

小微金服

性能测试报告



网新恒天软件有限公司

2014.06.18

撰稿人：宗婷婷

时间：2014/06/18

Version: 0.1

目录

1. 概论.....	5
1.1 测试结果汇总.....	5
1.2 所有场景性能问题汇总分析.....	7
1.3 系统整体评价.....	8
2. 测试简介.....	8
2.1 测试背景.....	8
2.2 测试目标.....	8
2.3 测试范围.....	8
2.4 性能测试指标要求.....	9
3. 测试环境.....	9
3.1 网络拓扑图.....	10
3.2 软硬件配置.....	10
3.3 软件参数及相关配置.....	11
4. 测试方案.....	11
4.1 主要场景.....	11
4.2 测试数据准备.....	13
4.3 测试资源监控指标.....	14
4.4 测试类型与场景分配.....	14
4.5 结果数据搜集方法.....	16
5. 测试实施情况.....	17

5.1	测试时间和地点.....	17
5.2	参加人员.....	17
5.3	测试约定.....	17
6.	测试结果.....	17
6.1	负载测试	17
6.2	尖峰冲击式测试	30
6.3	耐久度测试	32
6.4	其他数据分析和折算	33
7	系统评价.....	33

1. 概论

本次针对小微金服系统的性能测试开始于 2014-06-09，结束于 2014-06-25。本次测试的主要目的是为了评估小微金服系统的性能表现，同时也为了帮助小微金服系统项目组找到潜在的性能相关的软硬件瓶颈，系统 bug 以及程序错误等等。

我们模拟系统日常工作的 6 个主要场景和流程，以启发式的方法逐渐对系统加压，评估系统的平均响应时间，吞吐率，CPU 使用率，内存使用率等等性能指标。

本次性能测试及调优显著地提高了小微金服系统的质量，包括软件的可靠性，硬件的配置合理性，在上线前有效地减少了系统的潜在风险。

下面的章节将详细介绍本次测试的背景，场景，方法，结果，结论等等具体细节。

1.1 测试结果汇总

1.1.1 系统容量

1. 首页加载场景：当并发用户小于 300 人时，系统吞吐率随并发用户增多而线性增长；当并发用户达到 300 人时，系统达到最大吞吐率 275.34TPS；当并发用户大于 300 人时，系统吞吐率保持在 250TPS 左右。

2. 新建融资项目场景：当并发用户小于 50 人时，系统吞吐率随并发用户增多而线性增长；当并发用户达到 50 人时，系统达到最大吞吐率 275.34TPS；当并发用户大于 50 人时，系统吞吐率保持在 90TPS 左右。

3. 投资人申购场景：当并发用户小于 30 人时，系统吞吐率随并发用户增多而线性增长；当并发用户达到 30 人时，系统达到最大吞吐率 27.12TPS；当并发用户大于 30 人时，系统吞吐率成小幅下降趋势，总体保持在 24TPS 左右。

4. 自动申购场景：自动申购场景为单线程操作，并发数限定为 1 人，在 4 种类型的融资包数据下的系统吞吐率分别为 11.90TPS、5.99TPS、2.70TPS、0.15TPS。

5. 综合场景：当并发用户小于 500 人时，系统吞吐率随并发用户增多而线性增高；当并发用户达到 500 人时，系统达到最大吞吐率 755.51TPS；当并发用户大于 500 人时，系统吞吐率随并发用户增多而线性降低，当并发人数达到 1500 人时，吞吐率为 299TPS。

6. 登陆场景：当并发用户小于 10 人时，系统吞吐率随并发用户增多而线性增高；当并发用户达到 10 人时，系统达到最大吞吐率 14.76TPS；当并发用户大于 10 人时，系统吞吐率保持在 14.2TPS 左右。

对比“首页加载”场景和“登陆”场景可知，登陆的验证请求是影响用户登陆操作的主要因素，而首页静态资源的加载对用户登陆的影响较小。

1.1.2 系统响应性

场景	体验良好 (小于 3 秒)	体验一般 (3-5 秒)	体验较差 (大于 5 秒)
首页加载	小于 1600 人	1600-2000 人	大于 2000 人
新建融资项目	小于 50 人	50-70 人	大于 70 人
投资人申购	小于 40 人	40-60 人	大于 60 人
综合场景	小于 500 人	500-1000 人	大于 1000 人
登陆	小于 20 人	20-40 人	大于 40 人

