

电子商务系统结构

实验四报告

3160103785 叶梓成

3170102587 蒋仕彪

实验一

第一部分：

开始录制 - [WebHttpHtml1]

更多选项

操作选择:

录制到操作(R): * Action

录制模式:

录制: Web 浏览器

应用程序: * Microsoft Internet Explorer

URL 地址(U):

https://www.jianshu.com/p/541256e608b4

设置:

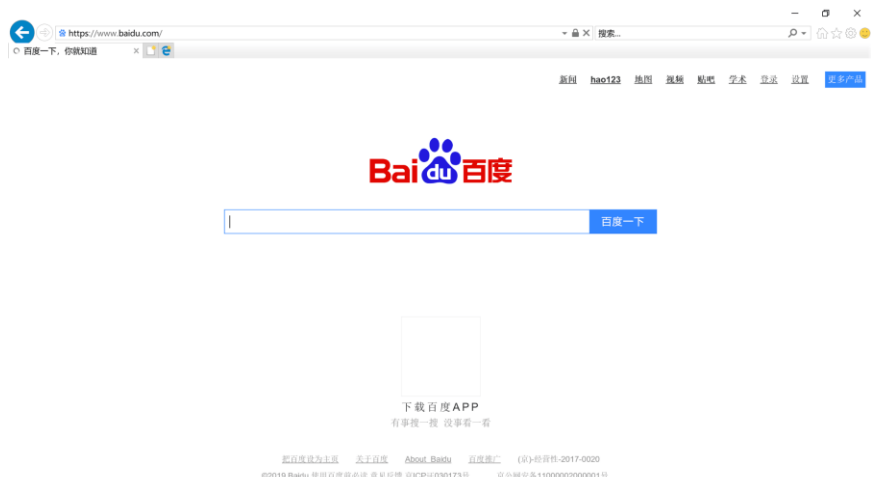
开始录制(S): ☒ 立即 ☐ 在延迟模式中

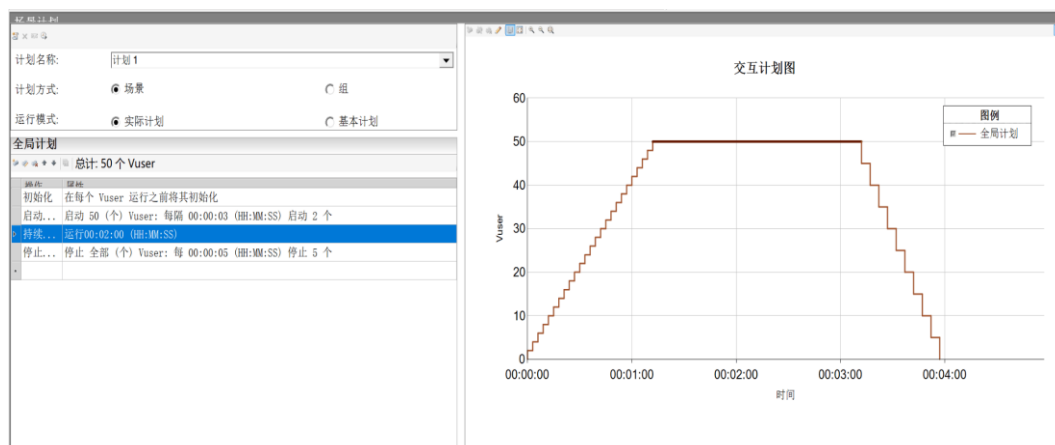
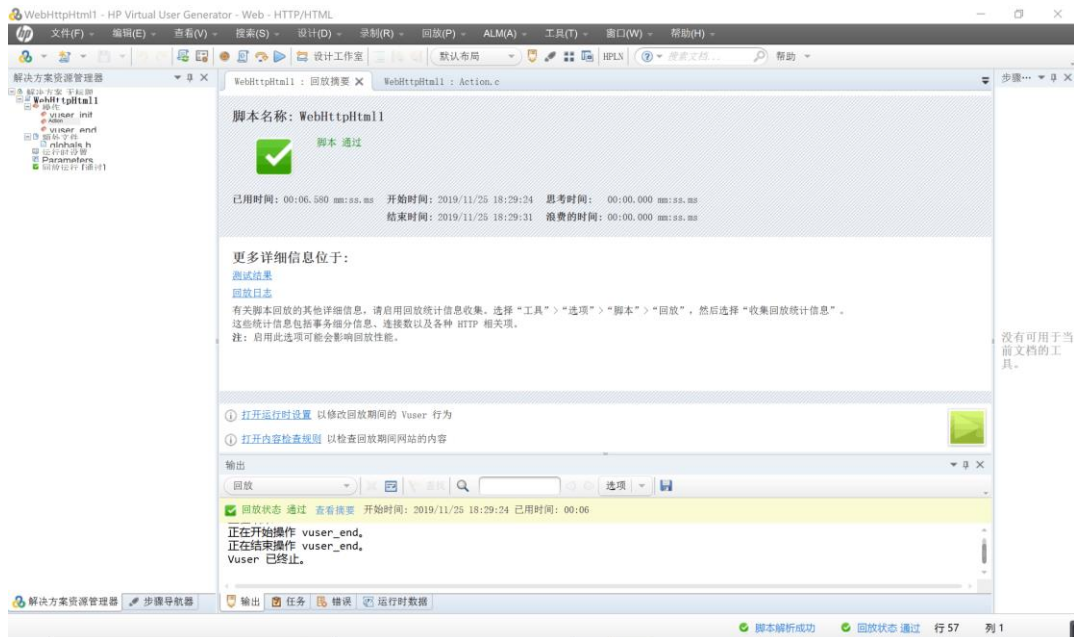
工作目录(W): * C:\Program Files (x86)\HP\LoadRunner\Bin

