**Jmeter实训报告**

## 1、录制（小组分模块进行性能测试，我负责查询部分）

使用Badboy进行录制

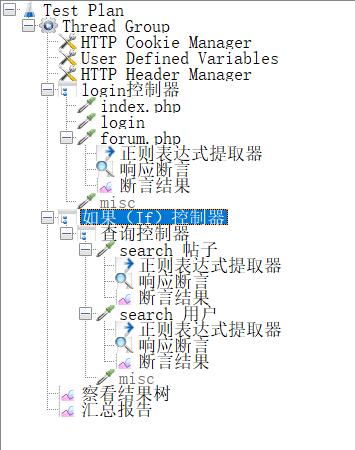
登录

查询帖子

查询用户

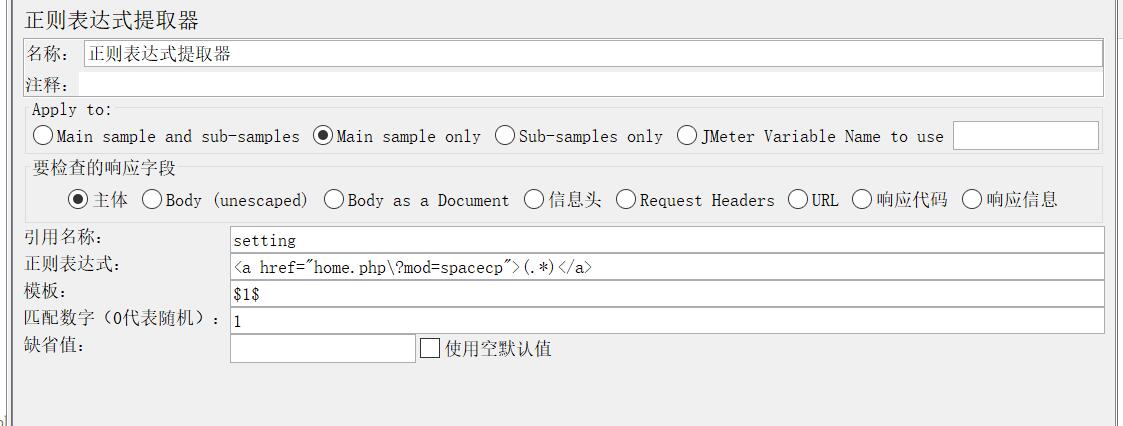
## 2、 加事务控制器

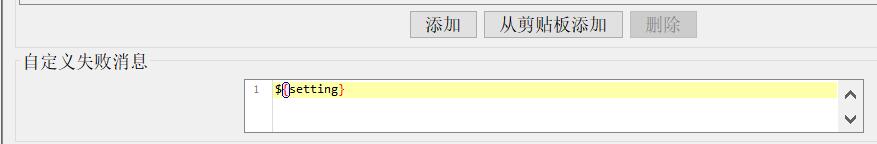
对录制好的脚本添加事务控制器

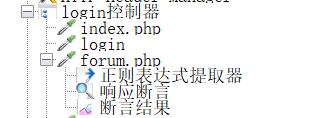


## 3、 加检查点

登录：登录成功后网页出现 设置 按钮，以此作为检查点







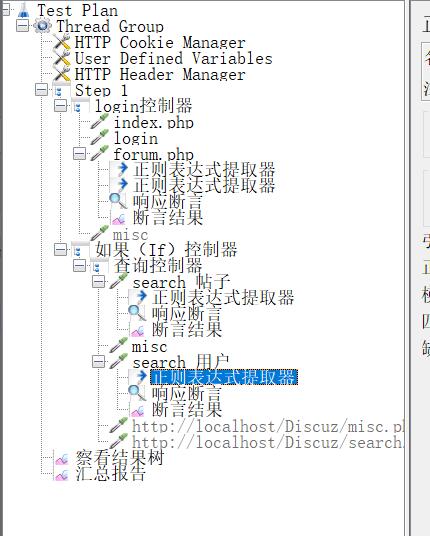
查询帖子：检查查询出的帖子标题



查询用户：检查查询出的用户



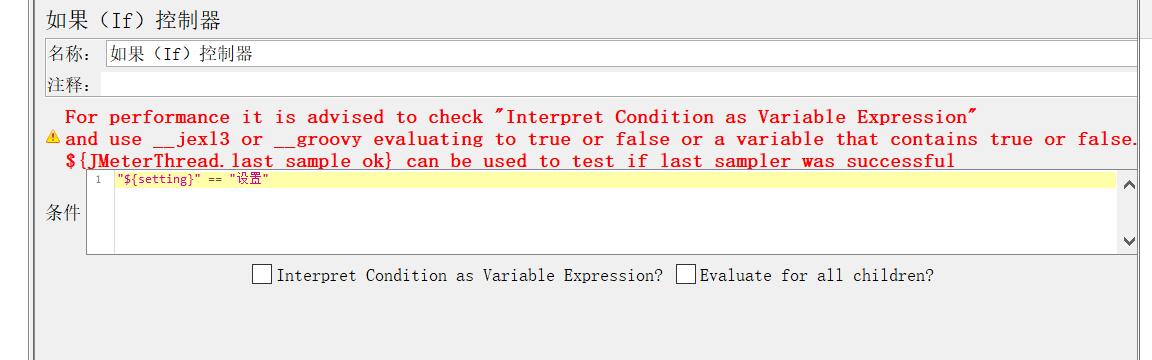
分布如下：

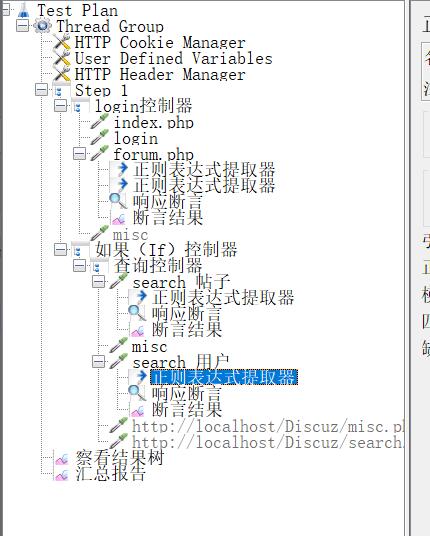


运行无错

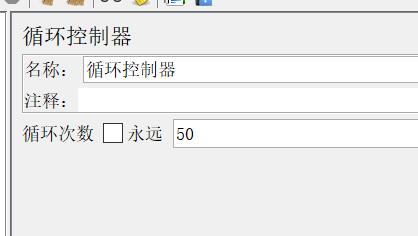
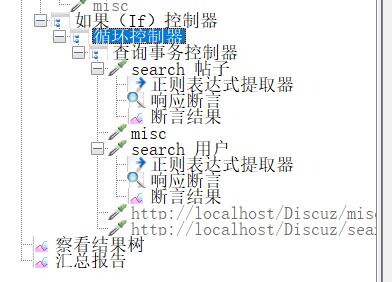
## 4、 加逻辑控制

