![Sun Microsystems logo]

# OpenSSP White Paper

Dan Anderson
Enterprise System Products

Adobe PostScript™

# Contents

# OpenSSP White Paper

This paper describes how to set up and verify third-party software on System Service Processor (SSP) workstations. SSP workstations host SSP software that monitors and controls a Sun Enterprise™ 10000 system. Applications that run on the SSP, other than the SSP software and the Solaris™ Operating Environment, are considered as third-party software. Third-party applications are expected to be lightweight, such as monitoring and backup agents, and to not demand intensive system resources. Described here are the SSP resource requirements, the maximum permitted resource consumption for third-party software, and techniques to help ensure that the SSP receives the resources it needs to function properly.

It is the responsibility of the customer (system administrator) to ensure that adequate SSP resources are available to support SSP and all other software packages that are being installed on the SSP workstation.

This paper is intended for system administrators, system engineers, and others who are installing and configuring an SSP system with third-party applications.

# SSP Functionality

SSP software runs on a workstation (running the Solaris Operating Environment) dedicated to controlling and monitoring a Sun Enterprise 10000 system. To function correctly, the SSP must be able to respond in a timely manner to incoming events from the control board for the system. This includes responding to environmental conditions, such as over-temperature boards, software problems (panics), and hardware problems (arbstops). If the SSP is delayed in responding, these events can be dropped or handled too late. Information about failures could be lost with no alerts or log trail. In a worst-case scenario, hardware can also be damaged due to a lack of timely action by the SSP.

# Impact of Third-Party Applications on the SSP

Traditionally, SSP software has been the only application allowed to run on the SSP workstation. This was required to prevent non-SSP software from interfering with the SSP critical mission of monitoring and controlling a Sun Enterprise 10000 system. This is especially important for SSP software running on SPARCstation™ 5 workstations, which are slower and have less memory. The only non-SSP processes previously allowed on an SSP box have been system background programs that run the Solaris Operating Environment and Common Desktop Environment (CDE).

However, there has been increasing demand to run third-party agents on the SSP. These agents gather information about the Sun Enterprise 10000 system and the SSP, and report back to a central server, allowing centralized control of multiple systems on the network. Backup servers are similarly structured, allowing centralized backup of multiple networked systems.

When running third-party software on an SSP system, the primary goal must be non-interference with SSP software. SSP software normally requires few hardware resources, but when it needs these resources, they must be immediately available.

In order for non-SSP agents or other software to run on the SSP workstation, you must take reasonable steps to ensure non-interference with the mission of the SSP—monitoring and controlling the Sun Enterprise 10000 system.

These steps include:

- Applying recommended patches for the Solaris Operating Environment and SSP software on a regular basis
- Using SSP 3.4 with SSP 3.4 patch 110193-01

  This SSP patch contains performance improvements that allow SSP to function under adverse conditions, namely limited CPU and memory. These improvements include retrying operations without creating a further excessive load on the SSP workstation and the use of real-time class processes.

- Measuring the resource requirements of third-party software and considering *as a whole* the impact of all applications on the SSP.

  Software resource requirements usually describe system impact for the standalone use of an application. These requirements typically provide only generic recommendations such as "Software XYZ requires a Sun Ultra™ 5 workstation with a minimum of 128 Mbytes memory." Many recommendations do not mention the cumulative impact of multiple software packages running on a single system. In other words, one needs to look at the whole forest, not just one tree.

■ Planning for sufficient hardware resources for the SSP workstation to run all SSP-resident software without slowing system response.

These steps are explained in the following sections.

# Recommended Patches

Sun produces a set of patches, Recommended and Security Solaris Patch Clusters, which contain Solaris software updates of universal interest for each version of the Solaris Operating Environment. These selected patches are important and highly recommended because they provide fixes for critical system, user, or security-related bugs. Some of these patches also fix performance problems. They are generally safe to apply, as opposed to higher-risk patches, or patches with new features, new drivers, or low-priority fixes, which are not included in these patch clusters. A prudent system administrator keeps systems current with the latest recommended patch level to protect against system problems. Solaris Recommended Patch Clusters are available at the SunSolve[SM] website, `http://SunSolve.Sun.COM/`.

An SSP workstation should be regularly updated with all the SSP patches available for the particular release of SSP software used, except for special-case patches noted in the patch README file. These patches should be applied only on a case-by-case basis. SSP patches are also available from the SunSolve website, `http://SunSolve.Sun.COM/`.

# Performance Improvements

SSP software has been hardened so that third-party applications can be run on the SSP workstation. The performance improvements include the use of real-time processes, which enable SSP to run with fewer resources and to function continuously, even when CPU and memory utilization on the SSP is high. Upgrade to SSP 3.4 with SSP 3.4 patch 110193-01 to take advantage of these performance improvements.

# Performance Improvements to the Solaris Operating Environment

To fully benefit from the SSP performance improvements, use the Solaris 8 Operating Environment on the SSP workstation, if possible. The Solaris 8 software improvement most relevant to the SSP involves better thread handling. When a real-time SSP thread is blocked by a lower-priority thread, the kernel temporarily assigns a higher priority to the blocking thread in order to quickly complete and release the blocked resource. This results in faster SSP response time to Sun Enterprise 10000 events.

