

性能测试常见面试题

性能测试流程

性能测试计划

一、准备工作

1、系统基础功能验证 性能测试在什么阶段适合实施？切入点很重要！一般而言，只有在系统基础功能测试验证完成、系统趋于稳定的情况下，才会进行性能测试，否则性能测试是无意义的。

2、测试团队组建 根据该项目的具体情况，组建一个几人的性能测试team，其中DBA是必不可少的，然后需要一至几名系统开发人员（对应前端、后台等），还有性能测试设计和分析人员、脚本开发和执行人员；在正式开始工作之前，应该对脚本开发和执行人员进行一些培训，或者应该由具有相关经验的人员担任。

3、工具的选择 综合系统设计、工具成本、测试团队的技能来考虑，选择合适的测试工具，最起码应该满足以下几点：

- ①支持对web（这里以web系统为例）系统的性能测试，支持http和https协议；
- ②工具运行在Windows平台上；
- ③支持对webserver、前端、数据库的性能计数器进行监控；
- 4、预先的业务场景分析 为了对系统性能建立直观上的认识和分析，应对系统较重要和常用的业务场景模块进行分析，针对性的进行分析，以对接接下来的测试计划设计进行准备。

二、测试计划 测试计划阶段最重要的是分析用户场景，确定系统性能目标。

1、性能测试领域分析 根据对项目背景，业务的了解，确定本次性能测试要解决的问题点；是测试系统能否满足实际运行时的需要，还是目前的系统在哪些方面制约系统性能的表现，或者，哪些系统因素导致系统无法跟上业务发展？确定测试领域，然后具体问题具体分析。

2、用户场景剖析和业务建模 根据对系统业务、用户活跃时间、访问频率、场景交互等各方面的分析，整理一个业务场景表，当然其中最好对用户操作场景、步骤进行详细的描述，为测试脚本开发提供依据。

3、确定性能目标 前面已经确定了本次性能测试的应用领域，接下来就是针对具体的领域关注点，确定性能目标（指标）；其中需要和其他业务部门进行沟通协商，以及结合当前系统的响应时间等数据，确定最终我们需要达到的响应时间和系统资源使用率等目标；比如：

- ①登录请求到登录成功的页面响应时间不能超过2秒；
- ②报表审核提交的页面响应时间不能超过5秒；
- ③文件的上传、下载页面响应时间不超过8秒；
- ④服务器的CPU平均使用率小于70%，内存使用率小于75%；
- ⑤各个业务系统的响应时间和服务器资源使用情况在不同测试环境下，各指标随负载变化的情况等；
- 4、制定测试计划的实施时间 预设本次性能测试各子模块的起止时间，产出，参与人员等等。

三、测试脚本设计与开发 性能测试中，测试脚本设计与开发占据了很大的时间比重。

1、测试环境设计 本次性能测试的目标是需要验证系统在实际运行环境中的性能外，还需要考虑到不同的硬件配置是否是制约系统性能的重要因素！因此在测试环境中，需要部署多个不同的测试环境，

在不同的硬件配置上检查应用系统的性能，并对不同配置下系统的测试结果进行分析，得出最优结果（最适合当前系统的配置）。

这里所说的配置大概是如下几类：

- ①数据库服务器
- ②应用服务器
- ③负载模拟器
- ④软件运行环境，平台 测试环境测试数据，可以根据系统的运行预期来确定，比如需要测试的业务场景，数据多久执行一次备份转移，该业务场景涉及哪些表，每次操作数据怎样写入，写入几条，需要多少的测试数据来使得测试环境的数据保持一致性等等。可以在首次测试数据生成时，将其导出到本地保存，在每次测试开始前导入数据，保持一致性。