如果登录成功，则可以进行查询用户

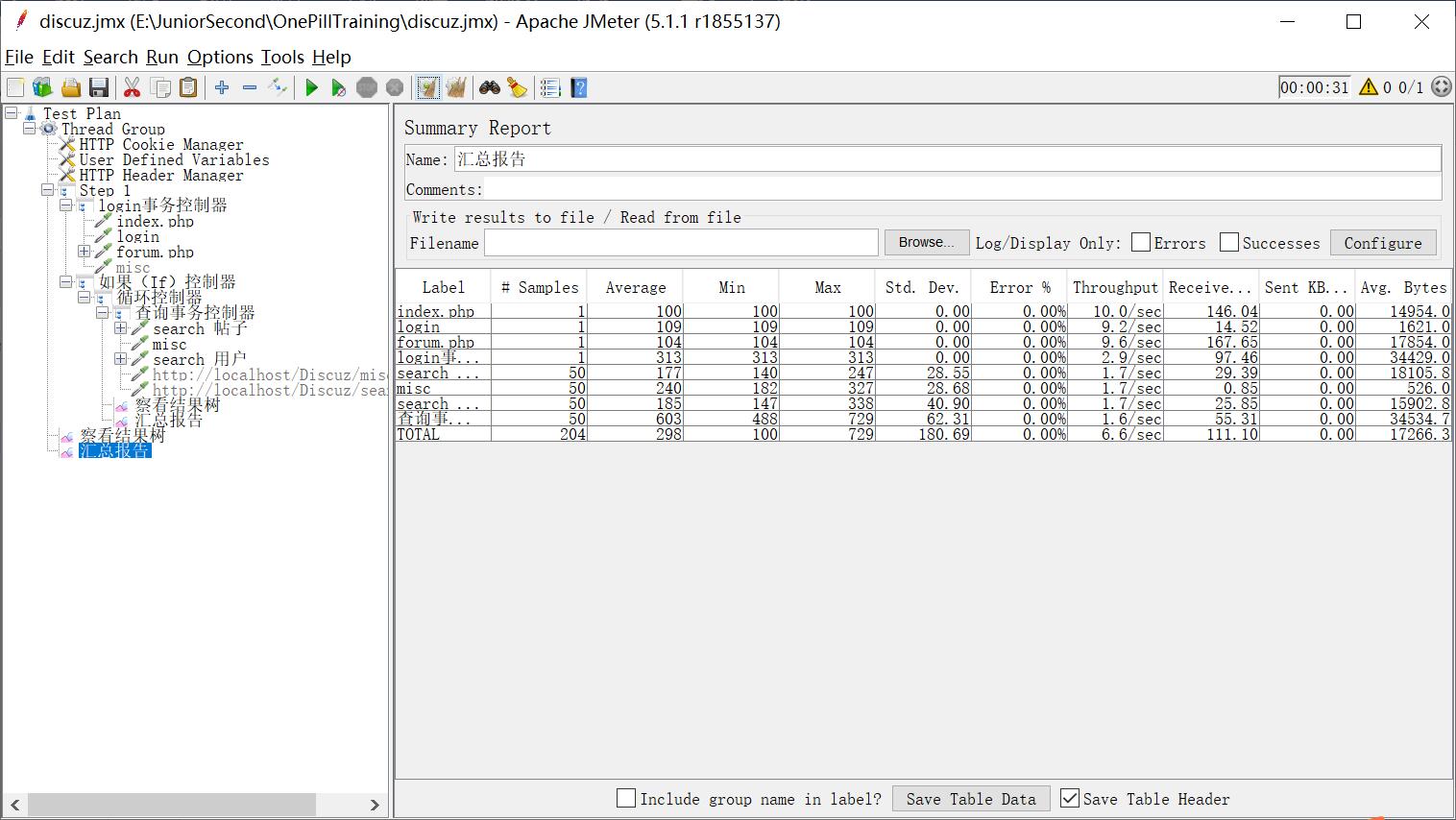


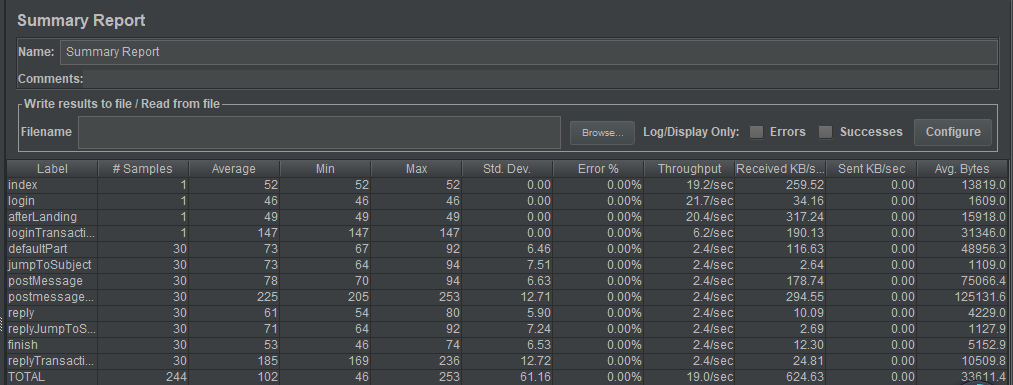


当成功登录后，连续做50次查询（在if 控制器里加一个循环控制器Loop Controller）



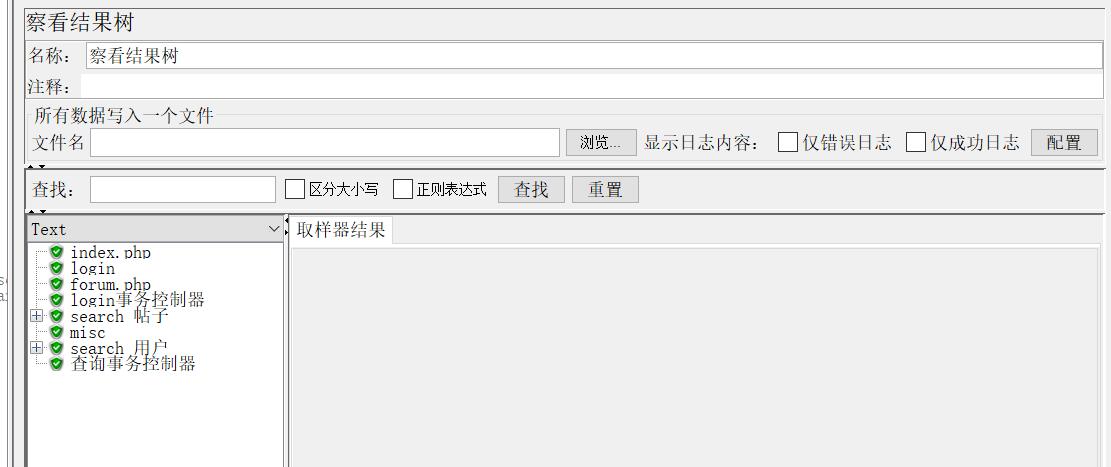
查询运行50次结果如下：



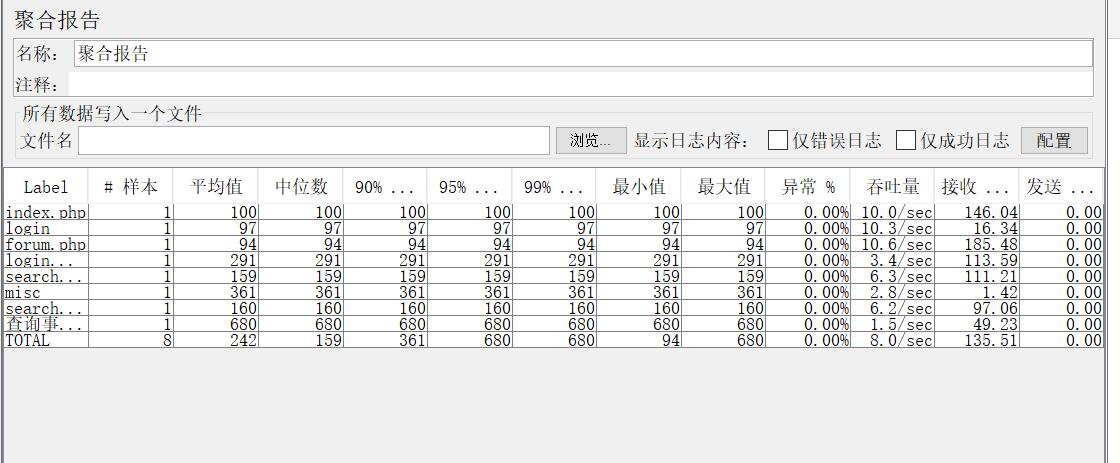