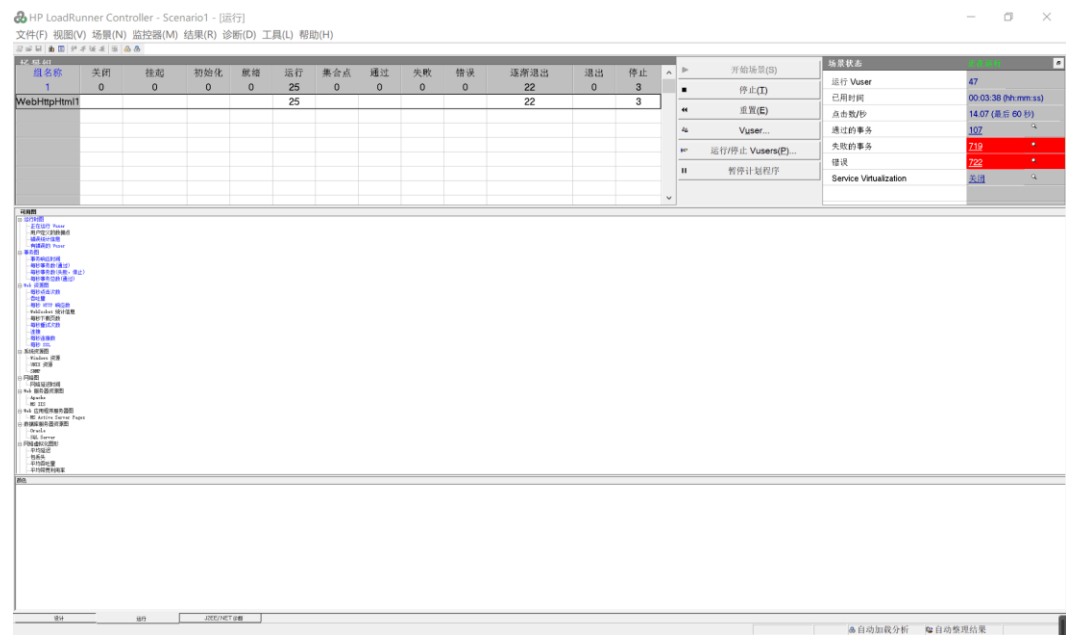
录制选项

开始录制

取消







统计信息概要表

运行 Vuser 的最大数目:	50
总吞吐量(字节):	97,927,672
平均吞吐量(字节/秒):	404,660
总点击次数:	3,019
平均每秒点击次数:	12.475
错误总数:	725

查看 HTTP 响应概要

您可以使用以下对象定义 SLA 数据 SLA 配置向导

您可以使用以下对象分析事务行为 分析事务机制

事务摘要

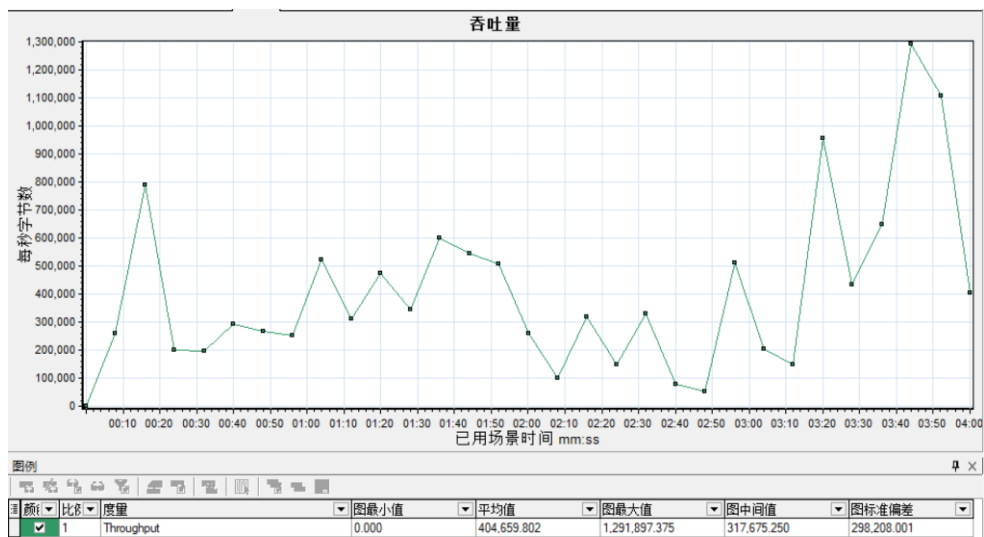
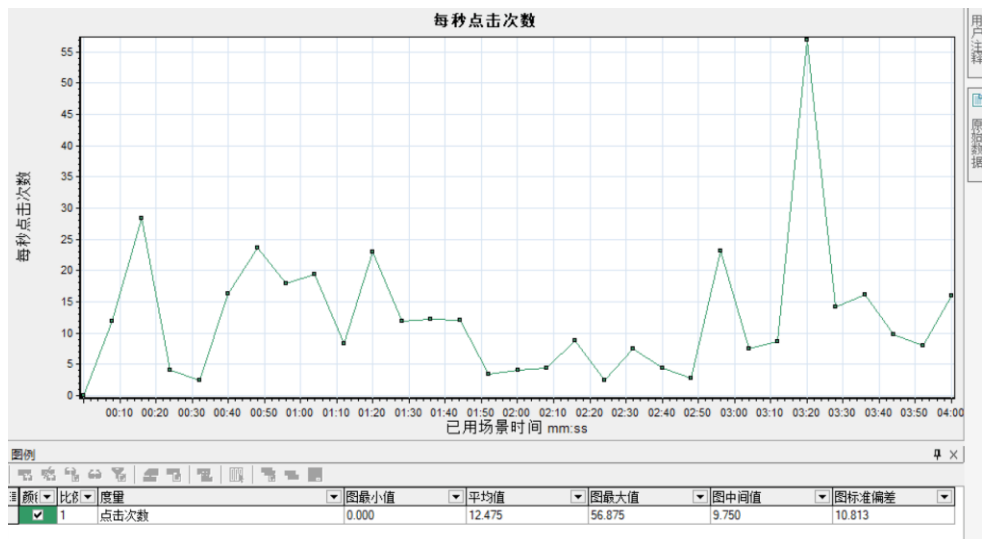
事务: 通过总数: 204 失败总数: 722 停止总数: 0 平均响应时间

