**性能压力测试**

## 一、概述

**1 系统吞吐量指单位时间内系统处理的请求数量**，体现系统的整体处理能力。对于网站，可以用"请求数/秒"或是"页面数/秒"来衡量，也可以用"访问人数/天"或是"处理的业务数/小时"等来衡量。TPS（每秒事务数）是吞吐量的一个常用量化指标，此外还有HPS（每秒HTTP请求数）、QPS（每秒查询数）等。

**（1）并发数**

指系统能够同时处理请求的数目，这个数字也反映了系统的负载特性。

对于网站而言，并发数即网站并发用户数，指同时提交请求的用户数目。

**（2）QPS和TPS**

TPS（transaction per second），每秒钟系统能够处理的交易或者事务的数量，它是衡量系统处理能力的重要指标。

QPS（TPS）= 并发数/平均响应时间，或者，

并发数 = QPS\*平均响应时间

一个系统吞吐量通常由QPS（TPS）、并发数两个因素决定，

每套系统这两个值都有一个相对极限值，在应用场景访问压力下，只要某一项达到系统最高值，系统的吞吐量就上不去了，

**如果压力继续增大，系统的吞吐量反而会下降，原因是系统超负荷工作，上下文切换、内存等等其它消耗导致系统性能下降。**

**（3）用户平均请求等待时间**

请求响应时间指的是从客户端发起的一个请求开始，到客户端接收到从服务器端返回的响应结束，这个过程所耗费的时间，在某些工具中，响应通常会称为“TTLB”，即"time to last byte"，意思是从发起一个请求开始，到客户端接收到最后一个字节的响应所耗费的时间，响应时间的单位一般为“秒”或者“毫秒”。

**一个公式可以表示：响应时间＝网络响应时间+应用程序响应时间。**

**（4）响应时间（Response time）**

**2-5-8原则：**（据统计当网站慢一秒就会流失十分之一的客户）

当用户再2-5秒之间得到响应时，会感觉系统的响应速度还可以；

当用户再5-8秒内得到响应时，感觉慢，但是还可以接受；

当用户大于8秒内得到响应时，感觉无法接受；

**（5）吞吐量（Throughput）**

指的是在单位时间内客户端和服务器成功传送数据的数量；

**（6）资源使用率（Resource utilization）**

常见的有：cpu占用率、内存使用率、磁盘I/O、网络I/O；

**（7）每秒点击数（Hits per Second）**

客户端每秒向服务器提交的请求数量，如果客户端发出的请求数量越多，与之相对的平均吞吐量也应该越大；

**（8）并发用户数（Concurrent users）**

客户端的同一批用户同时执行一个操作的数量。

**2 [Web](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)性能[测试](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)的部分概况一般来说，一个Web请求的处理包括以下步骤：**

（1）客户发送请求

（2）web [server](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)接受到请求，进行处理；

（3）web server向DB获取数据；

（4）webserver生成用户的object(页面)，返回给用户。给客户发送请求开始到最后一个字节的时间称为响应时间（第三步不包括在每次请求处理中）。

## 二 [性能测试](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)指标

**1 事务（Transaction）**

在web[性能测试](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)中，一个事务表示一个“从用户发送请求->web server接受到请求，进行处理-> web server向DB获取数据->生成用户的object(页面)，返回给用户”的过程，一般的响应时间都是针对事务而言的。

**2 请求响应时间**

请求响应时间指的是从客户端发起的一个请求开始，到客户端接收到从服务器端返回的响应结束，这个过程所耗费的时间，在某些工具中，响应通常会称为“TTLB”，即"time to last byte"，意思是从发起一个请求开始，到客户端接收到最后一个字节的响应所耗费的时间，响应时间的单位一般为“秒”或者“毫秒”。一个公式可以表示：响应时间＝网络响应时间+应用程序响应时间。标准可参考国外的3/5/10原则：

（1）在3秒钟之内，页面给予用户响应并有所显示，可认为是“很不错的”；

（2）在3~5秒钟内，页面给予用户响应并有所显示，可认为是“好的”；

（3）在5~10秒钟内，页面给予用户响应并有所显示，可认为是“勉强接受的”；

（4）超过10秒就让人有点不耐烦了，用户很可能不会继续等待下去；

**3 事务响应时间**

  事务可能由一系列请求组成,事务的响应时间主要是针对用户而言,属于宏观上的概念，是为了向用户说明业务响应时间而提出的.例如:跨行取款事务的响应时间就是由一系列的请求组成的.事务响应时间是直接衡量系统性能的参数.