## 5、 加监听器

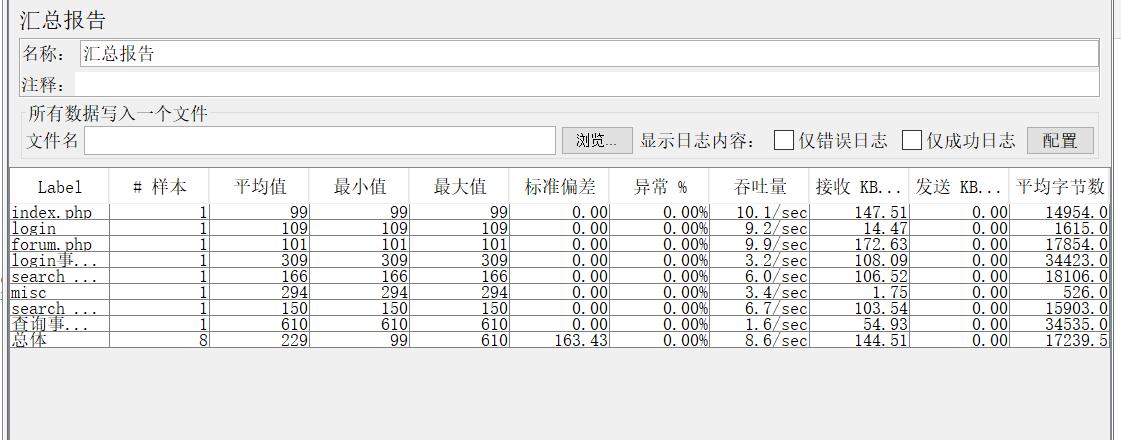
树形监听器（View Results Tree）：如下图，运行成功



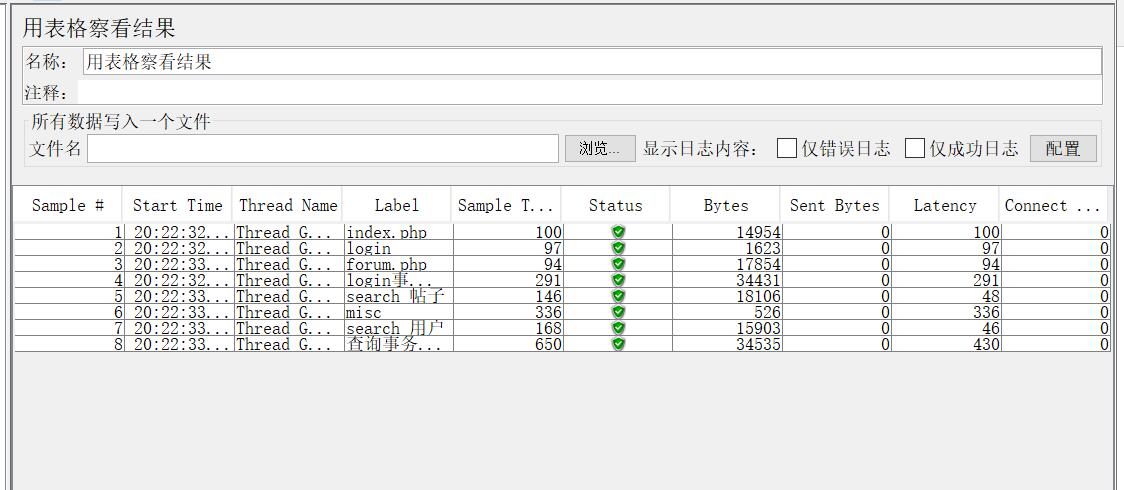
聚合报告（Aggregate Report）：如下图，统计每个请求的响应信息并提供请求数，平均值，最大，最小值，中位数、90%、95%用户响应时间、错误率，吞吐量(以请求数/秒为单位)和以kb/秒为单位的吞吐量



汇总报告(Summary Report)