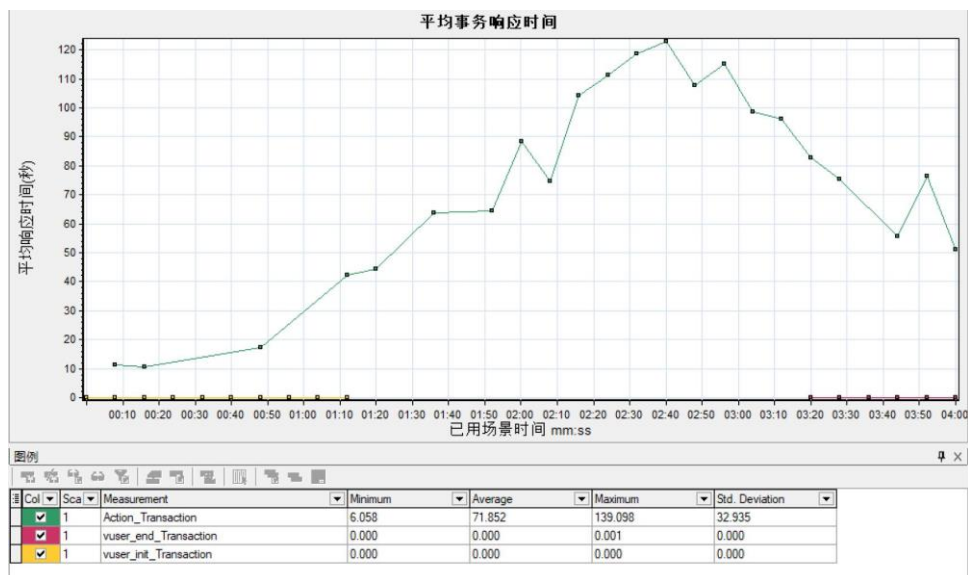
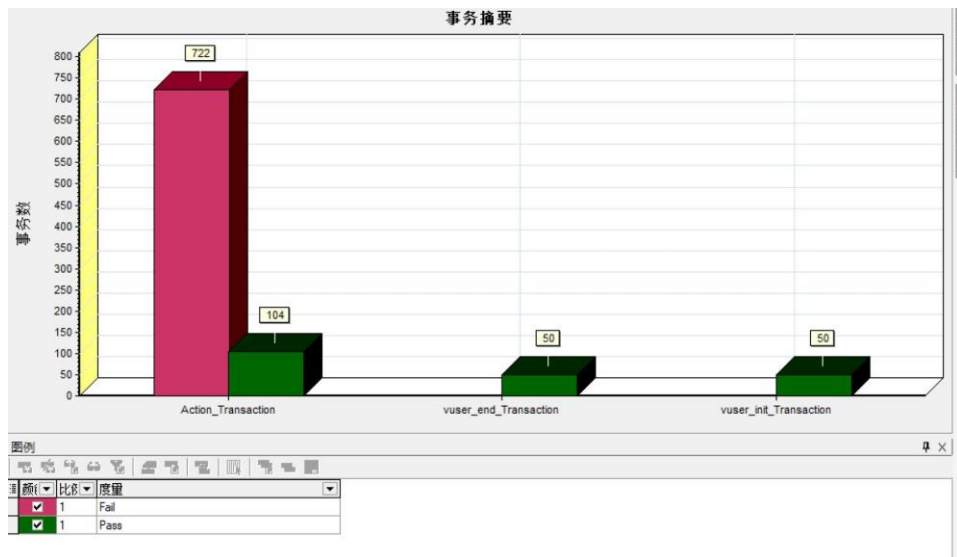
事务名称	SLA 状态	最小值	平均值	最大值	标准偏差	90 百分比	通过	失败	停止
Action Transaction		6.058	71.852	139.098	32.935	117.557	104	722	0
vuser_end Transaction		0	0	0.001	0	0	50	0	0
vuser_init Transaction		0	0	0	0	0	50	0	0

服务水平协议图例: 通过 失败 无数据

HTTP 响应概要

HTTP 响应	合计	每秒
HTTP 200	2,915	12.045
HTTP 204	104	0.43





具体参数：

详见上面一些截图。

一个 trick：

刚开始测试的时候，我们发现：线程一开多，电脑就会变得特别卡，而且经常出现 Fail。在尝试了很多操作以后我们发现：网速也是至关重要的一点。校园网好像限制了带宽，所以我们最后通过烧流量来完成实验。

分析：

首先，在开多了线程后，服务器响应的平均时间显著地变长了。

鼠标点击量和响应时间好像没有正向关联。

在并发数减少的时候，吞吐量会显著增加，说明网站依然以高负载的状态运行，即使并发数正在减少。

第二部分：

The screenshot shows the WebHttpHtml2 - HP Virtual User Generator - Web - HTTP/HTML interface. The main window displays the Action.c script for the WebHttpHtml2 script. The script includes various web actions such as web_set_sockets_option, web_add_cookie, and web_url. The output window shows the generated code and the execution results, including the transaction ID and the location flag.

```
1 Action()
2 {
3
4     web_set_sockets_option("SSL_VERSION", "TLS1.1");
5
6     web_add_cookie("euin_cookie=603DF0270431944608873CA4725F4BE2F86D12971D761E3F; DOMAIN=user.qzone.qq.com");
7
8     web_add_cookie("pgv_pvi=2678027264; DOMAIN=user.qzone.qq.com");
9
10    web_add_cookie("uin_cookie=2261709315; DOMAIN=user.qzone.qq.com");
11
12    web_add_cookie("RK=1N6Zhp6NwF; DOMAIN=user.qzone.qq.com");
13
14    web_add_cookie("p; DOMAIN=user.qzone.qq.com");
15
16    web_add_cookie("euin_cookie=603DF0270431944608873CA4725F4BE2F86D12971D761E3F; DOMAIN=qlogo4.store.qq.com");
17
18    web_add_cookie("pgv_pvi=2678027264; DOMAIN=qlogo4.store.qq.com");
19
20    web_add_cookie("uin_cookie=2261709315; DOMAIN=qlogo4.store.qq.com");
21
22    web_add_cookie("RK=1N6Zhp6NwF; DOMAIN=qlogo4.store.qq.com");
23
24    web_url("jump",
25           "URL=https://ssl.ptlogin2.qq.com/jump?ptlang=2052&clientuin=2261709315&clientkey=3583AF420373D22808700A8158CA448",
26           "Resource=0",
27           ...
28 }
```

输出

代码生成

web_add_cookie(pgv_pvi=2678027264; DOMAIN=026.photo.store.qq.com);

***** Add Event For Transaction With Id 0 ***** (Location Flag : tflagInsertBefore, Location ID : 555)

脚本解析成功 行 0 列 0

The screenshot shows the WebHttpHtml2 - HP Virtual User Generator - Web - HTTP/HTML interface. The main window displays the script execution results for the WebHttpHtml2 script. The results show that the script passed successfully, with a duration of 00:12.098 mm:ss.ms. The output window shows the generated code and the execution results, including the transaction ID and the location flag.