2、测试场景设计

通过和业务部门沟通以及以往用户操作习惯，确定用户操作习惯模式，以及不同的场景用户数量，操作次数，确定测试指标，以及性能监控等。

3、测试用例设计 确认测试场景后，在系统已有的操作描述上，进一步完善为可映射为脚本的测试用例描述，用例大概内容如下：

用例编号：查询表单_xxx_x1（命名以业务操作场景为主，简洁易懂即可） 用例条件：用户已登录、具有对应权限等。。。

操作步骤：

- ①进入对应页面。。。。。
- ②查询相关数据。。。。。
- ③勾选导出数据。。。。。
- ④修改上传数据。。。。。

PS：这里的操作步骤只是个例子，具体以系统业务场景描述；

4、脚本和辅助工具的开发及使用 按照用例描述，可利用工具进行录制，然后在录制的脚本中进行修改；比如参数化、关联、检查点等等，最后的结果使得测试脚本可用，能达到测试要求即可；

PS：个人而言，建议尽量自己写脚本来实现业务操作场景，这样对个人技能提升较大；一句话：能写就 绝不录制！！

四、测试执行与管理 在这个阶段，只需要按照之前已经设计好的业务场景、环境和测试用例脚本，部署环境，执行测试并记录结果即可。

1、建立测试环境 按照之前已经设计好的测试环境，部署对应的环境，由运维或开发人员进行部署，检查，并仔细调整，同时保持测试环境的干净和稳定，不受外来因素影响。

2、执行测试脚本 这一点比较简单，在已部署好的测试环境中，按照业务场景和编号，按顺序执行我们已经设计好的测试脚本。

3、测试结果记录 根据测试采用的工具不同，结果的记录也有不同的形式；现在大多的性能测试工具都提供比较完整的界面图形化的测试结果，当然，对于服务器的资源使用等情况，可以利用一些计数器或第三方监控工具来对其进行记录，执行完测试后，对结果进行整理分析。

五、测试分析

- 1、测试环境的系统性能分析 根据我们之前记录得到的测试结果（图表、曲线等），经过计算，与预定的性能指标进行对比，确定是否达到了我们需要的结果；如未达到，查看具体的瓶颈点，然后根据瓶颈点的具体数据，进行具体情况具体分析（影响性能的因素很多，这一点，可以根据经验和数据表现来判断分析）。
- 2、硬件设备对系统性能表现的影响分析 由于之前设计了几个不同的测试环境，故可以根据不同测试环境的硬件资源使用状况图进行分析，确定瓶颈是再数据库服务器、应用服务器抑或其他方面，然后针对性的进行优化等操作。
- 3、其他影响因素分析 影响系统性能的因素很多，可以从用户能感受到的场景分析，哪里比较慢，哪里速度尚可，这里可以根据258原则对其进行分析；至于其他诸如网络带宽、操作动作、存储池、线程实现、服务器处理机制等一系列的影响因素，具体问题具体分析，这里就不一一表述了。
- 4、测试中发现的问题 在性能测试执行过程中，可能会发现某些功能上的不足或存在的缺陷，以及需要优化的地方，这也是执行多次测试的优点。

你们项目的性能测试结果是什么样的，具体的吞吐量，响应时间，错误率

我们是普通制造业的项目，性能需求没有太高，响应时间是在4s左右，每秒能够处理50笔事务(tps是50)，错误率我们公司要求的是不超过千分之六，也就是成功率不得低于99.4，在实际测试的时候，错误率是99.5%。算是达标。

我们是互联网行业，测试的时候，响应时间在500ms左右，用户体验还是很不错的，然后每秒能够处理大约800tps的业务量。错误率我们公司要求的是不超过千分之六，也就是成功率不得低于99.4，在实际测试的时 候，错误率是99.5%。算是达标。

TPS:

参考标准: 金融行业：1000TPS~50000TPS，不包括互联网化的活动 保险行业：100TPS~100000TPS，不包括互联网化的活动 制造行业：10TPS~5000TPS 互联网电子商务：10000TPS~1000000TPS 互联网中型网站：1000TPS~50000TPS

