

LoadRunner工具使用培训

陈柳村

中科软科技股份有限公司

目录

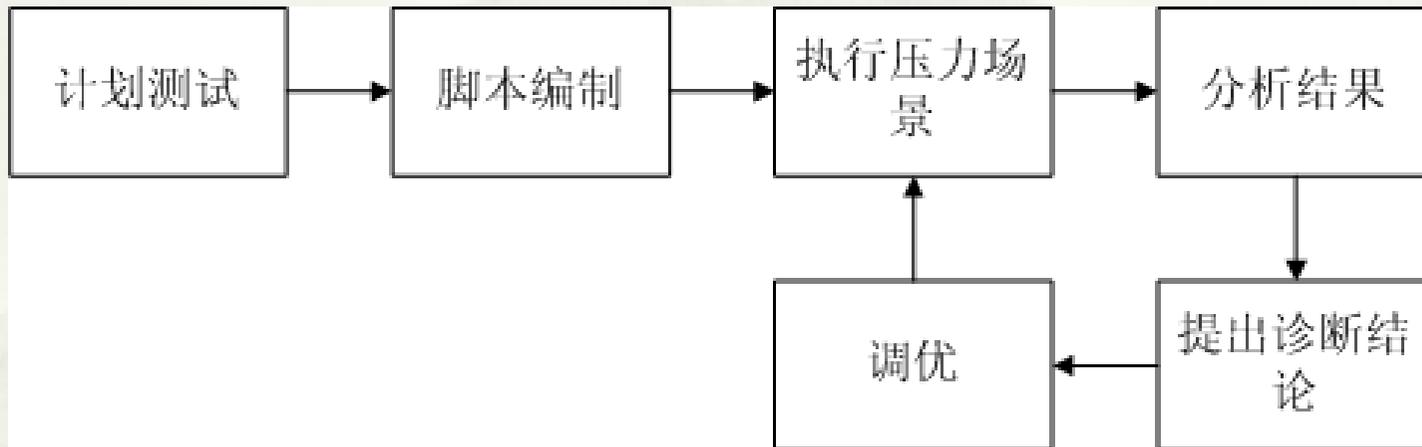
- 一、 工具简介
- 二、 工具方法论
- 三、 VuGen的使用
- 四、 Controller的使用
- 五、 Analysis的使用

工具简介

LoadRunner产品包括四个核心组件：

- * **VuGen**： 虚拟用户生成器是脚本编辑工具，可以通过捕获用户业务流程生成脚本，支持脚本的修改和调试；
- * **Controller**： 负载的组织、驱动、管理和监控；
- * **Runner**： 运行脚本形成访问压力；
- * **Analysis**： 结果分析器用于读取分析负载运行过程采集的监控数据生成图表以帮助进行分析；

工具方法论



1. 计划测试：测试前要规划测试对象的性能需求，定义压力指标；
2. 脚本编制：捕获实际用户的业务操作流程，形成自动脚本；
3. 执行场景：设定压力管理和执行压力场景，进行性能参数的监控；
4. 分析结果：使用分析器导出结果生成分析图表；分析结果；
5. 诊断结论：给出诊断结论；
6. 调优：针对诊断结果进行调优；