脚本名称: WebHttpHtml2

脚本通过

已用时间: 00:12.098 mm:ss.ms 开始时间: 2019/11/25 18:47:17 思考时间: 00:00.000 mm:ss.ms
结束时间: 2019/11/25 18:47:29 浪费的时间: 00:00.000 mm:ss.ms

更多详细信息位于:

[测试结果](#)

[回放日志](#)

有关脚本回放的其他详细信息, 请启用回放统计信息收集。选择“工具”>“选项”>“脚本”>“回放”, 然后选择“收集回放统计信息”。
这些统计信息包括事务细分信息、连接数以及各种 HTTP 相关项。
注: 启用此选项可能会影响回放性能。

① [打开运行时设置](#) 以修改回放期间的 Vuser 行为

① [打开内容检查规则](#) 以检查回放期间网站的内容

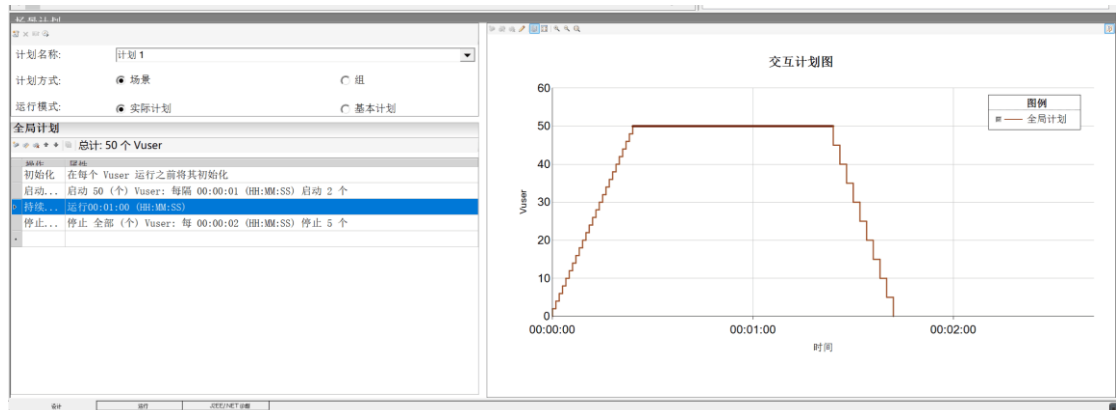
输出

回放

回放状态 通过 查看摘要 开始时间: 2019/11/25 18:47:17 已用时间: 00:12

正在开始操作 vuser_end。
正在结束操作 vuser_end。
Vuser 已终止。

脚本解析成功 回放状态 通过 行 421 列 1



统计信息概要表




运行 Vuser 的最大数目:	50	
总吞吐量(字节):	62,868,305	
平均吞吐量(字节/秒):	361,312	
总点击次数:	8,128	
平均每秒点击次数:	46.713	查看 HTTP 响应概要
错误总数:	41	

您可以使用以下对象定义 SLA 数据 [SLA 配置向导](#)
您可以使用以下对象分析事务行为 [分析事务机制](#)

事务摘要

事务: 通过总数: 133 失败总数: 41 停止总数: 0 [平均响应时间](#)

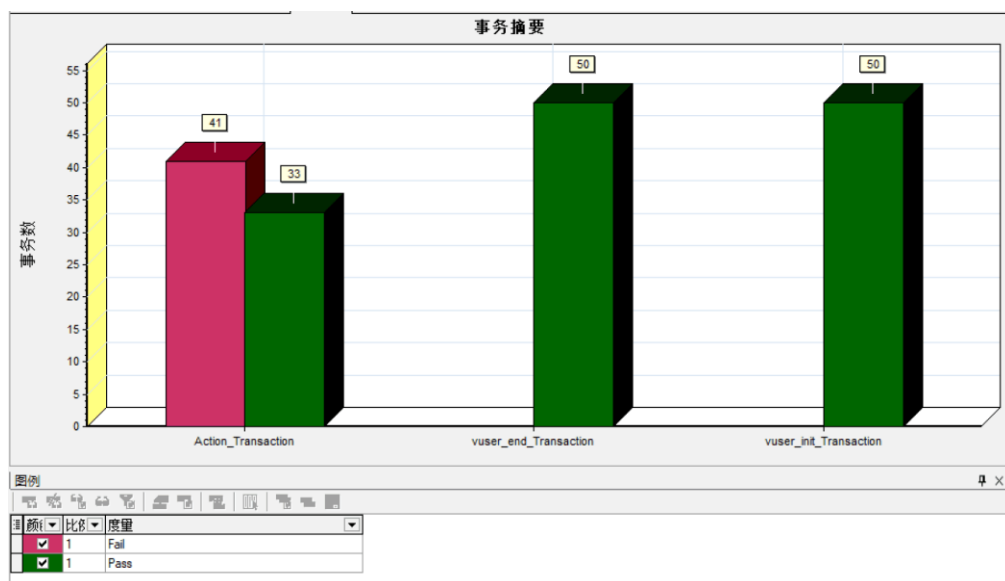
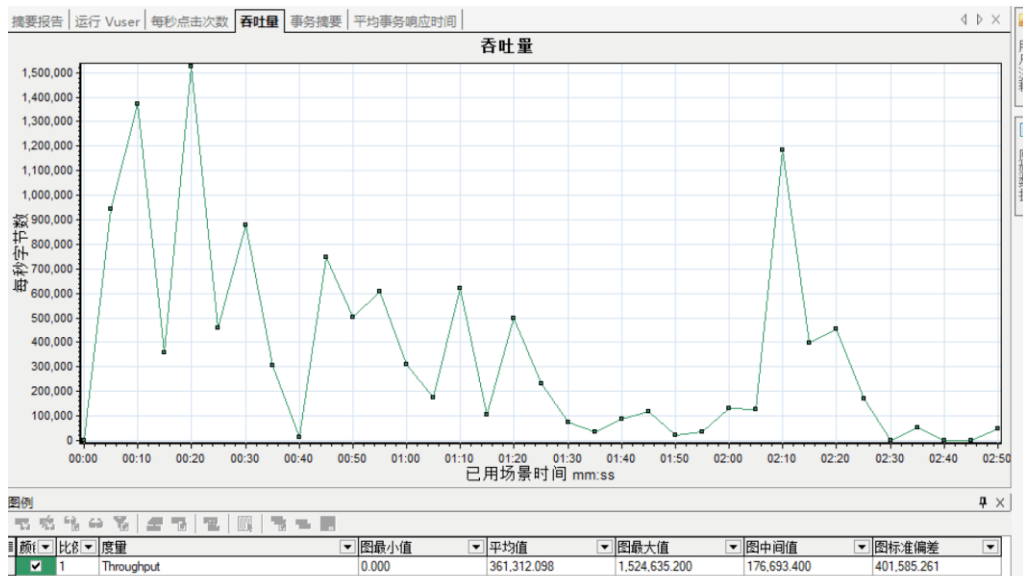
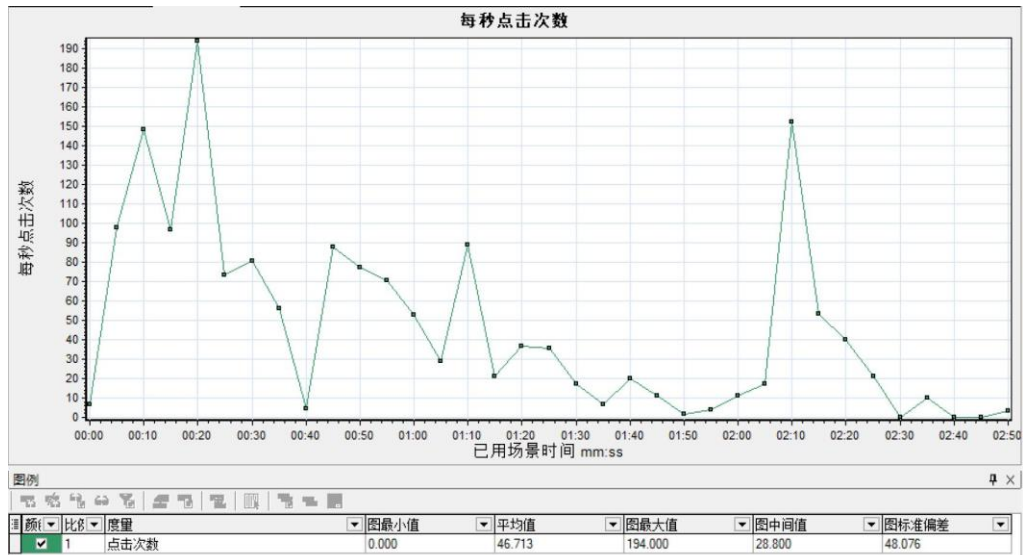
事务名称	SLA 状态	最小值	平均值	最大值	标准偏差	90 百分比	通过	失败	停止
Action Transaction	⊖	20.227	83.772	120.999	32.262	116.735	33	41	0
vuser_end Transaction	⊖	0	0	0	0	0	50	0	0
vuser_init Transaction	⊖	0	0	0.001	0	0	50	0	0

