34 | 站在巨人的肩膀：企业级实际性能测试案例与经验分享

2018-09-14 茹炳晟

讲述：茹炳晟
时长 14:01　大小 6.43M

你好，我是茹炳晟，今天我分享的主题是：站在巨人的肩膀之企业级实际性能测试案例与经验分享”。

在前面的四篇文章中，我介绍了前端性能测试和后端性能测试的理论与方法，还分享了使用 LoadRunner 实现后端性能测试的过程。有了这些内容的铺垫，我今天会和你聊聊传统的企业级软件企业如何开展性能测试工作。

其实，传统的企业级软件产品和互联网产品的性能测试，在原理和测试方法上基本一致，它们最大的区别体现在并发数量的数量级上，以及互联网软件产品的性能测试还需要直接在生产环境下进行特有的全链路压测。而全链路压测其实是在传统的企业级软件产品的性能测试基础上，又进行了一些扩展。

所以，在我看来，只要掌握了传统的企业级软件产品的性能测试的原理和方法，搞定互联网产品的性能测试自然不在话下。

言归正传，传统企业级软件产品的性能测试重点是在服务器端。为了达到不同的测试目标，往往会有多种不同类型的性能测试。今天，我就和你聊聊这其中都有哪些测试类型，以及每类测试的目的、所采用的方法。

所以，今天的分享，我会从以下四种测试类型的角度展开：

性能基准测试；

稳定性测试；

并发测试；

容量规划测试。

性能基准测试

性能基准测试，通常被称为 Performance Benchmark Test，是每次对外发布产品版本前必须要完成的测试类型。

性能基准测试，会基于固定的硬件环境和部署架构（比如专用的服务器、固定的专用网络环境、固定大小的集群规模、相同的系统配置、相同的数据库背景数据等），通过执行固定的性能测试场景得到系统的性能测试报告，然后与上一版本发布时的指标进行对比，如果发现指标有“恶化”的趋势，就需要进一步排查。

典型的“恶化”趋势，主要表现在以下几个方面：

同一事务的响应时间变慢了。比如，上一版本中，用户登录的响应时间是 2 s，但是在最新的被测版本中这个响应时间变成了 4 s；

系统资源的占用率变高了。比如，上一版本中，平均 CPU 占用率是 15%，但是在最新的被测版本中平均 CPU 占用率变成了 30%；

网络带宽的使用量变高了。比如，上一版本中，发送总字节数是 20 MB，接收总字节数是 200 MB，但是在最新的被测版本中发送总字节数变成了 25 MB，接收总字节数变成了 250 MB。

这里需要注意的是，这些“恶化”趋势的前提是：完全相同的环境以及测试负载。不同“恶化”指标的排查，有不同的方法。我以最常见的事务响应时间变慢为例，和你说明一下排查方法。

假设，通过性能基准测试的比较结果得知，用户登录的响应时间从 2 s 变成了 4 s。

那么，我们首先要做的是验证在单用户的情况下，是否会出现响应时间变长的问题。具体做法是，将用户登录的虚拟用户脚本单独拿出来，建立一个单用户运行的性能测试场景并执行，观察用户登录的响应时间是否变慢。

如果变慢了，就说明这是单用户登录场景就可重现的性能问题，后续的处理也相对简单了。解决方法是：分析单用户登录的后端日志文件，看看完成登录操作的时间具体都花在了哪些步骤上，相比之前哪些步骤花费的时间变长了，或者是多出了哪些额外的步骤。

如果没有变慢，则说明我们必须尝试在有压力的情况下重现这个性能问题。为此，我们要基于用户登录的虚拟用户脚本构建并发测试的场景，但是我们并不清楚在这个场景设计中到底应该采用多少并发用户、加入多长的思考时间。这时，通常的做法是，直接采用性能基准测试中的并发用户数和思考时间，去尝试重现问题。如果无法重现，我们可以适当地逐步加大测试负载，并观察响应时间的变化趋势。

这里需要注意的是，千万不要使用过大的测试负载。因为测试负载过大的话，系统资源也会成为性能瓶颈，一定会使响应时间变长。但这时，响应时间变长主要是由资源瓶颈造成的，而不是你开始要找的那个原因。

如果此时可以重现问题，那就可以进一步去分析并发场景下，用户登录操作的时间切片，找到具体的原因。如果此时还是不能重现问题的话，情况就比较复杂了，也就是登录操作的性能可能和其他的业务操作存在依赖，或者某种资源竞争关系，这就要具体问题具体分析了。