Other performance improvements in the Solaris 8 Operating Environment can affect third-party applications running on the SSP, especially if those applications communicate over a WAN or use a large number of open files or sockets. For details, see *What's New in the Solaris 8 Operating Environment* at `http://www.sun.com/software/solaris/whatsnew.html`.

# SSP Resource Requirements

The following table lists the SSP minimum resource requirements, obtained from the *SSP 3.4 Installation Guide and Release Notes*, and the OpenSSP resource requirements. OpenSSP resource requirements are higher than the SSP minimum requirements because of the extra load expected on the SSP system.

**TABLE 1**    Required SSP Resources

| Resource | SSP Minimum Requirement | OpenSSP Minimum Requirement |
|---|---|---|
| Solaris Operating Environment | Version 2.6, 7, or 8 | Version 7 or 8 |
| System Service Processor (SSP) | Version 3.1.1, 3.2, 3.3, or 3.4 | Version 3.4 with patch 110193-01 for the SSP performance improvement |
| Workstation | SPARCstation 5 (supports 1 to 8 domains), Sun Ultra 5, or Sun Enterprise™ 250 workstation (supports 1 to 16 domains) | Sun Ultra 5 or Sun Enterprise 250 workstation |

**TABLE 1** Required SSP Resources *(Continued)*

| Resource | SSP Minimum Requirement | OpenSSP Minimum Requirement |
|---|---|---|
| Disk space | 2.1 Gbytes for the Solaris Operating Environment and SSP software | 3.2 Gbytes for the Solaris Operating Environment and SSP software<br><br>This includes 1.1 Gbytes for the Solaris Operating Environment, 1 Gbyte for SSP software and log files (in `/var/opt/SUNWssp/` and `/var/adm/`), 512 Mbytes swap space, and 512 Mbytes or more for third-party applications |
| Processor speed | 170 MHz | 270 MHz |
| CPU utilization | None | 35% idle |
| Real memory | 64 Mbytes | 128 Mbytes |
| Swap space | 512 Mbytes | 512 Mbytes<br><br>Memory requirements will vary depending on the software used on the SSP. The amount of real memory required usually ranges from 128 Mbytes to 256 Mbytes. The swap space amount usually ranges from 512 Mbytes to 1042 Mbytes. For details, see "Sizing SSP Memory." |

# Sizing SSP Memory

To measure the memory requirements of the SSP workstation, you must consider the cumulative requirements of all applications *as a whole*, not just the impact of an individual application. To do this, first determine the type of run-time environment involved by answering the following questions:

- What is the maximum number of domains running on the Sun Enterprise 10000 system?
- How many Hostview applications could be running at the same time?
- Is Sun™ Management Center installed on the SSP?
- Are third-party applications running on the SSP?
    - If so, what are their virtual and real memory requirements?
    - What is the CPU overhead?

# Calculating Memory Usage by Third-Party Applications

If the SSP is running third-party applications, determine how much virtual and real memory is used by these applications. This memory amount can usually be found in the installation or administrator guide for the application. If this information is not available, it can be easily calculated using the memory usage output from the pmap command. Use this command when the system is not thrashing (paging at a high rate) , and the application is in an active running state so that the command output shows how much resident memory the application requires when it is active, but not thrashing. For information on determining whether a system is thrashing, see "Verifying That You Have Sufficient Real Memory."

The following example shows how to size an application called CST (Configuration and Service Tracker), which has one process, cstd. (What this application does is not relevant here, as it serves only as an example of measuring memory usage.)

To obtain the memory information needed to calculate the memory usage for the application, type pmap -x, followed by the process ID of the application, as shown in the following example:

```
# pgrep cstd
406
# /usr/proc/bin/pmap -x 406
406:   /opt/SUNWcstv/bin/cstd -b
Address  Kbytes Resident Shared Private Permissions   Mapped File
. . .
total Kb  2848  2496  1400  1096
```

The last line in this example shows that 1096 Kbytes of resident private memory is being used. To obtain the virtual memory amount, subtract the shared memory from the total memory, then round up the resulting value. For instance, the virtual memory is 1448 Kbytes, which is derived by subtracting the shared memory (1400 kbytes) from the total memory (2848 Kbytes). Round up this value to 2 Mbytes. In this example, CST requires 1 Mbyte of resident memory and 2 Mbytes of virtual memory.

Some third-party applications have their own application-specific shared libraries. For these applications, you must add the real and virtual memory sizes of these libraries. The virtual memory used for these shared libraries is approximately the same as the shared library (*.so) file size. The resident memory used by shared libraries is also shown in the pmap -x command output.

# Using the SSP Memory Worksheet

To calculate the virtual and real memory requirements for an SSP workstation, use the SSP memory worksheet. The following is an example of a completed worksheet, which contains sample entries in bold font.