用表格查看结果（View Results in Table）：

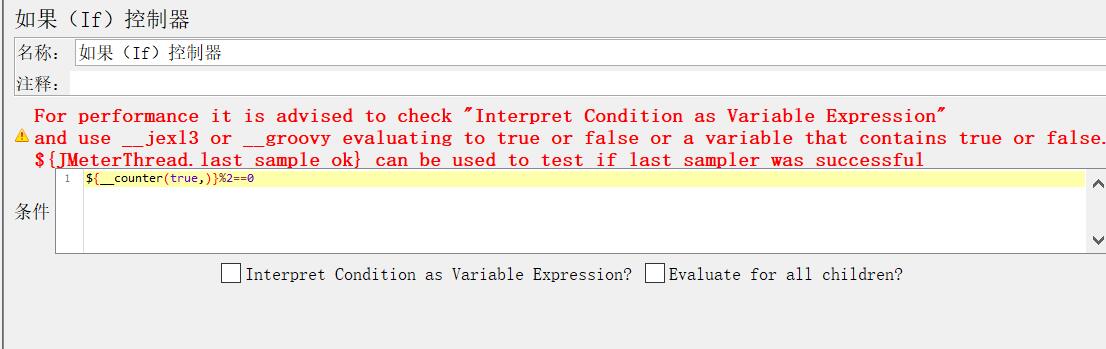


还可以进行图形化（以点击率为例）：

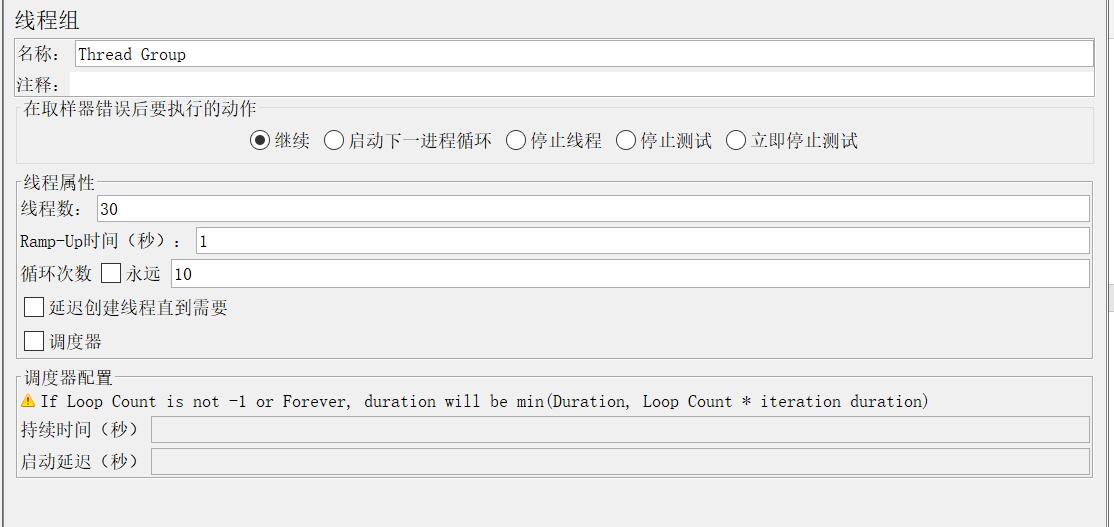
## 6、 创建场景，结果分析

通过If Controller和线程迭代计数函数（ ${\_\_counter(true,)} ）来实现用户组式场景

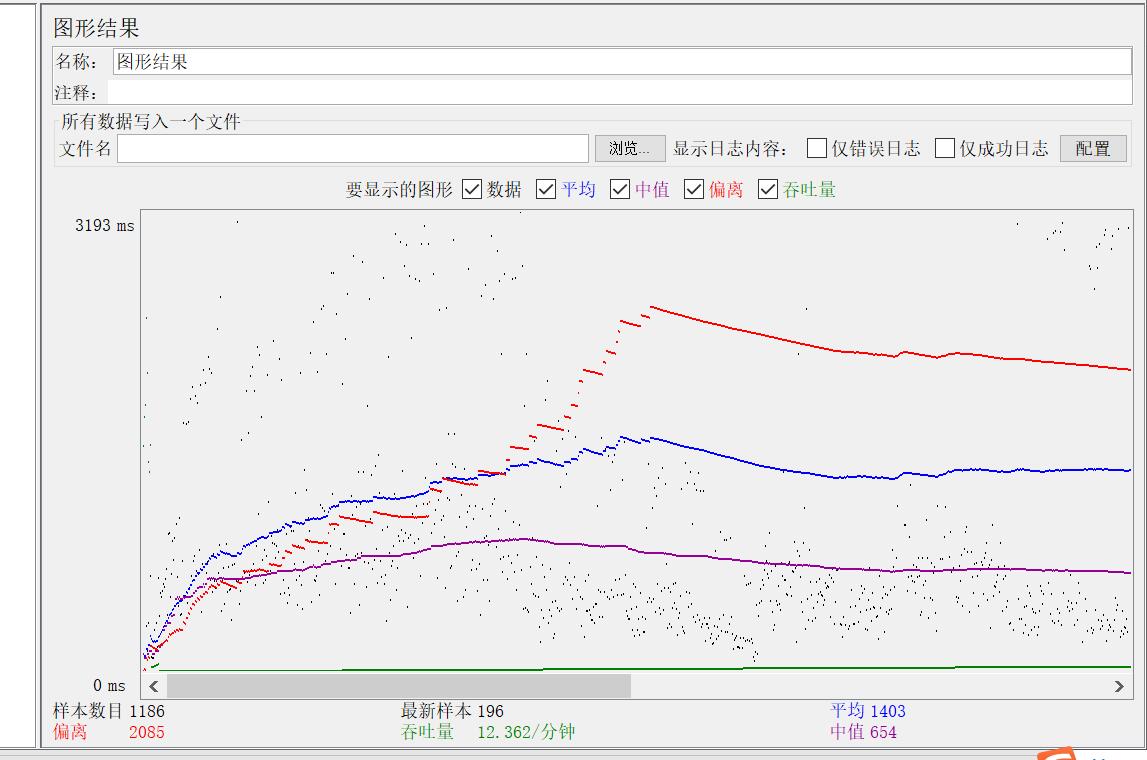
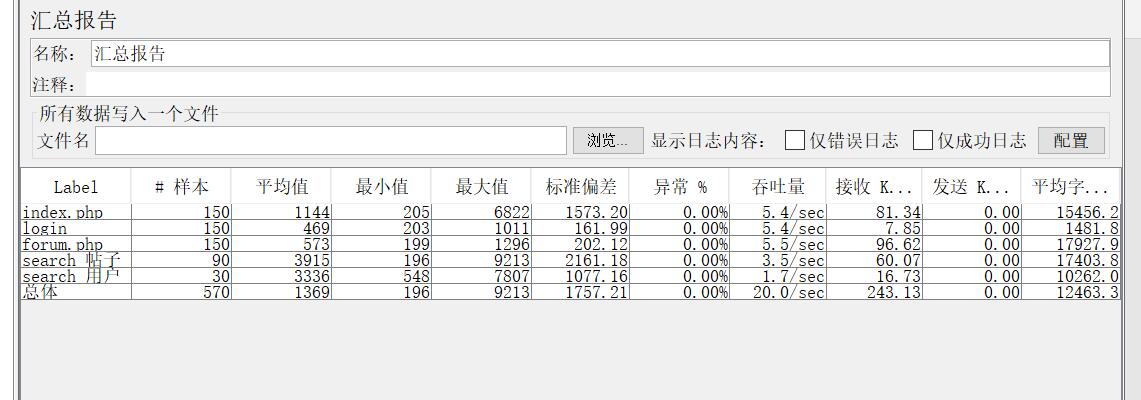
30并发量：20%用户做登录、30%查询帖子、50%查询用户