一般来说，当定位到性能“恶化”的原因并修复后，我们还会再执行一轮性能基准测试，以确保系统对外发布前的性能基准测试指标没有“变坏”。可以说，**通过对每个预发布版本的性能基准测试，我们可以保证新发布系统的整体性能不会下降，这也就是性能基准测试最终要达到的目的。**

很多大型的传统软件公司都有专门的性能测试团队，这个团队会建立标准的性能基准测试场景，并把性能基准测试的结果作为产品是否可以发布的依据之一。比如，我曾工作过的 HP

软件，就由性能测试卓越中心负责维护、执行性能基准测试，并分析测试结果。

从性能基准测试的设计角度来看，你需要特别注意以下三点：

1. 性能基准测试中虚拟用户脚本的选择以及配比，需要尽可能地匹配实际的负载情况；
2. 总体的负载设计不宜过高，通常被测系统的各类占用率指标需要控制在 30% 以内，尽量避免由于资源瓶颈引入的操作延时；
3. 每次性能基准测试前，一般需要对系统资源以及网络资源做一轮快速的基准测试，以保证每次被测环境的一致性，同时也要保证数据库的数据量在同一个级别上。总之，你需要采用一切可能的手段，来确保多次性能基准测试之间的环境一致性。

稳定性测试

稳定性测试，又称可靠性测试，主要是通过长时间（7*24 小时）模拟被测系统的测试负载，来观察系统在长期运行过程中是否有潜在的问题。通过对系统指标的监控，稳定性测试可以发现诸如内存泄漏、资源非法占用等问题。

很多企业级的服务器端产品，在发布前往往都要进行稳定性测试。稳定性测试，通常直接采用性能基准测试中的虚拟用户脚本，但是性能测试场景的设计和性能基准测试场景会有很大不同：

> 一般是采用“波浪式”的测试负载，比如先逐渐加大测试负载，在高负载情况下持续 10 多个小时，然后再逐渐降低负载，这样就构成了一个“波浪”，整个稳定性测试将由很多个这样的波浪连续组成。

稳定性测试成功完成的标志，主要有以下三项：

系统资源的所有监控指标不存在“不可逆转”的上升趋势；

事务的响应时间不存在逐渐变慢的趋势；

事务的错误率不超过 1%。

实际工程项目中，由于稳定性测试执行的时间成本很高，往往需要花费 3~7 天的时间，所以我们一般是在其他所有测试都已经完成，并且所有问题都已经修复之后才开始稳定性测试。

另外，有些企业为了缩短稳定性测试的执行时间，往往还会采用“时间轴压缩”的方法，具体的做法就是：在加大测试负载的前提下，适当缩短每个“波浪”的时间，从而减少整体的测试执行时间。

最后，需要强调的一点是，**虽然很多时候，尤其是产品版本已经逐渐走向成熟期时，稳定性测试并不会发现问题，但是千万不要小看稳定性测试带来的价值**。因为稳定性测试一旦发现问题，那么这些问题都是很严重而且非常隐蔽的大问题。

所以，很多大型的企业级软件企业都会执行严格的稳定性测试，并把稳定性测试的结果作为产品是否可以发布的硬性要求。比如，我曾经工作过的 HP 软件研发中心，它每次产品发布前都会由专门的性能测试团队完成严格的稳定性测试，并以此来决定是否要发布这个产品。

并发测试

并发测试，是在高并发情况下验证单一业务功能的正确性以及性能的测试手段。高并发测试一般使用思考时间为零的虚拟用户脚本来发起具有“集合点”的测试。

“集合点”的概念，我已经在《聊聊性能测试的基本方法与应用领域》中解释过了。如果你不清楚的话，可以再回顾一下这篇文章。如果你还有不理解的地方，也欢迎和我留言讨论。

并发测试，往往被当作功能测试的补充，主要用于发现诸如多线程、资源竞争、资源死锁之类的错误。要执行并发测试，就需要加入“集合点”，所以往往需要修改虚拟用户脚本。

加入“集合点”一般有两种做法：

1. 在虚拟用户脚本的录制过程中直接添加；
2. 在虚拟用户脚本中，通过加入 lr_rendezvous() 函数添加。

容量规划测试

容量规划测试，是为了完成容量规划而设计执行的测试。

那什么是容量规划呢？所谓容量规划，是软件产品为满足用户目标负载而调整自身生产能力的过程。

所以，**容量规划的主要目的是，解决当系统负载将要达到极限处理能力时，我们应该如何通过垂直扩展（增加单机的硬件资源）和水平扩展（增加集群中的机器数量）增加系统整体的负载处理能力的问题。**