互联网小型网站: 500TPS~10000TPS

响应时间: 互联网企业：500毫秒以下，例如淘宝业务10毫秒左右。 金融企业：1秒以下为佳，部分复杂业务3秒以下。

保险企业：3秒以下为佳。

制造业：5秒以下为佳。

如何理解压力、负载、性能测试？

性能测试是一个较大的范围，实际上性能测试本身包含了性能、强度、压力、负载等多方面的测试内容。

压力测试是对服务器的稳定性以及负载能力等方面的测试，是一种很平常的测试。增大访问系统的用户数量、或者几个用户进行大数据量操作都是压力测试。而负载测试是压力相对较大的测试，主要是测试系统在一种或者集中极限条件下的相应能力，是性能测试的重要部分。

100个用户对系统进行连续半个小时的访问可以看作压力测试，那么连续访问8个小时就可以认为负载测试，

1000个用户连续访问系统1个小时也可以看作是负载测试。实际上压力测试和负载测试没有明显的区分。测试人员应该站在关注整体性能的高度上来对系统进行测试。

JMeter为什么要参数化

第一点:多用户登录的时候，如果不进行参数化，就没法演示了，需要使用CSV将参数放到文件，来演示多用户登录

第二点:在进行录制的时候，有可能存在第二个请求的参数是从第一个请求中获取出来的，需要在第一个请求下，去将参数提出取来，再在第二个请求中进行参数化

你用什么机器对服务器进行压力测试

按照规范的话，需要使用一台性能比较好服务器来对服务器进行压力测试。在Linux系统下搭建测试环境，然后进行测试。可以说使用的JMeter进行的测试，前期需要搭建的环境包括Java MySQL 等环境如果我们测试的并发量比较大，单台机器没法满足，可以进行分布式压力测试。通过主从机方式部署分布式测试环境。

使用JMeter，在从机上，设置端口，启动JMeter-server服务，在主机上，配置从机ip，端口，然后调用远程从机，进行启动压力测试就可以。

吞吐量、QPS、并发数、响应时间（RT）概念

QPS 每秒查询率，因特网上，经常用每秒查询率来衡量域名系统服务器的机器的性能，其即为QPS。

对应请求数/sec，即每秒的响应请求数，也即是最大吞吐能力。响应时间(RT)

响应时间是指系统对请求作出响应的的时间 吞吐量 吞吐量是指对网络、设备、端口、虚电路或其他设施，单位时间内成功地传送数据的数量（以比特、字节、分组等测量）

服务端和客户端的性能分析从哪些角度来进行

服务端

- 1. 数值说明 测试完成的总事务数 平均请求响应时间 统计意义上的平均响应时间 除特殊情况之外的最大响应时间最短响应时间最大响应时间吞吐量，和ab的每秒处理请求数相同 流量，权衡
- 2.测试并发性能
- 3.测试获得结果分析
 - a)整个场景中的网络传输量
 - b) Request per second：每秒处理的请求数，即每秒事务数（TPS），一般来说100~200是 比较理想的 范围
 - c) Time per request：每个请求所花的时间，即平均事务时间。此数值一般有两行，一般关注后一行的数值，也就是计算请求平均响应的时间。
 - d) Transfer rate：平均每秒的网络流量，此数据可以帮助排除是否存在网络流量过大导致响应时间延长 的问题。

服务端性能测试的几个注意事项：

- a) 性能测试最好在本地进行，至少要保证服务器和测试机都在内网中，这样才能排除网络的干扰，更准确的 测出系统本身的问题。
- b) 必须根据服务端应用的实际情况选用合适的输入参数，这样可以预估出和目标性能相似的测试。

客户端 稳定性测试的三个要点：

- a) 应用的运行实际要尽可能的长，
- b) 保持运行时是多线程运行状态
- c) 尽可能使用多的机型或者操作系统进行测试

性能测试关注哪些指标

从外部看，性能测试主要关注如下三个指标 吞吐量：每秒钟系统能够处理的请求数、任务数。 响应时间：服务处理一个请求或一个任务的耗时。错误率：一批请求中结果出错的请求所占比例。从服务器的角度看，性能测试主要关注CPU、内存、服务器负载、网络、磁盘IO等