SSP Memory Worksheet Example

| Line | Item | Number | Real Memory (MB) | Virtual Memory (MB) | Real Memory Subtotal (MB) | Virtual Memory Subtotal (MB) |
|------|------|--------|------------------|---------------------|---------------------------|------------------------------|
| 1 | System | 1 | 60 | 236 | 60 | 236 |
| 2 | Base SSP | 1 | 22 | 35 | 22 | 35 |
| 3 | Domains (1-16) | **4** | 3 * no. of domains | 4 * no. of domains | **12** | **16** |
| 4 | Hostviews (0 or more) | **1** | 12 * no. of Hostviews | 17 * no. of Hostviews | **12** | **17** |
| 5 | Sun Management Center (0 or 1) | **0** | 0 or 26 | 0 or 31 | **0** | **0** |
| 6 | Third-party applications | | | | **0** | **0** |
| 7 | Subtotal (lines 1 to 6) | | | | **106** | **304** |
| 8 | Kernel buffer memory (Mbytes) | **128 MB RAM** | 15% of RAM | | **19** | |
| 9 | Recommended real memory (lines 7 and 8) | | | | **125** | |
| 10 | Reserved for `/tmp/` in swapfs | | | 512 | | 512 |
| 11 | Subtract amount of real memory | | | | | **-128** |
| 12 | Recommended swap space size | | | (Virtual memory subtotal – real memory total) + `/tmp/` reserved | | **688** |

This section that follows describes the steps to completing the worksheet for your SSP configuration. Each step also explains the sample entries made in the example worksheet (the entries in bold font). For information on how the predetermined values in the worksheet were derived, see "Appendix B: How Memory Sizing Values are Derived."

## ▼ To Complete the SSP Memory Worksheet

1. **In line 3:**

   a. **In the Number column, enter the highest number of domains (1 to 16) that you expect to have for your Sun Enterprise 10000 system.**

   b. **Multiply the number of domains by 3 Mbytes and enter the result in the Real Memory Subtotal column.**

   c. **Multiply the number of domains by 4 Mbytes and enter the result in the Virtual Memory Subtotal column.**

   *In line 3 of the example worksheet, 4 domains are specified, which results in 12 Mbytes for the Real Memory Subtotal and 16 Mbytes for the Virtual Memory Subtotal.*

2. **In line 4:**

   a. **In the Number column, enter the number of Hostview applications (the SSP GUI) that you expect to run at the same time. (This entry is usually 1).**

   b. **Multiply the number of Hostview applications by 12 Mbytes and enter the result in the Real Memory Subtotal column.**

   c. **Multiply the number of Hostview applications by 17 Mbytes and enter the result in the Virtual Memory Subtotal column.**

   *In line 4 of the example worksheet, 1 Hostview application is specified. The Real Memory Subtotal value is 12 Mbytes, and the Virtual Memory Subtotal value is 17 Mbytes (see the last two columns).*

3. **If Sun Management Center is installed and running, enter the following in line 5:**

   a. **In the Number column, enter 1.**

   b. **In the Real Memory Subtotal column, enter 26 Mbytes.**

   c. **In the Virtual Memory Subtotal column, enter 31 Mbytes.**

   *In line 5 of the example worksheet, Sun Management Center is not used, so 0 is entered in the Number, Real Memory Subtotal, and Virtual Memory Subtotal columns.*

4. **In line 6, enter the real and virtual memory amounts required for any third-party applications that will be running on the SSP workstation. For details on estimating these memory requirements, see "Calculating Memory Usage by Third-Party Applications."**

   *In line 6 of the example worksheet, 0 is entered in the Real Memory Subtotal, and Virtual Memory Subtotal columns because no third-party applications are being used.*

5. **In line 7, subtotal the values in the Real Memory Subtotal column and the Virtual Memory Subtotal column.**

   *In line 7 of the example worksheet, the subtotal for the Real Memory Subtotal values is 106, and the subtotal for the Virtual Memory Subtotal values is 304.*

6. **In line 8:**

   a. **In the Number column, enter the RAM that you will need. This number must be greater than 115% of the subtotal for the Real Memory entered in line 7. You must round this value up to the next 32 Mbyte increment. The Solaris Operating Environment uses 15% of the RAM for kernel buffer memory.**

   b. **In the Real Memory Subtotal column, enter 15% of the RAM specified in the Real Memory Subtotal column. This is the amount of buffer memory used by the kernel.**

   *In line 8 of the example worksheet, 128 Mbytes of RAM is specified in the Number column, which is greater than the 106 Mbytes of Real Memory Subtotal entered in line 7. Also, 15% of 128 Mbytes of RAM yields 19 Mbytes of kernel buffer memory.*

7. **In line 9, add the values from lines 7 and 8 and enter the resulting value in the Real Memory Subtotal column. This number typically ranges from 128 Mbytes to 256 Mbytes.**

   *In line 9 of the example worksheet, adding 106 Mbytes and 19 Mbytes results in the minimum memory requirement of 125 Mbytes.*

8. **In line 11, the Virtual Memory Subtotal column, enter the negative value of the RAM supplied in line 8.**

   *In line 8 of the example worksheet, the RAM value is 128 Mbytes, so –128 is specified in the Virtual Memory Subtotal column in line 11.*

9. **In line 12:**

   a. Add the subtotals from lines 7 and 10, then subtract the virtual memory total (negative RAM) in line 11 from that amount.

   b. Enter the resulting value in the Virtual Memory Subtotal column. This number is the minimum swap space size needed by the SSP workstation and typically ranges from 512 Mbytes to 1 Gbyte.

*In line 12 of the example worksheet, 304 Mbytes of virtual memory is added to 512 Mbytes (for /tmp/), which results in 816 Mbytes. The real memory value, 128 Mbytes, is subtracted from 816 Mbytes, which yields a minimum swapfile size of 688 Mbytes.*

**Note –** The size limit for a swap partition is 2 Gbytes. However, you can add multiple swap partitions if needed.

**Bottom line**: The minimum amount of memory needed for a SSP workstation is 128 Mbytes. If you do not want to calculate the exact amount required, 256 Mbytes of memory is more than sufficient if you are using other monitoring software. One Gbyte of swap space is more than sufficient for virtual memory and `swapfs` (`/tmp/`) space.

