

# 性能测试工具 LoadRunner 的工作机理 及关键技术研究

桑圣洪 胡 飞

(西北工业大学, 西安 710072)

**摘 要** 对软件性能测试的方法进行分析, 通过对大型软件测试工具 LoadRunner 三个测试组件的体系结构和工作机制进行研究, 详细介绍了软件性能测试的技术与方法, 对于编写软件性能测试工具有良较好的借鉴作用。

**关键词** 性能测试 负载测试 关联 响应时间

**中图分类号** TP315 **文献标识码** B

目前 IT 系统已经成为各个企业运转业务时最重要的支撑系统之一。IT 系统发展中最明显的特征之一就是所谓的“数据大集中”, 即数据越来越集中到后台的服务器中, 系统同时为成百上千, 乃至上万的用户提供服务。这样的例子在银行、保险、电信公司中随处可见。随着企业业务量的加大, 其 IT 系统承载的负荷越来越重, IT 系统性能的好坏直接影响到企业对外提供服务的质量。因此, 对 IT 系统的性能进行测试和调优就成为企业提高效率的重要因素。

目前, 典型的企业 IT 系统的架构如图 1 所示。



图 1 典型企业 IT 系统架构

系统由客户端、网络、防火墙、负载均衡器、Web 服务器、应用服务器（中间件）、数据库服务器等环节组成。根据木桶原理, 即木桶所能装的水量取决于最短的那块木板, 整个系统的性能要得到提高, 每个环节的性能都需要优化。在这样的 IT 系统中, 每个环节都是一个复杂的子系统, 对其性能调优都是一门专门的技能。Oracle 数据库的调优就

需要专家有专门的技能和多年的经验。对于整个 IT 系统的调优, 其复杂程度更是急剧增加。因此 IT 系统性能测试调优是一个复杂的项目, 需要拥有各种专门技能的专家来完成。这些专家包括: 操作系统专家、网络专家、数据库专家、应用服务器专家、应用软件和业务专家等。

## 1 软件测试中的性能测试

软件测试是保证软件质量的重要手段, 是软件开发过程中一个必不可少的环节。而性能测试则属于软件测试中的系统级测试, 它针对软件在集成系统中运行的性能指标进行测试, 旨在及早确定和消除软件中与系统构架有关性能瓶颈。

### 1.1 性能测试的含义

性能测试一般地, 主要是针对系统的性能指标制定性能测试方案、执行测试用例、得出测试结果来验证系统的性能指标是否满足既定值。性能指标里可能包括系统各个方面的能力, 如系统并发处理能力、批量业务处理能力等。

### 1.2 性能测试的分解

在性能测试的执行中, 可以根据具体的性能指标, 分解为多种子测试, 根据其关系, 可以在不同的时间和空间内执行。这些子测试通常包括以下几种:

### 1.2.1 压力测试

压力测试是指逐步增加系统负载,测试系统在不同负载下的运行能力,并确定在什么负载条件下系统性能处于失效的临界状态,并以此来获得系统所能提供的最大服务级别。通俗地讲,压力测试是为了发现系统运行的上限和下限支持条件。

### 1.2.2 负载测试

验证系统的负载工作能力。是通过逐步增加系统负载测试系统性能的变化,并最终确定在满足性能指标的情况下,系统所能承受的最大负载量的测试。

### 1.2.3 并发测试

并发性能测试是评估系统交易或业务在渐增式并发情况下处理瓶颈,以及能够接收业务的性能过程。即逐渐增加并发用户数负载,直到系统的瓶颈或者不能接收的性能点,通过综合分析交易指标、资源监控指标等来确定并发性能的过程。并发性能测试是负载压力测试中的重要内容。

### 1.2.4 配置测试

核实在操作条件保持不变的情况下,系统在使用不同配置时,系统的性能指标。

### 1.2.5 健壮性测试

健壮性是指在异常情况下,软件能够正常运行的能力,这里的异常指的是资源过少、用户数过多、突发故障等。

目前,市场上有很多软件性能测试工具,从功能简单、单一的开放源码的软件测试工具,到价格昂贵的商品化性能测试工具,种类繁多、价格差别很大。本文论述了性能测试工具 LoadRunner的一般体系架构和技术要点。

## 2 LoadRunner的工作原理研究

### 2.1 性能测试原理

性能测试的主要手段是通过产生模拟真实业务的数据或动作,对被测系统进行加压,研究被测系统在不同压力情况下的表现,找出其潜在的瓶颈问题。LoadRunner通过三个组件来满足进行压力