表 1.1 系统响应性

自动申购场景为单线程操作，并发数限定为 1 人，在 4 种类型的融资包数据下的平均响应时间分别为 342ms、1188ms、3603ms、80070ms。

1.1.3 系统稳定性

场景	错误率	资源占用	对应并发数
首页加载	0.02%	CPU 使用率<80% 内存使用率<80%	2000 人
新建融资项目	0%	CPU 使用率<80% 内存使用率<80%	100 人
投资人申购	0%	CPU 使用率<80% 内存使用率<80%	100 人
自动申购	0%	CPU 使用率<80% 内存使用率<80%	单线程操作，并发数 限定为 1 人
综合场景	0.03%	CPU 使用率<80%	1500 人



		内存使用率<80%	
登陆	0%	CPU 使用率>90% 内存使用率<80%	40 人

表 1.2 系统稳定性

1.2 所有场景性能问题汇总分析

场景中问题 url 汇总：

场景	问题 url	问题描述	调优情况
新建融资项目	/web/product/getproductlist	url 平均响应时间相对较长	未调优
投资人申购	/web/market/financing/search /web/market/financing/purchase	url 平均响应时间相对较长	未调优
自动申购	/web/market/autosubscribe/drawdone/融资包编号	url 平均响应时间相对较长	未调优
综合场景	/web/myaccount/accountoverview /web/myaccount/xwbooverview	url 平均响应时间相对较长	未调优
登陆	/web/members/signin/authc_int?userName=dztz1 &password=123456	url 平均响应时间相对较长	未调优

表 1.3 问题 url 汇总

场景中可疑 sql 语句汇总：

可疑 sql 语句	单次执行时间	调优情况
update um_user set login_failure_ct=0, last_login_ts=current_timestamp where lower(user_name) = lower(:1)	16.55s	未调优
* select generatedAlias0 from UserPo as generatedAlias0 where upper(generatedAlias0.username)=upper(:param0) */ select userpo0_.USER_ID as USER_ID1_63_, userpo0_.CREATE_TS		未调优



as CREATE_TS2_63_, userpo0_.CREATE_OPID as CREATE_OPID3_63_, userpo0_.EMAIL as EMAIL4_63_, userpo0_.LOGIN_FAILURE_CT as LOGIN_FAILURE_CT5_63_, userpo0_.LAST_LOGIN_TS as LAST_LOGIN_TS6_63_, userpo0_.LAST_MNT_OPID as LAST_MNT_OPID7_63_, userpo0_.LAST_MNT_TS as LAST_MNT_TS8_63_, userpo0_.MOBILE as MOBILE9_63_, userpo0_.NAME as NAME10_63_, userpo0_.OWNER_ID as OWNER_ID11_63_, userpo0_.PASSWORD as PASSWORD12_63_, userpo0_.REGION_CD as REGION_CD13_63_, userpo0_.STATUS as STATUS14_63_, userpo0_.USER_TYPE as USER_TYPE15_63_, userpo0_.USER_NAME as USER_NAME16_63_ from UM_USER userpo0_ where upper(userpo0_.USER_NAME)=upper(:1)	3.7s	
--	------	--

表 1.4 可疑 sql 语句汇总

1.3 系统整体评价

经过六个场景的多轮性能测试，系统整体表现稳定，错误率较低，能够满足目前的业务需求。

服务器的 CPU 等资源在部分场景中并未充分利用，部分请求的响应时间和 sql 语句的执行时间较长，系统仍有较大的优化空间。

2. 测试简介

2.1 测试背景

本次性能测试在恒天内网进行，小微系统部署在恒天 4 台虚拟机上。

2.2 测试目标

执行相关性能测试场景案例，以启发式的方法进行加压，来评估系统响应性、容量、稳定性三个方面的指标。

2.3 测试范围

全系统性能测试，覆盖主要场景，主要业务。

2.4 性能测试指标要求

2.4.1 响应性需求

条目	技术要求	设计目标
用户登录响应时间	$\leq 3-5$ 秒	≤ 3 秒
用户操作响应时间	$\leq 3-5$ 秒	≤ 3 秒
交易处理时间	$\leq 3-5$ 秒	≤ 3 秒
用户查询时间	$\leq 3-5$ 秒	≤ 3 秒
注册用户时间	$\leq 3-5$ 秒	≤ 3 秒

表 2.1 响应性需求

2.4.2 稳定型需求

条目	技术要求	设计目标
系统可使用率	\geq 总运行时间的 99.9%	\geq 总运行时间的 99.99%

表 2.4 稳定性需求

2.4.3 系统资源使用

条目	技术要求	设计目标
交易高峰期间 CPU 最高使用率	$\leq 80\%$	$\leq 80\%$
交易高峰期内存最高使用率	$\leq 80\%$	$\leq 80\%$