你这个项目做性能测试的时候，并发量设置多大？具体的数据访问量，在线用户数量

一般情况下，大型系统（业务量大、机器多）做压力测试，10000~50000个用户并发，中小型系统做压力测试，1000-5000个用户并发比较常见。

参考回答: 我们公司设置的并发量，在1000用户，或者2000用户。在线用户数量在峰值的时候，我们最高是1w用户。

参数化具体用在哪些路径上

凡是请求，基本都要带参数，带参数，就有可能进行参数化。

购物车:带参数，买的什么商品，买了几个。搜索，关键词，排序方式，类别，热度，价格区间 分类:参数信息参数化

性能测试中设置了哪些公共参数

公共参数实际上就是接口中的一些公共参数,我上家公司接口中的公共参数有appVersion版本号,deviceName设备名称,source,authSign签名信息等这样的字段 所以在性能测试的过程中要把这些参数进行设置

服务器Linux上能不能直接进行性能测试

不能，脚步需要通过windows调试好，然后放到linux上运行。在linux上运行的时候，只能通过non GUI形式进行启动JMeter，然后需要注意的是，csv文件在windows上和Linux上要统一路径，最好使用相对路径，放到统一目录下边。

性能测试如何保障JMeter 同时有100个用户在线

可以在JMeter中使用同步定时器功能，同步定时器可以设置集合点，比如设置100，这样当达到100并发的时候，才会执行相应的接口请求

性能测试指标

用户数

1注册用户数

注册用户数指软件中已经注册的用户，这些用户是系统的潜在用户，随时都有可能上线。这个指标的 意义在于让测试工程师了解系统数据中的数据总量和系统最大可能有多少用户同时在线。

2在线用户数

在线用户数是指某一时刻已经登录系统的用户数量。在线用户数只是统计了登录系统的用户数量，这些用户不一定都对系统进行操作，对服务器产生压力。

3并发用户数 不同于在线用户数，并发用户数是指某一时刻向服务器发送请求的在线用户数，他是衡量服务器并发

容量和同步协调能力的重要指标，从这个含义上讲，我们可能会如下两种理解：

同一时刻向服务器发送相同或者不同请求的用户数，也就是说，既可以包括对某一业务的相同请求，也可以 包括对多个业务的不同请求

同一时刻向服务器发送相同请求的用户数，仅限于某一业务的相同请求

事务的响应时间

事务是指用户在客户端做一种或多种业务所做的操作集，事务的响应时间就是衡量用户执行这些操作集所花费的时间。在性能测试中，一般通过计算事务的开始时间和结束时间的差值来获取事务的响应时间。一个事务表示一个“从用户发送请求->web server接受到请求，进行处理-> web server向DB获取数据-> 生成用户的object(页面)，返回给用户”的过程，一般的响应时间都是针对事务而言的。

每秒点击数

每秒点击数是指每秒钟像web服务器提交的HTTP请求数，它是衡量服务器处理能力的一个常用指标。需要注意的是，这里的相应时间并非鼠标的一次单击操作，因为在一次单击操作中，客户端可能向服务器发出多个HTTP请求，切勿混淆

吞吐量

吞吐量通常指单位时间内从服务器返回的字节数，也可以单位时间内客户提交的请求数。吞吐率是大型web系统衡量自身负载能力的一个重要指标，一般来说，吞吐率越大，单位时间内处理的数据就越多，系统的负载能力也强。吞吐率与很多因素有关，服务器的硬件配置，网络的宽带及拓扑结构，软件的技术架构等。

业务成功率

指多用户对某一业务发起操作的成功率。例如，测试网络订票系统的并发处理性能，在早上8:00——8:30半小时的高峰里，要求能支持10万比订票业务，其中成功率不少于98%。也就是说系统允许200笔订票业务超时或者因其他原因导致未能订票成功

TPS

TPS表示服务器每秒处理的事务数，他是衡量系统处理能力的一个非常重要的指标，在性能测试中，通过检测不同用户的TPS,可以估算出系统处理能力的拐点。

资源利用率

资源利用率就是指资源的使用情况

CPU使用率70%—80%，内存使用率80%以下 网络带宽利用率100Mbps=12.5MB/s

计算并发用户数

根据PV计算公式

比如一个网站，每天的PV大概1000w，根据2/8原则，我们可以认为这1000w pv的80%是在一天的9个小时内完成的（人的精力有限），那么TPS为：

$1000w \times 80\% / (9 \times 3600) = 246.92 \text{个/s}$,取经验因子3，则并发量应为：246.923=740