VuGen的使用

- * 录制业务流程

- * 修改强化脚本

- ① 参数化

- ② 关联

- ③ 检查点

- ④ 同步点

- ⑤ 事务化

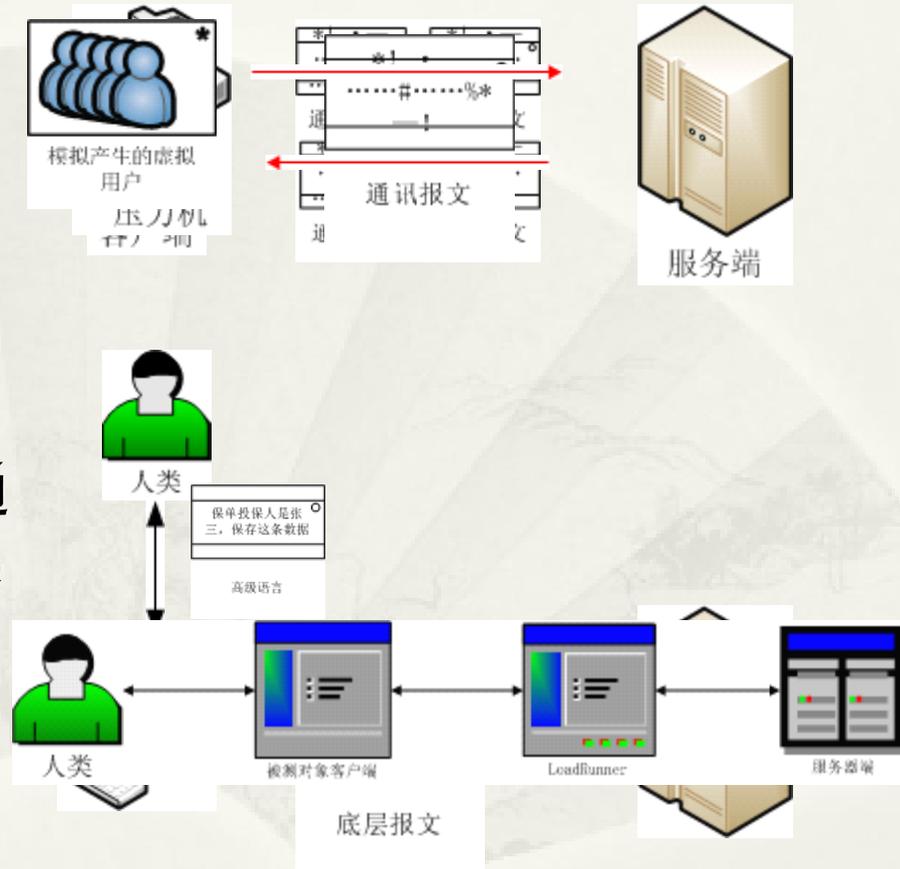
VuGen

—协议

* VuGen的脚本协议

LoadRunner对业务流程模拟的原理；

协议是VuGen的核心组成，协议相当于翻译官，是VuGen对录制对象的通讯包进行识别和模拟的基础；



VuGen

—参数化

* 参数化的作用

通过将一组待提交的数据进行变量化来达到在脚本运行时引入多样数据的目的;

```
User=Lakata *
Passwd=Lakata123
.....
Fromcity=Beijing
Tocity=Tokeyo
Number=1
```

录制时使用的数据生成脚本

```
User=Lakata *
Passwd=Lakata123
.....
Fromcity=Beijing
Tocity=Tokeyo
Number=1
```

录制时使用的数据生成脚本

```
User={PUser} *
Passwd={PPasswd}
.....
Fromcity={PFromcity}
Tocity={PTocity}
Number={PNumber}
```

录制时使用的数据生成脚本

```
User=LuluXi
Passwd=LuluXi123
.....
User=LiuYong
Passwd=LiuYong123
.....
User=SunXi
Passwd=SunXi123
.....
```

