

# 实验七 自动化性能测试用例设计

## 一、实验目的与要求

### 1. 实验目的

Virtual User Generator 是脚本设计组件，性能测试脚本在此基础上开发并优化。场景设计须在 LoadRunner 的 Controller 中完成。利用 LoadRunner 的“Analysis”功能组件对当前测试结果进行汇总分析。

本次实验练习使用 Controller 设计并执行场景，利用 LoadRunner 监控运行的虚拟用户、事务响应时间、每秒请求数和吞吐量等；利用 Analysis 对结果进行分析，结合报告给出性能问题解决方案。

### 2. 实验要求

本次实验包括以下练习：

- (1) 场景设计与实现
- (2) 场景执行与结果收集
- (3) 结果分析与报告输出
- (4) 性能调优与回归测试

## 二、实验环境

LoadRunner

## 三、实验内容和步骤

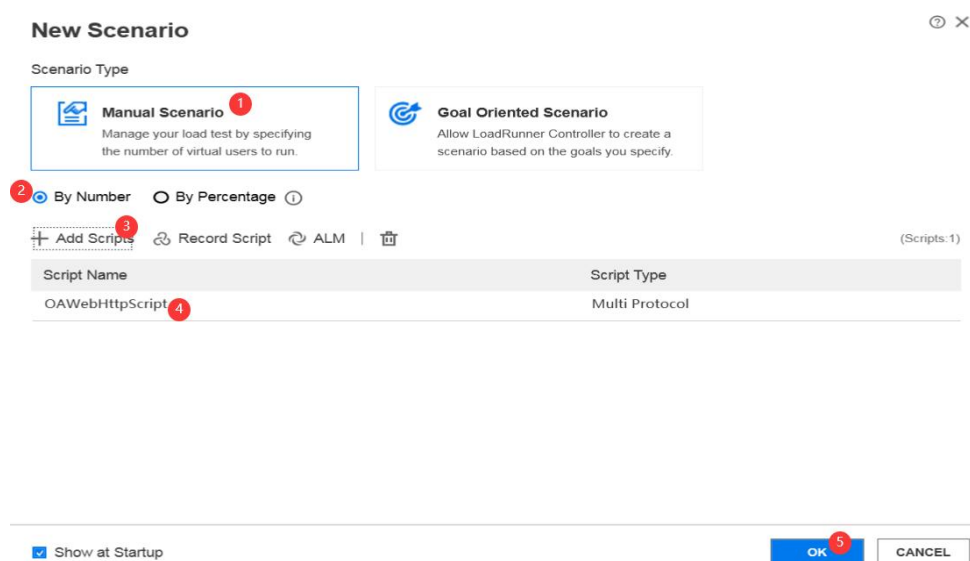
### 1. 场景设计与实现

Virtual User Generator 是脚本设计组件，性能测试脚本在此基础上开发并优化。场景设计须在 LoadRunner 的 Controller 中完成。