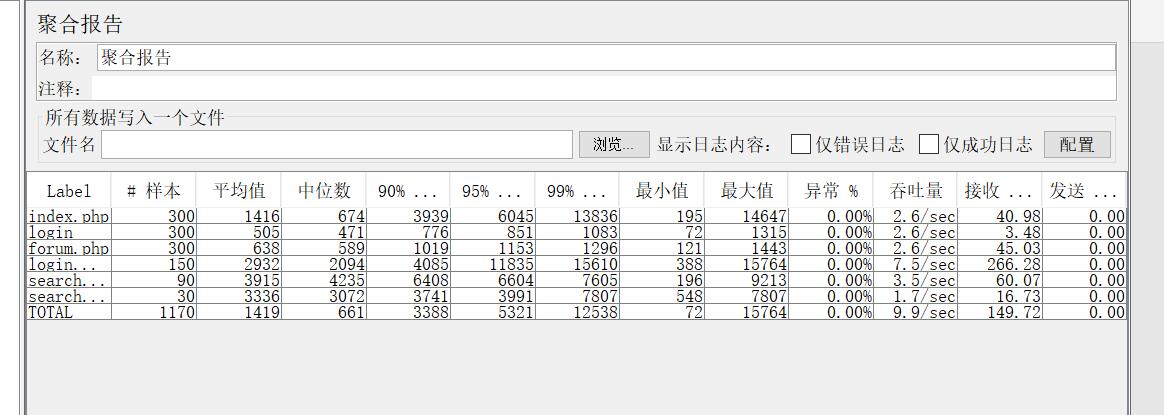


修改用户数为30，运行



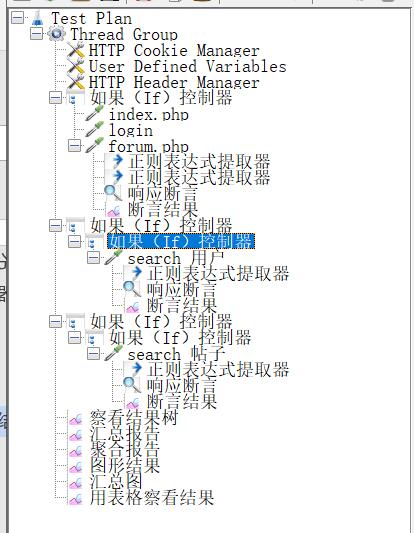
此时运行结果



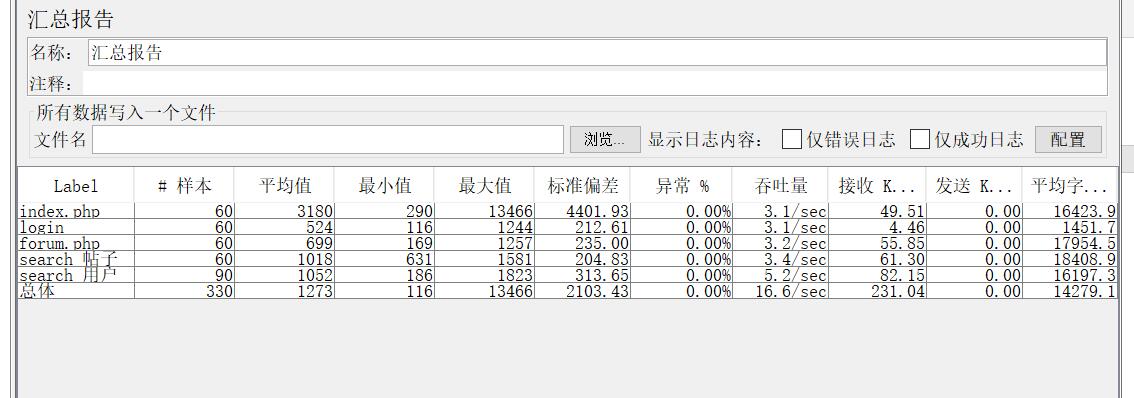


不符合20% 30% 50%,猜测是检查点的ifControler的问题，

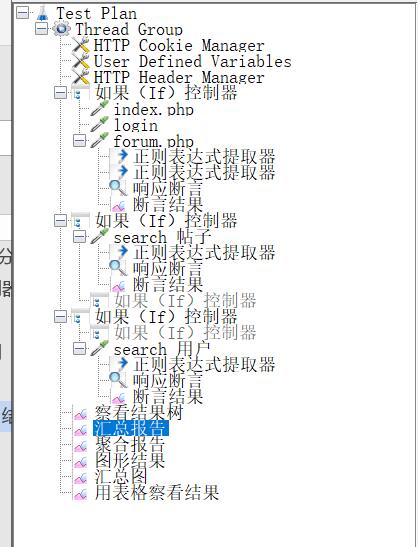