**4 并发用户数**

并发一般分为2种情况。

**一种是严格意义上的并发，即所有的用户在同一时刻做同一件 事情或者操作**，这种操作一般指做同一类型的业务。比如在信用卡审批业务中，一定数目的拥护在同一时刻对已经完成的审批业务进行提交；还有一种特例，即所有 用户进行完全一样的操作，例如在信用卡审批业务中，所有的用户可以一起申请业务，或者修改同一条记录。

　　另外一种并发是广义范围的并发。这种并发与前一种并发的区别是，尽管多个用户对系统发出了请求或者进行了操作，**但是这些请求或者操作可以是相同的，也可以是不同的。**对整个系统而言，仍然是有很多用户同时对系统进行操作，因此也属于并发的范畴。

　　可以看出，后一种并发是包含前一种并发的。而且后一种并发更接近用户的实际使用情况，因此对于大多数的系统，只有数量很少的用户进行“严 格意义上的并发”。对于WEB性能测试而言，这2种并发情况一般都需要进行测试，通常做法是先进行严格意义上的并发测试。严格意义上的用户并发一般发生在 使用比较频繁的模块中，尽管发生的概率不是很大，但是一旦发生性能问题，后果很可能是致命的。严格意义上的并发测试往往和[功能测试](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)关联起来，因为并发功能遇到异常通常都是程序问题，这种测试也是健壮性和稳定性测试的一部分。用户并发数量：关于用户并发的数量，有2种常见的错误观点。 一种错误观点是把并发用户数量理解为使用系统的全部用户的数量，理由是这些用户可能同时使用系统；还有一种比较接近正确的观点是把在线用户数量理解为并发用户数量。实际上在线用户也不一定会和[其他](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)用户发生并发，例如正在浏览网页的用户，对[服务器](http://server.chinaitlab.com/" \t "https://www.cnblogs.com/flyr/p/_blank)没有任何影响，但是，在线用户数量是计算并发用户数量的主要依据之一。

**5 吞吐量**

指的是在一次性能测试过程中网络上传输的数据量的总和.吞吐量/传输时间,就是吞吐率.

**6 TPS（transactionper second）**

每秒钟系统能够处理的交易或者事务的数量.它是衡量系统处理能力的重要指标.

**7 点击率**

每秒钟用户向WEB[服务器](http://server.chinaitlab.com/" \t "https://www.cnblogs.com/flyr/p/_blank)提 交的HTTP请求数.这个指标是WEB应用特有的一个指标:WEB应用是"请求-响应"模式,用户发出一次申请,服务器就要处理一次,所以点击是WEB应 用能够处理的交易的最小单位.如果把每次点击定义为一个交易,点击率和TPS就是一个概念.容易看出,点击率越大,对服务器的压力越大.点击率只是一个性 能参考指标,重要的是分析点击时产生的影响。需要注意的是,这里的点击并非指鼠标的一次单击操作,因为在一次单击操作中,客户端可能向服务器发出多个 HTTP请求.

**8 资源利用率**

指的是对不同的系统资源的使用程度,例如服务器的CPU利用率,磁盘利用率等.资源利用率是分析系统性能指标进而改善性能的主要依据,因此是WEB性能测试[工作](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)的重点.

