[性能测试知多少---性能分析与调优的原理](http://www.cnblogs.com/fnng/archive/2013/03/19/2970315.html)

2013-03-19 23:53 by 虫师, 17197 阅读, 6 评论, [收藏](http://www.cnblogs.com/fnng/archive/2013/03/19/2970315.html), [编辑](https://i.cnblogs.com/EditPosts.aspx?postid=2970315)

　　最近一直纠结性能分析与调优如何下手，先从硬件开始，还是先从代码或数据库。从操作系统（CPU调度，内存管理，进程调度，磁盘I/O）、网络、协议（HTTP， TCP/IP ），还是从应用程序代码，数据库调优，中间件配置等方面入手。

　　单一个中间件又分web中间件（apache 、IIS），应用中间件（tomcat 、weblogic 、webSphere ）等，虽然都是中间件，每一样拎出来往深了学都不是一朝一夕之功。但调优对于每一项的要求又不仅仅是“知道”或“会使用”这么简单。起码要达到“如何更好的使用”。

　　常看到性能测试书中说，性能测试不单单是性能测试工程师一个人的事儿。需要DBA 、开发人员、运维人员的配合完成。但是在不少情况下性能测试是由性能测试人员独立完成的，退一步就算由其它人员的协助，了解系统架构的的各个模块对于自身的提高也有很大帮助，同进也更能得到别人的尊重。

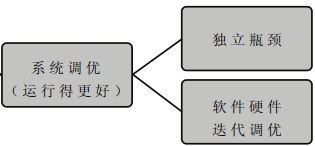
　　再说性能调优之前，我们有必要再提一下进行测试的目的，或者我们进行性能测试的初衷是什么？

**能力验证**：验证某系统在一定条件具有什么样的能力。

**能力规划**：如何使系统达到我们要求的性能能力。

**应用程序诊断**：比如内存泄漏，通过功能测试很难发现，但通过性能测试却很容易发现。

**性能调优**：满足用户需求，进一步进行系统分析找出瓶颈，优化瓶颈，提高系统整体性能。



**一般系统的瓶颈**

性能测试调优需要先发现瓶颈，那么系统一般会存在哪些瓶颈：

**硬件上的性能瓶颈**：

一般指的是CPU、内存、磁盘I/O 方面的问题，分为服务器硬件瓶颈、网络瓶颈（对局域网可以不考虑）、服务器操作系统瓶颈（参数配置）、中间件瓶颈（参数配置、数据库、web服务器等）、应用瓶颈（SQL 语句、数据库设计、业务逻辑、算法等）。

**应用软件上的性能瓶颈**：

一般指的是应用服务器、web 服务器等应用软件，还包括数据库系统。

例如：中间件weblogic 平台上配置的JDBC连接池的参数设置不合理，造成的瓶颈。

**应用程序上的性能瓶颈**：

一般指的是开发人员新开发出来的应用程序。

例如，程序架构规划不合理，程序本身设计有问题（串行处理、请求的处理线程不够），造成系统在大量用户方位时性能低下而造成的瓶颈。

**操作系统上的性能瓶颈**：

一般指的是windows、UNIX、Linux等操作系统。

例如，在进行性能测试，出现物理内存不足时，虚拟内存设置也不合理，虚拟内存的交换效率就会大大降低，从而导致行为的响应时间大大增加，这时认为操作系统上出现性能瓶颈。

**网络设备上的性能瓶颈**：

一般指的是防火墙、动态负载均衡器、交换机等设备。

例如，在动态负载均衡器上设置了动态分发负载的机制，当发现某个应用服务器上的硬件资源已经到达极限时，动态负载均衡器将后续的交易请求发送到其他负载较轻的应用服务器上。在测试时发现，动态负载均衡器没有起到相应的作用，这时可以认为网络瓶颈。

　　性能测试出现的原因及其定位十分复杂，这里只是简单介绍常见的几种瓶颈类型和特征，而性能测试所需要做的就是根据各种情况因素综合考虑，然后协助开发人员\DBA\运维人员一起定位性能瓶颈。