# Verifying That You Have Sufficient Real Memory

Virtual memory consists of real memory (RAM) and page file (swap) space on disk. Unlike some other systems, real memory for the Solaris Operating Environment is not mirrored in a swapfile. It is no longer necessary to duplicate a page of swap for each page of real memory, so the old rule that "swap space size should be twice real memory size" no longer applies. The only swap space required is the amount of virtual memory that exceeds the real memory for your system.

The amount of virtual memory required depends on the *working set model* for a process. The working set is the set of pages a process needs to work effectively. A working set needs to be in real memory or the program may thrash. Thrashing occurs when there is insufficient real memory for all the working sets of a process. As a result, the system spends an excessive amount of time paging the process working sets in and out of swap space.

The working set for a program is defined as $W(t, \omega)$, which is the set of pages referenced from time $(t - \omega)$ to time $t$. Typically, a working set for a program does not change much over time, although it can change drastically on occasion. Increasing the time period, $\omega$, does not have much effect on the working set. Pages currently in use are likely to be used in the near future. Memory outside the working set is rarely, if ever, used. Therefore, a program that uses only its working set in real memory and the remainder in swap space will perform almost as effectively as if all of its pages were in memory. This is true even though disk access time is about 100,000 times slower than RAM access time (about 10,000,000 nanoseconds versus 100 nanoseconds).

However, if there is insufficient real memory to keep the working set for a process in memory, the process can easily thrash and run more slowly. Running fewer processes or adding more real memory keeps the process working set in memory and stops the process from thrashing. Thrashing can affect the SSP ability to handle events in a timely manner, due to timeouts and lost SNMP traps. Thrashing can be prevented by properly sizing the SSP system for all applications that it runs.

How do you know if a system is thrashing? The easiest way is to check the paging scan rate (sr). The kernel for the Solaris Operating Environment uses a page scanner, which scans a circular list of pages in memory in order to reclaim memory and swap it out to disk. Pages not referenced since the last cycle are paged out of memory. The scanner runs faster when demand for memory increases. If the demand is too high, memory in a working set for a process can be removed from real memory, which slows those processes. This can prevent SSP processes from reacting quickly to real-time events. Also, as its scan rate increases, the page scanner uses more CPU time.

If you suspect the system is thrashing, use the vmstat command to sample and display virtual memory statistics. This command adds little overhead and can safely run for long periods of time, if required. To use this command, type vmstat followed by the number of times you want it to sample, optionally followed by the frequency to sample, in seconds.

For example,

```
% vmstat 5
```

prints results every five seconds, while

```
% vmstat 3 100
```

prints results every three seconds for 100 times.

Review the sr column in the output displayed by vmstat. Ignore the entries in the first row, as the values are cumulative based on when the system was booted. If subsequent values in the sr column are non-zero, the system is thrashing. Ignore the po (page-out) column, as those values includes swapfs (swap file system or /tmp/) activity.

The following example shows the `vmstat` output for a system that is thrashing:

```
% vmstat 3
procs     memory                      page                     disk          faults        cpu
r  b w   swap  free  re  mf  pi  po   fr  de   sr dd f0 s0 --  in   sy    cs  us sy id
0  0 0 669856 7336  41 233   5   5    5 136    0  4  0  0   0 234 3107 1459 5   8  87
3  4 0 597232 2136 131 717 354 317  533 128  302 48  0  0   0 399 4889 2252 65 33   2
7  1 0 597120 3408 175 745 197 133  218   0  137 61  0  0   0 430 4757 2130 67 33   0
11 0 0 595832 2456 145 757 184 221  376 424  272 26  0  0   0 378 5235 2380 65 35   0
^C
```

In the output above, note the `sr` column but ignore the first entry. The `sr` values are 302, 137, 272, and 121, which indicate the system is thrashing heavily.

The next example shows a system that is not thrashing. The values in the `sr` column are zero, indicating that there is no excessive page scanning by the Solaris kernel to free pages:

```
% vmstat 3
procs     memory             page                  disk          faults       cpu
r b w   swap free   re   mf pi po fr de sr dd f0 s0 --  in   sy    cs  us sy id
0 0 0 672728 8376  41  236  4  5  5  0  0  4  0  0   0 243 3358 1585  6  9 85
0 0 0 672472 6960  29   46  0  0  0  0  0  0  0  0   0 239 3858 1924  8  4 88
0 0 0 672488 6992  59  374  0  0  0  0  0  0  0  0   0 237 4215 1933  5 11 83
0 0 0 666968 7248  87  811  2  0  0  0  0 13  0  0   0 266 4971 1938 24 29 47
0 0 0 672520 7200  47  176  0  0  0  0  0  0  0  0   0 292 4043 2043  8  6 86
0 0 0 672520 7200   0    0  0  0  0  0  0  0  0  0   0 240 3516 1861  4  0 96
0 0 0 672520 7200  31   74  0  0  0  0  0  0  0  0   0 235 3726 1876 14  4 82
^C
```

# CPU Utilization

When some people notice a system with low CPU utilization, for example 25%, they might ask, "Why is this high-performance system not doing anything? Should we use a lower-end system or find more work for it to do?"

