
第四章 LoadRunner 场景监控

一、 场景监控的含义

- 整个监视过程由控制器来执行并在监控过程中收集相关数据。
- LoadRunner 更多的是监控每种服务器资源消耗的情况，对于各种服务器自身的某些特性需要使用第三方工具监控。
- 使用负载平衡技术时，负载机首先收集每个虚拟用户运行时的数据，再将这些数据发送到控制器，由控制器将数据保存在数据库中，最后由分析器来重新整理数据，画成不同的曲线图。

二、 监控系统资源

1 监控系统资源的含义

- 监控系统资源是为了分析是否由于系统资源引起了性能瓶颈，通常分析内存、硬盘和 CPU。
- 在 LoadRunner 中，通过添加计数器来监控系统资源。

2 重要计数器

2.1 内存计数器(Memory)

关注 Available Bytes、Pages/sec、Page Faults/sec。

2.1.1 Available Bytes

- 表示计算机上可用于运行处理的有效物理内存的字节数量。
- 是用零、空闲和备用内存表上的空间总值计算的。
 - ✓ 空闲内存指可以使用的内存。
 - ✓ 零内存指为了防止以后的处理看到以前处理使用的数据而在很多页内存中充满了零内存。
 - ✓ 备用内存是指从处理的工作集（它的物理内存）移到磁盘的，但是仍旧可以调用的内存。
- 这个计数器只显示上一次观察到的值，它不是一个平均值。如果该值低于 4MB 且 Pages/sec 持续地处于峰值状态下，那么说明让算机的物理内存不够。一般至少要有 10% 的剩余物理内存。

2.1.2 Page Faults/sec

- 表示每秒出错页的平均数量。

-
- 当处理器向内存指定的位置请求一页(可能是数据或代码)出错时, 就构成一个 Page Fault。
 - ✓ 如果该页在内存的其他位置, 称为软错误;
 - ✓ 如果该页必须从硬盘上重新读取, 则称为硬错误。
 - 许多处理器可以在有大量软错误的情况下继续操作。但是, 硬错误可以导致明显的拖延。

2.1.3 Pages/sec

- 表示为解决硬页错误从硬盘读取或写入磁盘的页数。
- 应该保持 0-20 或接近 0。
- Pages/sec 的值很大可能是内存问题, 应进一步研究内存页交换。

2.1.4 注意内存泄漏

- 一般以下两种情况表明出现内存泄漏的情况:
 - ✓ (1) 观察内存分配池, 如果内存分配池中可用内存消耗是呈不断上升的趋势, 说明可能出现内存泄漏的情况。
 - 建议 7*24 测试;
 - 观察 working set、handle count、private bytes 是否持续上升。
 - ✓ (2) 进程分配内存后, 但并未将用完的内存回收。

2.2 CPU 计数器(Processor)

- 一个进程对应于一个程序, 一个程序可以同时对应于多个进程。
- 当处理器不能即时处理进程时, 就会出现进程排队的现象, 如果出现持续排队的现象就说明 CPU 当前处于繁忙状态, 所以分析 CPU 是否处于繁忙状态的第一个指标是 CPU 的队列长度。
- 一般关注 ProcessorTime、Processor Queue Length、%UserTime。



2.2.1 %Processor Time

- 表示被消耗的处理器时间百分比。
- 此值持续高于 80%, 说明 CPU 存在压力, 需要继续分解以确定用户模式进程(User Time)还是内核模式进程(Privileged Time)消耗的时间更多。

2.2.2 %UserTime

- 若%UserTime 占比很大，说明应用程序出了问题，接下来需要确定是哪个进程消耗了 CPU 的时间。

2.2.3 Processor Queue Length

- 表示队列长度，一般不大于 2。
- 若 Processor Queue Length 显示的队列长度保持不变(≥ 2)，且处理器的利用率 Processor Time 超过 90%，则很可能存在处理器瓶颈。
- 若发现 Processor queue length 显示的队列长度超过 2，但处理器的利用率却一直很低，则应该去解决处理器阻塞问题，处理器一般不是瓶颈。