**一般性能调优步骤**

一般性能问题调优的步骤：

**步骤一**：确定问题

应用程序代码：在通常情况下，很多程序的性能问题都是写出来的，因此对于发现瓶颈的模块，应该首先检查一下代码。

数据库配置：经常引起整个系统运行缓慢，一些诸如oracle 的大型数据库都是需要DBA进行正确的参数调整才能投产的。

操作系统配置：不合理就可能引起系统瓶颈。

硬件设置：硬盘速度、内存大小等都是容易引起瓶颈的原因，因此这些都是分析的重点。

网络：网络负载过重导致网络冲突和网络延迟。

**步骤二**：确定问题

　　当确定了问题之后，我们要明确这个问题影响的是响应时间吞吐量，还是其他问题？是多数用户还是少数用户遇到了问题？如果是少数用户，这几个用户与其它用户的操作有什么不用？系统资源监控的结果是否正常？CPU的使用是否到达极限？I/O 情况如何？问题是否集中在某一类模块中？ 是客户端还是服务器出现问题？ 系统硬件配置是否够用？实际负载是否超过了系统的负载能力？ 是否未对系统进行优化？

通过这些分析及一些与系统相关的问题，可以对系统瓶颈有更深入的了解，进而分析出真正的原因。

**步骤三**： 确定调整目标和解决方案

得高系统吞吐理，缩短响应时间，更好地支持并发。

**步骤四**：测试解决方案

对通过解决方案调优后的系统进行基准测试。（基准测试是指通过设计科学的测试方法、测试工具和测试系统，实现对一类测试对象的某项性能指标进行定量的和可对比的测试）

**步骤五**：分析调优结果

系统调优是否达到或者超出了预定目标？系统是整体性能得到了改善，还是以系统某部分性能来解决其他问题。调优是否可以结束了。

　　最后，如果达到了预期目标，调优工作就基本可以结束了。

下面算是一个技巧，如面试官问到一个性能问题假设，我不知道性能问题出在哪儿时，可以按照这个思路回答^\_^

   • 查找瓶颈时按以下顺序，由易到难。  
    服务器硬件瓶颈---〉网络瓶颈（对局域网，可以不考虑）---〉服务器操作系统瓶颈（参数配置）---〉中间件瓶颈（参数配置，数据库，web服务器等）---〉应用瓶颈（SQL语句、数据库设计、业务逻辑、算法等）  
    注：以上过程并不是每个分析中都需要的，要根据测试目的和要求来确定分析的深度。对一些要求低的，我们分析到应用系统在将来大的负载压力（并发用户数、数据量）下，系统的硬件瓶颈在哪儿就够了。  
    • 分段排除法 很有效

性能测试调优应该注意的要点：

* 要点1： 在应用系统的设计开发过程中，应始终把性能放在考虑的范围内。
* 要点2： 确定清晰明确的性能目标是关键。
* 要点3： 必须保证调优后的程序运行正确。
* 要点4： 系统的性能更大程度上取决于良好的设计，调优技巧只是一个辅助手段。
* 要点5： 调优过程是迭代渐进的过程，每一次调优的结果都要反馈到后续的代码开发中去。
* 要点6： 性能调优不能以牺牲代码的可读性和可维护性为代码。

　　本文只介绍了一些性能调优的要关注的东西以及性能调优的一般要点。并没有具体说如何对系统的每个部件进行调优，如何要细说也不是一两书能说清的，对知识面的要求也非常高，是我目前的能力无法触摸的。

**这里做个总结**：

　　《性能测试知多少》系列基本完结，虽然时间拉得比较长，但我没有把它给太监。虽然内容都在空谈性能测试理论知识，但我认为这些东西对于你从事性能测试工作必不可少。当然，我在“ jmeter基础 ” 与“ loadrunner 技巧 ” 中讲解两个性能测试工具的使用。

　　如果我的这些文章对于想了解和学习性能的同学带来一丝的帮助，我将非常开心。我不是高手，只是和你一起热爱测试技术的初学者，只是比较喜欢总结；也时常为前途迷茫，但我知道只要断去学习，路就在前方。我后面会整理性能调优的相关文章。

性能测试所有文章汇总：http://www.cnblogs.com/fnng/archive/2012/08/17/2644878.html