**https://www.cnblogs.com/jaazz/p/8668214.html**

**[Loadrunner教程--常用操做流程](https://www.cnblogs.com/jaazz/p/8668214.html)**

**1loadrunner压力测试一般使用流程**

**1.1loadrunner压力测试原理**

本质就是在loadrunner上模拟多个用户同时按固定行为访问web站点。其中固定行为在loadrunner中是通过录制脚本定义的，多个用户同时访问的策略是在loadrunner的场景中定义的。

**1.2压测思路**

通过loadrunner进行压力测试web应用的主要思路分两步：

第一步 录制脚本

先录制用户请求访问的行为，如用户访问首页，或用户先登陆，后浏览页面，后搜索；录制完后会生成用户请求行为的脚本；然后我们按需要修改脚本，使脚本更符合实际请求情况；

第二步 运行场景

    在第一步中录制了单个用户请求流程的脚本，第二步主要是设置用户按照脚本同时访问的并发数以及并发访问的时间和策略，使其更接近真实高并发访问的场景；在模拟并发访问后，会形成测试报告，包括服务器数据流吞吐量、每秒请求次数、事务响应时间等。

**1.3压测步骤简介**

以测试A应用首页为例，主要分为以下5个步骤

1设置录制选项

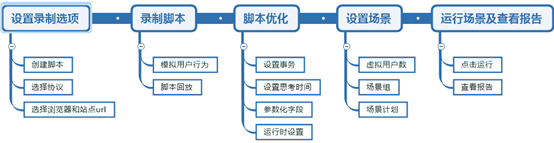
2录制脚本

3脚本优化

4设置场景

5运行场景及查看报告

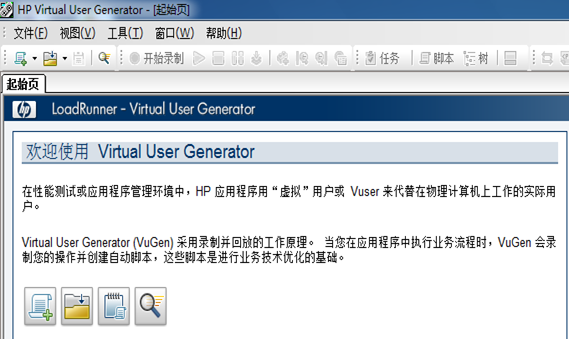
具体配置如下图所示，其中有些配置步骤可以用loadrunner默认的，不是必须项。