测试的手段:

- (1)提供产生压力数据的手段;
- (2)能够对后台系统进行监控;
- (3)对压力数据能够进行分析,快速找出被测系统的瓶颈问题。

### 2.2 测试工具 LoadRunner体系结构

LoadRunner的体系结构如图2所示。

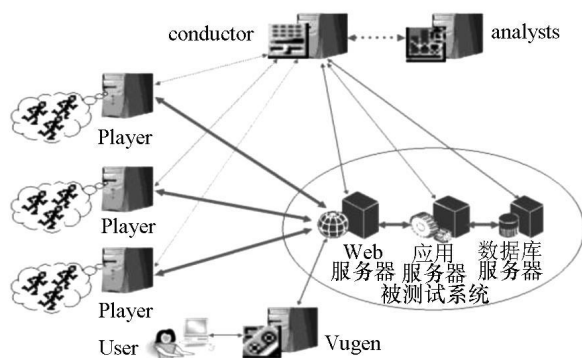


图2 LoadRunner体系结构

其中包括三个部分:

- (1)提供产生压力脚本的 Vugen;
- (2)提供运行脚本并对后台进行监控的 Controller 即图2中所示的 Conductor;
- (3)对压力数据进行分析的 Analysis

### 2.3 LoadRunner的工作原理

LoadRunner三个组件的工作机理及关键技术:

#### 2.3.1 虚拟用户产生器 Vugen

虚拟用户产生器通过录制客户端和后台服务器之间的通讯包,分析其中的协议,自动产生脚本。用户在自动产生的脚本的基础上进行修改,从而快速开发出一个逻辑功能和客户端软件完全一样的压力脚本程序。但是在脚本的回放过程中会遇到一些问题,如:用户 SessionId 的动态数据的过期、重复数据的插入、Ip 地址的限制等,LoadRunner 通过以下技术来解决脚本运行过程中的问题,具体包括:

a. 参数的关联:把服务器返回的动态数据,通过函数 `web_reg_save_param()` 保存到一个参数中如: `web_reg_save_param("CSRule_1_UD2", "LB=userSession value=", "RB=>", "Ord=1", "Search=Body", "RefFrameId=1", LAST)`,这个函数把服

务器产生的动态数据 Session value 保存到参数 CSRule\_1\_UID2中, 在以后需要用到该动态数据时可以直接引用该参数, 格式如下:

```
web_submit_data( "login.pf",
    "Action=http://localhost:1080/mercuryWebTours/login.pf",
    "Method=POST",
    "RedContenttype=text/html",
    "Referer=http://localhost:1080/mercuryWebTours/nav.pf?in=home",
    "Snapshot=ttl.inf",
    "Mode=HTTP",
    ITEM DATA,
```

“Name=userSession”, “Value={CSRule\_1\_UID2}”, END ITEM ), 这样在脚本回放时 useSession就用参数 CSRule\_1\_UID2来代替, 不会出现Session过期的问题。

b. 数据的参数化: 录制业务流程时, Vugen 生成一个脚本来包含录制期间所用到的实际值。假设, 您要使用不同于录制内容的值来执行该脚本的操作(查询、提交等)。要实现这一点, 可以使用参数替换已录制的值, 这被称为脚本参数化。通过脚本参数化可以避免重复数据的提交和模拟真实数据的场景等。数据的参数化可以通过数据文件, 在数据文件中通过定义列值和行值, 通过取值策略的不同来分配每次迭代, 或者每个虚拟用户取值参数化的参数值; 还可以定义查询语句, 从数据库中提取数据进行数据的参数化; 另外, 还可通过分配内部数据, 包括日期/时间、组名、迭代编号、负载生成器名、随机编号、唯一编号和 Vuser ID 以及调用用户定义的函数即 DLL函数生成的数据等。

### 2.3.2 负载生成控制器 Controller

在 LoadRunner的体系结构中, Player是负载生成器, 而 Controller在整个体系结构中起到了“指挥家”的角色, 即控制 Player的运行。Player如何运行脚本, 由 Controller来决定, Conductor和 Player实际上是一套框架程序。具体执行什么功能, 是由脚本来完成的。Conductor和 Player的体系结构如图 3 所示。Conductor上面有若干进程/线程, 每种进程的作用如下:

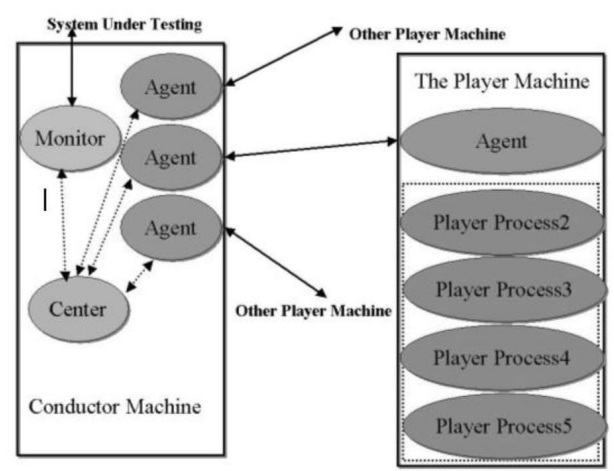


图 3 Conductor和 Player的体系结构

· Center进程是整个调度的核心进程, 它负责联系和用户界面交互的工作。

· Agent进程负责和远端的 Player 机器中对应的 Agent进程通讯。负责把编译好的脚本传送到 Player机器上。在脚本运行的时候, 定期从 Player机器上获取 Player的运行状态, 保存每个虚拟用户运行的日志。

· Monitor进程负责对被测试系统的各个环节进行监控, 并把监控的内容写入 Conductor机器的本地磁盘; 同时, 把监控的内容传送给 Center进程, 实时地显示在用户界面上。

Player的进程有两种, 一个是 Agent进程, 一个是 Player进程。Agent负责和 Conductor机器通讯, 它根据 Conductor的指示, 在本机器上派生出指定数目的 Player进程, 这些 Player进程负责具体执行相应的脚本。Player进程个数就是虚拟用户的个数。

Player需要解决的一个问题是 IP 问题。为了防止黑客的攻击, 某些后台的负载均衡设备一旦发现来自某一个 IP 的请求特别频繁时, 就会拒绝为该 IP 提供服务。这个功能的后果是 Player无法把真正的压力加到后台系统中。LoadRunner的解决方法就是在 Player机器上, 伪装多个 IP 地址进行发送请求。这项技术称为 IP 欺骗 (IP Spoofing), LoadRunner的多 IP 地址功能可以使用许多 IP 地址, 来标识一台计算机上运行的多个 Vuser 这样, 服务

器和路由器认为 Vuser来自不同的计算机,测试环境会更加真实。

### 2.3.3 进行数据进行分析的 Analysis

此组件其实是一个数据处理组件,通过对 Controller收集的大量数据进行综合分析,以图的形式展现给用户。用户通过分析各种参数指标,如 Web应用的页面吞吐量、每秒点击次数和事务平均响应时间,系统资源如 CPU利用率、内存利用率、磁盘 I/O,中间件服务器的各性能参数指标、Weblogic的 JVM内存使用情况、事务的响应时间、EJB的使用情况、数据库如 Oracle的高速缓存命中率、Sql语句的执行情况、锁资源的争用等,来确定系统的性能瓶颈问题。

## 3 结束语

本文通过对系统性能测试工具 LoadRunner的体系结构和工作机理进行研究,分别对 LoadRunner

的三个组件如何进行工作、以及部分关键技术进行了分析解释,可以看出 LoadRunner的设计具有代表性,目前市场上无论是开源,还是大型的商业测试软件的设计原理,也类似于 LoadRunner。所以,本文通过对 LoadRunner的工作原理的分析和介绍,将对设计测试软件的开发人员具有较好的借鉴意义。

### 参 考 文 献

- 1 柳纯录. 软件评测师教程. 北京:清华大学出版社, 2002
- 2 Zadrozny P. J2EE performance testing with BEA weblogic Server 北京:电子工业出版社, 2003
- 3 Pressman R S. 软件工程实践者的研究. 北京:机械工业出版社, 2001
- 4 Niemiec R J. Oracle9i performance tuning tips & techniques 北京:清华大学出版社, 2005
- 5 王玉亭. 性能测试. 软件测试杂志, 2004 (2): 26—29
- 6 陈绍英, 夏海涛, 等. Web性能测试实战. 北京:电子工业出版社, 2002

## Work Mechanism and Key Technique Study of Performance Test Tool Load Runner

SAN Sen hong HU Fei

(School of Software and Microelectronics, Northwestern Polytechnical University, Xi'an 710065, P. R. China)

**[Abstract]** With analysis of performance test methods, research of system structure and work mechanism of large test tool LoadRunner's three components, as well as description and analysis of its key technique, its useful and directive to design performance test tool are presented.

**[Key words]** performance test load test correlation response time