Surprisingly, the answer to the latter question may be "no." For a batch-processing system, where response times are not as critical, high utilization is usually preferred. However, for an interactive or a real-time controlling system such as SSP, response time is more critical than high CPU utilization. High utilization leads to slower response time, as noted in queuing theory. As utilization approaches 100%, the wait time increases exponentially.

Queuing theory uses models to predict utilization and wait time for a client/server system. The operations in a retail bank, hospital emergency room, or computer server are examples of a client/server system. One of the basic assumptions in queuing theory is that the arrival time between two customers is an exponential distribution. In other words, long periods between customer arrivals are more unlikely than short periods.

Queuing theory is best illustrated by an example. Assume there is a small town bank, the Bank of Ethel, which has one teller (Ethel), and several customers. Ethel's utilization (how busy she is) can be determined using the following formula for a single-server model (one teller):

$$\rho = \frac{\lambda}{\mu} < 1$$

where $\rho$ (rho) is the proportion of time the servers (tellers) are busy (on a scale of 0 to 1.0, where 0 is no customers at all and 1 indicates the server is completely busy), $\lambda$ (lambda) is the mean arrival rate, and $\mu$ (mu) is the mean service rate per server.

In this example, if two customers come to the Bank of Ethel every hour and Ethel serves an average of six customers an hour, $\lambda = 2$, $\mu = 6$, and Ethel's utilization is: $\rho = 2 / 6 = 1/3$, or about 0.33. Multiply $\rho$ by 100 to convert $\rho$ to percent, for example, 33%.

The number of customers expected in the bank at any one time is:

$$L = \frac{\lambda^2}{\mu(\mu - \lambda)}$$

In the above example, the expected number of customers is $L = 2^2 /6(6 - 2) = 1/6$ or about 0.17. That is, on average there will be about 0.17 customers at the Bank of Ethel, which is not very busy.

What if the number of customer arrivals increases from two to five per hour? Then $\lambda = 5$, and the utilization will be $\rho = 5/6$ or about 0.83. This means that Ethel will be serving a customer 83% of the time. However, this has a drastic effect on L, the expected number of customers in the bank. In this example, $L = 5^2/6(6 - 5) = 25/6$ or about 4.17 customers in the bank, on the average. The number of customers waiting for service is (L - 1), so in this example, there are about 3.17 customers waiting for service. This example shows why high utilization and immediate service are not possible at the same time.

For an SSP workstation, assume the CPU of the SSP workstation is the server. Utilization, ρ, is the percentage that the CPU is in use. The CPU "customers" are processes that are either being serviced by the CPU or waiting in the queue. The *load average* shown by various commands in the Solaris Operating Environment is (L - 1), which represents the number of waiting customers.

FIGURE 2 illustrates how increased utilization drastically increases customer wait time.
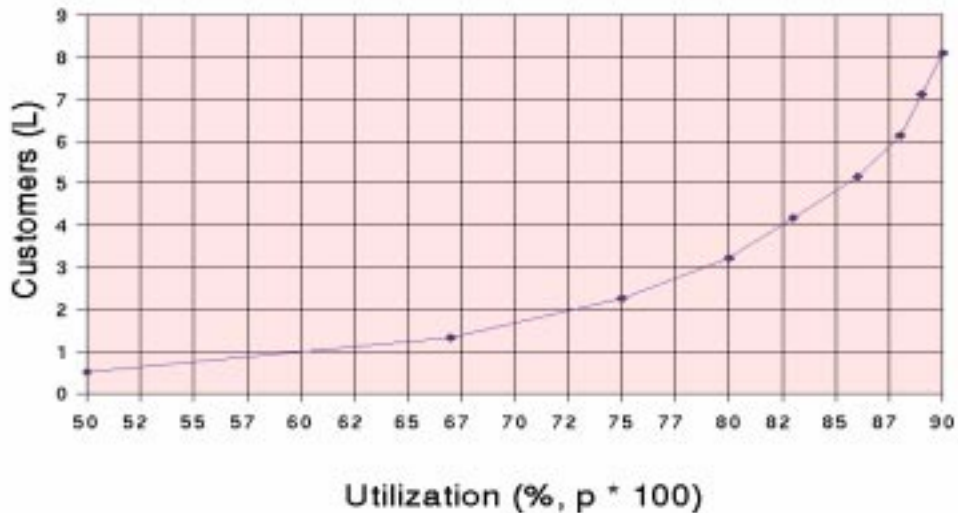


**FIGURE 2**    Utilization and Customers Expected

The X-axis is the utilization of a single server, ranging from 50% to 90%. The Y-axis shows L, the expected number of customers at any one time for a given level of utilization, ρ. If L is greater than or equal to 2, at least one customer is always waiting. At 60% utilization or less, almost no one is waiting for service. When utilization exceeds 72% or so, a customer is almost always waiting. When utilization exceeds 80%, multiple customers are usually waiting. In conclusion, to have quicker service, you must sacrifice high utilization. The highest utilization you can have without having customers wait for service is usually about 65%.