资源利用率主要针对WEB服务器,[操作系统](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank),[数据库](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)服务器,网络等,是测试和分析瓶颈的主要参考.在WEB性能测试中,更根据需要采集相应的参数进行分析。

**9 通用指标（指[Web](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)应用服务器、[数据库](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)服务器必需[测试](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)项)**

|  |  |
| --- | --- |
| 指标 | 说明 |
| ProcessorTime | 服务器CPU占用率，一般平均达到70%时，服务就接近饱和 |
| Memory Available Mbyte | 可用内存数，如果测试时发现内存有变化情况也要注意，如果是内存泄露则比较严重 |
| Physicsdisk Time | 物理磁盘读写时间情况 |

**10 Web服务器指标**

|  |  |
| --- | --- |
| 指标 | 说明 |
| Requests Per Second（Avg Rps） | 平均每秒钟响应次数＝总请求时间 / 秒数 |
| Avg time to last byte per terstion （mstes） | 平均每秒业务脚本的迭代次数 ,有人会把上面那个混淆 |
| Successful Rounds | 成功的请求 |
| Failed Requests | 失败的请求 |
| Successful Hits | 成功的点击次数 |
| Failed Hits | 失败的点击次数 |
| Hits Per Second | 每秒点击次数 |
| Successful Hits Per Second | 每秒成功的点击次数 |
| Failed Hits Per Second | 每秒失败的点击次数 |
| Attempted Connections | 尝试链接数 |

**11 [数据库](http://lib.csdn.net/base/14" \o "MySQL知识库" \t "https://www.cnblogs.com/flyr/p/_blank)服务器性能指标**

|  |  |
| --- | --- |
| 指标 | 说明 |
| User 0 Connections | 用户连接数，也就是数据库的连接数量 |
| Number of deadlocks | 数据库死锁 |
| Butter Cache hit | 数据库Cache的命中情况 |

**12 系统的瓶颈定义**

|  |  |  |
| --- | --- | --- |
| 性能项 | 命令 | 指标 |
| CPU限制 | vmstat | 当%user+%sys超过80%时 |
| 磁盘I/O限制 | Vmstat | 当%iowait超过40%(AIX4.3.3或更高版本)时 |
| 应用磁盘限制 | Iostat | 当%tm\_act超过70%时 |
| 虚存空间少 | Lsps，-a | 当分页空间的活动率超过70%时 |
| 换页限制 | Iostat, stat | 虚存逻辑卷%tm\_act超过I/O(iostat)的30%，激活的虚存率超过CPU数量(vmstat)的10倍时 |
| 系统失效 | Vmstat, sar | 页交换增大、CPU等待并运行队列 |

**13 稳定系统的资源状态**

|  |  |  |
| --- | --- | --- |
| 性能项 | 资源 | 评价 |
| CPU占用率 | 70% | 好 |
| 85% | 坏 |
| 90%+ | 很差 |
| 磁盘I/0 | <30% | 好 |
| <40% | 坏 |
| <50%+ | 很差 |
| 网络 | <30%带宽 | 好 |
| 运行队列 | <2\*CPU数量 | 好 |
| 内存 | 没有页交换 | 好 |
| 每个CPU每秒10个页交换 | 坏 |
| 更多的页交换 | 很差 |

## 三、举例说明

　　日访问量

　　常用页面最大并发数

　　同时在线人数

　　访问相应时间

1 案例：

　　最近公司一个项目，是个门户网站，需要做[性能测试](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)，根据项目特点定出了主要测试项和测试方案：

　　一种是测试几个常用页面能接受的最大并发数(用户名参数化，设置集合点策略)

　　一种是测试服务器长时间压力下，用户能否正常操作(用户名参数化，迭代运行脚本)

