

组件 1 : Unix 平台服务器操作系统 (CPU)

序号	指标名称 (中英文)	指标理解	建议值	访问方式	瓶颈判断
1	<ul style="list-style-type: none"> %usr us user mode cpu utiliza tion 	<p> 显示了用户方式下所花费CPU 时间的百分比。</p> <p> 用户使用CPU的进程包括：</p> <ul style="list-style-type: none"> n CPU运行常规用户进程 n CPU运行niced process n CPU 运行实时进程 <p>注解： 一个 UNIX 进程可以在用户方式下执行,也可以在系统(内核)方式下执行。当在用户方式下执行时,进程在它自己的应用程序代码中执行,不需要内核资源来进行计算、管理内存或设置变量。</p>	合理值： 60-85%	<p>监控工具： Loadrunner Quest等 下述指标使用监控工具同此</p> <hr/> <p>监控命令： sar -u vmstat</p>	<ul style="list-style-type: none"> 如果在一个多用户系统中us + sy 时间超过85%，则进程可能要花时间在运行队列中等待，响应时间和业务吞吐量会受损害； us 过大，说明有用户进程占用很多CPU 时间，需要进一步的分析其它软硬件因素； sy 过大，说明系统管理方面花了很多时间。需要进一步的分析其它软硬件因素。
2	<ul style="list-style-type: none"> %sys sy syste m mode cpu utiliza tion 	<p> 显示了系统方式下所花费CPU 时间的百分比。这包括内核进程(kprocs)和其它需要访问内核资源的进程所消耗的CPU 资源。</p> <p> 系统使用CPU的进程包括：</p> <ul style="list-style-type: none"> n 用于系统调用 n 用于I/O管理：中断和驱动 n 用于内存管理：paging 			

		<p>and swapping</p> <p>用于进程管理 : context switch and process start</p> <p>注解 : 如果一个进程需要内核资源 ,它必须执行一个系统调用 ,并由此切换到系统方式从而使该资源可用。</p>			
3	<ul style="list-style-type: none"> wa iow ait 	<ul style="list-style-type: none"> 显示了暂挂本地磁盘I/O 和NFS 加载的磁盘的CPU 空闲百分比, 是由于进程等待I/O 而使CPU 处于空闲状态的比率。 I/O 主要包括 block I/O,raw I/O,VM paging/swapins <p>注解 : 如果在wait 运行时至少有一个未完成的磁盘I/O ,该时间就归为I/O 等待时间。对磁盘的I/O 请求会导致调用的进程阻塞(或睡眠),直到请求完成为止。一旦进程的I/O 请求完成,该进程就放入运行队列中。如果I/O 很快完成,该进程可以使用更多的CPU 时间。</p>	小于 25%	<p>监控命令 :</p> <p>sar -u</p> <p>vmstat</p> <p>iostat</p>	<ul style="list-style-type: none"> 超过 25% 的wa 的值可以 表示磁盘子系统可能没有被正确平衡 ; 也可能是磁盘密集工作负载的结果 ; 系统的磁盘或其他I/O 可能有问题 ; 可用 iostat/SAR -C命令对系统调用进行进一步分解。
4	<ul style="list-style-type: none"> idle id 	<p>是除了上面的WIO 以外的空闲情况,显示了没有本地磁盘I/O 时CPU 空闲或等待的时间百分比。</p> <p>注解 :</p>	大于 40	<p>监控命令 :</p> <p>Vmstat</p> <p>iostat</p>	<p>如果 r经常大于 4 ,且id经常少于 40 ,表示cpu 的负荷很重</p>