# SSP CPU Requirements

Average CPU utilization should be under 65%, as explained in the previous section. This amount allows sufficient CPU resources for SSP software to immediately handle error conditions on a Sun Enterprise 10000 system. The following table shows the approximate utilization currently used by SSP software at its busiest state (bringup of domains):

TABLE 2    CPU Utilization for SSP Software

| Domains | Sun Ultra 5 Workstation Average CPU Usage | Sun Enterprise 250 Workstation Average CPU Usage |
|---------|-------------------------------------------|--------------------------------------------------|
| 1 to 4  | 24%                                       | 17%                                              |
| 5 to 8  | 29%                                       | 18%                                              |
| 9 to 16 | 31%                                       | 20%                                              |

If you are using the Sun Management Center, add the CPU overhead values in TABLE 3:

TABLE 3    Additional CPU Utilization for Sun Management Center

| Domains | Sun Ultra 5 Workstation Average CPU Usage | Sun Enterprise 250 Workstation Average CPU Usage |
|---------|-------------------------------------------|--------------------------------------------------|
| 1 to 8  | 10%                                       | 5%                                               |
| 9 to 16 | 20%                                       | 10%                                              |

If you are using SunScreen™, add the following CPU overhead to the values in TABLE 2:

TABLE 4    Additional CPU Utilization for SunScreen

| Sun Ultra 5 Workstation Average CPU Usage | Sun Enterprise 250 Workstation Average CPU Usage |
|-------------------------------------------|--------------------------------------------------|
| 2%                                        | 4%                                               |

# SSP Workstation Hardware

Your SSP workstation must have enough hardware resources to operate the maximum number of domains planned on the Sun Enterprise 10000 system it controls, as well as what is required to run third-party applications.

The proper hardware technology depends on the usage profile of the third-party software. Less intensive software requires less CPU and memory than resource-hungry applications. You must determine the CPU and real memory resources needed for these applications before determining the appropriate workstation hardware to use.

## Using a Faster Processor

If CPU utilization is too high for your SSP configuration (over 65%), using a faster processor will reduce CPU utilization dramatically. For example, when bringing up 16 domains on a Sun Ultra5 workstation, an UltraSPARC™ II processor running at 360 MHz has about one-half the CPU utilization as a slower CPU running at 270 MHz. A faster processor is also appropriate if you add monitoring or third-party software, such as Sun Management Center, which uses a lot of CPU.

When comparing CPUs, remember to consider the CPU family and L2 cache size, in addition to CPU speed. The UltraSPARC II processor on the Sun Enterprise 250 workstation is faster than the UltraSPARC IIi processor on the Sun Ultra 5 workstation. The UltraSPARC II processor comes with more L2 cache (1 to 4 Mbytes) than the UltraSPARC IIi processor (256 Kbytes to 2 Mbytes).

## Adding More Memory

If a system does not have sufficient memory, it will thrash—that is, swap pages excessively in and out of real memory. Having sufficient memory modules will prevent thrashing and enable software to run with fewer interruptions on the SSP workstation.

## Adding More Swap Space

Adding more swap space if needed will improve system reliability, even though it will not improve system performance. If a system runs out of memory, processes cannot allocate more memory and will begin to fail. Swap space is required to save

inactive processes and memory regions, and to handle overflow in `swapfs` (`/tmp/`). Additionally, automatic SSP failover may need to occasionally propagate large files, such as SSP log files and possibly user-specified files (listed in `$SSPVAR/.ssp_private/user_file_list`, which identifies various data files, including files used by third-party applications). Therefore, it is important to have an adequately-sized swap file to hold these files.

## Adding More Disk Space

The SSP requires 1 Gbyte of unused disk space for the file system containing `/var/`, which is used to store SSP log files (under `/var/opt/SUNWssp/` and `/var/adm/`) and SSP backup files (usually under `/var/tmp/`). If the file system fills up, the SSP can exhibit strange behavior, such as freezing, respawning processes, or login failures, and event information might not be saved in logs.

# Conclusion

The SSP workstation has the critical job of monitoring the Sun Enterprise 10000 system so that it stays up and running. Installing third-party software requires extra consideration to help ensure that these applications do not affect the proper operation of the SSP.

Installing third-party software on the SSP workstation requires you to perform the following steps:

- Determine that your SSP workstation meets the minimum hardware and software requirements for OpenSSP. These requirements are generally greater than the SSP minimum requirements.
- Estimate the amount of real memory and swap space required by your SSP and third-party software. This amount can be above the OpenSSP minimum requirements.
- Verify that your CPU utilization is not too high. If it is, consider using a faster CPU or a more powerful workstation, for example, a Sun Enterprise 250 over a Sun Ultra 5 workstation.

# Appendix A: Blank SSP Memory Worksheet

This section contains a blank SSP memory worksheet (below) that you can use to determine the memory requirements for your SSP configuration.

**FIGURE 3**    Blank SSP Memory Worksheet

| Line | Item | Number | Real Memory (MB) | Virtual Memory (MB) | Real Memory Subtotal (MB) | Virtual Memory Subtotal (MB) |
|---|---|---|---|---|---|---|
| 1 | System | 1 | 60 | 236 | 60 | 236 |
| 2 | Base SSP | 1 | 22 | 35 | 22 | 35 |
| 3 | Domains (1-16) | | 3 * no. of domains | 4 * no. of domains | | |
| 4 | Hostviews (0 or more) | | 12 * no. of Hostviews | 17 * no. of Hostviews | | |
| 5 | Sun Management Center (0 or 1) | | 0 or 26 | 0 or 31 | | |
| 6 | Third-party applications | | | | | |
| 7 | Subtotal (lines 1 to 6) | | | | | |
| 8 | Kernel buffer memory (Mbytes) | | 15% of RAM | | | |
| 9 | Recommended real memory (lines 7 and 8) | | | | | |
| 10 | Reserved for /tmp/ in swapfs | | | 512 | | 512 |
| 11 | Subtract amount of real memory | | | | | |
| 12 | Recommended swap space size | | | (Virtual memory subtotal – real memory total) + /tmp/ reserved | | |

# Appendix B: How Memory Sizing Values are Derived

The SSP memory worksheet (FIGURE 3) is used to size virtual and real memory for a SSP workstation, based on the number of Sun Enterprise 10000 domains controlled by the SSP and the number of SSP, Sun Management Center, and third-party applications involved.

