFS for either permanent SAS data files or for the SAS workarea, build the file system with 64K blocks (default is 8K) and 8K fragments (default is 1K).
- If using JFS for either permanent SAS data files of for the SAS workarea, build the file system with 8K blocks (the maximum) instead of the default 1K blocks.
- For HFS, consider decreasing the free space percentage (**minfree**) from 10% to 2% or 3%.
- If using JFS for the SAS workarea, make sure that the file system is built with both logging and caching turned OFF (**nolog**, **nodatainlog**, and **mincache=tmpcache** options on the **mount** command). **NOTE**: Use these options **ONLY** for the SAS workareas as these options disable file system recovery! Note: for more information on the various JFS mount options and the associated impact, refer to the JFS Adminstrator's Guide at http://docs.hp.com.
- If using JFS for permanent SAS data files, mount the file system with the following options: **delaylog**, **nodatainlog**, **mincache=direct**, **convosync=direct**.

This will tune the JFS file system to eliminate double inode writes and minimize intent log writes while preserving file system recovery.

- If using and creating files larger than 2GB, use the **largefiles** option when creating the file system. This option should probably be used when creating all file systems.

Disk Striping Guidelines

- Disk striping is recognized as a good way to balance I/O, which might be destined for a single disk volume across multiple spindles and improve overall throughput. Much more detail on disk striping can be found in the HP-UX documentation at http://docs.hp.com.
- Stripe the data across multiple drives using a 64K stripe. When using striping, file system block size should equal stripe size for maximum performance.

There have been many papers published on disk I/O optimization both generally and in a SAS software environment. The following are just a few papers that you may want to reference:

- "Disk I/O Considerations for SAS Processing" located at http://www.sas.com/partners/directory/hp/whitepap.html

- "Configuring Scaleable Performance Data Server Wide Mode on HP-UX: A Database Achievement" located at http://www.sas.com/partners/directory/hp/whitepap.html

PART II: Tuning HP-UX Kernel Parameters (32 bit)

Unlike some application software, the SAS System does not have a large number of requirements for kernel tuning. This section does outline kernel parameter changes that should be considered for best performance.

- **maxdsiz** specifies the maximum data segment size, in bytes, for an executing process. The default is 64 MB. This parameter must be large enough to be able to handle the largest SAS program. You can get this information from the SAS logfile using the **FULLSTIMER** option. For example, the **maxdsiz** parameter will need to be larger if using MDDB, since the MDDB is built in main memory.
- Dynamic buffer cache is controlled by a number of different parameters: **bufpages**, **nbuf**, **dbc_min_pct**, and **dbc_max_pct**. If using dynamic buffer cache, both **bufpages** and **nbuf** need to be set to 0 (zero). The value of **dbc_min_pct** specifies the minimum percentage of physical memory that is reserved for use by the dynamic buffer cache. The default value is 5 and is probably good for most environments. The value of **dbc_max_pct** sets the maximum percentage of physical memory that can be allocated to the dynamic buffer cache. This default is 50 and should be lowered to perhaps 15 or 20, depending on the size of main memory and the type of workload on the system.
- For a predictable workload which is fairly static, the buffer cache can be set to a fixed size by setting **dbx_min_pct** and **dbc_max_pct** to the same value, thereby effectively disabling the dynamic nature of the buffer cache. This has been found to be a performance gain in some SAS application environments.

For more information on these and other HP-UX Kernel parameters, refer to the "Configurable Kernel Parameters" manual at http://docs.hp.com

PART III:Tuning SAS Parameters

There are a number of SAS parameters that should also monitored and perhaps changed depending on your environment. The following guidelines are only a few of the many. Contact SAS Institute for additional tuning recommendations.

- **MEMSIZE** sets the maximum memory that a SAS session can use. The default is 32 MB (SAS V6.12) or 64 MB (SAS V8). This parameter needs to be increased to allow programs to run without swapping. Use the option **FULLSTIMER** to determine how much memory the program is actually using and adjust this parameter accordingly.. Note that this parameter, as well as many others, can be specified on a per-user basis, allowing for specific 'power' users to be allocated more system resources than more casual users.
- **MEMSIZE=0** implies that SAS can use all of available main memory if needed, up to the value of **maxdsiz**. IMPORTANT: setting this parameter to 0 on a multi-user system allocates ALL system memory to each SAS session – use with care!
- **SORTSIZE** sets the maximum amount of memory that a sort can use. The default is 16 MB (6.12) or 48 MB (8). Setting **SORTSIZE** too large can cause performance degradation when sorting a large file. Never set **SORTSIZE** greater than **MEMSIZE**.
- By default, all SAS workareas for all users go to the same directory. It is recommended that these workareas be in different directories, and possibly different file systems, depending on the workload and configuration of the system. These areas can be broken up by user into different SAS workareas by making a change in the *config.sas6.12* (6.12) or *sasv8.cfg* (8) file or by entering an option on the command line.
- **SUMSIZE** (8 only) sets the maximum amount of memory that PROC SUMMARY or PROC MEANS uses. The default value is 8 MB.