#### 【考勤业务场景设计】

根据场景用例，本次考勤业务测试场景设计过程如下。

(1) 创建场景。启动 Controller，选择“Manual Scenario”，选“By Number”，加载测试脚本。



(2) 场景计划。根据分析，本次测试需 30 分钟，8 个 Vuser，一开始即加载所有 Vuser。  
场景执行计划：

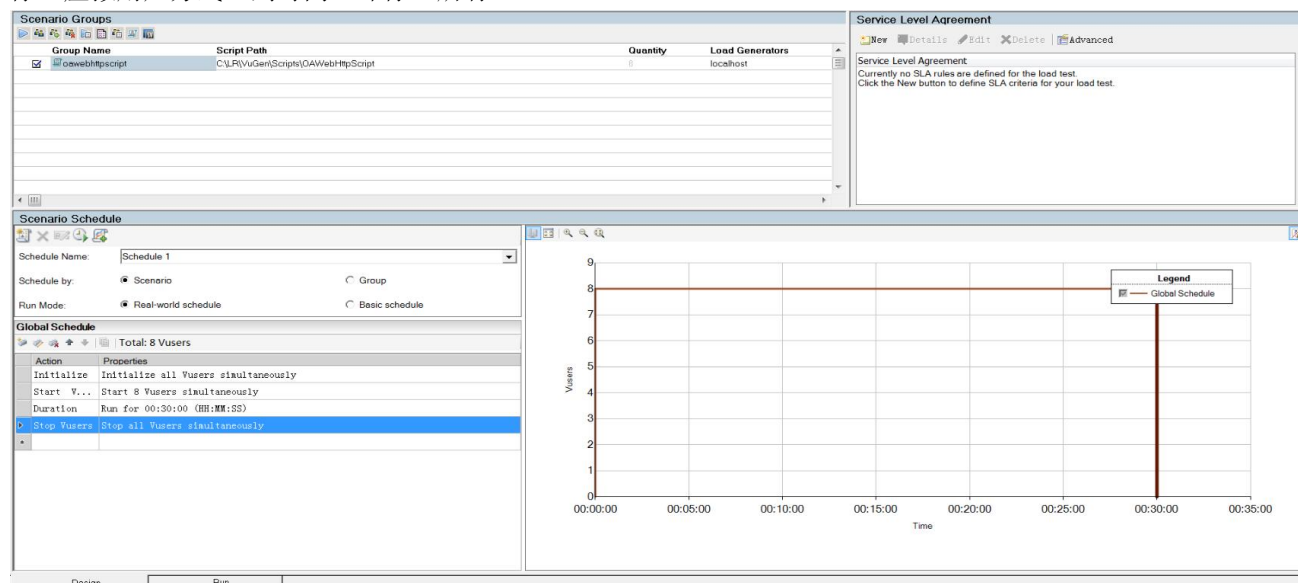
Vuser 虚拟用户数：8 个

初始化：在运行前初始化所有用户

开始虚拟用户方式：一开始即加载所有 Vusers

持续时间：30 分钟

停止虚拟用户方式：到时间立即停止所有 Vusers



(3) 设置测试结果保存目录，保存测试结果。

Result——Result Settings

**Result Settings**

Result name: \* OAWebHttpRes

Directory: \* C:\LR\controller [BROWSE](#)

☐ Automatically create a results directory for each scenario execution

☐ Automatically overwrite existing results directory without prompting for confirmation

[OK](#) [CANCEL](#) [HELP](#)

