

## 28 | 带你一起解读不同视角的软件性能与性能指标

2018-08-31 茹炳晟

软件测试52讲

[进入课程 >](#)



讲述：茹炳晟

时长 21:33 大小 8.64M



你好，我是茹炳晟。今天我和你分享的主题是：带你一起解读不同视角的软件性能与性能指标。

我用三篇文章的篇幅，和你分享了代码级测试的一些基本概念和测试方法，希望可以帮助你解决在实际工作中遇到的问题，如果你感觉还有一些问题没有解决的话，欢迎你给我留言，我们一起去讨论、解决。

从今天开始，我将和你分享一个新的测试主题：性能测试。这个系列，我准备了七个主题，要和你详细聊聊软件性能测试相关的知识点。

因为性能测试的专业性很强，所以我会以从 0 到 1 的入门者视角，系统性地阐述性能测试的方法以及应用领域，用实例去诠释各种性能指标；我还会分享一些前端和后端性能测试工

具的基本原理，并基于 LoadRunner 去分析一些大型企业性能测试的规划、设计、实现的具体实例；我还会和你分享一些跨国软件公司性能测试卓越中心的实际案例。

希望这个系列的内容，可以完善你的性能测试知识体系，真真正正地帮你解决实际项目中遇到的问题。

那么，作为性能测试系列的第一次分享，我会站在全局的视角，帮你梳理软件性能、软件性能测试相关的知识点，让你对那些你或许已经耳熟能详的性能指标有一个更清晰的理解，为你完成后续的性能测试工作打好基础。

如果你对软件性能的理解还停留在响应时间的快慢上，那么赶紧和我一起开始今天的内容吧。

**在开始下面的内容前，请你先思考一个问题：当我们谈及软件性能的时候，我们到底谈的是什么？**

目前，对软件性能最普遍的理解就是软件处理的及时性。但其实，从不同的系统类型，以及不同的视角去讨论软件性能，都会有所区别。

对于不同类型的系统，软件性能的关注点各不相同，比如：

Web 类应用和手机端应用，一般以终端用户感受到的端到端的响应时间来描述系统的性能；

非交互式的应用，比如典型的电信和银行后台处理系统，响应时间关注更多的是事件处理的速度，以及单位时间的事件吞吐量。

这很容易理解。同样地，对同一个系统来说，不同的对象群体对软件性能的关注点和期望也不完全相同，甚至很多时候是对立的。这里，不同的对象群体可以分为四大类：终端用户、系统运维人员、软件设计开发人员和性能测试人员。

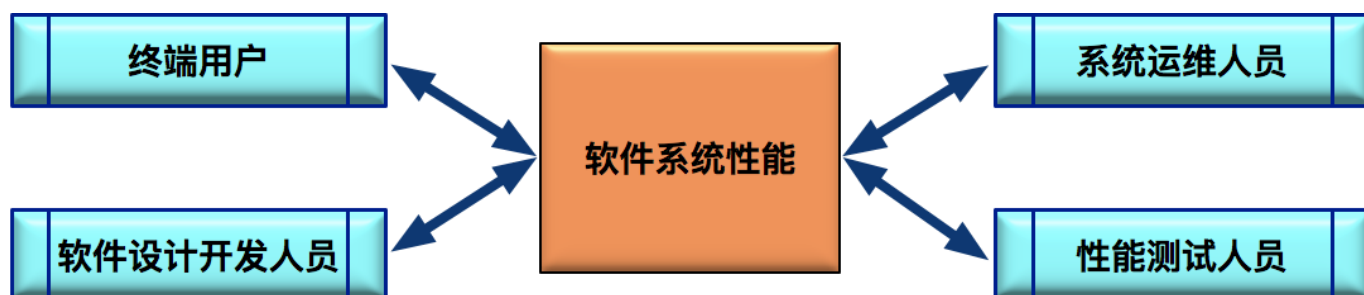


图 1 衡量软件性能的四个维度

终端用户是软件系统的最终使用者，他们对软件性能的反馈直接决定了这个系统的应用前景；而，软件开发人员、运维人员、性能测试人员，对性能测试的关注点则直接决定了一个系统交付到用户手中的性能。