2.3 磁盘计数器

- 为了确定磁盘读写时间是否遇到瓶颈，需要先获得物理磁盘读写一次所花费的时间，即磁盘服务时间，磁盘服务时间由 3 部分组成：
 - ✓ 寻道时间
 - ✧ 是指将读写磁头移动到正确的磁道上所需要的时间，寻道时间越短，I/O 操作越快，目前磁盘的平均寻道时间为 3-15ms。
 - ✓ 旋转延迟时间
 - ✧ 是指盘片旋转将请求数据所在扇区移至读写磁头下方所需要的时间，旋转延迟取决于磁盘转速，通常为磁盘旋转一周所需要时间的 1/2。现在的磁盘转速一般为 5400r/min、7200r/min、10Kr/min 和 15Kr/min。以 7200r/min 为例，可以转 120r/s，那么磁盘转一周的时间为 1/120s，大约为 8.33ms，旋转延迟时间为旋转一周所需时间的 1/2，即大概为 4.17ms。
 - ✓ 数据传输时间
 - ✧ 是指完成传输请求的数据所需要的时间，它取决于数据传输率，该值等于数据大小除以数据传输率，目前 IDE/SATA 的传输率达到 133MB/s，SATAII 可达到 300MB/s，所以数据传输的时间很短，相对于寻道时间和旋转延迟时间，数据传输时间可以忽略不计。
- 以 7200r/min 为例，读写一次平均时间为 $3\text{ms} + 4.17\text{ms} = 7.17\text{ms}$ ，如果监控到的平均读写时间比计算出来的平均读写时间值大，则说明磁盘可能出现瓶颈。
- 一般关注 Disk Time、Average Disk Queue Length、Average Disk Seconds/Read、Average Disk Seconds/Write。



2.3.1 %Disk Time

- 表示所选磁盘忙于为读或写入请求提供服务所用时间的百分比。正常值小于 10，此值过大表示耗费太多时间来访问磁盘，可考虑增加内存、更换更快的硬盘、优化读写数据的算法等。

2.3.2 Average Disk Queue Length

- 表示读写请求的平均队列数，正常值小于 0.5，此值过大表示磁盘 I/O 太慢，需更换更快硬盘。
 - ✧ 如驾车等信号灯，如果当前信号灯有很多车在等，那么排在后面的车就必须等待下一个信号灯才有可能通过。

2.3.3 Average Disk Seconds/Read

- 表示读取数据所需的平均时间。

2.3.4 Average Disk Seconds/Write

- 表示写入数据所需的平均时间。

3 监控 Windows

- 被监视计算机设置密码
- 被监视计算机开启以下服务(services.msc)
 - ✓ Remote Registry
 - ✓ Remote Procedure Call (RPC)
- 关闭防火墙

4 监控 Linux

- Linux 设置 ip、关闭防火墙
 - ✓ eth0 变成 eth1 后出错的解决方案
 - ✧ 当服务器更换完主板或者网卡时经常出现网卡乱序的问题。

-
- vi /etc/sysconfig/network-scripts/ifcfg-eth0, 注释 HWADDR 一行, 或者删除此行
 - rm -fr /etc/udev/rules.d/70-persistent-net.rules
 - 重启 Linux
 - ✧ 只能临时解决, 虚拟机粘贴到其他目录时, 仍会出错
 - 按顺序安装 gcc-c++、rsh、rsh-server
 - ✓ 通过 CentOS 系统光盘安装即可
 - 安装 rstat (在 LR 光盘中)
 - ✓ tar 解压 (tar -zvxf rstatd.tar.gz -C ~) 后进入 rstatd→./configure→make→make install
 - 直接输入 rpc.rstatd 启动该服务
 - 进入/etc/xinetd.d 目录
 - ✓ 分别进入文件 rsh、rexec、rlogin, 将 disabled=yes 改为 no
 - 执行 service xinetd restart 重启 rsh、rexec、rlogin 服务

