**性能测试规范**

**目录**

[1 概述 3](#_Toc32400)

[1.1 编写目的 3](#_Toc14635)

[1.2 适用范围 3](#_Toc12855)

[2 性能测试指标 3](#_Toc16816)

[2.1 响应时间 3](#_Toc987)

[2.1.1 定义 3](#_Toc2626)

[2.1.2 测试方法 4](#_Toc29766)

[2.1.3 分析评估 5](#_Toc21188)

[2.2 TPS（QPS）、并发用户数 7](#_Toc25883)

[2.2.1 定义 7](#_Toc23120)

[2.2.2 测试方法 7](#_Toc5361)

[2.2.3 分析评估 8](#_Toc31104)

[2.3 请求成功率 9](#_Toc9451)

[2.3.1 定义 9](#_Toc5139)

[2.3.2 测试方法 9](#_Toc15440)

[2.3.3 分析评估 9](#_Toc17585)

[2.4 CPU使用率、内存使用率、IO WAIT 9](#_Toc16822)

[2.4.1 定义 9](#_Toc31706)

[2.4.2 测试方法 10](#_Toc7872)

[2.4.3 分析评估 11](#_Toc21319)

[2.5 GC 11](#_Toc27258)

[2.6 进程级别的资源占用 11](#_Toc8509)

# 概述

## 编写目的

本文档在对性能指标的概念、测试及分析方法、评判标准以及工具的使用进行说明，旨在指导性能测试工程师更好的理解各个性能指标，并对系统的性能质量做出准确的评价和分析。

## 适用范围

本规范适用范围：性能测试、性能调优和性能验收活动。

# 性能测试指标

## 响应时间

### 定义

响应时间通常是指客户发出请求到得到响应的整个过程所耗费的时间，通常被定义TTLB（Time to Laster Byte），代表从发起一个请求开始，到客户端收到响应的最后一个字节所耗费的时间。

响应时间根据所耗费的时间段可以做细致的拆解，我们可以把它拆解为三部分，系统处理时间、数据传输时间、呈现时间（Web页面特有，接口类请求无呈现时间），每个部分的时间消耗影响的因素有所不同。

呈现时间：主要是浏览器对接收到的数据渲染展示的过程，呈现时间不止于浏览器有关，和操作系统、电脑的硬件配置也有关系。

数据传输时间：请求、响应数据在网络中传输消耗的时间，和网络的时延、带宽有关系。

系统处理时间：系统接收到请求后，对请求处理，并将结果返回的时间，和系统服务器的软硬件配置有关系。

### 测试方法

#### 测试前提

1. 前提一：性能测试中响应时间的测试，需要保持一个稳定的网络环境。

不建议在办公网络中搭建“施压设备”，不稳定的办公网络环境会影响对测试结果的评判。建议在以下两种环境下测试：

①施压设备与被测系统在同一局域网中，更能够排除网络情况对响应时间的影响，能够更准确的衡量“系统处理时间”。

②施压设备和被测系统在不同的机房环境中通过公网测试，这种场景更能准确的模拟并评估系统在生产环境中的表现。

测试工程师可以根据测试的目的，选择后两种环境进行测试。

1. 前提二：确定一定的并发量来测试响应时间

最优并发用户场景、最高并发用户场景两种场景测试，响应时间的表现是不同的，最高并发场景的响应时间将会比最优并发的响应时间大得多，测试前我们需要确定我们测试的场景是最优并发还是最高并发。

#### 测试步骤

1. 找到最高的吞吐量(TPS)。

* 测试前确定一个响应时间的标准（如：小于100ms），然后进行基准测试，通过虚拟并发用户数为1的方式测试，记录测试的TPS、响应时间测试结果，将该响应时间与标准比较，若大于标准响应时间，那么则说明系统有问题无法满足标准，若该响应时间小于标准时间，则继续下面的测试。
* 通过压力测试找到最大的吞吐量：在基准测试响应时间的限制下，找到系统最大的吞吐量（TPS），该状况下响应时间满足要求、吞吐量最大，可确定为“最佳并发用户数”。方法是按照一定的步长，不断增加虚拟并发用户数，直至响应时间超过限制、吞吐量不在增长、任意节点资源使用率超过要求（如：70%）。

1. 负载测试：保持最大吞吐量，执行负载测试，持续30分钟，记录测试TPS、响应时间测试结果。
2. 稳定性测试：保持最大吞吐量，执行稳定性测试，持续3\*24小时，记录测试TPS、响应时间。

#### 测试对象的分类

1. 接口

接口类响应时间只包含数据传输时间、系统处理时间，不包含呈现时间，Apache Jmeter支持该类响应时间的统计，共有min、max、avg三种统计结果，分别代表最小、最大、平均值，其他的性能测试工具均有对接口类响应时间的精确统计。

1. Web页面

有3种方法可以统计Web页面的响应时间：

①浏览器抓包工具统计页面响应时间

②录屏软件抓取屏幕计算响应时间

③JS打点统计页面响应时间。

注意：目前还无法通过大量并发访问的采样统计页面的响应时间，在通过浏览器测试Web页面响应时间时，要确保通过Jmeter对系统相对应接口保持一定压力的并发用户访问（通常在最优并发下测试）。