只有全面了解各类群体对软件系统的不同需求，才能保证这个系统具有真正高可靠的性能。所以，接下来我会从这四类人的视角和维度去分享软件性能到底指的是什么。

## 终端用户眼中的软件性能

从终端用户（也就是软件系统使用者）的维度来讲，软件性能表现为用户进行业务操作时的主观响应时间。具体来讲就是，从用户在界面上完成一个操作开始，到系统把本次操作的结果以用户能察觉的方式展现出来的全部时间。对终端用户来说，这个时间越短体验越好。

这个响应时间是终端用户对系统性能的最直观印象，包括了系统响应时间和前端展现时间。

系统响应时间，反应的是系统能力，又可以进一步细分为应用系统处理时间、数据库处理时间和网络传输时间等；

前端展现时间，取决于用户端的处理能力。

从这个角度来看，你就可以非常容易理解性能测试为什么会分为后端（服务器端）的性能测试和前端（通常是浏览器端）的性能测试了。

## 系统运维人员眼中的软件性能

从软件系统运维（也就是系统运维人员）的角度，软件性能除了包括单个用户的响应时间外，更要关注大量用户并发访问时的负载，以及可能的更大负载情况下的系统健康状态、并发处理能力、当前部署的系统容量、可能的系统瓶颈、系统配置层面的调优、数据库的调优，以及长时间运行稳定性和可扩展性。

大多数情况下，系统运维人员和终端用户是站在同一条战线上的，希望系统的响应速度尽可能地快。但，某些情况下他们的意见是对立的，最常见的情况就是，系统运维人员必须在最大并发用户数和系统响应时间之间进行权衡取舍。比如，当有两套系统配置方案可以提供以下系统能力的时：

配置方案 A 可以提供 100 万并发访问用户的能力，此时用户的登录响应时间是 3 秒；

配置方案 B 可以提供 500 万并发访问用户的能力，此时用户的登录响应时间是 8 秒。

这时，从全局利益最大化角度来看，系统具有更大并发用户承载能力的价值会更大，所以运维人员一般都会选择方案 B。

**目前，有些系统为了能够承载更多的并发用户，往往会牺牲等待时间而引入预期的等待机制。**比如，火车票购票网站，就在处理极大并发用户时采用了排队机制，以尽可能提高系统容量，但却增加了用户实际感受到的响应时间。

## 软件设计开发人员眼中的软件性能

从软件系统开发（也就是软件设计开发人员）的角度来讲，软件性能关注的是性能相关的设计和实现细节，这几乎涵盖了软件设计和开发的全过程。

在大型传统软件企业中，软件性能绝不仅仅是性能测试阶段要考虑的问题，而是整个软件研发生命周期都要考虑的内容，我们往往把围绕性能相关的活动称为“性能工程”（Performance Engineering）。我曾在惠普软件研发中心的性能测试卓越中心负责这方面的技术工作，所以感受颇深。