表 2.5 系统资源使用

3. 测试环境

本章节主要描述小微金服系统性能评估项目所涉及到的测试环境。测试环境包括物理架构、应用系统逻辑架构和物理部署、相关操作系统、中间件、硬件型号、数据库版本等简要描述。性能评估结果的有效性主要与两方面相关：1) 测试环境与生产环境的相似度，2) 测试负载与真实负载的相似度。因此，对测试数据的分析和测试结果的推理要充分考虑到测试环境的因素，过分和不科学的推理都将影响结论的可信度。

3.1 网络拓扑图

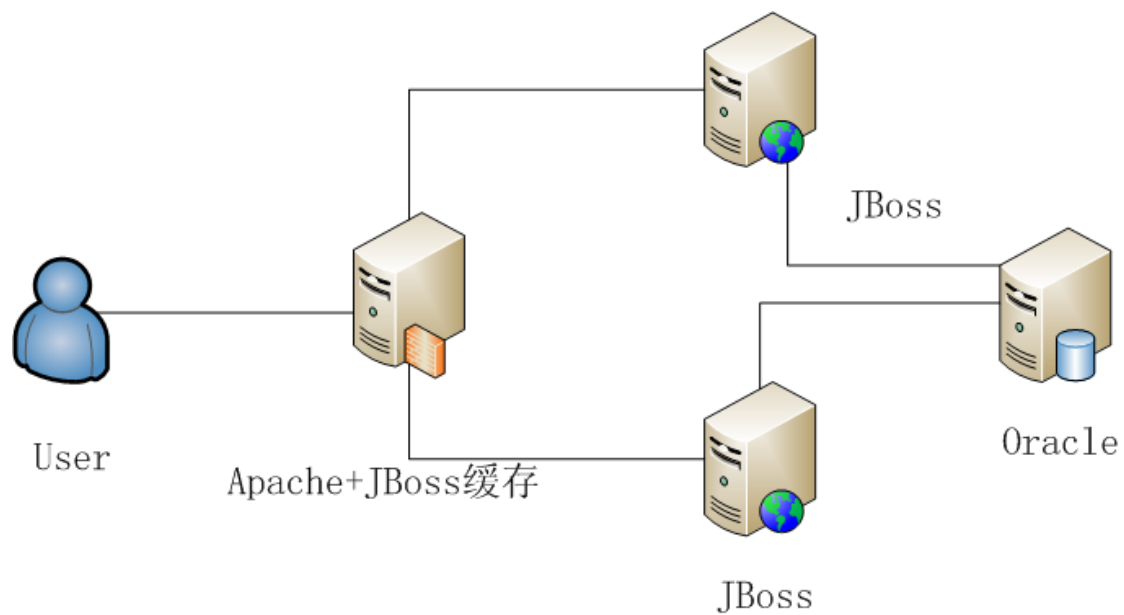


图 3.1 网络拓扑图

3.2 软硬件配置

集群服务器（172.17.3.42）	
CPU	2 QEMU Virtual CPU
内存	4G
硬盘	43G
操作系统	linux Ubuntu
服务器	Apache-2.2.22,JBoss-eap-6.2
web 服务器（172.17.3.43）	
CPU	4 QEMU Virtual CPU
内存	8G
硬盘	47G
操作系统	linux Ubuntu
服务器	JBoss-eap-6.2
web 服务器（172.17.3.44）	
CPU	4 QEMU Virtual CPU
内存	8G
硬盘	47G
操作系统	linux Ubuntu

服务器	JBoss-eap-6.2
数据库服务器（172.17.3.45）	
CPU	2 QEMU Virtual CPU
内存	4G
硬盘	50G
操作系统	linux Ubuntu
服务器	Oracle Database 11g Enterprise Edition

表 3.1 软硬件配置

3.3 软件参数及相关配置

Jre 版本	Jre1.7.0_51
Apache 最大连接数	1500
JBoss 最大连接数	默认配置
Oracle 连接池	1000

表 3.2 软件参数及相关配置

4. 测试方案

4.1 主要场景

根据不同用户的使用方式和工作时间特点，以及合约中对系统 SLA 的要求，我们设计了以下场景。这些场景涵盖了主要流程，可能存在性能问题的业务场景，以及专门针对系统稳定性的长时间测试。这些场景的测试通过可以保证系统在速度，可扩展性和稳定性等方面都可以达到预期效果。