(4) 设置 Runtime setting。

think time 思考时间

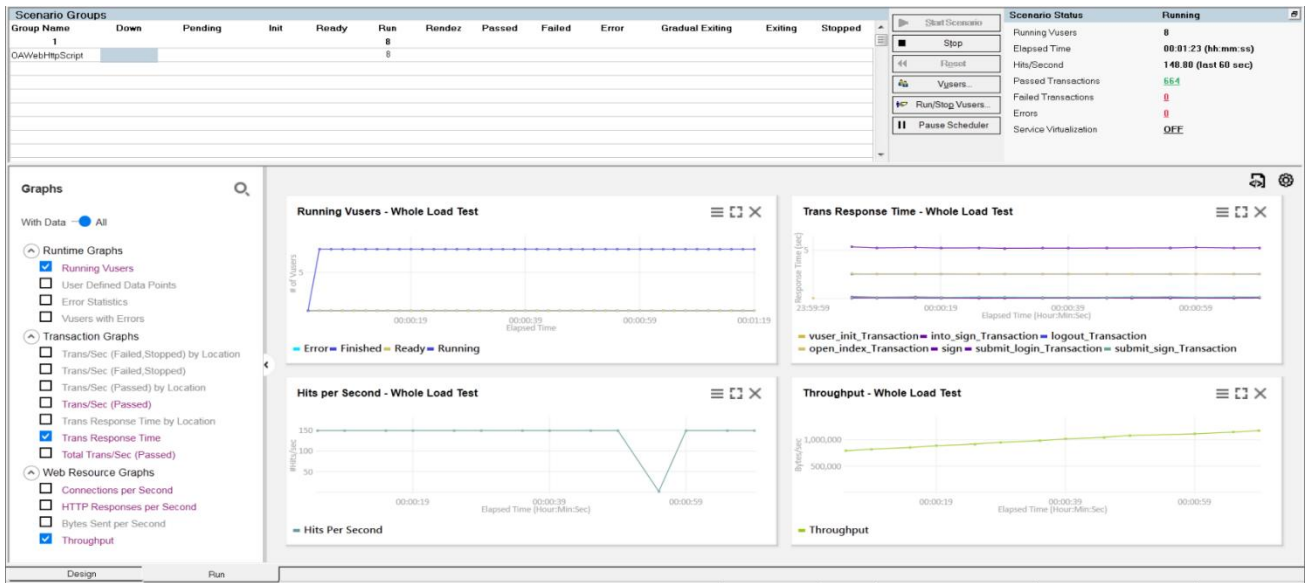
log 日志

Miscellaneous: 将每一个 Action 当成了一个事务执行

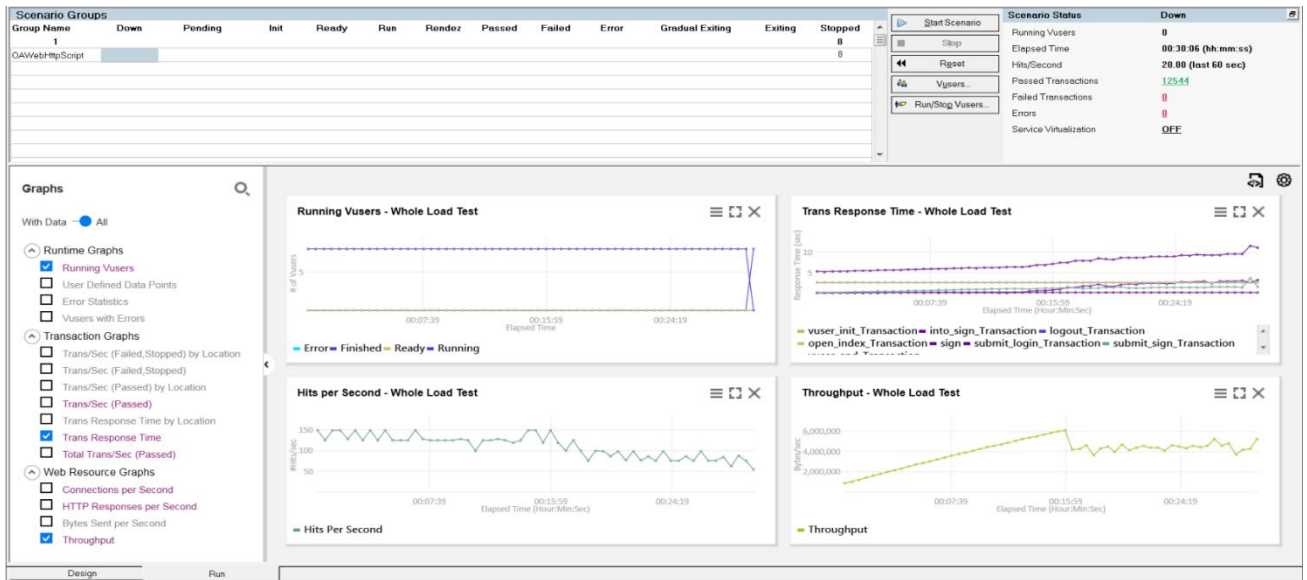
## 2. 场景执行与结果收集

利用 LoadRunner 监控运行的虚拟用户、事务响应时间、每秒请求数和吞吐量等。

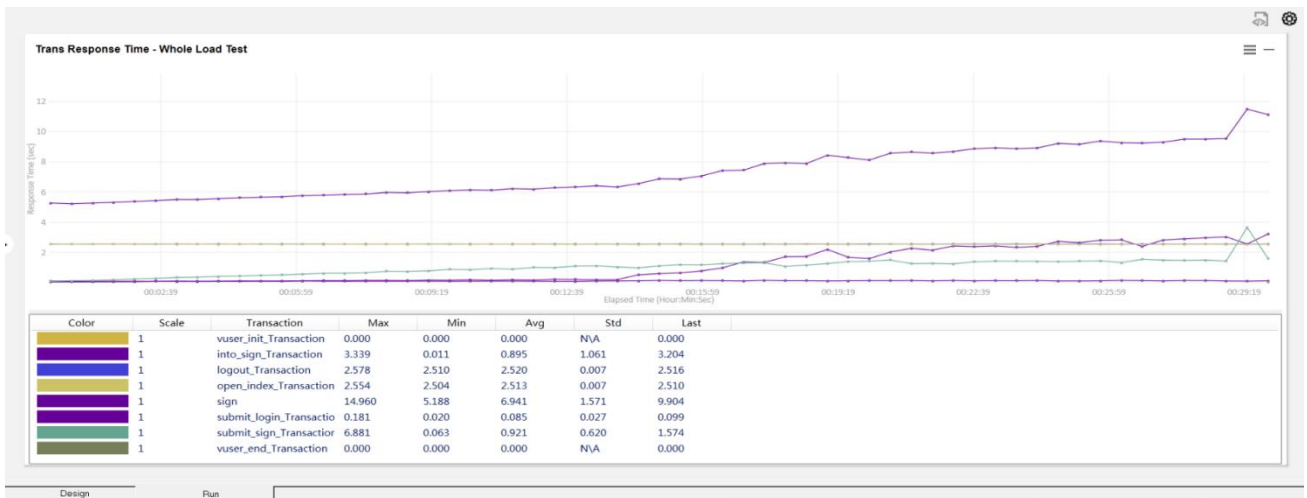
OA 考勤业务场景刚开始运行（1 分 23 秒），监控如下：



OA 考勤业务场景运行结束（30 分 06 秒），监控如下：



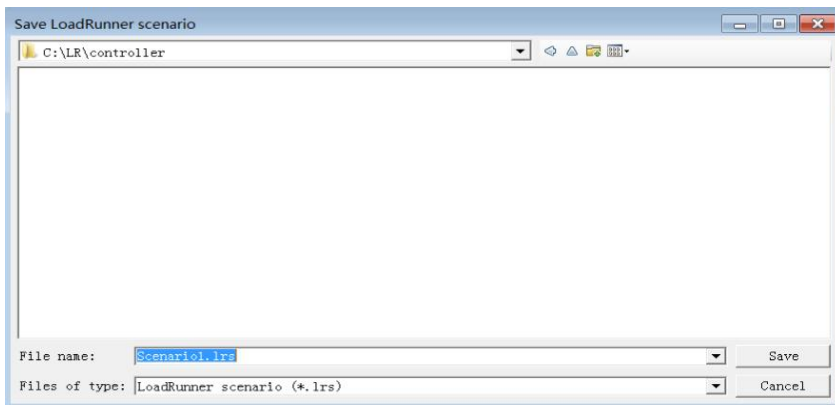
每个 Action 事务响应时间趋势图，如下：



这里没有失败事务，无 error；若有 error，分析 error 原因。

通过事务数 12544 个，失败事务数 0 个，error 数 0。

最后，保存场景（.lrs）。



### 3. 结果分析与报告输出

利用 LoadRunner 的“Analysis”功能组件对当前测试结果进行汇总分析。在 Controller 主界面单击工具栏中的【Analysis Results】按钮进行结果分析。