一种则需要测试服务器能否接受10万用户同时在线操作，如果是用IIS做应用服务器的话，单台可承受的最大并发数不可能达到10万级，那就必须 要使用集群，通过多台机器做负载均衡来实现；如果是用websphere之类的应用服务器的话，单台可承受的最大并发数可以达到10万级，但为性能考虑还 是必须要使用集群，通过多台机器做负载均衡来实现；通常有1个简单的计算方式，1个连接产生1个session，每个session在服务器上有个内存空 间大小的设置，在NT上是3M，那么10万并发就需要300G内存，当然实际使用中考虑[其他](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)程序也占用内存，所以准备的内存数量要求比这个还要多一些。还有10万个用户同时在线，跟10万个并发数是完全不同的2个概念。这个楼上已经说了。但如何做这个转换将10万个同时在线用户转换成多少个并发数呢？这就必须要有大量的历史[日志](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)信息来支撑了。系统日志需要有同时在线用户数量的日志信息，还需要有用户操作次数的日志信息，这2个数据的比例就是你同时在线用户转换到并发数的比例。另外根据经验统计，对于1个[JAVA](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)开发的[WEB](http://www.51testing.com/javascr%C4%ABpt:;" \t "https://www.cnblogs.com/flyr/p/_blank)系统（别的我没统计过，给不出数据），一般1台双CPU、2G内存的服务器上可支持的最大并发数不超过500个（这个状态下大部分操作都是超时报错而且服务器很容易宕机，其实没什么实际意义），可正常使用（单步非[大数据](http://lib.csdn.net/base/20" \o "Hadoop知识库" \t "https://www.cnblogs.com/flyr/p/_blank)量 操作等待时间不超过20秒）的最大并发数不超过300个。假设你的10万同时在线用户转换的并发数是9000个，那么你最少需要这样的机器18台，建议不 少于30台。当然，你要是买个大型服务器，里面装有200个CPU、256G的内存，千兆光纤带宽，就算是10万个并发用户，那速度，也绝对是嗖嗖的。

2 另外暴寒1下，光设置全部进入运行状态就需要接近6个小时。具体的可以拿1个系统来压一下看看，可能会出现以下情况：

　　1、服务器宕机；

　　2、客户端宕机；

　　3、从某个时间开始服务器拒绝请求，客户端上显示的全是错误；

　　4、勉强测试完成，但网络堵塞或测试结果显示时间非常长。假设客户端和服务器之间百兆带宽，百兆/10000=10K，那每个用户只能得到10K，这个速度接近1个64K的MODEM上网的速度；另外以上分析全都没考虑系统的后台，比如[数据库](http://www.51testing.com/javascr%C4%ABpt:;" \t "https://www.cnblogs.com/flyr/p/_blank)、中间件等。

　　1、服务器方面：上面说的那样的PC SERVER需要50台；

　　2、网络方面：按每个用户50K，那至少5根百兆带宽独享，估计仅仅网络延迟就大概是秒一级的；

　　3、如果有数据库，至少是ORACLE，最好是SYSBASE，[SQL](http://blog.csdn.net/qiguiting/article/details/11584397" \t "https://www.cnblogs.com/flyr/p/_blank)SERVER是肯定顶不住的。数据库服务器至少需要10台4CPU、16G内存的机器；

4、如果有CORBA，那至少再准备10台4CPU、16G内存的机器；再加上负载均衡、防火墙、路由器和各种软件等，总之没个1000万的资金投入，肯定搞不定。

3 这样的门户系统，由于有用户权限，所以并不象jackie所说大多是静态页面。但只要是多服务器的集群，那么我们就可以通过1台机器的测试结果 来计算多台机器集群后的负载能力的，最多额外考虑一下负载均衡和路由上的压力，比如带宽、速度、延迟等。但如果都是在1台机器上变化，那我们只能做一些指 标上的计算，可以从这些指标上简单判断一下是否不可行，比如10万并发用户却只有1根百兆带宽，那我们可以计算出每个用户只有1K带宽，这显然是不可行 的。但实际的结果还是需要测试了才知道，毕竟系统压力和用户数量不是线性变化的。

　　这一类系统的普遍的成熟的使用，以及很多软件在方案设计后就能够大致估算出系统的性能特点，都导致了系统在软件性能方面调优的比例并不大（当然 不完全排除后期针对某些代码和配置进行优化后性能的进一步提高），更多的都是从硬件方面来考虑，比如增加内存、硬盘做RAID、增加带宽、甚至增加机器 等。

　　网络技术中的10M 带宽指的是以位计算， 就是 10M bit /秒 ，而下载时的速度看到的是以字节（Byte）计算的，所以10M带宽换算成字节理论上最快下载速度为： 1.25 M Byte/秒!