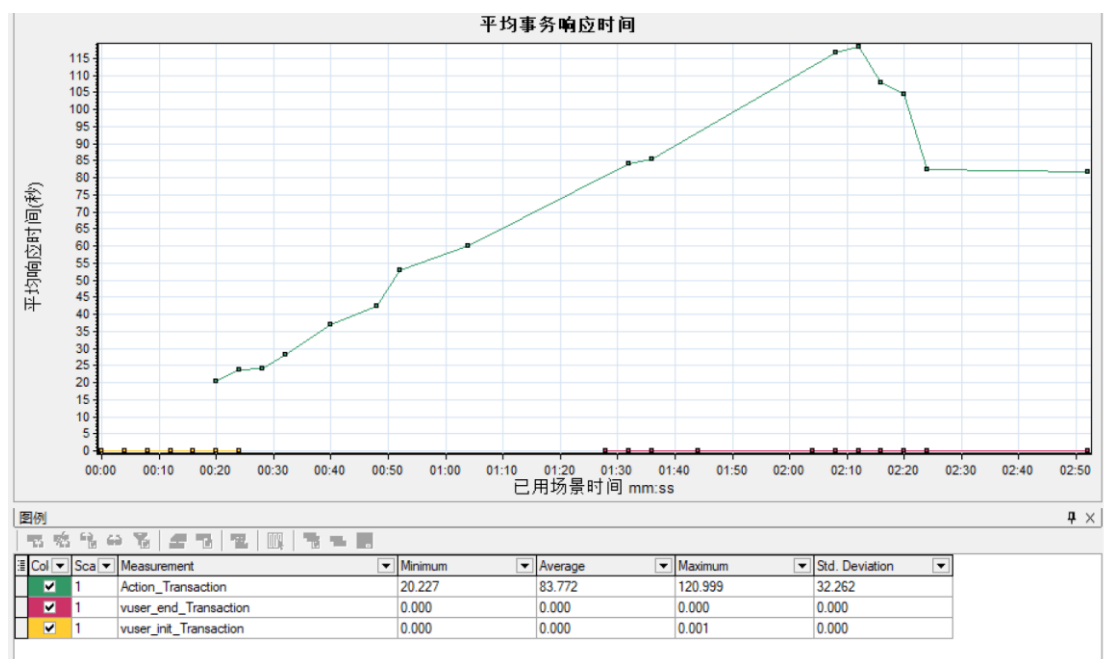
服务水平协议图例:  通过  失败  无数据

HTTP 响应概要

HTTP 响应	合计	每秒
HTTP 200	7,834	45.023
HTTP 302	294	1.69







具体参数：

详见上面一些截图。

分析：

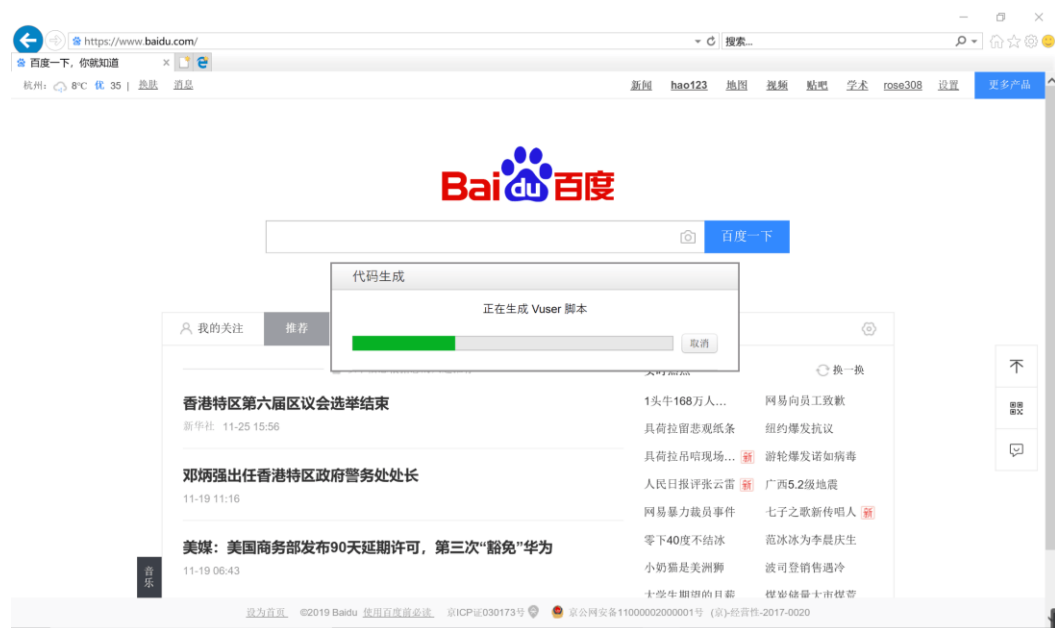
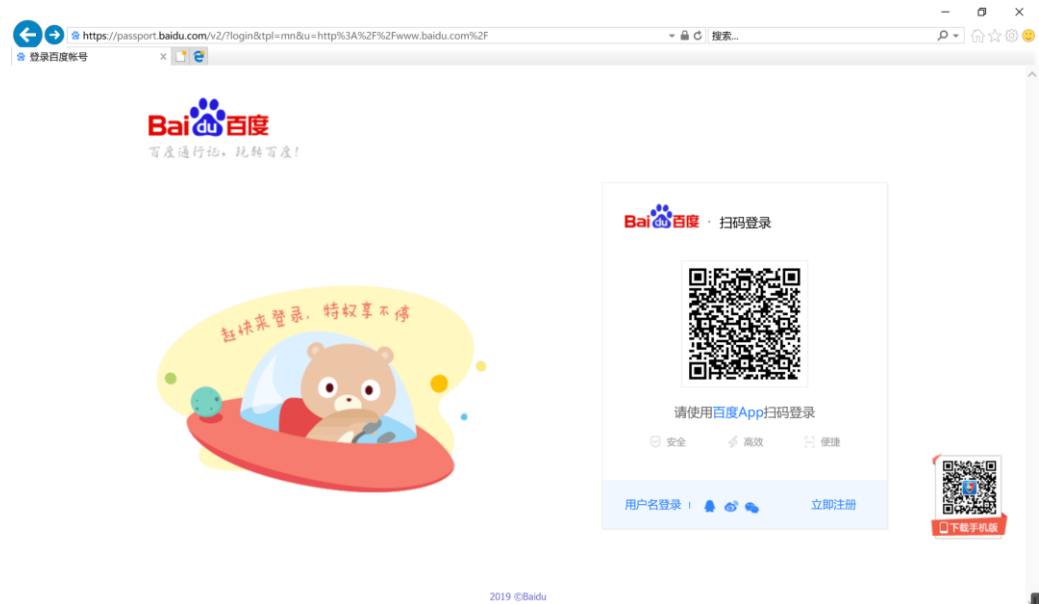