### 分析评估

#### Web页面响应时间分析遵循258原则

在互联网上对于用户响应时间有一个普遍的标准（2/5/8原则），一般认为响应时间超过5s是系统是需要优化，如果超过8s是不可接受的。

* 2s之内响应被认为非常有吸引力的用户体验。
* 5s之内响应被认为比较不错的用户体验。
* 8s之内响应被认为非常糟糕的用户体验。
* 超过8s没有响应，用户通常认为请求失败。

需要特殊说明的一点，对于用户来说，响应时间是否被接受带有一定的主观色彩，例如一个系统报表的功能，每个月才会有用户使用一次，那么每次花费1个小时，用户是可以接受的，但是一个常用的登录按钮提交1分钟后才返回登录成功，我们也难以接受。因此响应时间的“长”和“短”并没有绝对的定义，合理的响应时间取决于用户实际需求，而不能依据测试人员的设想或者标准的硬性规定。

#### Web页面响应时间分析评估时需要考虑有无浏览器缓存的两种情况

Web页面响应时间测试，要分为浏览器有缓存和无缓存的两种情况（无缓存的情况由于资源的下载响应时间会稍长），一般通过有浏览器缓存的场景的结果表现来评估响应时间对用户体验的影响。

#### 接口类响应时间，参考系统需求规格定义评估

最优并发情况下，性能测试结果平均响应时间不得高于系统需求规格定义。

建议：需求规格的定义，单接口响应时间应小于100ms。

响应时间的标准一般定义：99.9%响应时间必须在100ms以下（非平均值，99.9%取样响应时间均在100ms以下）或者平均响应时间在100ms以下，目前工具只能统计平均响应时间指标。

#### 响应时间与历史版本比较

当前系统实测响应时间的指标不得高于历史版本的实测结果。

注意：两者的测试结果的比较，一定是在相同条件下测试的结果（环境对性能的影响较大）。

#### 参考同类系统功能的响应时间

对于新开发的系统，在没有生产环境数据、历史版本参考的情况下，可参考其他类似系统的响应时间的实测结果，对比本系统实测的结果，经过产品经理、开发、运营运维共同评审确定该系统的性能需求标准，并按照达成一致的需求标准进行评估。

## TPS（QPS）、并发用户数

### 定义

**TPS**：每秒事务数，指系统每秒能够处理的事务数量（一个事务可能是有多个请求组成）。

**QPS**：每秒查询率，只系统每秒能够处理的查询（通常指一个request请求）数量。

**并发用户数**：在同一时刻（任一时刻）与服务器进行交互（服务器正在处理）的在线用户的数量。对于并发用户数避免两种错误的理解，一种错误的理解是把并发用户数理解为系统注册用户数，还有一个错误的理解是把并发用户数理解为系统在线用户数。

### 测试方法

1. 找到稳定运行的最高的吞吐量(TPS)。

* 测试前确定一个响应时间的标准（如：小于100ms），然后进行基准测试，通过虚拟并发用户数为1的方式测试，记录测试的TPS、响应时间测试结果，将该响应时间与标准比较，若大于标准响应时间，那么则说明系统有问题无法满足标准，若该响应时间小于标准时间，则继续下面的测试。
* 通过压力测试找到最大的吞吐量：在基准测试响应时间的限制下，找到系统最大的吞吐量（TPS），该状况下响应时间满足要求、吞吐量最大，可确定为“最佳并发用户数”。方法是按照一定的步长，不断增加虚拟并发用户数，直至响应时间超过限制、吞吐量不在增长、任意节点资源使用率超过要求（如：70%）。

1. 负载测试：保持最大吞吐量，执行负载测试，持续30分钟，记录测试TPS、响应时间测试结果。
2. 稳定性测试：保持“最佳并发用户数”，执行稳定性测试，持续3\*24小时，记录测试TPS、响应时间。
3. 在成功率100%的限制下（不考虑响应时间长短）找到系统的极限值。不断增加并发用户数，能够持续运行30分钟不出错误的并发量即为系统的极限值。

### 分析评估

#### 最大吞吐量和系统资源使用的分析

在明确响应时间要求的限制下，压力测试过程中，找到最大吞吐量的拐点时，分析系统资源（CPU、内存）的使用率，若使用率过低，则继续加大并发用户量，若系统的所有节点的任一资源均无法达到70%使用率，说明系统存在系统类、软件类问题和瓶颈，需要调优。