		如果没有线程可以执行（运行队列为空），系统分派一个叫做wait的线程，也称为idleproc。如果ps报告显示这个线程的总计时间较高，这表明存在时间段，其中没有其它线程准备在CPU上运行或等待执行。系统因此大部分时间空闲和等待新任务。			
5	runq-s z r	运行进程队列的长度。 注解： 即处于可运行状态的进程个数的大小，这些进程在内存中准备就绪	小于 4	监控命令： sar -q vmstat	队列大于4时，表明系统的CPU或内存可能有问题，如果r经常大于4，且id经常少于40，表示cpu的负荷很重；当队列变长时，队列中进程在等待CPU调度执行时所花的时间变长。
6	Swpq- sz w	VMM队列中的进程长度，这些进程被交换出去但已准备就绪 注解： VMM指Virtual Machine Monitor，实际的硬件平台通过VMM与各虚拟机打交道。虚拟技术是指在一个CPU上运行多个操作系统，同时又能在多个操作系统上运行多个程序	小于 5	监控命令： sar -q vmstat	当队列变长时，队列中进程在等待CPU调度执行时所花的时间变长
<p>名词解释：</p> <p>用户级：系统用户可以与操作系统进行交互操作，如运行应用和系统命令。用户级通过系统调用接口访问内核级。</p> <p>内核级：操作系统自动运行一些功能，它们主要对硬件进行操作。</p>					
<p>使用命令：</p> <p>sar -q：查看系统的运行进程队列和交换队列长度</p>					

runq-sz: Average length of the run queue(s) of processes (in memory and runnable)

%runocc: The percentage of time the run queue(s) were occupied by processes (in memory and runnable);

swpq-sz: Average length of the swap queue of runnable processes (processes swapped out but ready to run);

%swpocc: The percentage of time the swap queue of runnable processes (processes swapped out but ready to run) was occupied.

fork/s: Number of fork() and/or vfork() system calls per second;

exec/s: Number of exec() system calls per second;

rchar/s: Number of characters transferred by read system calls (block devices only) per second;

wchar/s: Number of characters transferred by write system calls (block devices only) per second.

SAR -c: Report system calls:

scall/s: Number of system calls of all types per second;

sread/s: Number of read() and/or readv() system calls per second;

swrit/s: Number of write() and/or writev() system calls per second;

命令使用经验：

如果scall/s列的值很大，那么这么多的系统调用的原因就必须仔细分析了。

可以查看fork/s和exec/s列的值，看看系统是否在创建大量新的进程。

用time命令测试某个命令和程序的执行效率

可以利用time命令来测试一个命令的执行效率，语法为：time command

可以利用top命令来查看最耗CPU资源的进程。top命令还会根据进程占用CPU资源的多少而动态改变。

它的语法为：top [-s time] [-d count] [-q] [-u] [-h] [-n number]

判断瓶颈的方法以及调优方案：

CPU资源成为系统性能的瓶颈的征兆：

很慢的响应时间(slow response time)

CPU空闲时间为零(zero percent idle CPU)

过高的用户占用CPU时间(high percent user CPU)

过高的系统占用CPU时间(high percent system CPU)

长时间的有很长的运行进程队列(large run queue size sustained over time)

processes blocked on priority

必须注意的是，如果系统出现上面的这些症状并不能说一定是由于CPU资源不够，事实，有些症状的出现很可能是由于其他资源的不足而引起，如内存不够时，CPU会忙

内存管理的事，这时从表面上，CPU的利用是100%，甚至显得不够，如果据此就简单地认为增加CPU就可以解决问题是大错特错了。对CPU需求密集型系统的性能调试

调优方法：

1) 基于硬件的方法：

升级到更快的CPU；

升级到更大的高速缓存；

增加CPU个数；

把应用分布到多个系统中；
 使用无盘结点；
 增加浮点处理器；
 2) 基于软件的方法：
 在不是高峰时间运行批处理；
 Nice unimportant application;
 使用rtpio 命令来帮助重要的应用；
 使用plock命令来帮助重要的应用；
 Turn off system accounting;
 Consider using Taskbroker or DCE;
 优化应用；

组件 1：Unix 平台服务器操作系统 (Memory)

序号	指标名称 (中英文)	指标理解	建议值	访问方式	瓶颈判断
1	buffer cache size	在 windows 操作系统中有，需要查明			
2	page out/in rates po/pi paging-out/in rate	page out是每秒钟写入页面文件 (在硬盘) 中和从物理内存删除的页数 ,page in是读入物理内存的页数	0	监控命令： Sar Vmstat loadrunner	page in 的平均值大于 80表示物理内存不足。 HP-UX 是一个按需调页的操作系统，通常情况下，它只执行调入页面进入内存的操作，以让进程能够运行。只有操作系统觉

