[性能测试知多少---测试环境搭建](http://www.cnblogs.com/fnng/archive/2012/10/20/2732786.html)

2012-10-20 23:36 by 虫师, 14765 阅读, 2 评论, [收藏](http://www.cnblogs.com/fnng/archive/2012/10/20/2732786.html), [编辑](https://i.cnblogs.com/EditPosts.aspx?postid=2732786)

　　在进行性能则试前，需要完成性能测试的搭建工作，一般包括硬件环境、软件环境及网络环境，可以要求配置和开发工程师协助完成，但是作为一个优秀性能测试工程师，这也是你的必备技能之一。

**性能测试环境与功能测试环境的区别**

　　那么性能测试环境与功能测试环境有什么不同呢？性能测试对测试环境的干净、独立性要求更高，更为严格。对于一个相对较规范的公司，都会建立其独立的研发环境、测试环境、线网环境（最终运行软件的环境）。

**（**

　　这里多扯一点，系统可以分为C/S架构的系统与B/S架构的系统，C/S架构的系统又可以分为两种，第一种是基本不用与服务器连接的，比如我们用到的java虚拟机JVM，photo shop平面处理软件，我们可以开启软件更新功能，这时软件向服务器发请求，查当前版本是否是服务器端发布的最新版本，然后，提示用例是否需要更新或下载最新版本的软件。当然，我们也可以关闭更新功能或不检测更新。那么这个软件一样可以在电脑上运行。对于这类软件，我的主要测试环境就是用户的电脑。不同硬件配置、不同操作系统下对软件一系列，从安装使用到卸载。除了验证软件与硬件和系统的兼容性能，还需要验证与其它软件是否兼容。

　　第二种类型的C/S软件要时刻与服务器与连接，比如我的在线网游，QQ聊天工具等。从软件的启动就需要与服务器进行连接，对于此类软件，我们测试环境的重点依然是用户电脑，但服务器端必须也有一个相对应的测试环境支撑。

　　对于B/S的系统，我们测试环境的重点就要由用户电脑转为服务器端了，因为系统的所有功能都是由服务器端传递给用户的，所以需要验证服务器传递来的功能是否可用，以及功能的容错能力等。

**）**

　　再回到测试环境的问题上，对于一些企业为了节约资源，进行功能测试的测试环境，一台服务器可以运行多个系统，通过技术手段可以使系统之间是不会相互影响的（以前公司就是一台服务器上跑多个tomcat）。因为功能测试的重点大于系统对客户端发来的请求是否可以进行正确的处理。

　　那么性能测试为什么对系统的环境要求干净、独立呢？性能测试是要对整个系统运行的软件硬件环境进行测试的，如果某环境下运行多个系统，就很难判断其中的某个环境对资源的占用情况。

**性能测试环境包含内容**

**一般web应用系统分为3层架构**（在系统架构一章中有介绍）

\* 表现层（web服务器）

\* 业务逻辑层（应用服务器）

\* 数据层（数据库服务器）

**性能测试环境包含内容**：

硬件：服务器、客户端、交换机等。

软件：数据库、中间件、被测系统、操作系统等。

网络：有线/无线/宽带、网络协议等。

**如何保证测试环境与真实生产的一致性**

保证性能测试与真实生产环境的一致性，具体从以下三个方面来看：

**1、硬件环境，包括服务器环境、与网络环境**

　　如服务器的型号以及是否和其它应用程序共享此服务器，是否在集群环境下，是否通过BIGIP进行负载均衡，客户使用的硬件配置情况，使用的交换机型号，网络传输速率。

**2、软件环境**

版本一致性

　　包括包括操作系统、数据库、中间件的版本，被测系统的版本。

配置一致性

　　系统（操作系统/数据库/中间件/被测试系统）参数的配置一致，这些系统参数的配置有可能对系统造成巨大的影响。所以，除了保证测试环境与真实环境所使用的软件版本一致，也要关注其参数的配置是否一致。

**3、使用场景的一致性**

基础数据的一致性

　　包括预测的业务数据量，以及数据类型的分配。很简单的一个列子，一个系统的数据库只有10条数据和一条数据库里几千万条数据，我们在对其进行性能测试时，得到的性能指标可能会有非常大的差别。