在软件设计开发人员眼中，软件性能通常会包含算法设计、架构设计、性能最佳实践、数据库相关、软件性能的可测试性这五大方面。其中，每个方面关注的点，也包括很多。

### 第一，算法设计包含的点：

核心算法的设计与实现是否高效；

必要时，设计上是否采用 buffer 机制以提高性能，降低 I/O；

是否存在潜在的内存泄露；

是否存在并发环境下的线程安全问题；

是否存在不合理的线程同步方式；

是否存在不合理的资源竞争。

### 第二，架构设计包含的内容：

站在整体系统的角度，是否可以方便地进行系统容量和性能扩展；

应用集群的可扩展性是否经过测试和验证；

缓存集群的可扩展性是否经过测试和验证；

数据库的可扩展性是否经过测试和验证。

### **第三，性能最佳实践包含的点：**

代码实现是否遵守开发语言的性能最佳实践；

关键代码是否在白盒级别进行性能测试；

是否考虑前端性能的优化；

必要的时候是否采用数据压缩传输；

对于既要压缩又要加密的场景，是否采用先压缩后加密的顺序。

### **第四，数据库相关的点：**

数据库表设计是否高效；

是否引入必要的索引；

SQL 语句的执行计划是否合理；

SQL 语句除了功能是否要考虑性能要求；

数据库是否需要引入读写分离机制；

系统冷启动后，缓存大量不命中的时候，数据库承载的压力是否超负荷。

### **第五，软件性能的可测试性包含的点：**

是否为性能分析（Profiler）提供必要的接口支持；

是否支持高并发场景下的性能打点；

是否支持全链路性能分析。

需要注意的是，软件开发人员一般不会关注系统部署级别的性能，比如软件运行目标操作系统的调优、应用服务器的参数调优、数据库的参数调优、网络环境的调优等。

系统部署级别的性能测试，目前一般是在系统性能测试阶段或者系统容量规划阶段，由性能测试人员、系统架构师，以及数据库管理员（DBA）协作完成。

## 性能测试人员眼中的软件性能

从性能工程的角度看，性能测试工程师关注的是算法设计、架构设计、性能最佳实践、数据库相关、软件性能的可测试性这五大方面。

在系统架构师、DBA，以及开发人员的协助下，性能测试人员既要能够准确把握软件的性能需求，又要能够准确定位引起“不好”性能表现的制约因素和根源，并提出相应的解决方案。

一个优秀的性能测试工程师，一般需要具有以下技能：

性能需求的总结和抽象能力；

根据性能测试目标，精准的性能测试场景设计和计算能力；

性能测试场景和性能测试脚本的开发和执行能力；

测试性能报告的分析解读能力；

性能瓶颈的快速排查和定位能力；

性能测试数据的设计和实现能力；

面对互联网产品，全链路压测的设计与执行能力，能够和系统架构师一起处理流量标记、影子数据库等的技术设计能力；

深入理解性能测试工具的内部实现原理，当性能测试工具有限制时，可以进行扩展二次开发；

极其宽广的知识面，既要有“面”的知识，比如系统架构、存储架构、网络架构等全局的知识，还要有大量“点”的知识积累，比如数据库 SQL 语句的执行计划调优、JVM 垃圾回收（GC）机制、多线程常见问题等等。

看到如此多的技能要求你可能有点害怕，的确，性能测试的专业性比较强，我经常把优秀的性能工程师比作是优秀的医生，也是这个原因。你需要在实际项目中积累大量的实际案例，才能慢慢培养所谓的“性能直觉”，从我个人的学习路径来讲也是如此。

天下无难事只怕有心人，所以抓住一切可以充实自己的机会吧，我们终将会破茧成蝶。



这就是终端用户、系统运维工程师、软件开发工程师，以及性能测试工程师眼中的性能测试了，至此我们也就非常容易理解，不同的群体对同一个系统的性能要求为什么会如此不同。

现在，**我再来和你说说衡量软件性能的三个最常用的指标：并发用户数、响应时间，以及系统吞吐量**。只要你接触过性能测试，或者你的团队开展过性能测试，你都应该听说这三个指标。但其实很多人对它们的理解还都停留在表面，并没有深入细致地考虑过其本质与内涵，这也导致了性能测试很多时候并没有发挥应有的作用。

因此，接下来我会和你深入地聊聊这三个指标的内涵和外延，帮助你获得一个全新的认识。

## 并发用户数

**并发用户数，是性能需求与测试最常用，也是最重要的指标之一。它包含了业务层面和后端服务器层面的两层含义。**

业务层面的并发用户数，指的是实际使用系统的用户总数。但是，单靠这个指标并不能反映系统实际承载的压力，我们还要结合用户行为模型才能得到系统实际承载的压力。

后端服务器层面的并发用户数，指的是“同时向服务器发送请求的数量”，直接反映了系统实际承载的压力。

为了让你更好地理解这两层含义之间的区别，我们先一起来看一个实例：一个已经投入运行的 ERP 系统，该系统所在企业共有 5000 名员工并都拥有账号，也就是说这个系统有 5000 个潜在用户。

根据系统日志分析得知，该系统最大在线用户数是 2500 人，那么从宏观角度来看，2500 就是这个系统的最大并发用户数。但是，2500 这个数据仅仅是说在最高峰时段有 2500 个用户登录了系统，而服务器所承受的压力取决于登录用户的行为，所以它并不能准确表现服务器此时此刻正在承受的压力。