					得系统需要释放一些内存空间时，才会执行从内存调出页面的操作，而过高的调出页面操作说明内存缺乏
3	fre	<p>fre显示空闲内存页面的平均数量。一个页面是实内存中的4 KB 的区域。</p> <p>注解：</p> <p>VMM 在空闲列表上保存的最少页面数由vmtune 命令的 minfree 参数决定；</p> <p>当一个应用程序终止时，它所有的工作页面会立即返回给空闲列表。然而，它的持久页面（或文件）仍然在RAM 中，不会添加回空闲列表中，直到由VMM 窃取用于其它程序为止。如果删除相应的文件，也会释放其持久页面；</p> <p>fre 的值可能不表示进程可以使用的所有实内存。如果需要页帧，那与已终止应用程序相关的持久页面将会最先移交给另一个程序。</p>	fre 的值大于 maxfree 的值	监控命令： Vmstat	<p>如果 fre 的值远大于 maxfree 的值，那不太可能出现系统颠簸。系统颠簸意味着系统一直在进行页面调进调出。然而，如果系统正在经历抖动，可以肯定 fre 值很小。</p>

4	avm	活动虚拟内存avm 列显示在收集 vmstat 样本时存在的活动虚拟内存页面数。	?	监控命令： vmstat	
		注解： 缺省策略为延迟页面空间策略。 在该策略下 ,avm 的值可能比使用的调页空间页面数要高。avm 统计信息并不包含文件页面。			

名词解释：

按用途来分，内存可以分成两部分：预留内存和动态内存。

预留内存主要用于存放：

system table
 data structures
 buffer cache

其中系统表和数据结构占用的数量一般很小，但缓存则可能占到很大一部分。

动态内存主要用于存放：

process text
 data stack
 share memory segments

其中各进程锁定的内存会影响动态内存的大小。

使用命令：

利用vmstat 命令分析内存的利用率

vmstat-report virtual memory statistics ;The vmstat command reports certain statistics kept about process, virtual memory, trap, and CPU activity. It also can clear the accumulators in the kernel sum structure.

它的语法： vmstat [-dnS] [interval [count]] vmstat -f | -s | -z

它的选项的说明：

-d: Report disk transfer information as a separate section, in the form of transfers per second.

-n: Provide an output format that is more easily viewed on an 80-column display device. This format separates the default output into two groups: virtual memory information and CPU data. Each group is displayed as a separate line of output. On multiprocessor systems, this display format also provides CPU utilization on a per CPU basis.

-S: Report the number of processes swapped in and out (si and so) instead of page reclaims and address translation faults (re and at)

interval: Display successive lines which are summaries over the last interval seconds. If interval is zero, the output is displayed once only. If the -d option is specified, the column headers are repeated. If -d is omitted, the column headers are not repeated.

count: Repeat the summary statistics count times. If count is omitted or zero, the output is repeated until an interrupt or quit signal is received.

-f: Report on the number of forks and the number of pages of virtual memory involved since

boot-up.

-s: Print the total number of several kinds of paging- related events from the kernel sum structure that have occurred since boot-up or since vmstat was last executed with the -z option.

-z: Clear all accumulators in the kernel sum structure. This option is restricted to the super user.

命令使用经验：

在不带参数的vmstat的命令时，我们首先要关注的是avm(active virtual memory)列和free(free list size)列的值。如果avm 的值很大，而free的值却很小，这时，系统可能有内存瓶颈，我们必须用带-S 选项的 vmstat 命令查看系统是否有 deactivation/reactivation 活动正在发生。

判断瓶颈的方法以及调优方案：

当内存资源成为系统性能的瓶颈时，它有一些典型的症状：

很高的换页率(high pageout rate):HP-UX是一个按需调页的操作系统，通常情况下，它只执行调入页面进入内存的操作，以让进程能够运行。只有操作系统觉得系统需要释放一些内存空间时，才会执行从内存调出页面的操作，而过高的调出页面操作说明内存缺乏；

进程进入不活动状态(process deactivation activity):当自由的内存页面数量小于MINFREE时，很多进程将强制进入不活动状态

自由内存的数量很小，但活动的虚拟内存却很大(very small free memory and large active virtual memory)

交换区所有磁盘的活动次数可高(high disk activity on swap devices)