目前来讲，容量规划的主要方法是基于水平扩展。但是，具体应该增加多少机器，以及增加后系统的负载处理能力是否会线性增长，这些问题都需要通过容量规划测试进行验证。

那么，容量规划测试具体要怎么做呢？

我们可以使用性能基准测试中的虚拟用户脚本，以及各个业务操作脚本的百分比，压测单机部署的被测系统。我们会采用人工的方式不断增加测试负载直到单机系统的吞吐量指标到达临界值，由此就可以知道单台机器的处理能力。

理论上讲，整个集群的处理能力将等于单台机器的处理能力乘以集群的机器数，但是实际情况并不是这样。实际的集群整体处理能力一定小于这个值，但具体小多少就是要靠实际的测试验证了。

理想的状态是，集群整体的处理能力能够随着集群机器数量的增长呈线性增长。但是，随着机器数量的不断增长，总会在达到某个临界值之后，集群的整体处理能力不再继续呈线性增长。这个临界值是多少，我们也需要通过容量规划测试找出来了。

另外，容量规划测试的测试结果还可以被用作系统容量设计的依据。比如，企业级软件产品的目标用户规模通常是可以预估的，那么我们就可以通过这些预估的系统负载计算出软件部署的集群规模，并且可以在具体实施后通过容量测试的方式进行验证。

总结

在前面的两篇文章中，我和你分享了如何基于 LoadRunner 开展性能测试，但是并没有具体去讲解要开展哪些类型的性能测试。所以，今天我就挑选了最重要的四类性能测试方法，和你分享如何在实际项目中完成这些测试，确保软件的性能。

性能基准测试，可以保证新发布系统的整体性能不会下降；

稳定性测试，主要通过长时间模拟被测系统的测试负载，观察系统在长期运行过程是否存在问题；

并发测试，往往被当作功能测试的补充去发现多线程、资源竞争、资源死锁之类的问题。

容量规划测试，主要用于确定给定负载下的系统集群规模，其测试结果可以被用作系统容量设计的依据。

思考题

你所在企业，还会采用哪些性能测试方法，又是如何展开具体的测试工作的呢？

感谢你的收听，欢迎你给我留言。

© 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　33 | 无实例无真相：基于LoadRunner实现企业级服务器端性能测试的实践（下）

下一篇　35 | 如何准备测试数据？

精选留言 (6)

写留言

小老鼠

2018-11-07

1

1、作基准测试发现了问题，进行分析，若以前一些详细数据没有记录，比如单用户时的资

源数据，我们立刻如何处理？
2、可靠性测试在质量模型中属于效率还是可靠性
3、稳定性测试是否可以提前暴露问题，按文中所言，最后阶段测试，发现又是大问题。我以前在爱立信在Scrum team 中最后都要作一次75%并发 + 容量的48小时的测试。

展开

Ken
2018-10-23

1

你好，请教一下
1.全链路性能测试怎么做，
2.如果是微服架构的又怎么做全链路测试呢？

展开

口水窝
2019-05-07

自己所在的小公司，就我一个测试，所以在功能测试之余做过单接口测试，并发测试，以及相关的持续集成环境部署，因为精力有限，加上绩效考核，只能说在目标明确的基础上首先满足领导的意愿，在关注个人技能的提升吧。

展开

小老鼠
2018-11-07

如何作全链路压力测试

展开

平凡的人_
2018-09-26

负责性能的同事离职了后来我帮忙这块功能（后来由于其他的事情导致离开了，最终没有完成也是给别人），当时测的是多场景接口压，跑的vuser数也是分接口百分比，有个问题请教下就是压力机开始压，压到最后，我在loadrunner里看到的vuser数和开发查看后台日志中请求的并没有我看的那么多，经常可以看到我这一万多他那就几百左右

展开

Struggling
2018-09-25

之前公司一般稳定性测试后面会紧接着做弹力测试，即关掉依赖的服务或者数据库然后再重新打开，看被测服务是否能正常恢复