将判断是否登录成功的if逻辑控制器，分别作为带有\_\_counter函数的控制器的子节点，如下图所示



结果不符合20% 30% 50%



此时考虑去掉检查点（可能是系统的问题）

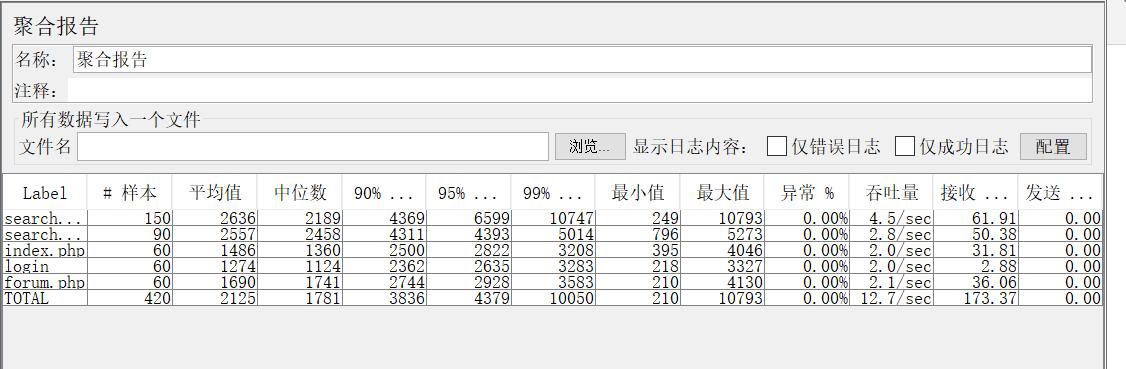


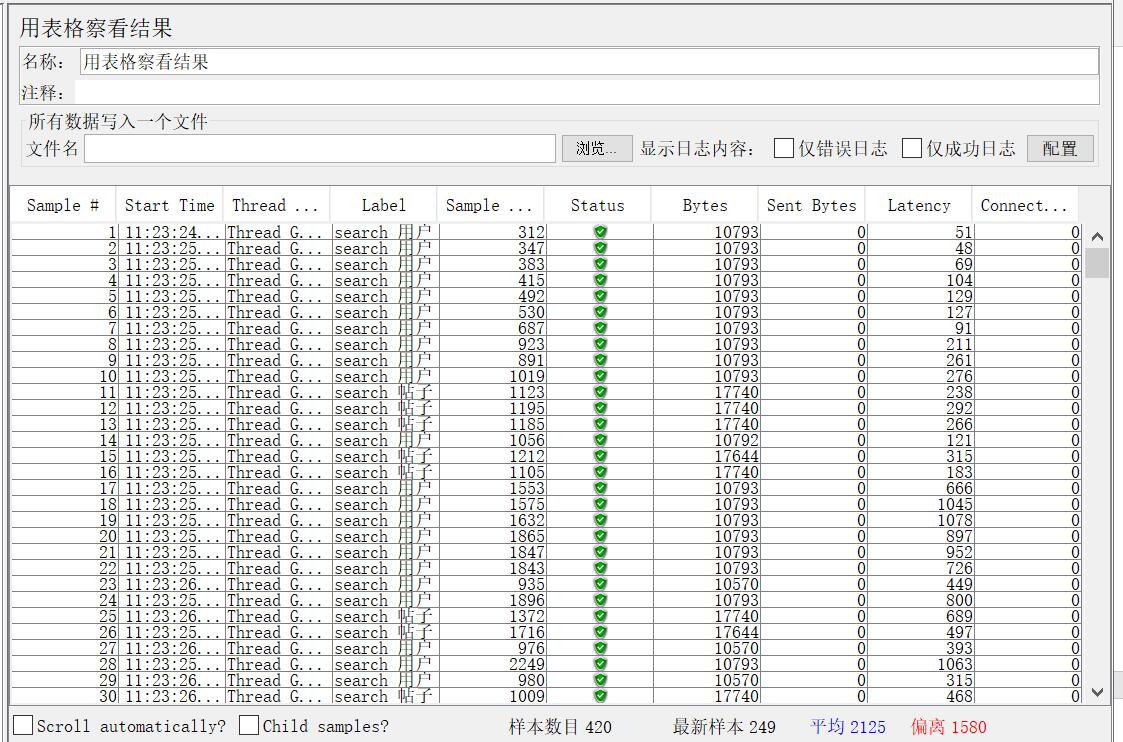
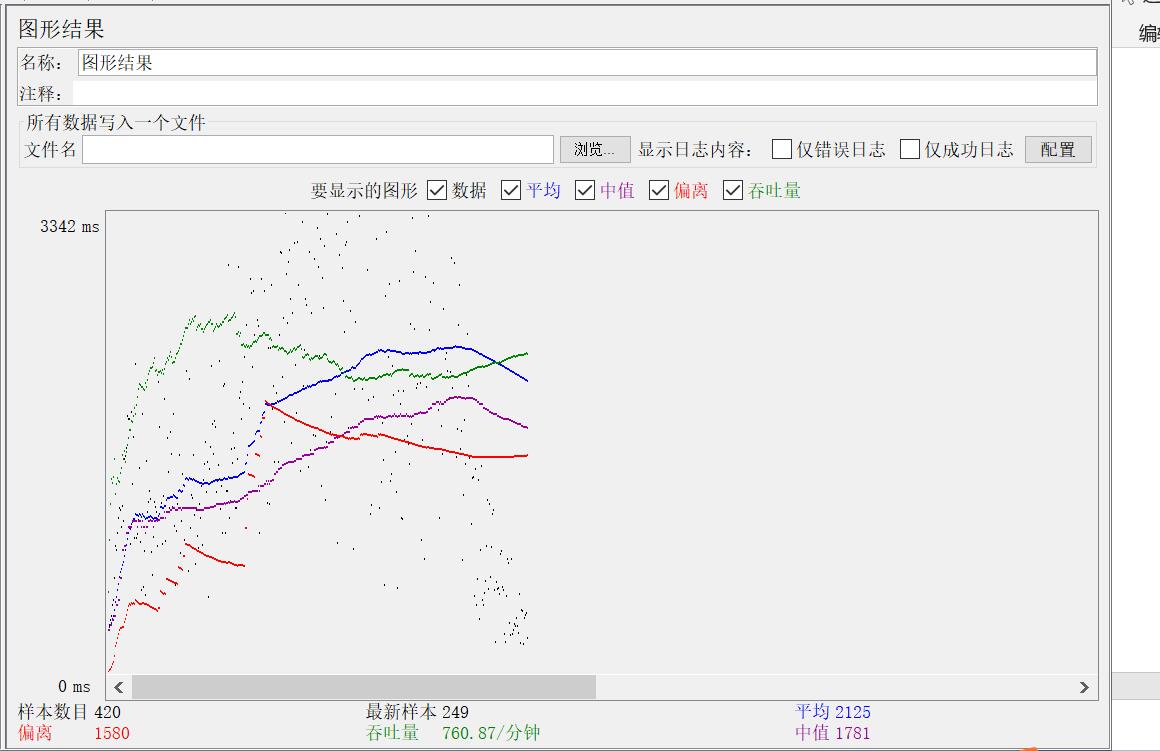