This section provides details on how the predetermined memory values in the worksheet are derived. The `pmap` command provides output that is used to calculate the following memory values in the worksheet.

- Process private resident memory
- Process private virtual memory
- System memory (line 1 of the worksheet)
- Base SSP memory (line 2 of the worksheet)
- Each domain SSP memory (line 3 of the worksheet)
- Hostview memory (line 4 of the worksheet)
- Sun Management Center memory (line 5 of the worksheet)
- Kernel buffer memory (line 8 of the worksheet)
- `swapfs` (line 10 of the worksheet)

The following sections describe how each of these values is calculated.

## Process Private Resident Memory

Use the `pmap -x` *processID* command to obtain the amount of private resident memory used by a process, where *processID* is the number of the process ID (for example, 406). In the output displayed, note the `Private` column in the last line. This value is the total private memory, in kilobytes, for a process that is resident in real memory.

## Process Private Virtual Memory

To find the approximate amount of private virtual memory that a process uses, run the `pmap -x` *processID* command. In the output displayed, locate the total resident shared memory (shown on the last line under the `Shared` column) and subtract it from the total process memory (shown on the last line under the `Kbytes` column).

The resulting value is the approximate total private virtual memory for a process. It is slightly higher than the actual amount needed, as it includes nonresident shared memory (paged-out memory that is also used by other processes). The resulting value is a reasonable measurement of the private virtual memory required, which does not underestimate the required memory.

For further information on memory sizing, see Richard McDougall's paper, "*The Solaris Memory System: Sizing, Tools and Architecture.*"

# System Memory (Line 1)

System memory usage consists of memory used by system processes, system shared libraries, CDE libraries, and CDE processes. TABLE 5 shows the values for these items and the resulting totals for system memory: 60 Mbytes of real memory and 60 Mbytes of virtual memory. (The kernel buffer memory used is discussed in a subsequent section.)

**TABLE 5**     System Memory Values

| Memory | Resident Memory (Mbytes) | Virtual Memory (Mbytes) | Description |
|---|---|---|---|
| System processes | 21 | 56 | Use `pmap -x` to find the private resident and virtual memory of root processes, that is `/usr/lib/*`, `/usr/sbin/*`, `/usr/sadm/*`, `rpc.*`, and `mibiisa`. |
| System shared libraries | 15 | 25 | The total resident and virtual memory for system shared libraries, which are under `/usr/lib/`, is approximately 15 and 25 Mbytes (see McDougall, pp. 34-36). |
| CDE libraries | 5 | 10 | Use `pmap -x` to find the highest shared memory among `/usr/dt/bin/*` and `/usr/openwin/bin/*` processes. |
| CDE processes | 19 | 145 | Use `pmap -x` to find the private resident and virtual memory of `Xsun`, `speckeysd`, `fbconsole`, `sdt_shell`, `ttsession`, and `dt*` processes. |
| System subtotal | **60** | **236** | |

# Base SSP Memory (Line 2)

Base SSP memory consists of memory used by the SSP shared libraries and SSP platform-wide processes. TABLE 6 shows the subtotals for base SSP memory: 22 Mbytes of real memory and 35 Mbytes of virtual memory.

**TABLE 6**    Base SSP Memory Values

| Memory | Resident Memory (Mbytes) | Virtual Memory (Mbytes) | Description |
| --- | --- | --- | --- |
| SSP shared libraries | 3 | 4 | Sum the file sizes of the *.so files under `/opt/SUNWssp/lib/` and `/opt/SUNWssp/release/` |
| | | | This gives the approximate virtual memory used by the SSP libraries. |
| SSP platform software | 19 | 31 | Use `pmap -x` to find the private resident and shared memory of SSP platform-wide processes. That is, `snmpd`, (but not `snmpdx`), `ssp_startup.*`, `fad`, `edd`, `straps`, `machine_server`, `fod`, `cbs`, and `datasyncd`. |
| SSP platform subtotal | *22* | **35** | |

# Each Domain SSP Memory (Line 3)

Each domain on the Sun Enterprise 10000 system requires three processes on the SSP:

- `obp_helper`
- `netcon_server`
- `netcon` client

Output from the `pmap -x` command shows that these three processes take 3 Mbytes of private resident memory and 4 Mbytes of private virtual memory.

# Hostview Memory (Line 4)

Each instance of Hostview, with all three status subwindows open, requires four processes. The output from the `pmap -x` command shows that these four processes use 12 Mbytes of private resident memory and 17 Mbytes of private virtual memory.

Hostview is not required for the SSP to function, but is useful in monitoring alerts and events if no other monitoring software, such as Sun Management Center, is being used.