很高的全局系统CPU利用率(high global system CPU utilization):

很长的运行进程队列，但CPU的空闲时间却很多(large run queue with idle CPU)

内存不够出错(out of memory errors)

CPU用于vhand和swapper 两中守护进程的时间(CPU time to vhand and swapper)

必须注意的是，有时候我们发现CPU 很忙，这似乎是CPU 资源成为系统性能的瓶颈，但如果进一步分析，发现vhand和swapper 守护进程占用了大量的系统CPU时间，很显然，这时系统性能瓶颈真正所在可能是内存。

调优：

对内存需求密集型系统的性能调试

1) 基于硬件的方法：

增加物理内存

使用无盘工作站替代X-terminal

2) 基于软件的方法：

减小内核参数maxdsiz的值；

减少内存锁定的使用；

杀死不必要的进程；

识别出需要大量内存的进程；

重新设计应用；

减小内核的大小；

减小系统表的大小；

减小缓存区的大小；

组件 1 : Unix 平台服务器操作系统 (磁盘 I/O)					
序号	指标名称 (中英文)	指标理解	建议值	访问方式	瓶颈判断
1	%util %busy %disk time	一秒中设备有百分之多少的时间用于 I/O 操作, 或者说一秒中有多少时间 I/O 队列是非空的	50%	监控命令 : Iostat -x sar -d	如果某个磁盘的 %busy 列的值大于 50%, 则说明该磁盘可能存在瓶颈; 如果 %util 接近 100%, 说明产生的 I/O 请求太多, I/O 系统已经满负荷, 该磁盘存在瓶颈。
2	await(毫秒)+ svctm(毫秒) await+ avserv	平均每次设备 I/O 操作的等待时间 平均每次设备 I/O 操作的服务时间	await 接近 svctm await 接近 avserv	监控命令 : Iostat -x sar -d	await 远大于 svctm 或 avserv, 说明 I/O 队列太长, 应用响应时间变慢, 该

					磁 盘 可 能 存 在 瓶 颈
3	disk + b	b表示在等待io的进程数 disk 显示每秒的磁盘操作	b小于3	监控命令： vmstat	如果 disk 经常不 等于0，且 在 b 中的队 列大于 3，表示 io 性能 不好

使用命令：

iostat [-t] [interval [count]]

其选项的含义为：

-t:Report terminal statistics as well as disk statistics.

interval: Display successive lines which are summaries of the last interval seconds. The first line reported is for the time since a reboot and each subsequent line is for the last interval only.

count: Repeat the statistics count times.

命令sar -d，我们可以分析系统中的每个磁盘和磁带的活动情况。

每行结果数值如下：

device：设备名；

%busy: Portion of time device was busy servicing a request; statistics.

avque: Average number of requests outstanding for the device;

r+w/s: Number of data transfers per second (read and writes) from and to the device;

blks/s: Number of bytes transferred (in 512-byte units) from and to the device;

await: Average time (in milliseconds) that transfer requests waited idly on queue for the device;

avserv: Average time (in milliseconds) to service each transfer request (includes seek, rotational latency, and data transfer times) for the device.

判断瓶颈的方法以及调优方案：

当I/O 成为瓶颈时，会出现下面这些典型的症状：

过高的磁盘利用率(high disk utilization)

太长的磁盘等待队列(large disk queue length)

等待磁盘I/O的时间所占的百分率太高(large percentage of time waiting for disk I/O)

太高的物理I/O 速率:large physical I/O rate(not sufficient in itself)

过低的缓存命中率(low buffer cache hit ratio(not sufficient in itself))

太长的运行进程队列，但CPU却空闲(large run queue with idle CPU)

下面是一些占用大量I/O 资源的活动：

换页(paging):paging不仅会引起内存问题，还可能引起磁盘问题；

open,creat,and stat system calls：系统调用会引起大量的磁盘I/O；

multiuser I/O and random I/O

relational database

core dumps

必须注意的是，有时候我们发现CPU 很忙，这似乎是CPU 资源成为系统性能的瓶颈，但如果进一步分析，发现vhand和swapper 守护进程占用了大量的系统CPU时间，

很显然，这时系统性能瓶颈真正所在可能是内存。