理论上思考，当操作/秒增加时，响应时间应该会有一个肉眼可见的上升（点击率越大,对服务器的压力越大），但第一部分和第二部分的两个例子告诉我们：**实际上它们没有关联**。我们怀疑网站会无视那些无用的点击。

吞吐量也和第一部分表现差不多：在并发数大量减少时，会产生一个峰值。

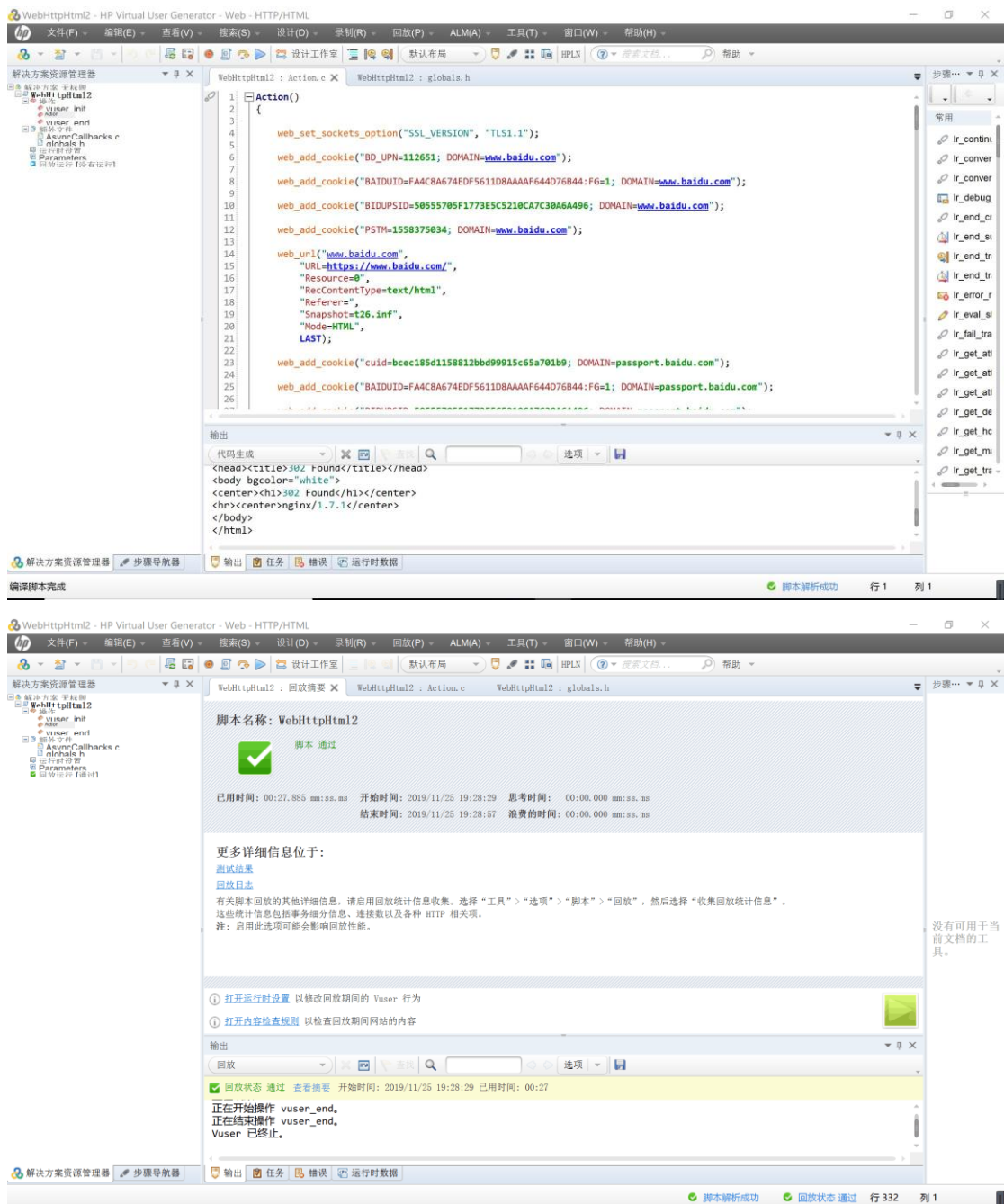
观察本页的那张图，响应时间在前 2 分半左右随着并发数的增加而飙升，说明并发效果特别差。设计者应该没有专门去处理并发问题，或者说访问数据库需要固定的时间，所以增加了并发会线性增加时间。