假设在某一时间点上，这 2500 个用户中，30% 用户处于页面浏览状态（对服务器没有发起请求），20% 用户在填写订单（也没有对服务器发起请求），5% 用户在递交订单，15% 用户在查询订单，而另外的 30% 用户没有进行任何操作。那么此时，这 2500 个“并发用户”中真正对服务器产生压力的只有 500 个用户（ $(5\% + 15\%) \times 2500 = 500$ ）。

在这个例子中，5000 是最大的“系统潜在用户数”，2500 是最大的“业务并发用户数”，500 则是某个时间点上的“实际并发用户数”。而此时这 500 个用户同时执行业务操作所实际触发的服务器端的所有调用，叫作“服务器并发请求数”。

从这个例子可以看出，在系统运行期间的某个时间点上，有一个指标叫作“同时向服务器发送请求的数量”，这个“同时向服务器发送请求的数量”就是服务器层面的并发用户数，这个指标同时取决于业务并发用户数和用户行为模式，而且用户行为模式占的比重较大。

因此，分析得到准确的用户行为模式，是性能测试中的关键一环。但，获得精准的用户行为模式，是除了获取性能需求外，最困难的工作。

目前，获取用户行为模式的方法，主要分为两种：

对于已经上线的系统来说，往往采用系统日志分析法获取用户行为统计和峰值并发量等重要信息；

而对于未上线的全新系统来说，通常的做法是参考行业中类似系统的统计信息来建模，然后分析。

## 响应时间

**通俗来讲，响应时间反映了完成某个操作所需要的时间，其标准定义是“应用系统从请求发出开始，到客户端接收到最后一个字节数据所消耗的时间”，是用户视角软件性能的主要体现。**

响应时间，分为前端展现时间和系统响应时间两部分。其中，前端时间，又称呈现时间，取决于客户端收到服务器返回的数据后渲染页面所消耗的时间；而系统响应时间，又可以进一步划分为 Web 服务器时间、应用服务器时间、数据库时间，以及各服务器间通信的网络时间。

除非是针对前端的性能测试与调优，软件的性能测试一般更关注服务器端。但是，服务器端响应时间的概念非常清晰、直接，就是指从发出请求起到处理完成的时间，没有二义性；而前端时间的定义，在我看来存在些歧义。所以，接下来我会和你详细聊聊前端时间这个话题。



虽然前端时间一定程度上取决于客户端的处理能力，但是前端开发人员现在还会使用一些编程技巧在数据尚未完全接收完成时呈现数据，以减少用户实际感受到的主观响应时间。也就是说，我们现在会普遍采用提前渲染技术，使得用户实际感受到的响应时间通常要小于标准定义的响应时间。

鉴于此，**我认为响应时间的标准定义就不尽合理了，尤其是对于“接收到最后一个字节”。**

我来举个实际案例吧。加载一个网页时，如果 10 秒后还是白屏，那你一定会感觉很慢、性能无法接受。但是，回想一下你曾经上新浪网的经历，当加载新浪首页时，你应该不会感觉速度很慢吧。其实，实际情况是，新浪首页的加载时间要远大于 10 秒，只是新浪采用了数据尚未完全接收完成时进行呈现的技术，大大缩短了用户主观感受到的时间，提升了用户体验。

所以，**严格来讲，响应时间应该包含两层含义：技术层面的标准定义和基于用户主观感受时间的定义。而在性能测试过程中，我们应该使用哪个层面的含义将取决于性能测试的类型。**显然，对于软件服务器端的性能测试肯定要采用标准定义，而对于前端性能评估，则应该采用用户主观感受时间的定义。

当然，我们在前端性能测试中，会利用一些事件的触发（比如 DOM-Load、Page-load 等）来客观地衡量“主观的前端性能”。这部分内容我会在后面介绍前端性能测试时，和你详细讨论。

## 系统吞吐量

系统吞吐量，是最能直接体现软件系统负载承受能力的指标。

这里需要注意的是，**所有对吞吐量的讨论都必须以“单位时间”作为基本前提。**其实，我认为把“Throughput”翻译成吞吐率更贴切，因为我们可以这样理解：吞吐率 = 吞吐量 / 单位时间。但既然国内很多资料已经翻译为了“吞吐量”，所以通常情况下我们不会刻意去区分吞吐量和吞吐率，统称为吞吐量。

对性能测试而言，通常用“Requests/Second”“Pages/Second”“Bytes/Second”来衡量吞吐量。当然，从业务的角度来讲，吞吐量也可以用单位时间的业务处理数量来衡量。