性能测试需求分析：业务响应时间、某业务成功率

性能风险分析：风险、优化和回归

#### 【OA 系统性能测试目标】

验证在 30 分钟内，完成 2000 次用户登录系统，进行考勤业务，最后退出；在业务操作过程中登录，考勤业务的服务器响应时间不超过 3 秒，并且服务器的 CPU 使用率、内存使用率均不超过 80%。

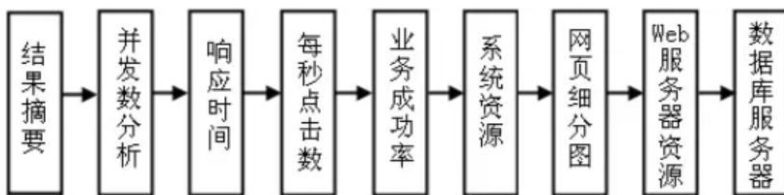
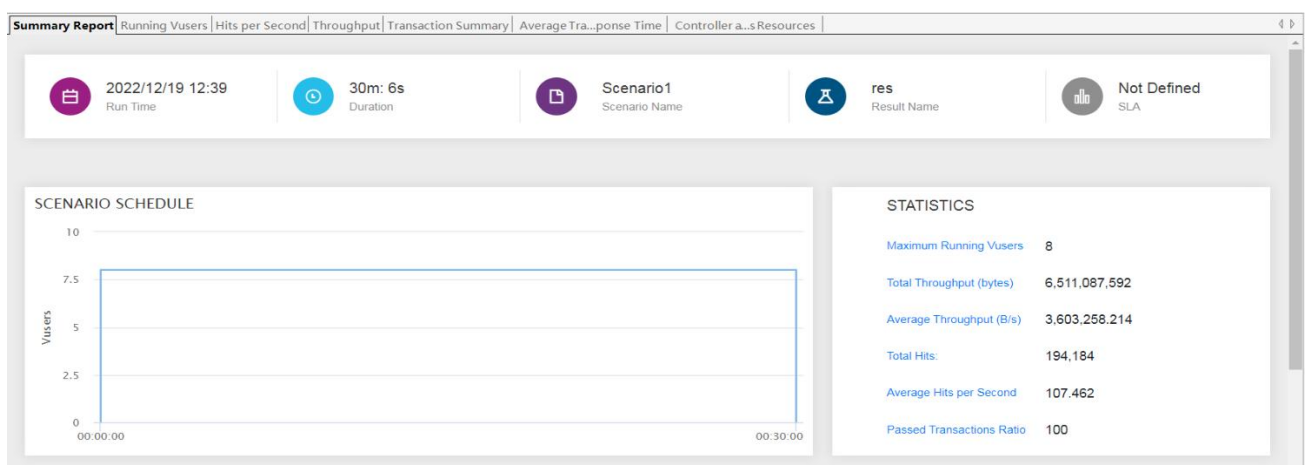


图 8-24 性能测试结果分析流程图

### 1. 摘要

概要中列出了场景执行情况、Statistics Summary（统计信息摘要）、Transaction Summary（事务摘要）以及 HTTP Responses Summary（HTTP 响应摘要）。

Summary Report 包括以下信息：



#### (1) 场景执行情况

测试场景的名称：Scenario1

结果名称: res

运行时间和持续时间: 2022. 12. 19, 30 分 06 秒 (与场景设计时间基本吻合)

场景计划: 虚拟用户数 8 个, 一开始即加载 8 个 Vusers, 到 30 分钟立即停止所有 Vusers。

(2) Statistics Summary (统计信息摘要)

最大运行 Vusers (并发数): 8 个

总吞吐量: 6 511 087 592 bytes

平均每秒吞吐量: 3 603 258 214 B/s

总请求数: 194 184

平均每秒请求数: 107 462

事务通过率: 100%(业务性事务全部通过)

对于吞吐量, 单位时间内吞吐量越大, 说明服务器的处理能力越好。请求数仅表示客户端向服务器发出的请求数, 与吞吐量一般成正比关系。

(3) Transaction Summary (事务摘要)

该部分给出了场景执行结束后相关 Action 的平均响应时间、通过率等情况。

Transaction Name	Minimum	Average	Maximum	Std. Deviation	90 Percent	Pass	Fail	Stop
vuser_init_Transaction	0	0	0.001	0	0.001	8	0	0
open_index_Transaction	0.001	0.003	0.074	0.004	0.004	2,088	0	0
submit_login_Transaction	0.013	0.085	0.299	0.054	0.162	2,088	0	0
into_sign_Transaction	0.005	0.89	3.352	1.069	2.708	2,088	0	0
submit_sign_Transaction	0.045	0.899	6.881	0.527	1.435	2,088	0	0
sign	0.131	1.888	9.951	1.504	4.229	2,088	0	0
logout_Transaction	0.002	0.011	0.153	0.013	0.025	2,088	0	0
vuser_end_Transaction	0	0	0	0	0	8	0	0

从该图得到每个 Action 的平均响应时间与业务成功率。

事务执行情况:

通过事务: 12 544

失败事务: 0

停止: 0

注意: 因为在场景的 “Run-time Settings” 的 “Miscellaneous” 选项中将每一个 Action 当成了一个事务执行, 故这里的事务其实就是脚本中的 Action。但原来的事务点 **sign** 并未删除, 故出现在统计图中。

(4) HTTP Responses Summary (HTTP 响应摘要)

该部分显示在场景执行过程中, 每次 HTTP 请求发出去的状态是成功还是失败。

Codes	Total	Per second
HTTP_200	192,096	106.307
HTTP_404	2,088	1.156

LoadRunner 共模拟发出了 194 184 (192 096+2 088) 次请求 (与 “统计信息摘要” 中的 “Total Hits” 一致)

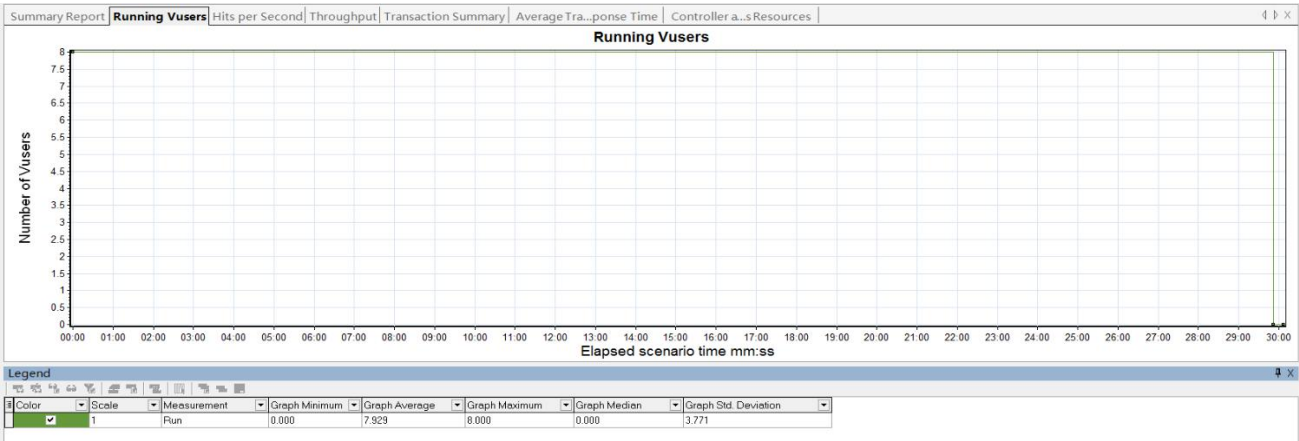
其中, “HTTP 200” 有 192 096 次, 而 “HTTP 404” 则有 2 088 次, 说明在本次过程中, 发出的请求大部分都能正确响应, 有部分请求失败了, 但未影响测试结果。

“HTTP 200” 表示请求被正确响应, 而 “HTTP 404” 表示文件或者目录未能找到。出现这种问题的原因是脚

本有些页面的请求内容并非关键点，不会影响最终的测试结果。

2. Running Vusers

Running Vusers（并发数）显示了在场景执行过程中并发数的执行情况，包括 Vuser 的状态、完成脚本的 Vuser 数量以及集合点统计信息。



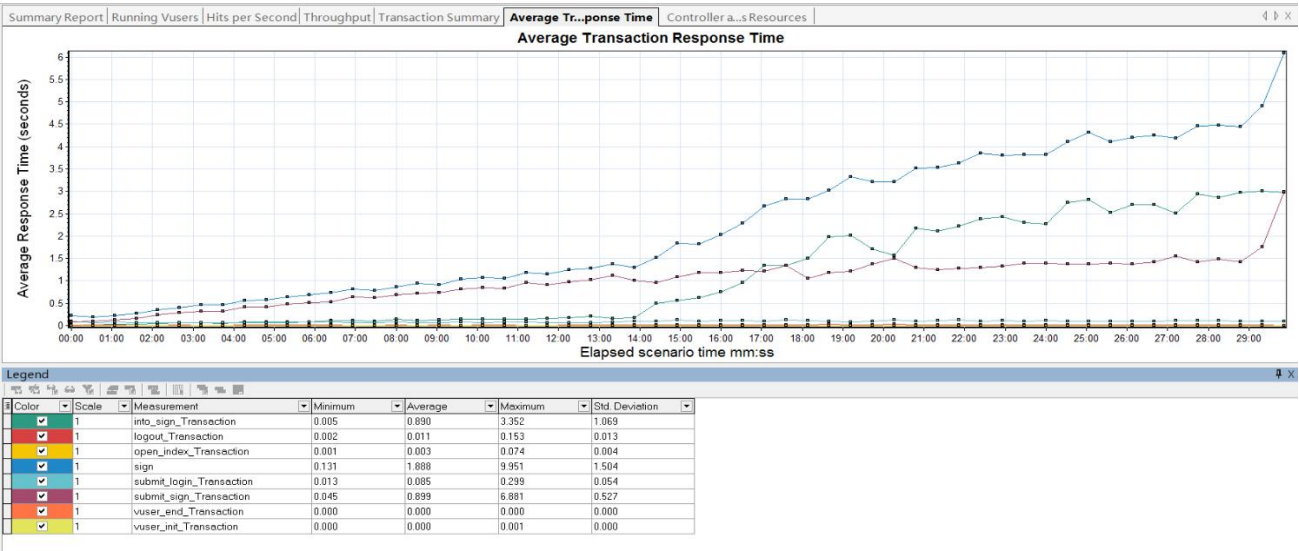
图显示 OA 系统考勤业务性能测试过程中 Vusers 的运行情况。Vusers 运行趋势与场景执行计划一致：最大虚拟用户数 8 个，一开始即加载 8 个 Vusers，到 30 分钟立即停止所有 Vusers。