# Sun Management Center Memory (Line 5)

On the SSP, Sun Management Center requires two agents to send information back to the Sun Management Center server. One agent monitors the SSP workstation itself, while the other agent monitors the Sun Enterprise 10000 platform. Sun Management Center appears in the `ps` command as two processes, both named `esd`. TABLE 7 presents the output from the `pmap -x` command for the Sun Management Center processes and the resulting subtotals for resident and virtual memory:

**TABLE 7**    Sun Management Center Memory Values

| Memory | Resident Memory (Mbytes) | Virtual Memory (Mbytes) |
|---|---|---|
| Sun Management Center workstation agent | 14 | 16 |
| Sun Management Center platform agent | 12 | 15 |
| Sun Management Center subtotal | **26** | **31** |

Due to the large footprint of the Sun Management Center server or console, running Sun Management Center on the same system with the SSP is not recommended. Sun Management Center is neither required nor used by SSP software.

# Kernel Buffer Memory (Line 8)

The following quote from page 37 of Richard McDougall's paper, *The Solaris Memory System: Sizing, Tools and Architecture*, provides a general rule of thumb for determining kernel RAM:

> The amount of memory that the kernel uses varies significantly, based on the size of the tunable parameters. A lot of the tunable parameters are set at boot in proportion with the amount of physical RAM in the system. As a general rule of thumb, if all of the parameters are standard, you can allow about 15% of physical RAM for the kernel.

## `swapfs` (Line 12)

The `swapfs` file system implements a file system in virtual memory, using both RAM and, if necessary, swap space, to store files. However, because this memory is not preserved across reboots, `swapfs` is used only for temporary files, usually `/tmp/`. SSP 3.4 automatic failover uses `/tmp/` to propagate modified files to the other SSP in a dual configuration, mainly log files and recently changed configuration files, as part of the process of failing over to the other SSP. It is important to allocate enough space for `swapfs`, as log files can become quite large. Otherwise, the SSP workstation might behave strangely or even fail. Allocating 512 Mbytes of swap space for `/tmp/` should be sufficient for SSP 3.4 automatic failover. This amount is in addition to the amount of required virtual memory that exceeds available RAM.

If the system is generating a large number of diagnostic messages to the SSP logs, or the system is generating a lot of log messages due to an error situation, the SSP log file sizes under directory `/var/opt/SUNWssp/adm/` need to be monitored. If, in rare instances, log files in this directory grow to hundreds of megabytes in size, trim or remove them from the `/var/opt/SUNWssp` directory.

# References

Sun reference manuals are available at `http://docs.sun.com/` .

1. "Managing Quotas (Tasks)," *System Administration Guide, Volume II*, part number 805-7229-10 (2000).

   Provides information on managing user quotas on file systems.

2. Sun performance information Web page at `http://www.Sun.COM/sun-on-net/performance.html`

   Provides links to papers and books on Sun performance.

3. *Sun Performance and Tuning: Java and the Internet* by Adrian Cockcroft & Richard Pettit, 2nd ed. (Prentice Hall, 1998; ISBN 0-13-95249-4).

   Practical book by two Sun engineers on how to measure Solaris Operating Environment and application performance, and how to improve it.

4. "The Solaris Memory System: Sizing, Tools and Architecture," by Richard McDougall (rev. D, May 1998), at `http://www.Sun.COM/sun-on-net/performance/vmsizing.pdf`

   Offers practical information on measuring system and application memory usage.

5. "Comparing swap space: Solaris challenges traditional Unix Implementations," Inside Solaris (July 2000) by Richard McDougall, at `http://www.Sunworld.COM/sunworldonline/swol-07-2000/swol-0728-insidesolaris-rm_p.html`

6. *Configuration and Capacity Planning for Solaris Servers* by Brian Wong (Prentice Hall, 1997; ISBN 0-13-349952-9).

   Chapter 9 has practical information on virtual memory, file systems, and scheduling. Other chapters cover real-life examples on various classes of servers.

7. *Solaris Internals: Core Kernel Components* by Richard McDougall & Jim Mauro (Prentice Hall, 2000; ISBN 0-13-022496-0).

   Chapter 5, "Solaris Memory Architecture," includes information on Virtual Memory and the page-scanning algorithm. See `http://www.Sun.COM/books/`

8. Solaris Resource Manager (SRM)

   Reserves or limits CPU and other system resource usage for a set of processes.

   ■ Solaris Resource Manager (SRM) Web page
     `http://www.Sun.COM/solaris/ds/ds-resourcemgr/`
   ■ *Solaris Resource Manager 1.2 System Administration Guide*, part number 806-4053-10 (2000).
   ■ *Solaris Resource Manager 1.2 Reference Manual*, part number 806-4777-10 (2000).
   ■ *Solaris Resource Manager 1.2 Release Notes*, part number 806-4778-10 (2000).
   ■ *Solaris Resource Manager 1.2 Installation Guide*, part number 806-4776-10 (2000).

9. System Service Processor (SSP)

   ■ System Service Processor (SSP) Web page
     `http://www.Sun.COM/servers/highend/10000/Tour/ssp.html`
   ■ *Sun Enterprise 10000 SSP 3.4 User Guide*, part number 806-4870-10 (2000).
   ■ *Sun Enterprise 10000 SSP 3.4 Reference Manual*, part number 806-4871-10 (2000). Man pages.
   ■ *Sun Enterprise 10000 SSP 3.4 Installation Guide and Release Notes*, part number 806-4872-10 (2000).