此时满足并发条件

对于各个请求期间得出的样本 平均值 最小值 标准偏差 异常百分比 吞吐量 平均每秒接受与发送数据量，以及平均字节数均可以清晰的得出。

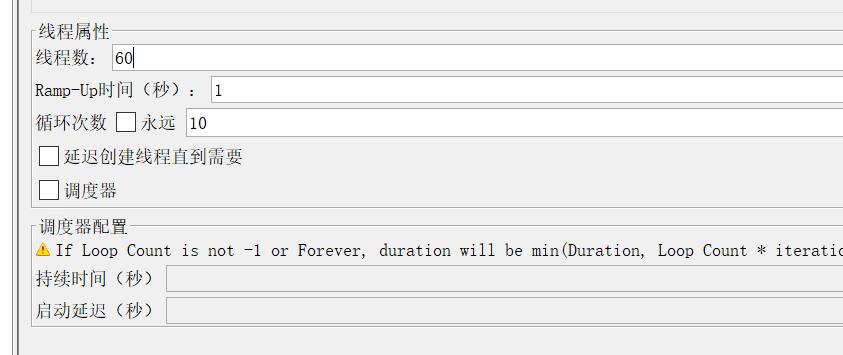
其他监听器结果如下：

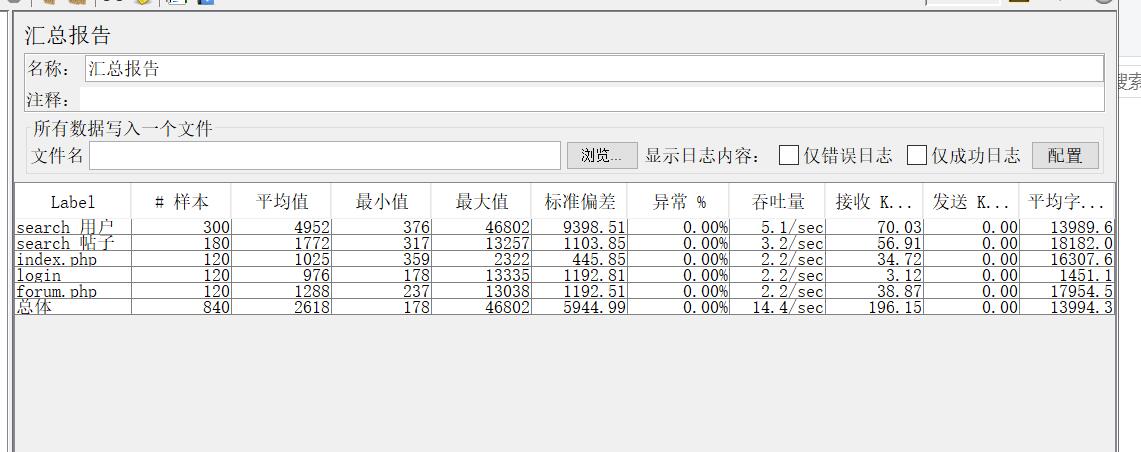


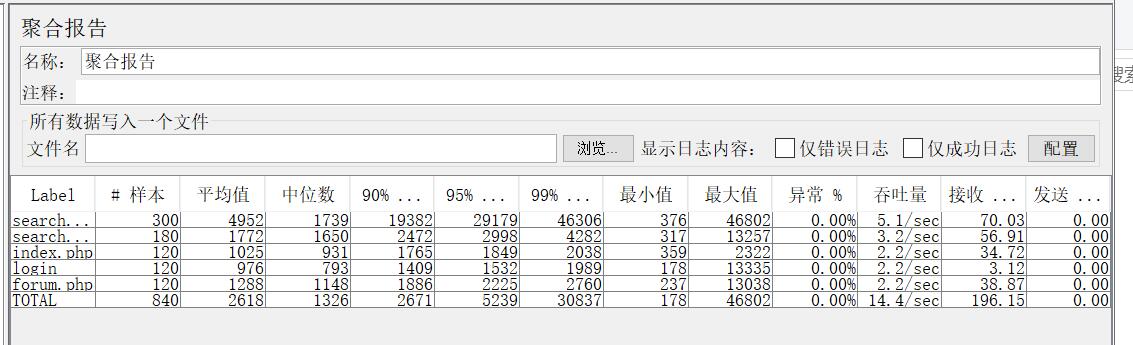