表明在场景执行过程中，Vusers 是按照预期的设置运行的，没有 Vuser 出现运行错误。

3. 响应时间

性能测试指标要求登录、考勤业务操作的页面响应时间不超过 3 秒。

Average Transaction Response Time（平均事务响应时间图），如下：



登录业务对应的 Action 是 submit\_login，Average Time（平均响应时间）是 0.085 秒，小于 3 秒，达到要求。最大值是 0.299 秒，由图可见，随着访问量变大，响应时间递增，性能下降。

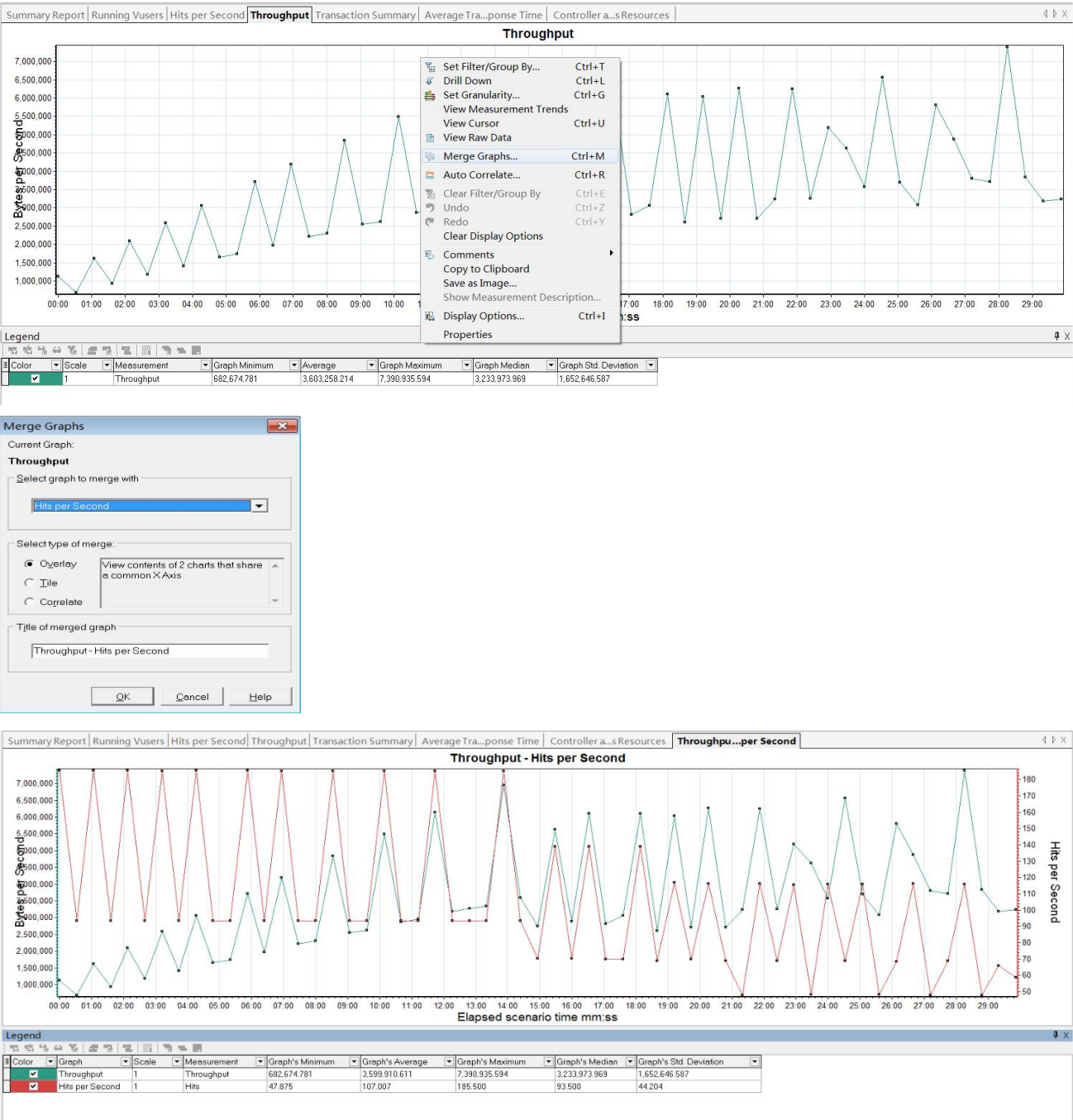
考勤业务提交对应的 Action 是 submit\_sign，Average Time（平均响应时间）是 0.899 秒，小于 3 秒，达到要求。最大值是 6.881 秒，由图可见，随着访问量变大，响应时间递增，性能下降。