4.1.1 首页加载

首页是所有用户登入系统的必经页面，也是使用频率最高的页面。首页加载速度直接影响了客户的使用感受。基于以上理由，我们模拟了以下场景。

首页加载的操作流程图如下：



图 4.1 首页加载操作流程图

4.1.2 新建融资项目

本场景在每一次融资客户需要融资时由担保公司代为发布融资包。目前系统已有担保公

司只有 100 多家，故此场景的并发操作压力不会很大。

本场景操作流程图如下：



图 4.2 新建融资项目操作流程

4.1.3 投资人申购

在系统正常工作期间，申购是整个系统的主要操作流程，在高峰时间段，此操作所占比例高达 95%，故将此场景作为关键场景。

场景流程图如下：



图 4.3 投资人申购操作流程

4.1.4 自动申购

本场景发生在每天闭市以后，由交易经理本人亲自操作，为单线程。且每天可供自动申购的融资包不足 10 个。用户操作方面，本场景并发压力较小，但是在数据计算方面压力较大。故设计此场景，并计划单线程执行此场景，同时监控系统各项性能指标以及资源消耗。

场景流程图如下：



图 4.4 自动申购操作流程

4.1.5 综合场景

结合系统特点，将综合场景分为高峰期和非高峰期两部分，分别确定测试流程和执行测试计划。

4.1.5.1 高峰期

在系统交易高分期，同一时间，所有操作中“申购”占据 95%以上，故将此场景视为与 4.1.3 场景相同，具体流程图参见 4.1.3 节。

4.1.5.2 非高峰期

在系统非高峰期阶段，用户主要执行各种查询操作，为了接近真实效果，在场景流程中

融入了实际各种操作的配比数据。

场景流程如下：

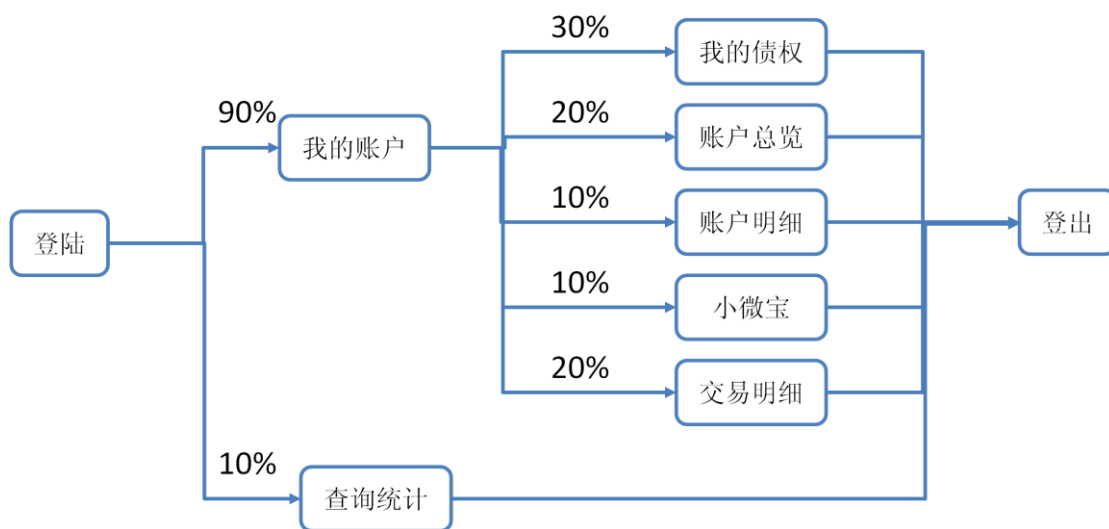


图 4.5 综合场景-非高峰期操作流程图

4.1.6 登陆

任何用户使用系统都要进行登陆操作，登陆的响应时间将会影响每位用户的体验。登陆场景包括登陆相关的请求，但不包括登陆后首页资源的加载。

4.1.7 耐久度测试

通过长时间的运行系统，来检查系统的稳定性。一般这个流程会持续3天到一周，以检查系统有无内存泄露，是否会发生崩溃等潜在性能瓶颈。

针对这个目的，我们专门设计了每次为期2天的系统综合场景运行测试。在测试期间，系统日夜不停的以500并发用户且无思考时间的方式跑综合场景，检查系统的稳定性。

4.2 测试数据准备

场景	是否需要准备数据	准备数据内容
首页加载	是	一个投资用户
新建融资项目	是	100 个担保机构账户
申购	是	5w 投资人账户，每个账户 500W。5w



		个非自动申购融资包。
自动申购	是	一个交易经理账户，
综合场景	是	符合不同申购条件的投资用户共 2W。 可以执行自动申购的包 3000 个。
耐久度测试	是	5w 投资人账户，每个账户 500W。5w 个非自动申购融资包。