5 监控 SQL Server

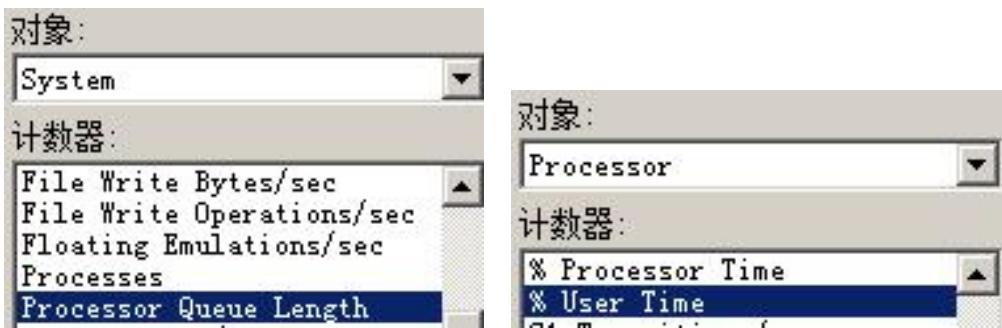
5.1 监控 CPU

5.1.1 System-Processor Queue Length (处理器的队列长度)

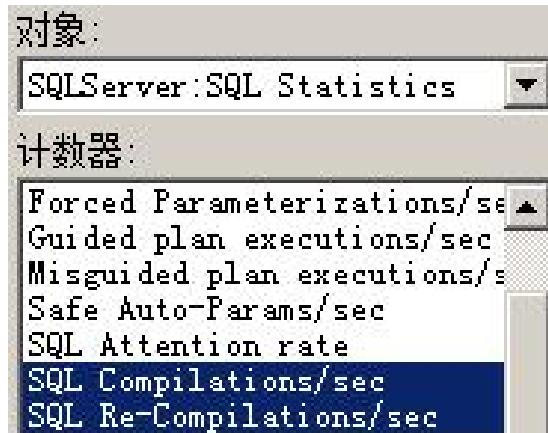
- 正常情况下服务器不忙碌的时候不可能出现处理器排队的现象, 当处理器出现排队现象时, 则说服务器的性能受到影响, 一般情况下处理器的队列长度不超过 CPU 个数。

5.1.2 Processor-%UserTime

- 显示 SQL Server 进程所消耗的 CPU 时间数量。
- 如果发现 CPU 的使用率过高, 通常是高于 85%, 那么很有可能是 CPU 出现瓶颈, 接下来需要进一步分析是什么原因导致 CPU 的使用率过高。



5.1.3 SQL Server:SQL Statistics-SQLCompilations/sec 与 SQLRe-Compilations/sec



- 这两计数器表示执行计划编译与重编译率；
- 对于一台服务器，执行计划的重用率至少应该在 90% 以上，即在执行过程中，最多只有 10% 的查询计划需要重新进行编译，如果当执行过程中需要重新编译的查询计划过多，那么将导致消耗过高的 CPU 资源。

5.2 监控内存

5.2.1 SQL Server:Buffer Manager-Total Pages、Target Pages 与 stolen pages



- Total Pages 表示 SQL Server 已获得的页面总数。
- Target Pages 表示 SQL Server Buffer Pool 所需要的理想页面数。
 - ✓ 如果 Target Pages 与 Total Pages 的值相同，则表示 SQL Server 拥有足够的内存，如果 Total Pages 值小于 TargetPages 则说明存在外部内存压力，SQL Server 不能获得足够的内存数量。
- stolen pages 表示内存中被其他进程挪用的页面，如果该值较高，那么说明服务器正在经受内存压力。

5.2.2 SQL Server:Buffer Manager-Page Life Expectancy 与 Buffer cache hit ratio

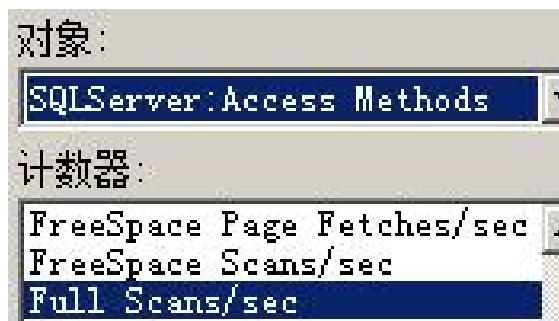


- Page Life Expectancy 表示数据页在缓冲池中驻留的时间长度，单位为 s。
 - ✧ 值越大，说明服务器的性能越好，如果值小于 300s，说明服务器存在内存压力，那么很有可能是因为缺少内存引起的。
- Buffer cache hit ratio 表示命中率，显示待查询请求页面在 SQLServer 缓冲池(即物理内存中) 中被找到的数目。
 - ✧ 值越大越好，一般应>98%，否则表示服务器可能没有足够的空闲内存。