**2压力测试步骤详细说明**

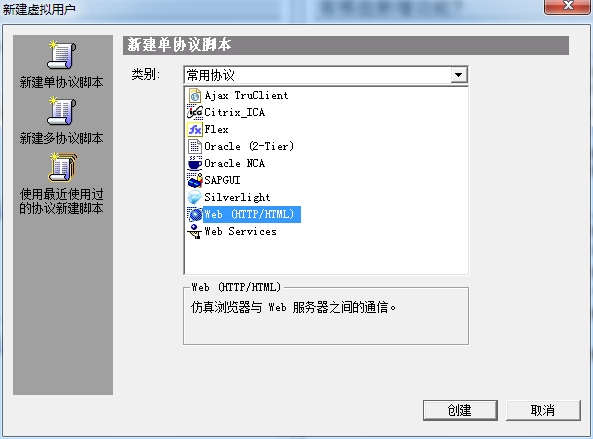
**2.1设置录制选项**

**2.1.1选择首页中的创建/编辑脚本**

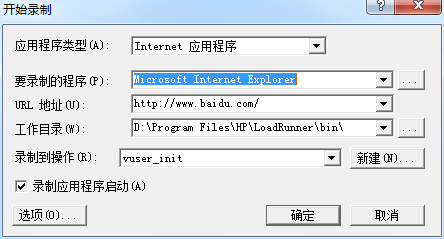


**2.1.2选择协议**

对web站点进行压力测试，选择web（HTTP/HTML）



**2.1.3输入系统url，设置录制选项**



【要录制的程序】：这个是打开url的浏览器，默认为IE，不过loadrunner11支持到IE8，可以自定义浏览器。

【URL地址】：需要进行测试的链接

**2.2录制脚本**

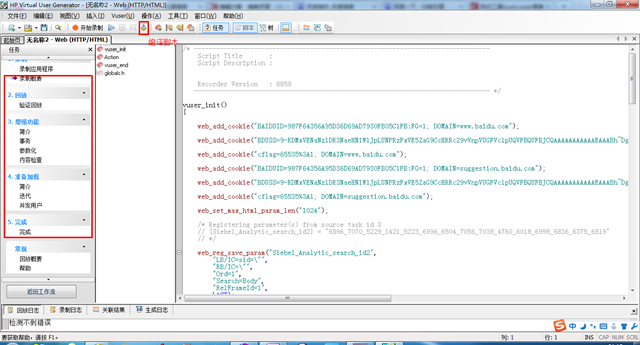
**2.2.1录制脚本**

在上一步点击"确定"后，会自动打开浏览器并定位到配置的url地址中。然后需要在浏览器中模拟用户的行为，我们执行一个搜索功能。



在模拟用户行为完后，点击蓝色按钮停止录制，会生成脚本，如下一步所示

**2.2.2验证回放脚本**



在生成脚本后，可以点击验证回放，会根据生成的脚本在浏览器中回放用户的行为，来验证用户的行为是否符合要求。

**2.3脚本优化**

回放验证成功后，可以进行事务、参数、并发用户的设置、脚本运行时设置。此操作为可选项，可以不用配置，系统采用默认配置。

**事务**：根据项目需要，除了衡量整个脚本的性能外，还想获取到脚本中的某一段或几段操作，例如：想知道从单击"确定"按钮到显示首页这一登录过程的性能数据，以便更详细的知道具体是用户的哪些动作对系统性能的影响比较大。这时候，可以将登录的运行脚本标记在一个事务内。

在脚本中没有事务时，可以在脚本开始和结束的地方分别加上

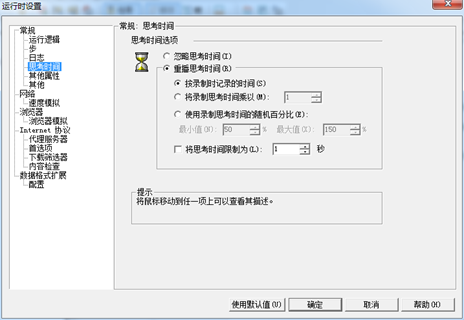
lr\_start\_transaction("xxx");

lr\_end\_transaction("xxx", LR\_AUTO);

表示整个脚本为一个事务。否则，可能监控不到事务响应时间

**参数化**：所谓参数化，针对的是脚本中的某些常量。让不同的虚拟用户在执行相同的脚本时，分别使用单数数据源中的不同数据代替这些常量，从而达到模拟多用户真实使用系统的目的。例如在'用户搜索'的事务脚本中，搜索的关键字是'AAA'。但实际情况中，每个并发访问的用户搜索的关键字是不一样的。所以要使每个用户执行脚本时的关键字都不一样，这个过程就需要对关键字参数化。

**脚本运行时设置**：设置每个action的执行次数、设定思考时间、步等



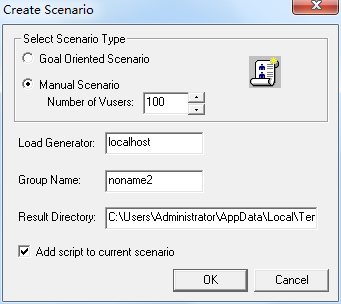
**【步】：**每个用户请求事务的时间间隔，一般选第一个，上一个事务一结束就开始下一个