实验二：

首先我们登陆百度知道（这一步试了很多次，电脑经常卡死）。



一些截图：



结果的呈现和分析:

统计信息概要表

运行 <u>Vuser</u> 的最大数目:	50	
总吞吐量(字节):	562,883,963	
平均吞吐量(字节/秒):	582,093	
总点击次数:	28,274	
平均每秒点击次数:	29.239	查看 HTTP 响应概要
错误总数:	111	

您可以使用以下对象定义 SLA 数据 [SLA 配置向导](#)
您可以使用以下对象分析事务行为 [分析事务机制](#)

事务摘要

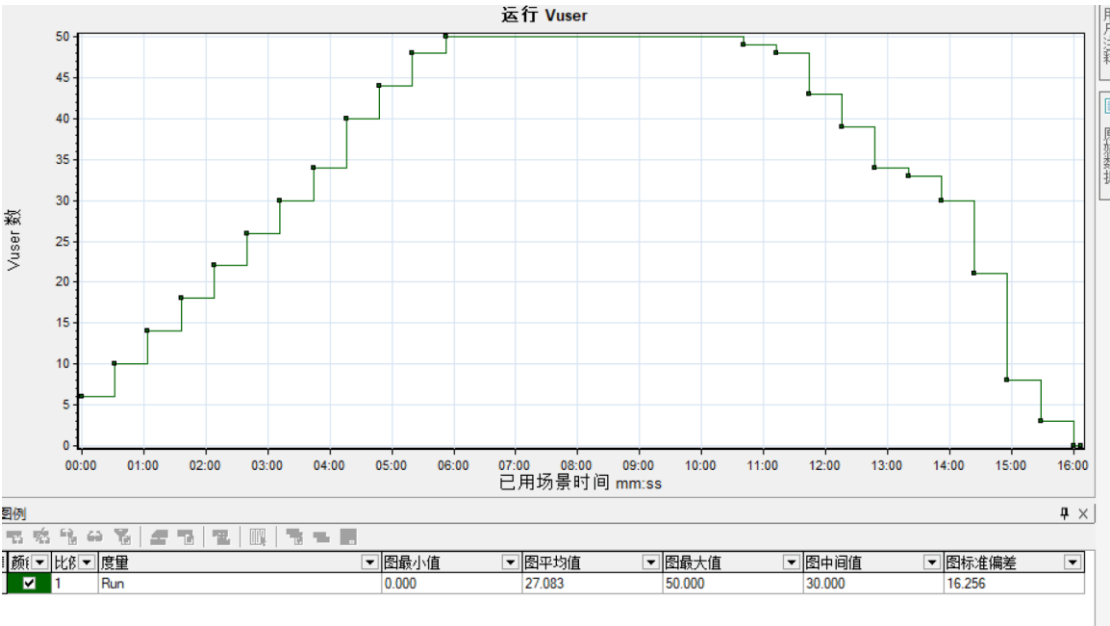
事务: 通过总数: 308 失败总数: 109 停止总数: 0 [平均响应时间](#)

