**[性能——理念篇](http://wiki.primeton.com/pages/viewpage.action?pageId=12419363)**

2013年4月24日

20:19

     开始写性能相关的事情，算是把我过去的几年的工作做一个汇总，一些好的经验，能给后来人做参考，一些坏的教训，给后来人吸取经验，前车之鉴而已，准备写如下章节，分为理念，环境，工具，积累，案例。

**1. 性能是什么？**

     冠冕堂皇的定义，大家可以自己去google，有说法。对于做企业IT的同行来说，性能是另外一种东西。  
     我这里想说的是，好的性能是一个让客户能接受的系统的运行状态。请注意，我用的词不是客户满意，用的是客户能接受。  
     所以这里提出了一个问题，什么是客户能接受的？  
     换言之，什么是需求？  
     说个最近发生的打了一把酱油的故事。  
     某行的项目要求系统的CPU控制在22%以下，这个是核心的性能需求，当然，还有其他的关于内存，响应时间等等需求。  
     然后同学们开始努力，做应用出身的哥儿们，想到的一定是代码优化，然后测试的同志们也给出个很谬误的推测（其他公司的基础测试），说由于页面Post的数据的数据量多少导致CPU占用率过高，然后就沿着这个错误的方向开始努力优化。代码，系统的一通乱搞。  
     电话会议把我接入，我问了一串问题：为什么要控制在22%以下？如何保证一定在22%以下？如果出现应用数据浪涌，CPU高过22%怎么处 理？CPU高于22%的时候我们的JVM在做什么事情？测试时大于22%的CPU使用率是怎么测试出来的？多大的并发，是否有用到集合点？测试方式是否合 理？数据库的资源是个什么情况？  
     问题问完，一一回答， 整体思路基本就已经清晰。  
     问题很快就搞定了，果然不是平台问题，代码问题，是数据库的相关配置问题，当然，这是后话。  
     如果从最大的性能需求开始分解，只要不是不可能完成的任务，其实都没那么复杂。

**2. 什么是性能测试？**

     当然，如果是标准答案，请大家google。  
     我所理解的性能测试，无非就是拿出一份上下都满意的，由工具生成的性能数据，如此。  
     为了拿出来一份对上对下都有交代的数据，会让人煞费苦心。  
     问自己几个问题：

1. 这数据合理么？
2. 这数据有优化的空间么？优化10%需要有多大的投入？   
        如果这俩问题都能找到让自己满意的答案，并且能说服自己，那么，这个性能测试就是成功的，否则，还需要继续努力。

* 数据合理性：
  1. 资源占用合理么？CPU为什么占满了？为什么没占满？内存使用满了么？为什么没使用满？带宽用完了么？为什么没用完？硬件有木桶效应，最大的处理能力由最短的那块板决定。  
     极限测试，一定会摸到其中的一块板。  
     当然，可能会有人问，你前面不是说需求么？需求也是我的一块板。如果达到需求的板，那性能测试也能宣布完成，拿出一份合理的数据来。
  2. 测试出来的数据合理么？一个用户，一次action算作一个TPS，那一个用户能提供的TPS就是  1/响应时间， 那么多少并发，多少TPS，多少响应时间，这三者是有一定的关系。  
     一个报文多大，带宽多少，那么这个带宽下能提供的最大的TPS是一定的，如果你的数据大于这个TPS，那不又是摔自己一个巴掌么？
* 优化空间
  + 改掉某个算法，系统就能提高多少？如果这里优化了10%，对于整个系统的性能提升是多少？  
         做性能测试，最怕的是面对的是黑盒，把一次请求调用拆解开来，把时间分片，就能知道每个地方的时间消耗，找到耗时最多的地方，优化掉50%，对 于整个系统的性能就可以提升了40%，而找个不那么耗时的地方，死磕改出来50%，加了俩通宵的班，发现整体的系统性能提升只有5%不到，这事一种很悲催 的投入产出。  
         没有写什么工具，没有写什么方法，如果出发点就是错的，可能能力越强错的越远。  
         写这个章节只是为了说明这个道理，如此。

To be continued……