根据系统用户数计算

如果是新系统，就要根据多种因素考量，相关的说法有：

1、在线用户或者终端数：即同时在使用应用系统的用户，可能在浏览，可能在做交易。如果是知道了，一般并发用户数取在线用户的10%-30%。（思考时间适当短或者不用，不要使用回放录制的思考时间）

2、交易总数和预期响应时间：如果你知道了每天（8小时）要完成20000笔交易，每笔交易希望在5秒内，那么可以预估并发用户数为 $20000 / (8 \times 3600) \times 5$ ，当然这个并发数还要根据你实际测出的响应时间再调整

3、八二原则：一般可以认为80%的用户在20%的时间内完成工作，所以峰值压力的时候，一般并发数要乘以 $80\% / 20\% = 4$

4、如果你的系统终端数量是固定的，比如就有100个客户端，那么极限压力就是这100个客户端都疯狂工作，所以峰值并发数也就是100，去掉脚本中的思考时间。

JMeter之前用的时什么版本，如何安装的

JMeter用的5.2.27版本，安装如下

现在电脑上安装jdk1.8，然后从官网上下载最新的安装包，进行环境变量配置，配置好后完成即可

JMeter描述

一般情况下，我们提压力测试，通常指的是指负载测试和压力测试。

我们做压力测试,基本上会使用到工具进行测试,我常用的工具,一个是JMeter,另外一个loadRunner。我先介绍一下JMeter吧,JMeter是Apache组织开发的基于java的压力测试工具,支持接口测试,压力测试,还可以做录制回放操作,操作比较简便。

在项目中如何使用JMeter进行接口测试

- 1)把线程组数量设置为1,循环次数设置为1
- 2)配置好全局变量url 通过配置元件--用户自定义的变量添加
- 3)增加配置元件http请求默认值,放置在用户定义的变量之后
- 4)添加事务控制器管理和组织测试用例
- 5)在事务控制中添加http请求添加测试用例中的接口请求信息
- 6)添加对应的断言元件进行断言

如何使用JMeter进行测试

我先说一下JMeter的操作的整体流程吧,我们测试的时候,通常是创建一个线程组,指定并发的线程数量,然后指定要测试的接口,创建相应的监听器,比如表格结果,结果树和聚合报告信息,通过监听器来监听测试是否通过或者接口是否存在什么问题

其中在结果树中可以监测到整体的请求信息,就拿Http请求这种来讲,其实就是整个http协议的所有信息,包括请求头,请求参数,请求路径,还有响应头,响应结果等信息。对于表格查看结果,可以看到每个请求的简单信息,本次请求的时间,以及平均的时间。在聚合报告中,我们就可以看到整体的信息了。比如可以看到平均响应时间,90%Line也就是90%的用户请求低于的时间。还有吞吐量TPS,还有错误率,还有用流量来计算的吞吐量。这些都可以看到。通常,聚合报告就是反应整体的数据。

JMeter参数化

做压力测试时,我们经常需要替换参数,在JMeter中,有多种参数化的形式。可以在测试计划中设置全局参数,可以设置用户参数,还可以在前置处理器中设置用户参数。在进行多线程并发的时候,如果需要多个参数,可以使用csv配置元件。比如做登陆操作,后台有可能会限制一个用户不能重复登陆多次,如果演示登陆的并发操作,可以使用JMeter中csv配置元件,将用户信息导出来,放到文件中,就可以让线程共享这些数据。另外,对于一些随机变化的参数,可以使用JMeter中的函数助手,生成随机函数,进行参数化测试。比如注册这样的操作,用户名要求唯一的,那就可以使用随机函数来模拟出来。

JMeter参数化的方式有几种可以实现

配置元件--用户定义的变量元件可以设置全局变量

函数助手对话框中可以选择比如随机字符串、随机日期、随机数字作为参数化 可以使用csv文件作为参数化,通过配置元件中的csv data set config元件进行设置即可