**【思考时间】**：思考时间是为了模拟时更加接近用户的真实行为。实际中，用户进行一系列的操作之间会有停顿，即思考时间，例如：加载出数据后，用户会阅读数据，而浏览数据的过程对服务器是没有压力的。因此，在脚本中用函数lr\_think\_time(double time)来模拟用户停顿过程(未执行任何操作的过程)，执行该函数时，用户线程会按照相应的time值进行等待。如果选择忽略思考时间，会对服务器造成更大的压力。

**2.4创建场景及运行场景**

**2.4.1创建场景**

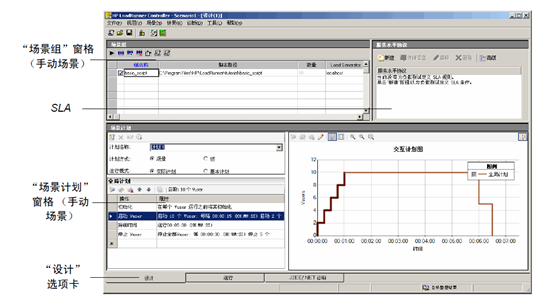
在录制完脚本页面中点击tools->create controller scenario，会弹出以下窗口



number of vusers：共要创建的虚拟用户100个；

Load gennrator：创建虚拟用户的机器，localhost本地；

点击ok后，跳转到创建场景的主页面



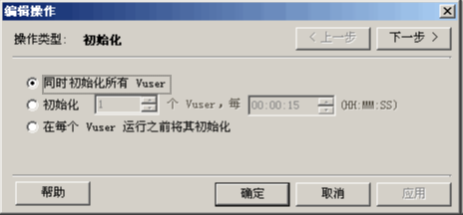
1"场景组"窗格。在"场景组"窗格中配置虚拟用户组及用户组来源，可以用来做分布式压力测试。我们知道一台测试机可以产生的虚拟用户是有限的（通常普通配置的主机能产生200个），当需要更大的压力即更多虚拟用户时，我就需要从别的测试机产生虚拟用户，并在此处配置，指定运行的 Vuser 数目以及运行时使用的计算机。

2"场景计划"窗格。在"场景计划"窗格中，设置加压方式以准确模拟真实用户

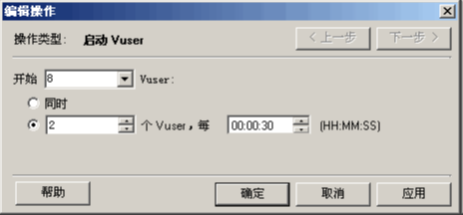
行为。可以设置初始化用户策略、逐步增加并发用户的速度、并发用户持续请求时间、逐步减少并发用户的速度。

3设置 **Vuser**初始化。

初始化是指通过运行脚本中的**vuser\_init**操作，为负载测试准备虚拟用户。在Vuser 开始运行之前对其进行初始化可以减少CPU占用量，并有利于提供更加真实的结果。