5.3 监控硬盘

- 在执行查询计划过程中，最多的是对磁盘进行写入与读出数据的操作，所以 SQL Server 服务器应该尽可能地避免频繁地在磁盘与内存间进行传递数据，以降低对服务器性能的影响，为了解决这个问题，SQLServer 使用缓冲高速缓存 (Buffercache) 和计划高速缓存 (Plancache)。
 - ✓ Buffer cache 用于预载数据；
 - ✓ Plan cache 用于加载检索数据的方法是否是最优计划。

5.3.1 SQL Server:Access Methods-Full scans/sec



- 该计数器表示每秒钟完全扫描索引或完全扫描基本表的数目，在数据库设计过程中应该尽量降低全表扫描的次数，特别是对于那些大表，如果进行全表扫描将会直接导致性能下降，如果扫描频率大于一次，则说明缺少索引或索引较差。

5.3.2 SQLServer:Access Methods-Page Splits/sec



- 该计数器表示每秒钟页面拆分数量，在执行插入或更新计划时，如果当前的数据页没有足够的空间来完成这些操作，那么就必须增加新页来完成插入或更新操作，过多的拆分页会损害服务器的性能。

5.4 监控 SQL Server 的操作

- 被监视计算机设置密码
- 被监视计算机开启以下服务(services.msc)
 - ✓ Remote Registry
 - ✓ Remote Procedure Call (RPC)
- 关闭防火墙

6 监控 Apache

6.1 修改配置文件

- 需要首先确保已经安装并启动了Apache。
- 监控Apache的状态需要启用如下模块
 - ✓ LoadModule status_module modules/mod_status.so
- 配置Apache Server Status

```
<location /server-status>
    SetHandler server-status
</location>
```

 - ✓ SetHandler server-status 不能省略，服务器的状态信息通常是在 server-status 中的 handler 调用。
- 显示所有的状态信息，需设置
 - ✓ ExtendedStatus On
- 重启Apache

6.2 查看 Apache 服务器状态

http://localhost/server-status

http://localhost/server-status?auto

- ✓ 显示访问状态下的动态信息（动态监控）

6.3 添加监控

- 报错信息：【Monitor name :Apache. Parsing error, cannot find token: BusyServers. Measurement: BusyServers|192.168.0.186. Hints: 1) Such a measurement does not exist, or the html page may be different from the supported one. 2) Try to replace the Apache.cfg with appropriate Apache_<version>.cfg file in <Installation>\dat\monitors and rerun the application (entry point: CApacheMeasurement::NewData).
 - ✓ 这是由于要监视Apache的版本提供的计数器与LoadRunner默认的计数器不一致导致的。
 - ✓ 先关闭Controller，打开<Installation>HP\LoadRunner\dat\monitors下的apache.cfg文件
 - ❖ 修改Counter0=IdleServers为Counter0=IdleWorkers，同时修改注释信息 Label0=#Idle Servers (Apache) 为 Label0=#Idle Workers (Apache)，描述信息也建议修改；
 - ❖ 修改Counter4=BusyServers为Counter4=BusyWorkers，同时修改注释信息 Label4=#Busy Servers (Apache) 为 Label4=#Busy Workers (Apache)，描述信息也建议修改。
 - ✓ 保存并关闭该文件，重新打开Controller并添加计数器。

6.4 计数器

- Apache CPU Usage
 - ✓ 服务器CPU的占用率
- Kbytes Sent/sec
 - ✓ 服务器每秒发送的字节数
- Hits/sec
 - ✓ Apache服务每秒的点击率
- Busy Workers
 - ✓ Apache服务占用率（正在工作数）

-
- Idle Workers
 - ✓ Apache服务空闲率（空闲工作数）

7 监控 MySQL

7.1 安装并配置 SiteScope

- 注意
 - ✓ 需要使用 64 位操作系统。
 - ✓ SiteScope 和 LoadRunner 安装到同一台计算机。
- 安装 MySQL
 - ✓ MySQL 在被测服务器上。
 - ✓ 与 Lr、SiteScope、JRE 可以不在同一计算机。
- 安装 JRE
- 安装 SiteScope