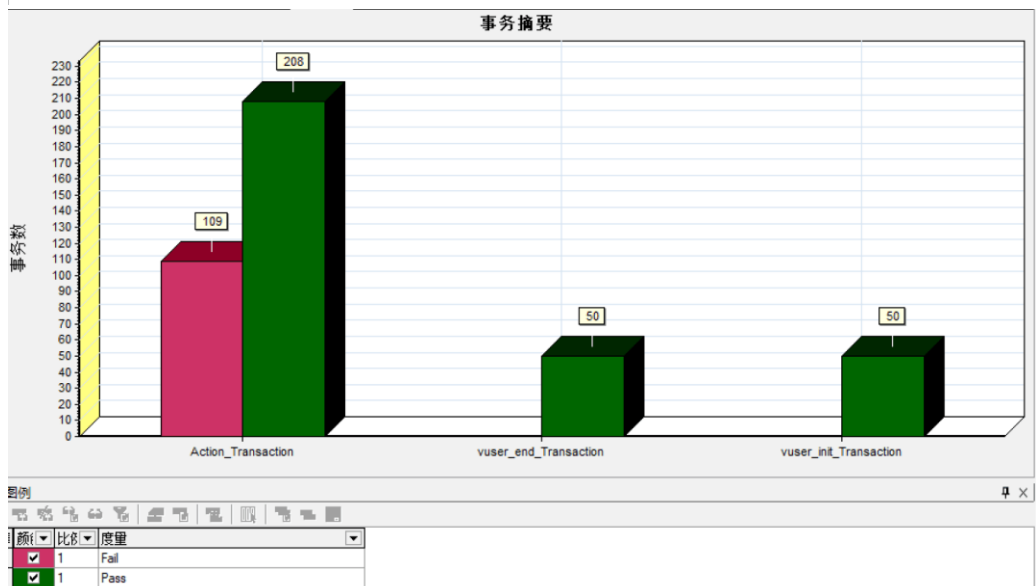
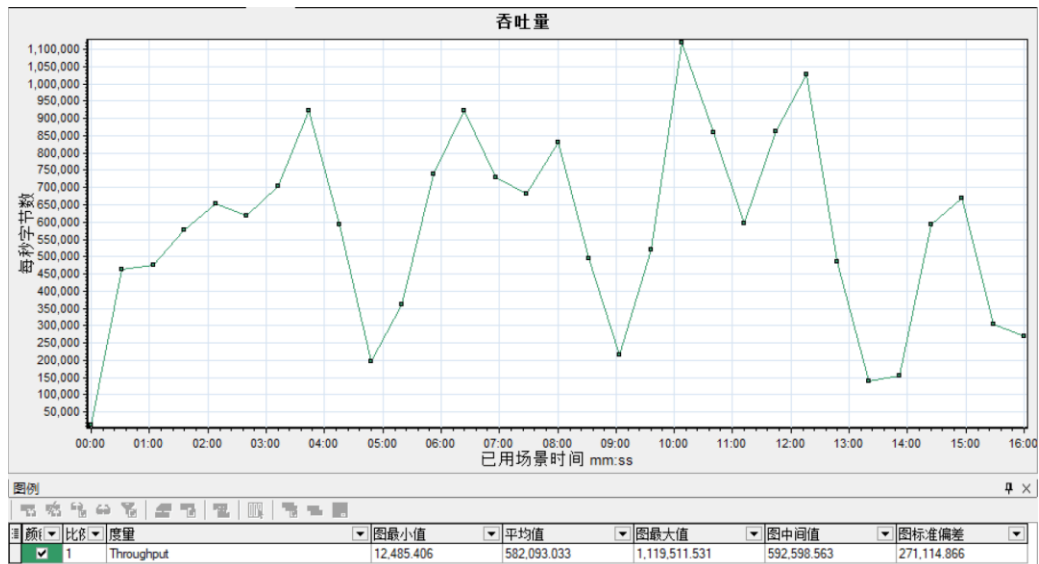
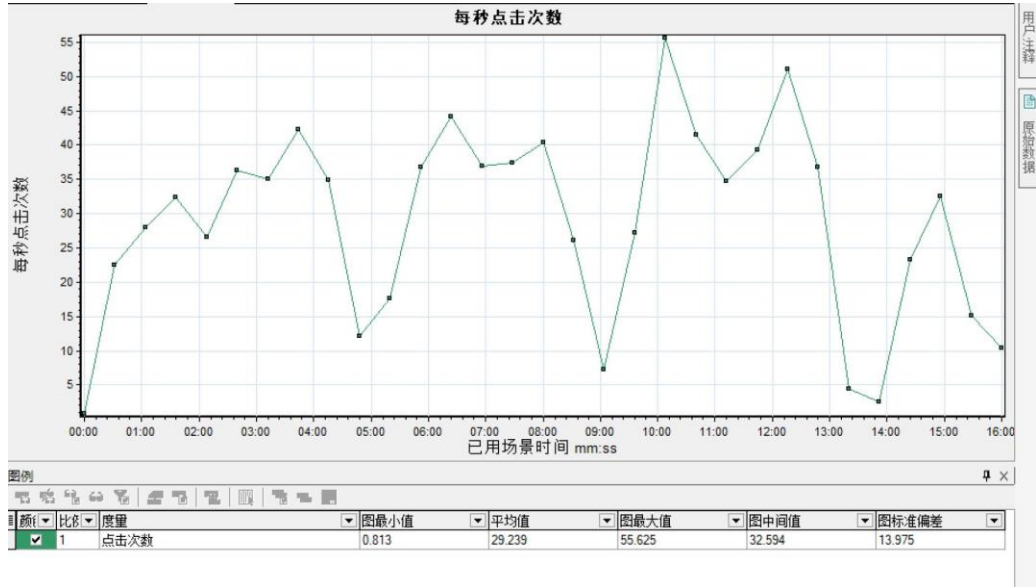
事务名称	SLA 状态	最小值	平均值	最大值	标准偏差	90 百分比	通过	失败	停止
Action Transaction	⊖	4.054	72.998	212.313	49.751	139.687	208	109	0
vuser_end Transaction	⊖	0	0	0	0	0	50	0	0
vuser_init Transaction	⊖	0	0	0.001	0	0.001	50	0	0

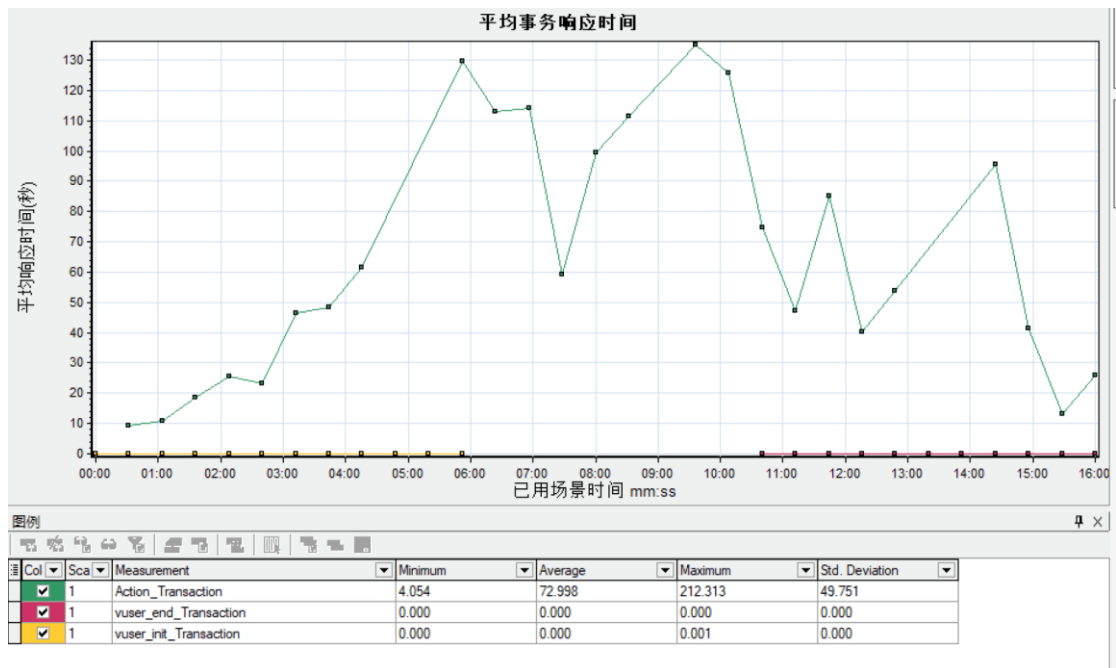
服务水平协议图例: ☒ 通过 ☒ 失败 ☒ 无数据

HTTP 响应概要

HTTP 响应	合计	每秒
HTTP 200	28,054	29.011
HTTP 302	220	0.228







具体参数：

详见上面一些截图。

分析：

登陆了之后，吞吐量的变化十分明显：以前是比较稳定、偶有震荡，现在是一直震的比较厉害，而且值比登陆前大：说明登陆后有更多的操作要做。

平均响应时间在中间出反而出现了一个谷，我们猜测登陆后网站会优化掉大量无用的并发。

每秒点击次数似乎依然没啥关联。