以不同方式表达的吞吐量可以说明不同层次的问题。比如：

“Bytes/Second” 和 “Pages/Second” 表示的吞吐量，主要受网络设置、服务器架构、应用服务器制约；

“Requests/Second” 表示的吞吐量，主要受应用服务器和应用本身实现的制约。

这里需要特别注意的是：虽说吞吐量可以反映服务器承受负载的情况，但在不同并发用户数的场景下，即使系统具有相近的吞吐量，但是得到的系统性能瓶颈也会相差甚远。

比如，某个测试场景中采用 100 个并发用户，每个用户每隔 1 秒发出一个 Request，另外一个测试场景采用 1000 个并发用户，每个用户每隔 10 秒发出一个 Request。显然这两个场景具有相同的吞吐量，都是 100 Requests/second，但是两种场景下的系统性能拐点肯定不同。因为，两个场景所占用的资源是不同的。

这就要求性能测试场景的指标，必然不是单个，需要根据实际情况组合并发用户数、响应时间这两个指标。

## 总结

作为性能测试系列的第一篇文章，我和你一起梳理了软件性能、软件性能测试相关的知识点，旨在你对加深软件性能指标的理解，为后续的性能测试实战打好基础。

首先，我从终端用户、系统运维人员、软件设计开发人员和性能测试人员，这四个维度介绍了软件系统的性能到底指的是什么：

终端用户希望自己的业务操作越快越好；

系统运维人员追求系统整体的容量和稳定；

开发人员以 “性能工程” 的视角关注实现过程的性能；

性能测试人员需要全盘考量、各个击破。

然后，我介绍了软件性能的三个最常用的指标：并发用户数，响应时间，系统吞吐量：

并发用户数包含不同层面的含义，既可以指实际的并发用户数，也可以指服务器端的并发数量；

响应时间也包含两层含义，技术层面的标准定义和基于用户主观感受时间的定义；

系统吞吐量是最能直接体现软件系统承受负载能力的指标，但也必须和其他指标一起使用才能更好地说明问题。

## 思考题

系统吞吐量的表现形式有很多，比

如“Requests/Second” “Pages/Second” “Bytes/Second” 等，你在性能测试项目中，选择具体的系统吞吐量指标时，会考虑哪些因素呢？

欢迎你给我留言。

 极客时间

# 软件测试52讲

## 从小工到专家的实战心法



茹炳晟

eBay中国研发中心  
测试基础架构技术主管

新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 27 | 深入浅出之动态测试方法

下一篇 29 | 聊聊性能测试的基本方法与应用领域

## 精选留言 (17)

 写留言



annie



2018-08-31

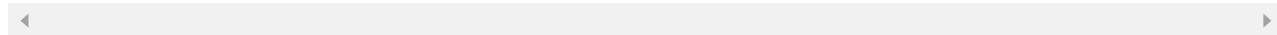


10

如果考察HTTP或者业务层面，可以选择“Requests/Second” “Pages/Second”

如果考察系统层面或网络层面，可以选择“Bytes/Second”，即网卡每秒接收/发送到字节数

作者回复: 非常好的方法，实践中的确就是这么做的👍



**sylan215**

2018-08-31



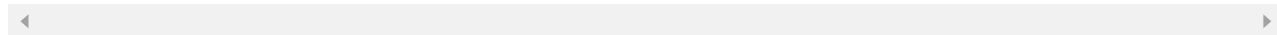
5

之前面试时，会问一下面试者的职业规划是什么，有不少人都说是做性能测试，但是问会代码么？不会；数据库呢？了解基本的 SQL 语句；网络知识呢？了解基本的协议。

好吧，不是说这些同学能力上有啥问题，感觉是对性能测试的认识上还存在一些不明确，唯一明确的可能就是性能测试很值钱。...