#### TPS与需求规格定义（生产环境负载）比较

1. 需求规格说明书中有明确的标准定义

将吞吐量的实测结果和需求规格定义（生产环境负载）比较，若大于需求规格定义则为通过。一般最大吞吐量对应生产环境的平均负载，系统极限值仅用来应对生产环境的突发高峰。

1. 需求规格说明书中无明确的标准定义

若需求规格说明书中没有关于性能指标的明确定义，在性能测试方案设计阶段性能测试工程师应推动测试、开发、产品经理和运维运营一起，明确相关性能指标。

性能指标可参考生产环境交易量统计数据来评估，评估结果一般应略高于当前生产环境的负载（预留半年到一年访问量增长的余量）。

若新开发产品，无生产环境度量数据，可参考同类产品、本产品运营推广计划来评估本产品的性能指标，性能指标确认的结果可通过提单结论归档。

#### 负载测试、稳定性测试采样分析

在负载测试、稳定性测试过程中，保持最高吞吐量情况下压测，TPS曲线、响应时间曲线应该是趋于稳定的，如出现大的波动（骤升或骤降），则视为异常的拐点（问题），需要进行问题定位。

## 请求成功率

### 定义

顾名思义，请求成功率代表所有请求中，成功接收到响应的请求所占的比例。系统的吞吐量和请求成功率是挂钩的。

### 测试方法

请求成功率是响应时间、TPS等指标的前提，在成功率满足大于99.9%的前提下，响应时间、TPS满足预期。成功率的测试方法，可参考2.1、2.2章节中关于响应时间、TPS的测试方法。

### 分析评估

标准要求：负载测试、稳定性测试，请求成功率要求大于99.9%。

## CPU使用率、内存使用率、IO WAIT

### 定义

#### CPU使用率：

CPU时间的百分比，共分为以下几个维度，我们通常认为的CPU使用率是us（用户态）+sy（系统态）使用的CPU百分比之和。

* Us：用户态使用的cpu时间百分比
* Sy：系统态使用的cpu时间百分比
* Ni：用做nice加权的进程分配的用户态cpu时间百分比
* Id：空闲的cpu时间百分比
* Wa：cpu等待IO完成时间百分比，指通常我们讲的IO WAIT
* Hi：硬中断消耗时间百分比
* Si：软中断消耗时间百分比

#### 内存使用率

内存使用率通常是指已使用的内存在总体内存中所占的比例。

* Total：内存总数
* Used：已使用内存数
* Free：空闲的内存数
* Shared：多个进程共享的内存总额，查询结果总是0
* Buffers：Buffer Cache 磁盘缓存的大小
* Cached：cached Page Cache磁盘缓存的大小

可用内存=free+buffer+cached

所以计算内存使用率公式为：

内存使用率=（total-free-buffer-cached）/total \* 100%

### 测试方法

1. 通过Nmon采集记录（可以同时监控CPU、内存、IO等各种丰富的性能资源指标），可以持续记录测试过程每个时间点的资源使用情况，对于多核系统，也可以分别监控每个CPU的使用情况。可通过Nmon采集的资源使用曲线、结合系统的性能表现，初步完成系统的评估（判断是否存在问题）。
2. 通过Linux命令来进行系统问题的详细分析（补充中）。

* top：查看进程活动状态以及一些系统状况，没办法记录所有时间点的资源使用情况，好处是可以查看到进程级别的资源使用。
* vmstat：查看系统状态、硬件和系统信息等，通过vmstat查询，查询整体的CPU使用情况，同时也可以查询进程、内存、交换页面、IO的情况。
* iostat：查看CPU 负载，硬盘状况
* sar(同类的tsar阿里开源工具）：综合类工具，比较全面的查看系统状况的工具，如文件的读写情况、系统调用的使用情况、[磁盘](http://lovesoo.org/tag/%e7%a3%81%e7%9b%98" \t "http://lovesoo.org/_blank" \o "View all posts in 磁盘)[I/O](http://lovesoo.org/tag/io" \t "http://lovesoo.org/_blank" \o "View all posts in I/O)、[CPU](http://lovesoo.org/tag/cpu" \t "http://lovesoo.org/_blank" \o "View all posts in CPU)效率、[内存](http://lovesoo.org/tag/%e5%86%85%e5%ad%98" \t "http://lovesoo.org/_blank" \o "View all posts in 内存)使用状况、进程活动及IPC有关的活动。
* mpstat：多核CPU的可以通过该命令查看某个cpu的情况
* netstat：查看网路情况
* tcpdump\tcptrace：抓取网络数据包和分析网络数据包工具
* dstat：综合工具，综合了 vmstat, iostat, ifstat, netstat 等多个信息
* ps：进程查询工具

### 分析评估

1. 通过比较“最大吞吐量”和“系统资源使用率”之间的关系进行分析，系统必须在CPU、内存资源使用小于70%的情况下，提供最大的吞吐量和用户能接受的响应时间限制。
2. 通过并发量的增加和资源使用率的增加比较分析，判断并发的资源开销，通过历史经验对比判断资源消耗是否过高。
3. 负载测试、稳定性测试长时间运行，CPU、内存使用率不的超过70%。
4. 通过IO Wait值初步判断，一般将IO Wait指标的标准定为2，但并不是说IO Wait超过2就是有问题，而是作为条件触发测试人员检查IO对业务是否有影响，如果对业务没有影响，就不算问题，如果有影响就进一步分析IO的问题。

## GC

待补充

## 进程级别的资源占用

待规划工具后完成内容补充