JMeter断言--检查点

在测试中,断言操作经常用到,JMeter的断言操作可以在请求后边设置一个断言结果,判断响应结果或者是响应状态码是否和预期的一致。

JMeter中如何设置断言

右键请求--断言--响应断言--响应断言界面输入要检查对比的项,设置好断言后,执行接口测试如果是通过的,查看结果树不会有任何提示,如果断言失败,就会有红色报错,如果接口返回的数据是json数据,也可以添加json断言

JMeter如何实现关联

先从一个接口通过正则表达式提取器或jsonpath解析器截取下一个接口需要的参数保存到变量,然后写一个接口中通过\${变量}去获取

JMeter逻辑控制器

在JMeter中,逻辑控制器的应用比较广泛,我们可以把一组操作放到简答控制器中,用来声明范围。也可以设置循环控制器,比如我们注册1次,登陆10次,就可以使用循环控制器。还有事务控制器,对于一个完整的事务,可以使用事务控制器来控制。

JMeter定时器

在JMeter中定时器类型也比较多,我们会经常用到固定定时器,可以设置启动线程组的延迟时间是固定的。还有高斯定时器,这种时间是可以设置随机值。另外,还有同步定时器,同步定时器可以用来做集合点,比如设置某个线程数,等并发到了该线程值的时候,才开始执行并发任务。

JMeter后置处理器

对于JMeter后置处理器,我们常用到的是正则表达式提取器,可以用来提取上一个请求的响应结果,用在下一个请求中。比如项目中会有条目展示,如果我们想要继续查看详情,就需要提取条目id,用来做下一个请求。这里还可以用到foreach循环控制器来完成。

JMeter插件

对于JMeter插件,常用的有场景设计插件,这个还是非常不错的一个功能。我们创建线程组的时候,会设置线程组,而线程组太死板,不灵活。可以安装场景控制的插件,比如一开始延迟启动多少线程,如何递增启动线程,线程总数多少,总的测试时间多长,都可以使用场景控制插件来完成。

对于结果监听,结果树,表格,聚合报告都不够直观,我们可以通过安装图表插件来进行结果收集。看起来更加直观一些。比如可以获取到每秒事务量的图表,获取到平均响应时间的图表,都比较人性化。这里有一点需要注意,就是监测服务器数据的时候,需要在服务器上启动一个插件,serverAgent,才能看到服务器的cpu和内存等数据。

JMeter脚本录制

对于JMeter脚本录制,我掌握的有两种方式,第一,可以使用JMeter内置的http代理服务器录制脚本,这种配置方便,但是会录制无用信息。不过好处是可以录制手机端的请求。也可以使用第三方工具,badboy录制,这种录制相对人性化一些,也会过滤掉一些请求,比如jpg,css等。

JMeter数据库压力测试

有些时候,还会用到JMeter去压测数据库,当然我们不会随便压测数据库。一般是当我们压测接口的时候,发现某个接口性能比较差,再进一步判定问题的时候,会压测数据库。

压测数据库,需要配置驱动,需要设置连接池大小,需要使用sql去操作数据库。如果我们想要看具体的哪条sql问题的话,还需要从开发那里拿到具体的sql进行压测。

NON GUI形式测试

所谓non GUI,就是我们不需要通过页面进行测试,这个也是我们推荐的测试方式,会提高测试机性能。或者在linux上,我们也没法打开图形化页面,就可以使用NON GUI形式的指定进行。可以指定脚本,可以指定生成jtl文件的位置,还可以生成html报告文档。

JMeter -n -t xxx.jmx(脚本文件)-l xx.jtl(生成结果文件)-e -o xxx.html

JMeter分布式压力测试

在进行压测的时候，如果我们进行大并发的压测，往往一台测试机是达不到这么高的并发量的。那就可以使用 分布式压测方式，设置一台控制器，然后几台压力机，咱们通常叫“肉鸡”，进行压测，这种压测也并不复杂，只需要做一些简单配置。

比如在肉鸡身上配置连接的端口，配置启动JMeter-server，就可以被连接测试了。在控制机master上我们，关联上肉鸡的ip和端口，就可以控制肉鸡进行分布式压力测试了

JMeter添加http请求默认值元件有什么作用

添加并设置好后，相当于给所有的http请求取样器都设置了默认值，即不用填写取样器中的比如主机地址、端口、代理等，都可以使用http请求默认值设置的

Loadrunner面试题

什么是负载测试?什么是性能测试?