**4**设置启动vusers，可以设置逐步增加并发用户的速度，下图表示表示每30 秒启动 2 个Vuser。



通过按照一定的间隔启动Vuser，可以让Vuser 对应用程序施加的负载在测试

过程中逐渐增加，帮助准确找出系统响应时间开始变长的转折点。

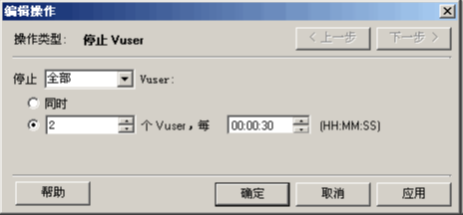
5设置用户并发访问持续时间。设置用户并发持续访问10分中。

中的时间显示为**00:11:30**。设置Vuser 运行10 分钟。

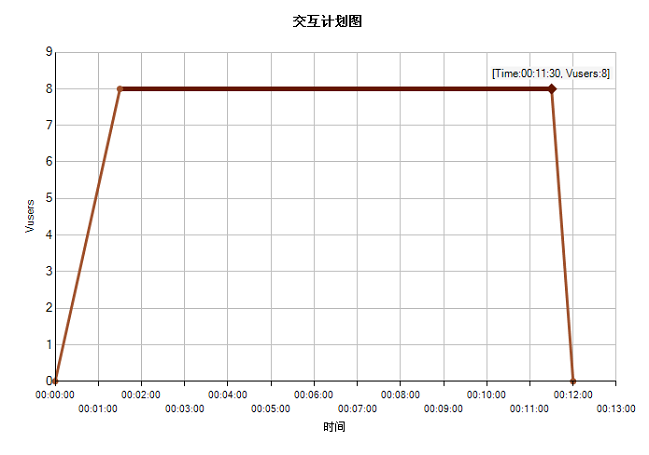
6安排逐渐减少用户访问。

建议逐渐停止Vuser，以帮助在应用程序到达阈值后，检查系统恢复情况。

如下图所示，表示每隔30 秒停止 2 个 **Vuser**。

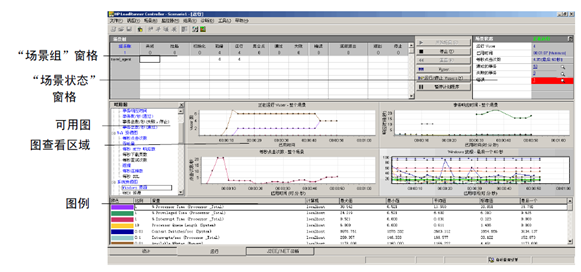


7用户并发访问策略，可以在下图中表现出来。



2.4.2创建及运行场景

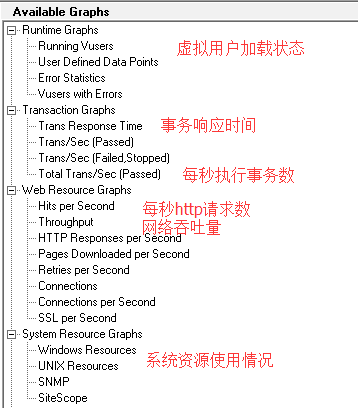
场景设计完后点击运行->运行场景，运行场景主页面如下图所示



1"场景组"窗格。位于左上角的窗格，可以在其中查看场景组内 Vuser 的状态。使用该窗格右侧的按钮可以启动、停止和重置场景，查看各个 Vuser 的状态，通过手动添加更多 Vuser 增加场景运行期间应用程序的负载。

2"场景状态"窗格。位于右上角的窗格，可以在其中查看负载测试的概要信息，包括正在运行的 Vuser 数量和每个 Vuser 操作的状态。

3可用图树。位于中间偏左位置的窗格，您可以在其中看到一列 LoadRunner 图。在树中选择一个图，会在图查看区域中显示。如下是常用的图

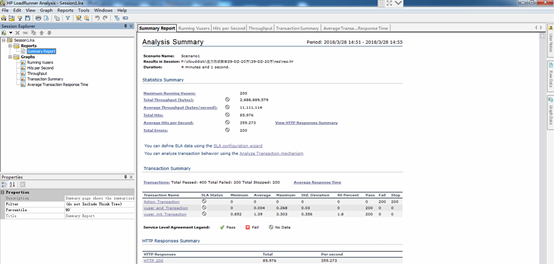


4图查看区域。位于中间偏右位置的窗格，用来显示可用图数中的图表。

5图例。位于底部的窗格，可以在其中查看所选图的数据。

**2.5查看报告**

在场景运行完后，可以在当前页面看到部分测试结果，如网络吞吐量，每秒请求数，事务平均响应时间，也可以在results->analyze results中查看详细的结果报告和图表，如下所示。



作者：[vps精选网](http://www.vpsjxw.com/)

分类: [压力测试](https://www.cnblogs.com/jaazz/category/1188192.html)

标签: [loadrunner](https://www.cnblogs.com/jaazz/tag/loadrunner/), [压力测试](https://www.cnblogs.com/jaazz/tag/%E5%8E%8B%E5%8A%9B%E6%B5%8B%E8%AF%95/)