#### 4. 点击数与吞吐量

Hits per Second（每秒点击数）反映了客户端每秒钟向服务器端提交的请求数量；如果客户端发出的请求数量越多，与之相对的 Average Throughput (bytes/second) 也应该越大；发出的请求会对平均事务响应时间造成影响。所以在测试过程中，往往将 Hits per Second（每秒点击数）、Average Throughput (bytes/second) 平均吞吐量、平均事务响应时间这三者结合起来分析。

生成 Hits per Second 与 Average Throughput (bytes/second) 的复合图：



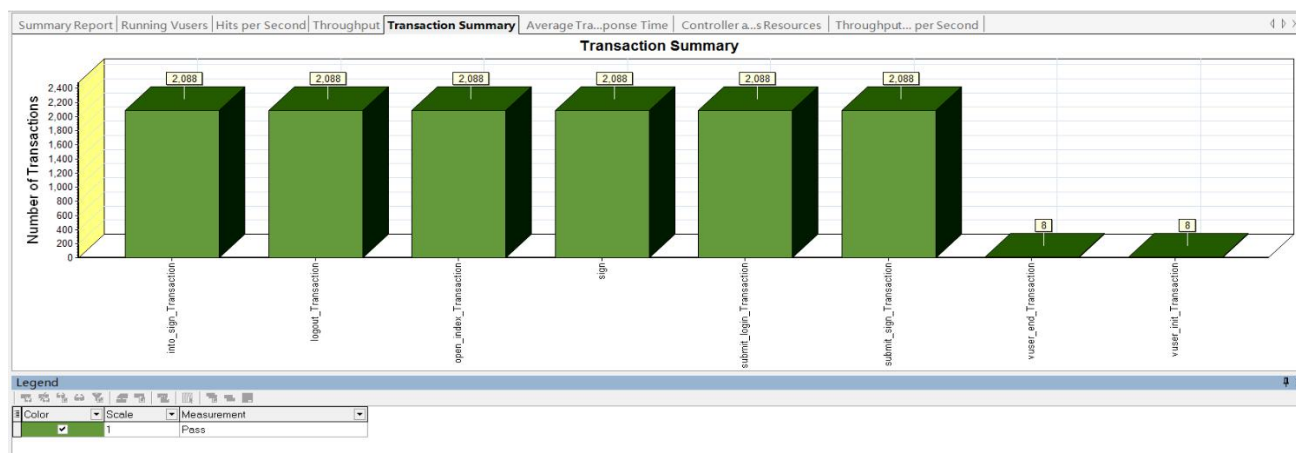
如果两种图形的曲线都正常并且基本一致，说明服务器能及时接受客户端的请求，并能够返回结果。

如果 Hits per Second 正常，而 Average Throughput (bytes/second) 不正常，则表示服务器虽然能够接受服务器的请求，但返回结果较慢，可能是程序处理缓慢。（上图为这种情况）