实际的机票订单

```
*
参数值
调度守
卫进程
```

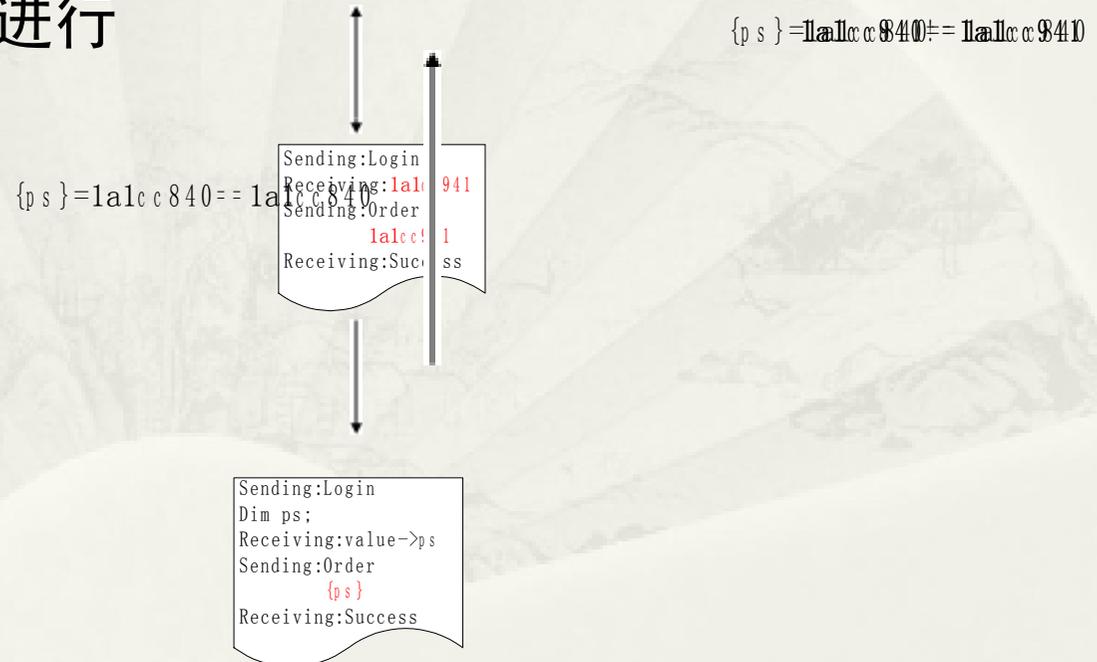
VuGen

—关联

录制时

* 关联的作用

将执行过程中从服务端获取即时数据保存在某个变量中，在后续业务中进行提交的机制



VuGen

—检查点

* 检查点的作用

从业务过程的角度检查用例结果是否正确；

运行正确的两个层次：服务器响应正确；业务处理正确；

[http状态码](#)