展开 ∨

作者回复: 感谢认可，有收获就好👍



**Struggling**

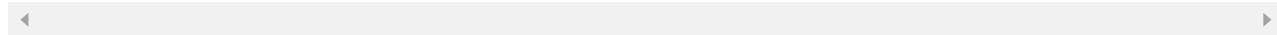
2018-09-03



2

终于迎来性能测试篇，一直很认同好的性能测试工程师不但知识面要广，还要在某些知识点比开发更有深度，以及快速定位问题的能力，期待老师的讲解

作者回复: 性能测试其实是个很难的领域，需要大量实际的积累，知识面不仅要广还要深入，比如jvm调优，数据库执行计划调优等，都需要对jvm和数据库内部原理有很清晰的认知



**假装乐**

2018-08-31



1

说到性能测试一直有个疑问，性能测试何时开展，实际环境开展感觉有点晚了，生产环境又不知道衡量指标是否具有借鉴价值，不知道老师的团队实际工作中，各阶段如何开展

展开 ∨





任欣

2019-05-19



考虑request/second比较多一些，因为前后端异步通信比较多，所以请求的数量会大大影响页面的加载速度。老师有没有什么更好的办法



口水窝

2019-04-24



比如，总共有100个用户，每隔2秒启动10个用户。我觉得这种场景设计，千人千面，每个人的理解不一样，所以设计的场景就不一样，关键看用户最关注的是什么，要求测试那些场景吧。

展开 ∨



秋月

2019-02-14



我们的性能测试不能在生产环境中进行，在集成环境测试的时候，开发又不认可数据，说集成环境和生产环境的部署相差太大，不具有代表性，性能测试都不知道要如何开展才能提现出来，好苦恼

展开 ∨



苦行僧

2019-02-01



不同业务形态的性能测试指标分别需要关注哪些点？比如测试中间件和测试web站点



年轻人的瞎...

2019-01-01



想入门性能测试有什么好的方法进行操作训练呢？

展开 ∨

作者回复: 入门的话我觉得可以学着用loadrunner，商业产品入门容易



关什么事

2018-12-20



留言马上改版了

展开 ▾



**Shine**

2018-11-09



干货就是读了一遍，还会想要再读一遍。思考不止。

展开 ▾



**sharon**

2018-09-27



请问老师，对于web业务流程通过LR进行压力测试后形成的报告不是也有网页细分报告吗，这里不能理解为页面下载的前端性能，为什么还要用Yslow等工具呢？

展开 ▾



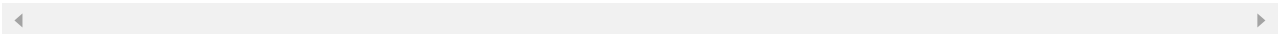
**Faris** 

2018-09-12



我是测试小程序的，很多前段加载和页面跳转很难定义标准，导致不知道性能指标是什么，我们公司又是自研的小公司。功能指标也只能通过测试人员定义

作者回复: 这个的确没有明确的标准定义，参考的做法是参考大公司类似app或者小程序的加载时间



**涅槃Ls**

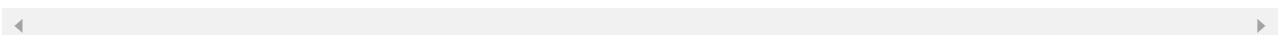
2018-09-09



打卡28，性能测试

展开 ▾

作者回复: 感谢支持



**hmc**

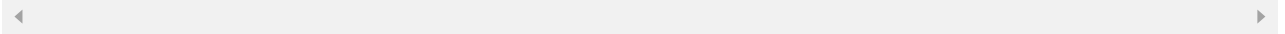
2018-09-04



请问有什么推荐的性能测试工具吗？

展开 ▾

作者回复: 后端性能工具推荐loadrunner和jmeter, 前端性能工具推荐webpagetest和Yslow, 后面文章会有讲



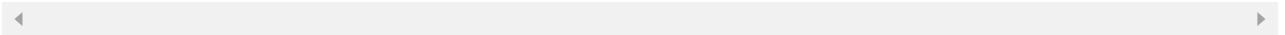
**在路上**

2018-09-01



希望老师能性能测试监控工具nmon, 做进一步分析, 工作中也在用, 但一般就只会关注CPU和内存, 具体里面的指标, 还是很茫然

作者回复: nmon本身不是问题的本质, 本质是你需要监控什么, 你监控的目的又是什么, 所以还是从最原始的测试目标来规划。



**威廉姆**

2018-08-31



性能测试报告分析, 以及相关的性能瓶颈排查, 包括性能调优, 建议老师可以多涉及一些?