负载测试是通过改变系统负载方式、增加负载等来发现系统中所存在的性能问题。负载测试是一种测试方法，可以为性能测试、压力测试所采用。负载测试的加载方式也有很多种，可以根据测试需要来选择。性能测试是为获取或验证系统性能指标而进行测试。多数情况下，性能测试会在不同负载情况下进行。压力测试通常是在高负载情况下来对系统的稳定性进行测试，更有效地发现系统稳定性的隐患和系统在负载峰值的条件下功能隐患等。

性能测试包含了哪些测试(至少举出3种)

压力测试、负载测试、并发测试、可靠测试、失效恢复测试。

简述性能测试的步骤

- Step 1. Identify the Test Environment
- Step 2. Identify Performance Acceptance Criteria
- Step 3. Plan and Design Tests
- Step 4. Configure the Test Environment
- Step 5. Implement the Test Design
- Step 6. Execute the Test
- Step 7. Analyze Results, Report, and Retest

简述使用Loadrunner的步骤

脚本录制设置—录制脚本—调试脚本—场景设置—结果分析

什么时候可以开始执行性能测试?

在产品相对比较文档，功能测试完成后。灵活性比较强。

loadrunner target="_blank">LoadRunner由哪些部件组成?

virtual user generator controller analysis

你使用LoadRunner的哪个部件来录制脚本?

virtual user generator

LoadRunner的哪个部件可以模拟多用户并发下回放脚本?

controller

什么是集合点?设置集合点有什么意义?Loadrunner中设置集合点的函数是哪个?

集合点：设置多个用户到达某个用户数量点集合，同时触发一个事务,以达到模拟真实环境下同时多个用户操作,同时模拟负载,实现性能测试的最终目的

LR_rendezvous(“集合点名称”)

什么是场景?场景的重要性有哪些?如何设置场景?

场景：模拟真实环境中，用户运行状况。

- 1.通过场景来模拟实际用户的操作，性能测试结果才具有代表性。
- 2.在运行过程中也需要关注场景性能测试值，测试过程是否正常。

- 1.选择场景中需要的脚本
- 2.选择为目标场景，还是指定的手工场景
- 3.设置用户数、设置产生负载的设备
- 4.设置执行策略

请解释一下如何录制web脚本?

LR通过转发请求，来捕获数据包，来形成脚本

为什么要创建参数?如何创建参数?

参数：在环境变化时必须脚本具有环境变化的能力，就需要参数化(客户端发送到服务器端)

- 1.确定要参数化的数据
- 2.设定规则形式来取值

什么是关联?请解释一下自动关联和手动关联的不同。

关联：很多构架用sessionid等方法标识不同任务和数据，应用在每次运行时方式发送数据不完全相同，需要利用的机制对录制的脚本进行处理，这种机制叫做关联(服务端发送到客户端)

你如何找出哪里需要关联?请给一些你所在项目的实例。

用户登陆，客户端发送请求后，服务端验证正确性后，发送给客户端sessionid，是某种规则产生。

你在哪里设置自动关联选项?

两地方可以设置

- 1.设置允许录制时进行自动关联，可以自定义规则
- 2.录制完成后，vuser-scan action for correlations

你在VuGen中何时选择关闭日志?何时选择标准和扩展日志?