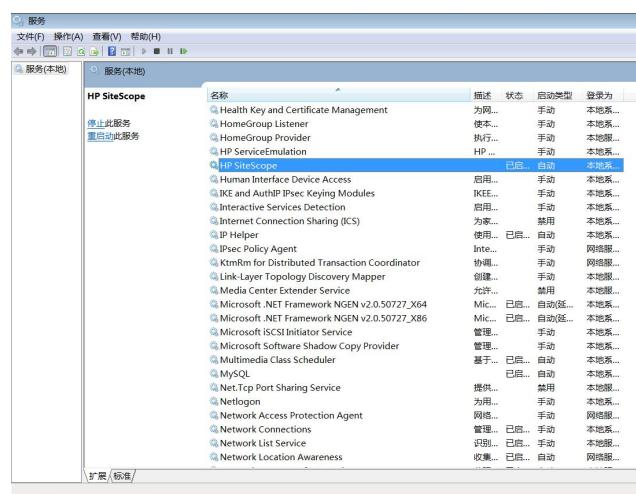


- 访问 SiteScope
 - ✓ <http://localhost:8080>
 - ✓ IE 浏览器建议使用 32 位
 - ✓ 在弹窗中选择“稍后”，不要阻止



➤ 复制 mysql 驱动

- ✓ 将 mysql 驱动程序拷贝放在 sitescope 安装目录下的 java/lib/ext 下
- ✓ 重启 SiteScope 服务



➤ SiteScope 设置

- ✓ 新建一个组
 - ↳ 选择 SiteScope, 点击鼠标右键, 点击新建, 点击组, 输入组的名称点击保存
- ✓ 新建一个监控器
 - ↳ 选择刚刚新建的组, 点击鼠标右键, 点击新建, 点击监控器进入选择监控器页面
- ✓ 点击数据库计数器

◆ 进入计数器配置页面，在数据库配置页面填写好内容

数据库计数器监控器设置

* 数据库连接 URL:	jdbc:mysql://localhost/mysql
* 查询:	show status
* 数据库驱动程序:	com.mysql.jdbc.Driver
数据库计算机名称:	
数据库端口:	
数据库实例名称:	
除数查询:	
凭据:	<input checked="" type="radio"/> 使用用户名和密码
	用户名: root
	密码: *****
	<input type="radio"/> 选择预定义的凭据
	凭据配置文件: <input type="button" value="添加凭据"/>
	<input checked="" type="checkbox"/> 无累积计数器
	<input checked="" type="checkbox"/> 无除法计数器
计数器:	<input type="button" value="获取计数器"/>

◆ 数据库连接 URL

- jdbc:mysql://IP:端口号/数据库名称

■ 默认端口: 3306

◆ 点击“获取计数器”

◆ 选择计数器后，保存

7.2 mysql 常用计数器

计数器	计数器分析
Aborted_clients	客户端被异常中断的数值，异常中断连接可能表明一些应用程序有问题
Opened_tables	指表缓存没有命中的数量，如果该值很大，就需要增加 table_cache 的数值
Questions	每秒获得的查询数量，也可以是全部查询的数量
Select_full_join	联合查询的执行数量，该值可能是零，这样的查询有可能降低系统的性能
Select_scan	执行全表搜索查询的数量，如果发现该值持续增长，说明需要优化，缺乏必要的索引或其他问题
Slow_queries	超长查询的数量，或没有使用索引查询数量。如果该值增长，表明系统有性能问题
Threads_connected	表示当前有多少个客户连接该 mysql 服务器，连接数是否过多，网络是否存在问题是动态变化的。当达到最大连接数时，数据库系统就不能提供更多的连接数了，这时，程序还想新建连接

	线程，数据库系统就会拒绝，如果程序没做太多的错误处理，就会出现报错信息
Threads_created	该值一般较低。较高的值意味着可能遇到了持续增加的连接，表明存在潜在的问题
Threads_running	如果数据库超负荷，将会得到一个正在增长的数值。若这个值超过预设值时且 5 秒内没有回落，要同时监视其他的一些值

7.3 LoadRunner 监控 MySQL

- 打开 LoadRunner，在 Controller 的 run 页面添加 sitescope，即可在运行中监控到 mysql 数据库的信息