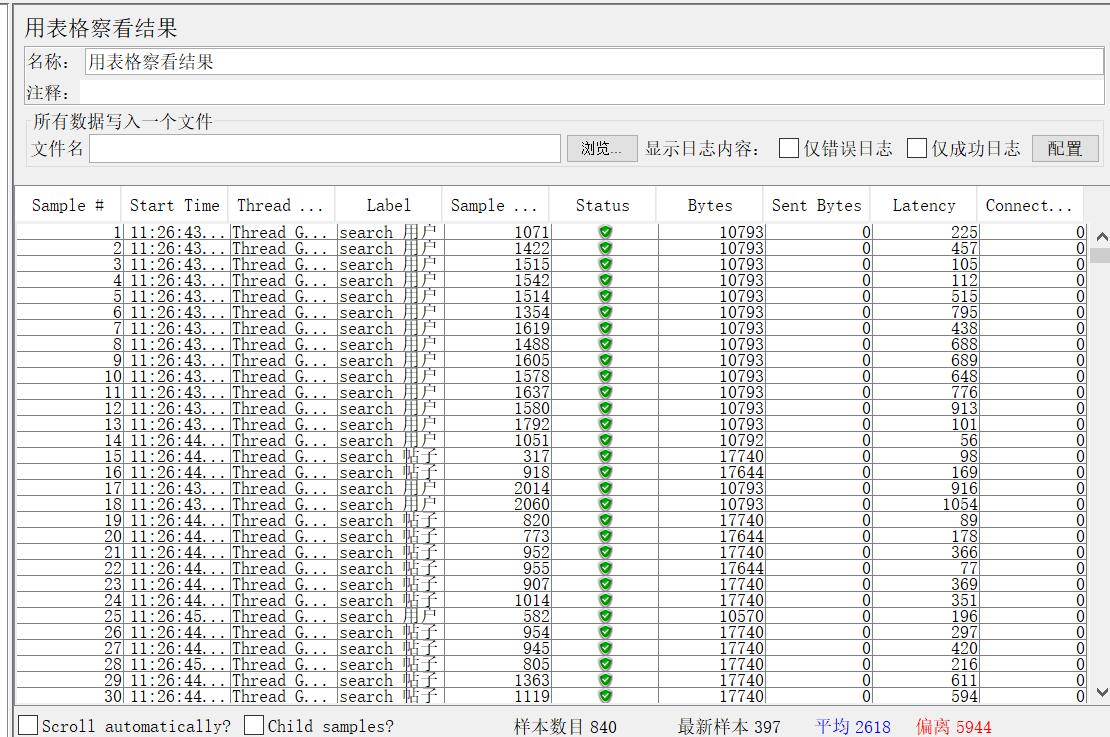
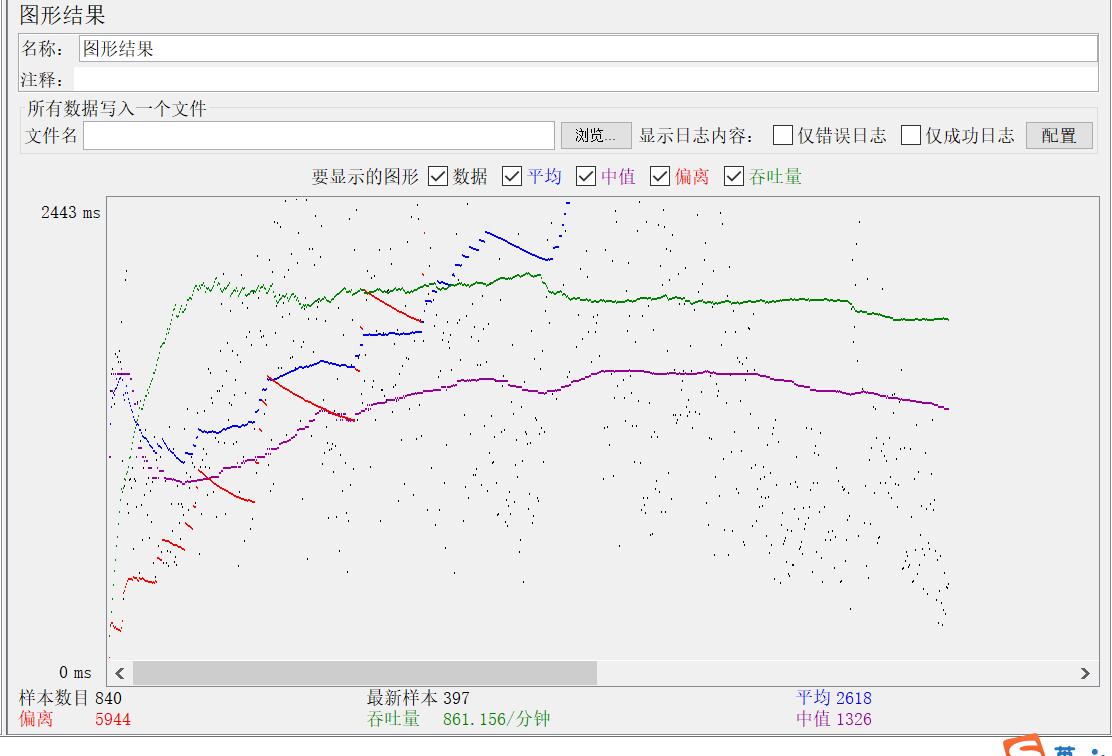
60并发量：20%用户登录、30%查询帖子、50%查询用户

修改用户数为60，监听器报告如下：









100并发量：20%用户做登录、30%做发帖、50%做回帖

修改用户数为60，监听器报告如下：