Run-time、log、

当调试脚本时，可以只输出错误日志，当在场景找你管加载脚本时，日志自动变为不可用。

Standard Log Option：选择标准日志时，就会在脚本执行过程中，生成函数的标准日志并且输出信息，供调试用。大型负载测试场景不用启用这个选项。扩展日志包括警告和其他信息。大型负载测试不要启用该选项。用扩展日志选项，可以指定哪些附加信息需要加到扩展日志中

你如何调试LoadRunner脚本？

VuGen有两个选项帮助调试Vuser脚本。Run Step by Step 命令和断点(breakpoints)。Option对话框中的调试设置(Debugsetting)项，可以确定在场景执行过程中执行轨迹范围。调试信息写在output窗口。可以用 lr_set_debug_message函数在脚本中手工设置信息类型。如果我们只想接收到一小段脚本的调试信息。

你在LR中如何编写自定义函数?请给出一些你在以前进行的项目中编写的函数。

在创建用户自定义函数前我们需要和创建DLL(external library)。把库放在VuGen bin 目录下。一旦加了库，把自定义函数分配做一个参数。该函数应该具有一下格式： declspec (dllexport) char* (char*, char*)。

什么是逐步递增?你如何来设置?

Ramp up这个选项用于逐渐增加[服务器]

(http://www.ltesting.net/ceshi/ruanjianceshikaifajishu/rjcsjh)的虚拟用户数或负载量。设置一个初始值而且可以在两个迭代之间设置一个值等待。设置Ramp up，请到‘Scenario Scheduling Options’。

以线程方式运行的虚拟用户有哪些优点?

VuGen提供了用多线程的便利。这使得在每个生成器上可以跑更多的虚拟用户。如果是以进程的方式跑虚拟用户，为每个用户加载相同的驱动程序到内存中，因此占用了大量的内存。这就限制了在单个生成器上能跑的虚拟用户数。如果按进程运行，给定的所有虚拟用户数(比如100)只是加载一个驱动程序实例到内存里。每个进程共用父驱动程序的内存，因此在每个生成器上可以跑更多的虚拟用户。

当你需要在出错时停止执行脚本，你怎么做?

lr_abort函数放弃虚拟用户脚本的执行。说明虚拟用户停止Action的执行，直接执行vuser_end 然后结束 执行。在出现错误情况下想手工放弃脚本的执行，这个函数是有用的。用这个函数停止脚本时，Vuser被指定 为“Stopped”状态。为了这个函数起作用，开始时候就不能选择Run-Time Settings中的Continue on error选项

响应时间和吞吐量之间的关系是什么?

吞吐量图显示的是虚拟用户每秒钟从服务器接收到的字节数。当和响应时间比较时，可以发现随着吞吐量的降低，响应时间也降低，同样的，吞吐量的峰值和最大响应时间差不多在同时出现。

说明一下如何在LR中配置系统计数器?

通过We资源监视器，利用这些监控器可以分析web服务器的吞吐量、点击率、每秒http响应数以及每秒下载的页面数

什么是think time?think_time有什么用?

思考时间是真实用户在action之间等待的时间。例如：当一个用户从服务器接收到数据时，用户可能需要在响应之前等待几分钟回顾数据，这种推迟被称为思考时间。

标准日志和扩展日志的区别是什么?

Standard Log Option：选择标准日志时，就会在脚本执行过程中，生成函数的标准日志并且输出信息，供调试用。大型负载测试场景不用启用这个选项。扩展日志包括警告和其他信息。大型负载测试不要启用该选项。用扩展日志选项，可以指定哪些附加信息需要加到扩展日志中

解释以下函数及他们的不同之处。

Lr_debug_message

lr_debug_message 函数在指定的消息级别

//处于活动状态时发送一条调试消息。如果指定的

//消息级别未处于活动状态，则不发出消息。 Lr_output_message 要发送不是特定错误消息的特殊通知， Lr_error_message

函数将错误消息发送到

//输出窗口和 Vuser 日志文件

Lrd_stmt lrd_exec 函数执行 lrd_stmt 设置的SQL语句。

Lrd_fetch 函数从结果集中提取后续若干行

result set.

函数准备用于

//通过光标输出字符串(通常为 SQL 语句)

//的下一结果集。对于 CtLib，它发出 ct_result

//命令，并且在 ODBC 中它运行用于当前数据库

什么是吞吐量?

单位时间内系统处理客户端的请求数

场景设置有哪几种方法?

目标场景，手工场景