如果“Hits per Second”不正常，则说明客户端存在问题，这种问题一般是网络引起的或者录制的脚本有问题，未能正确模拟用户行为。

## 5. Transaction Summary

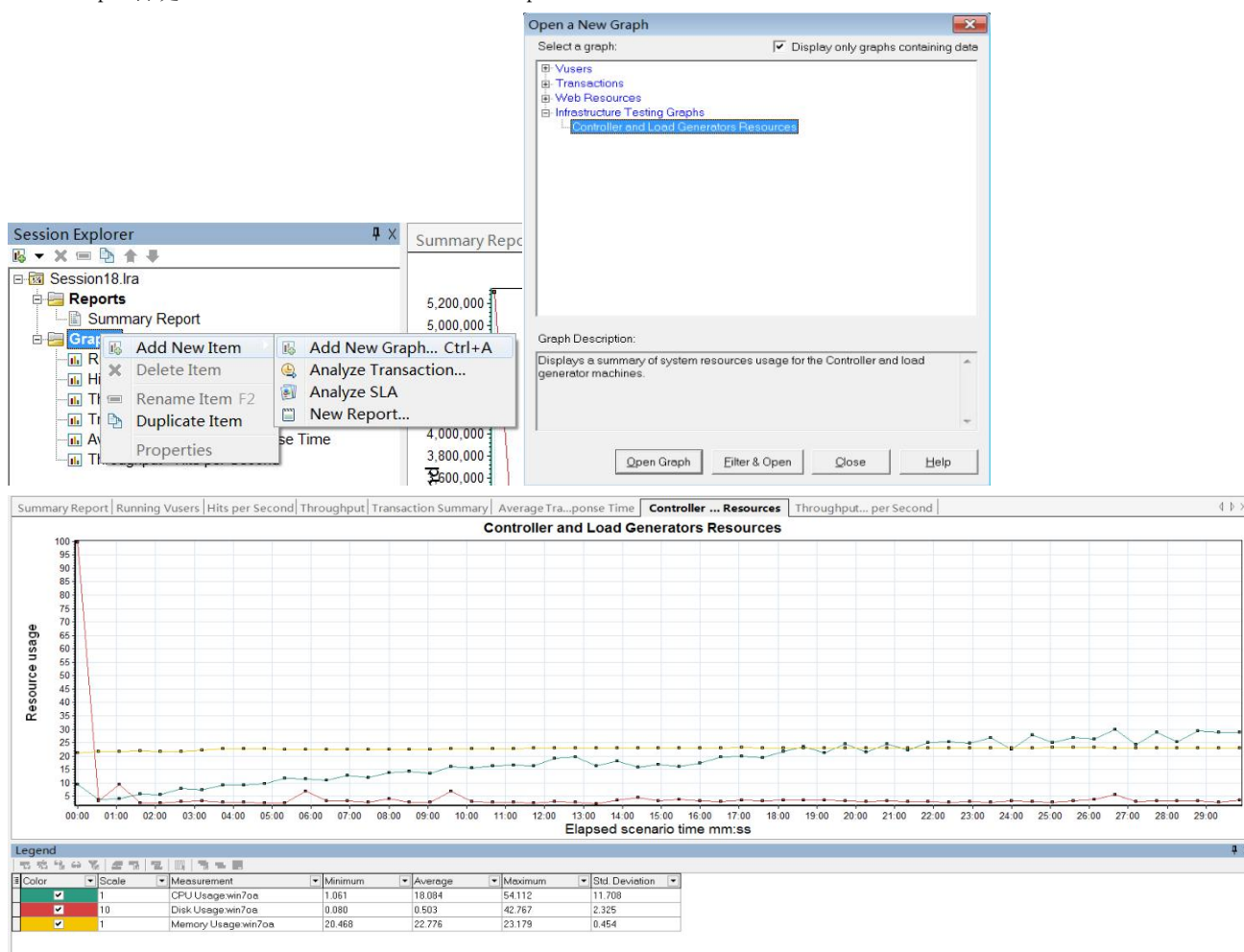
从图中可以看出，所有 Action 都是绿色的，即表示业务全部 Pass。同时，事务通过数为 2088，也就表明在 30 分钟的时间里，共完成了 2088 次考勤业务操作。



## 6. 系统资源

系统资源图显示场景执行过程中被监控的机器系统资源使用数据。一般情况下监控机器的 CPU、内存、网络、磁盘等各个方面。本次测试监控的是测试服务器的 CPU 使用率与内存使用率。

Graph 右键->Add New Item->Add New Graph





从图中可以看出，CPU 使用率递增，可用物理内存曲线都较为平滑。CPU 使用率平均值为 18.084%（<80%），最大值为 54.112%（超过 80%）。内存使用率平均值为 22.776%，最大值为 23.179%。

根据本次性能测试要求的“CPU 使用率不超过 80%，物理内存使用率不超过 80%”这两点来看，CPU 使用率平均值低于预期的 80%，内存使用率明显低于预期的 80%，故 CPU 及内存都达标。

测试结果记录表：

测试项	目标值	实际值	是否通过
登录业务响应时间	≤3 秒	0.085	Y
考勤业务响应时间	≤3 秒	0.899	Y
登录业务成功率	100%	100%	Y
考勤业务成功率	100%	100%	Y
登录业务总数	30 分钟完成 2000	2088	Y
考勤业务总数	30 分钟完成 2000	2088	Y
CPU 使用率	≤80%	18.084%	Y
内存使用率	≤80%	22.776%	Y

从上述数据来看，本次测试所有指标达到预期性能指标，测试通过。

#### 4. 性能调优与回归测试

测试结果分析完成后，即可进行性能问题确定与优化操作。通常情况下，系统出现性能问题的表象特征有以下几种。

- (1) 响应时间平稳但较长。测试一开始，响应时间就很长，即使减少 Vuser 数量，减少负载，场景快执行结束，响应时间仍然很长。
- (2) 响应时间逐步变长。测试过程中，负载不变，但运行时间越长，响应时间越长，直至出现很多错误。
- (3) 响应时间随着负载变化而变化。负载增加，响应时间变长；负载减少，响应时间下降，资源使用率也下降。
- (4) 数据积累导致锁定。起初运行正常，但数据量积攒到一定量，立刻出现错误，无法消除，只能重启系统。
- (5) 稳定性差。特定场景或运行周期很长以后，突然出现错误，系统运行缓慢。

以上是性能测试过程中碰到的几种性能有问题的特征。一旦出现上述几种情况，基本可以判定系统存在性能问题。接下来针对具体问题具体分析，从而发现问题并提出解决办法。

响应时间长，系统越来越慢，出现业务错误，通常由以下几种情况造成。

- (1) 物理内存资源不足；
- (2) 内存泄露；
- (3) 资源争用；
- (4) 外部系统交互；
- (5) 业务失败时频繁重试，无终止状态；
- (6) 中间件配置不合理；
- (7) 数据库连接设置不合理；
- (8) 进程/线程设计错误。

## 四、实验报告要求

请记录场景设计与实现步骤，尝试分析场景执行结果。

## 五、实验注意事项

提交场景设计文件和结果分析文件。

例如：