　　为了保证每次测试环境的更加一致性，磁盘的使用情况以及磁盘的碎片情况也会或多或少的影响的性能。

使用模式的一致性

　　尽量模拟真实场景下用户的使用情况，其实，我们在做性能测试前期的需求分析，其主要目的也就是为了更真实的模拟用户的使用情况。

**性能测试环境的实施策略**

　　上面讲测试环境与生产环境保持一致所需要注意的内容。其实在实际的测试中，我们很难搭建出与生产环境完全一致的一个测试环境，除非我们暂停生产环境用户于进行性能测试，这往往是不可能。一方面某些生产环境是不允许被暂停的，另一方面也为生产环境的安全性考虑。

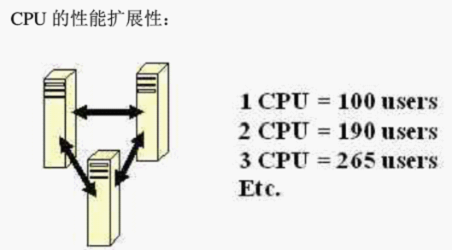
　　性能测试环境并不像功能测试环境，为了节省资源可以一台服务器上运行多个系统。由于性能测试的特殊性，整个测试环境需要在严格的独立监控下管理，在很多情况下，我们很难申请到足够的且一致的资源（说白了就是老板是否愿意出钱给你买服务器搭建系统）。对于一个并未上线的项目，其生产环境的配置也属于暂定状态，性能测试的目的就是为了确定具体生产环境的硬件配置。这个时候更不可能用过高的配置来搭建性能环境（除非现成的环境放着不用）。

我们一般通过两种策略来搭建性能测试环境（预估方式均有误差）

**1、通过建模的方式实现低端硬件对高端硬件的模拟**

　　通过配置测试来计算不同配置下的硬件性能和系统处理能力的关系，从而推导出满足系统性能的真实配置情况，这种模拟需要精确的建模，模型的采样点越多，那么得到的结果越精确，从而将在低端配置下的性能指标通过该模型转化为高端配置下的最终预计性能指标。

　　例如：搭建一个低端环境，首先需要对这个环境的CPU和内存进行单独的性能基准测试，同过在不同的配置的性能测试，得到一个基准信息列表，当然，在进行这个性能测试的过程中，我们要确定硬件是系统的瓶颈。如果只用一个CUP，在性能测试过程中，其使用率很低，但得到的性能数据都非常底，这起码说明CUP不是系统的平静，这种情况下就无法得到想要的基准值。



　　如上图，在一颗CPU情况下，运行100个用户且CUP使用率接近饱和（100%）。在增加至两颗CUP的情况下，可以运行190个用户且UPU使用率接近饱和（100%），以此做记录，那么我们就可以推算出运行800个用户需要多少颗CUP。

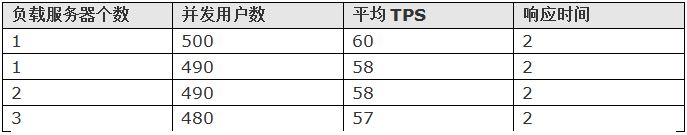
　　如果你在实际应用中使用的CUP型号及其频率并非完全一样，这个时候可以使用EVEREST工具计算每种CUP的得分，对其性能进行评估。

　　内存也可以使用此方法进行测试推导，这里需要我们多进行试验，对硬件的性能以及对整个项目的结构都要做深入的了解，以便尽量减少误差。

**2、通过集群的方式计算**

　　对于较大的系统来说，单台服务器的处理能力是有限的，通常都会采用集群的方式来进行负载均衡，完成对海量请求的处理。虽然无法获得整体集群的测试环境，但是可以对集群上的一个节点进行性能测试，得出该节点的处理能力，再计算每增加一个节点的性能损失，同样也可以能过建模的方式得到大型负载均衡情况下的预计性能指标。

　　例如:首先在单台服务器上获得具体的性能指标，每台服务器能够承受500用户并发，平均TPS为60，响应时间为2秒，接着，添加负载均衡策略，再次测试负载策略下的数据损耗。得出数据后添加1台负载均衡服务器，测试在两台服务器下每台服务器的性能指标，以此类推，可以得到下表：



　　随着负载均衡服务器的添加，平均每台服务器的处理能力会逐渐稳定，从而了解在什么情况下需要多少台负载均衡服务器。

　　对于测试环境的搭建，建议生成专门的文档进行管理，并进行配置管理，确保对测试环境做到基线控制。

------------------------------------------

　　这个性能测试系列以理论与性能测试的整体讲解为主，市面上的大部分书籍借着性能测试的表皮在讲性能测试工具loadrunner，那我何不找份loadrunner使用手册来看更好。

如果你感觉此文对你有所帮助，请推荐下或留言鼓励批评。后面将分享更多的内容。^\_^