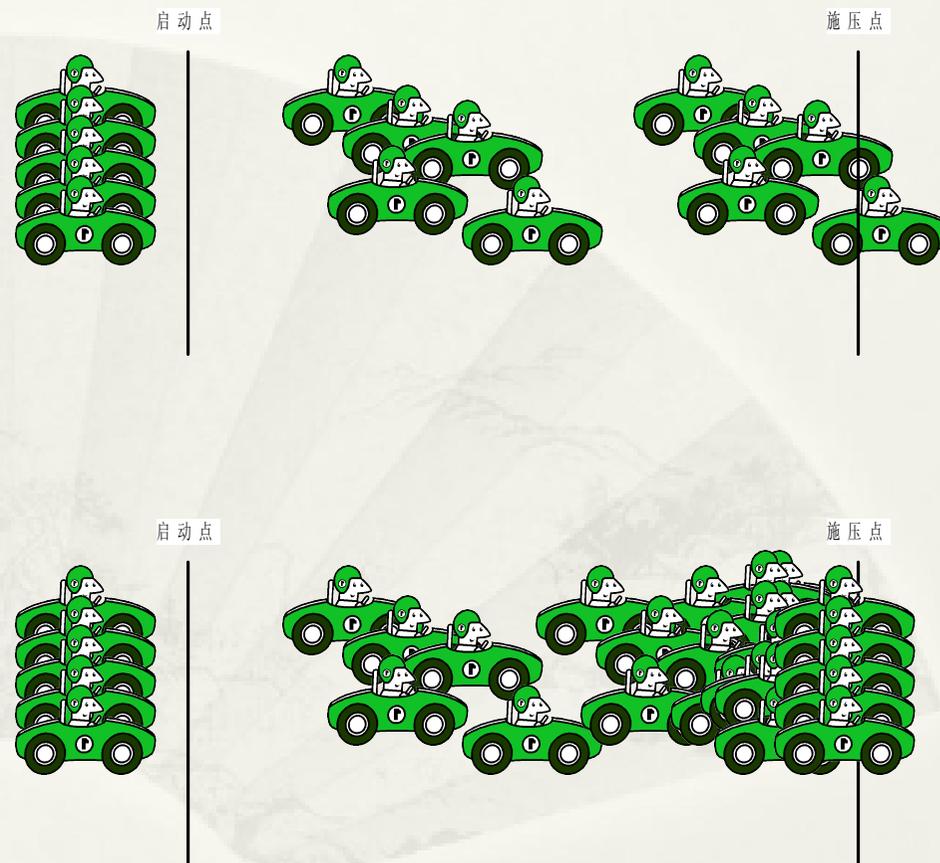
服务响应正确不等于业务处理正确

VuGen

—同步点

* 同步点的作用

保证用户的绝对并发执行；在模拟多用户对业务流程进行操作时，能够模拟绝对的并发压力；



VuGen

—事务化

* 事务化的作用

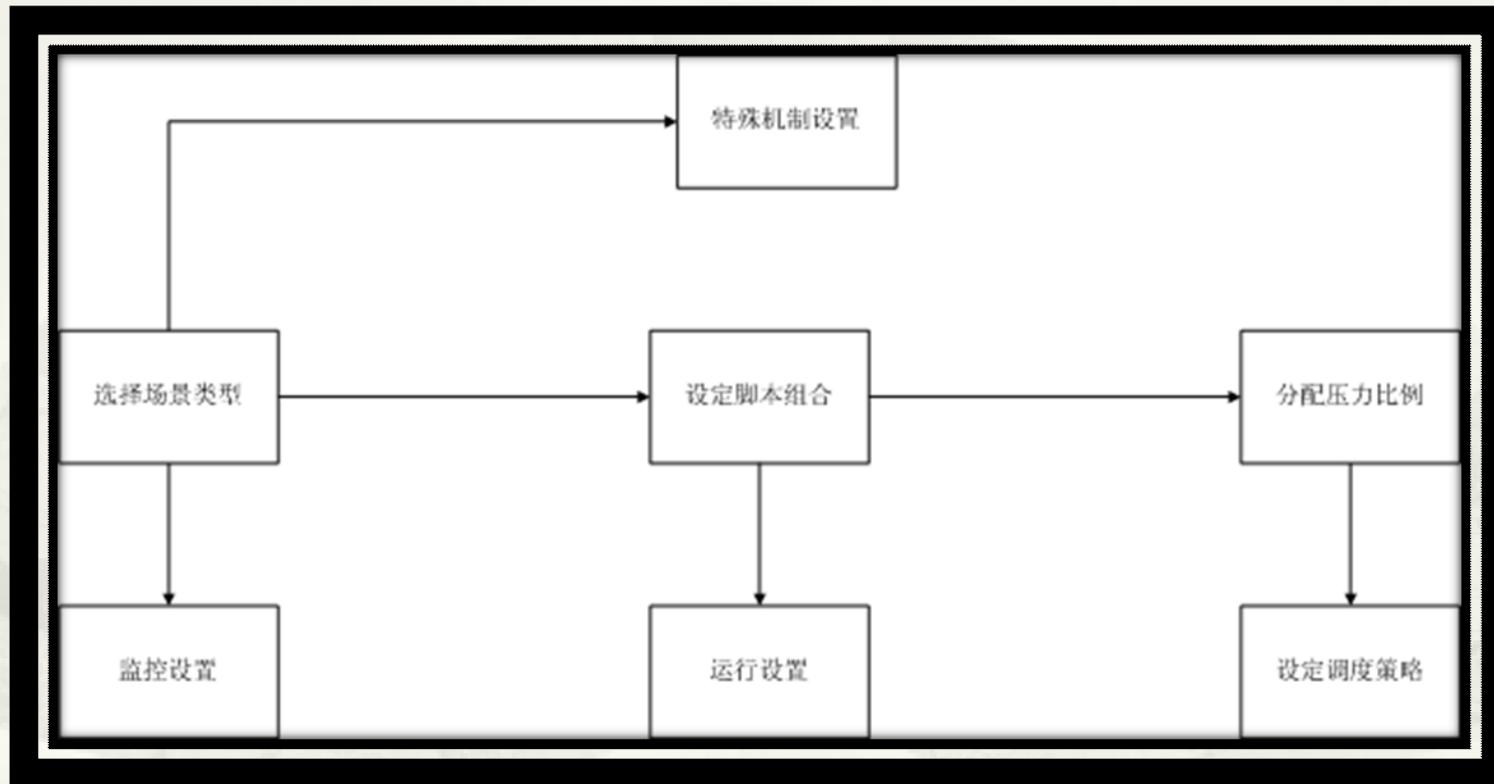
将连续的业务过程进行事务化分割，能够明确统计每一步骤的处理性能；

LoadRunner中默认一个action是一个事务，统计事务处理时间是以整个action来统计

Controller的使用

- * 场景的设定
- * 运行时设定
- * 监控设定
- * 网络模拟及IP欺骗

Controller



Controller

—场景设定

* Manual Scenario（手工场景设置）

手工设置压力的分布和增长、降低的策略，测试时工具根据设定的压力分配各条脚本的访问用户数量分布，可以选择根据百分比或直接设置绝对数量两种；

适用于对预计访问压力的性能评估需要；

* Goal-Oriented Scenario（目标场景设置）

设定压力运行的性能目标，由工具自动调整访问压力来尝试达到；

适用于对系统性能需求明确进行验证需要；

Controller

—场景设定

* Scenario的设置

starttime
convert scenario.....
load generators.....
schedule
enable ip spoofer

* Results

* Rendezvous

Controller

—运行时设定

- * Run Logic
- * Pacing
- * Think Times
- * Miscellaneous
- * Speed Simulation
- * Proxy
- * Preferences
- * ContentCheck

监控设定

* 性能监控

trans response time
trans / sec (passed)

* 网络监控

throughput

* 服务器监控

%processor time
(Memory) available bytes
%disk time
processor queue length

* 专项监控

Analysis的使用

- * 导出结果
- * 读取图表
- * 组合图表
- * 取样间隔

总结

使用LoadRunner配合合理的方案，能够对系统进行有效的性能测试；性能测试是一项团队合作的工作，需要各个技术角色的参与，从被测系统的开发人员到被测系统的软硬件维护人员多方的配合；

HTTP协议状态码的含义

* HTTP协议状态码的含义

号码 含义

"100" : Continue
"101" : switching Protocols
"200" : OK
"201" : Created
"202" : Accepted
"203" : Non-Authoritative Information
"204" : No Content
"205" : Reset Content
"206" : Partial Content
"300" : Multiple Choices
"301" : Moved Permanently
"302" : Found
"303" : See Other
"304" : Not Modified
"305" : Use Proxy
"307" : Temporary Redirect
"400" : Bad Request
"401" : Unauthorized

* HTTP协议状态码的含义

号码 含义

"402" : Payment Required
"403" : Forbidden
"404" : Not Found
"405" : Method Not Allowed
"406" : Not Acceptable
"407" : Proxy Authentication Required
"408" : Request Time-out
"409" : Conflict
"410" : Gone
"411" : Length Required
"412" : Precondition Failed
"413" : Request Entity Too Large
"414" : Request-URI Too Large
"415" : Unsupported Media Type
"416" : Requested range not satisfiable
"417" : Expectation Failed
"500" : Internal Server Error
"501" : Not Implemented
"502" : Bad Gateway
"503" : Service Unavailable
"504" : Gateway Time-out
"505" : HTTP Version not supported

Run Logic

—概念

* 脚本块

由一组脚本组成的集合，脚本块区别于脚本组在于工具的脚本组是固定的，而脚本块使客户自定义添加的；脚本是可以含有脚本块；

* 脚本组

脚本组是脚本或脚本块的组合，工具有三类固有的脚本组；

1. Init

Init脚本组在一个虚拟并发用户的整个执行周期内只会运行一次；所有该组的脚本或脚本块都只会顺序执行一次；

2. Run

Run脚本组在一个虚拟并发用户的整个执行周期内可以运行多次；

3. End

同Init脚本组End脚本组在一个虚拟并发用户的整个执行周期内只会运行一次；

* Iteration

Iteration特指的是Run脚本组的迭代；设置Iteration迭代次数时,将影响整个Run脚本组的迭代次数；当Run脚本组需要内迭代时，则可以通过脚本块来进行内循环的设置；

Preferences

* advanced-〉 Winlnet replay instead of Sockets (Windows only)

1. Socket和winsocket, socket是通用的tcpip的协议实现, winsocket是socket在windows下的针对windows特性的封装实现, 能发挥windows的特性但通用性较不足;
2. Winlnet是实现了winsocket的动态库;

ContentCheck

- * 将系统的错误封装成便于识别和定位的日志记录；
检索服务器返回的数据包是否含有目标内容，存在即说明错误发生；