表 4.1 数据准备

4.3 测试资源监控指标

指标名称	说明
CPU使用率	应用、数据库的cpu资源使用率
MEM使用率	应用、数据库的内存资源使用率
I/O使用率	应用、数据库的I/O资源使用率

表 4.2 资源监控指标

4.4 测试类型与场景分配

并发测试是为了检测出各系统在多并发量情况下，服务器的硬件资源利用情况、网络使用情况、应用服务运行情况。经过并发测试，可以发现系统的服务是否健壮（是否会出现原本逻辑正确的交易在并发情况下出现逻辑错误），硬件资源是否满足要求，系统是否存在占用资源过多（占用服务器 CPU、内存、存储器资源及网络带宽等），系统平台级的参数配置是否合理，各系统配置的数据库是否需要调整，架构是否符合实际应用等。

4.4.1 负载测试（Stress Test）

性能测试分为很多种类，根据目的不同我们要为不同的测试场景配合不同的测试方法。负载测试，通过测试系统在资源超负荷情况下的表现，以发现设计上的错误或验证系统的负载能力。在这种测试中，将使测试对象承担不同的工作量，以评测和评估测试对象在不同工作量条件下的性能行为，以及持续正常运行的能力。负载测试的目标是确定并确保系统在超出最大预期工作量的情况下仍能正常运行。此外，负载测试还要评估性能特征。例如，响应时间、事务处理速率和其他与时间相关的方面。

4.4.1.1 测试方法

模拟方式：

Jmeter 模拟客户端进行系统操作。

加压策略：

页面思考时间为 3s，每隔 1s 增加 10 个并发用户，压力变化策略为平均负载，随机发送。加压方式遵循 1, 200, 400, 600, 800, 1000... 个并发用户，以启发式的方式进行压力测试。并针对系统不同的表现情况适当修改加压条件和加压策略，测出系统的上限。

4.4.1.2 测试场景

本次测试中，我们在以下场景中使用了负载测试的方法：

1. 新建融资项目
2. 申购
3. 综合场景

4.4.2 尖峰冲击式测试（Spike Test）

尖峰冲击测试是模拟系统在短时间内突然加压，为了验证网站在用户突然极具增加的情况下能否正常工作。

在实际情况下，系统所产生的负载并不一定会遵循最大负载的限制，很可能在短时间内就会超过，这时系统并不一定会出现问题，尖峰冲击测试就是为了验证此时网站的应付能力。

4.4.2.1 测试方法

模拟方式：

Jmeter模拟客户端进行系统操作。

压测策略：

页面思考时间为 0s，所有并发用户在 1s 内启动。加压方式遵循 1, 200, 400, 600, 800, 1000... 个并发用户，以启发式的方式进行压力测试。并针对系统不同的表现情况适当修改加压条件和加压策略，测出系统的上限。

4.4.2.2 测试场景

本次测试中，我们在以下场景中使用了尖峰冲击式测试的方法：

1. 首页加载

4.4.3 耐久度测试（Endurance Test）

疲劳强度测试就是长时间不间断地运行软件，测试软件是否出现性能瓶颈。我们使用系统正常业务情况下并发用户数作为基础，进行一点时间的疲劳测试。

4.4.3.1 测试方法

模拟方式：

Jmeter模拟客户端进行系统操作。

压测策略：

页面思考时间为 0s，每隔 1s 启动 10 个并发用户，共 400 个并发用户，持续执行主流程，运行整整两天即 72 小时。

4.4.3.2 测试场景

本次测试中，我们在以下场景中使用了耐久度测试的方法：

1. 系统耐久度测试

4.5 结果数据搜集方法

1. 典型业务应用的平均响应时间，吞吐量，错误率由Jmeter监听器搜集。
2. 服务器的CPU使用率，内存使用情况，线程工作细节由Top, Vmstat, Jstat, Jmap, Eclipse Mat, Jconsole等工具配合综合搜集。
3. 数据库工作情况数据由Spotlight on oracle搜集。
4. 网络数据传输情况由Tast Manager, Netstat, Iostat, Firebug等工具综合搜集。

5. 测试实施情况

5.1 测试时间和地点

时间： 2014-06-09到2014-06-20

地点： 三墩

5.2 参加人员

参加本次性能测试的人员包括：

性能测试主管：周军赞

性能测试人员： 宗婷婷，沈利达，李光举

5.3 测试约定

性能测试计划中可执行部分已全部完成，被测系统符合性能需求或经过一次优化复测后得到客户方认可，已分析系统性能可能存在的瓶颈；

6. 测试结果

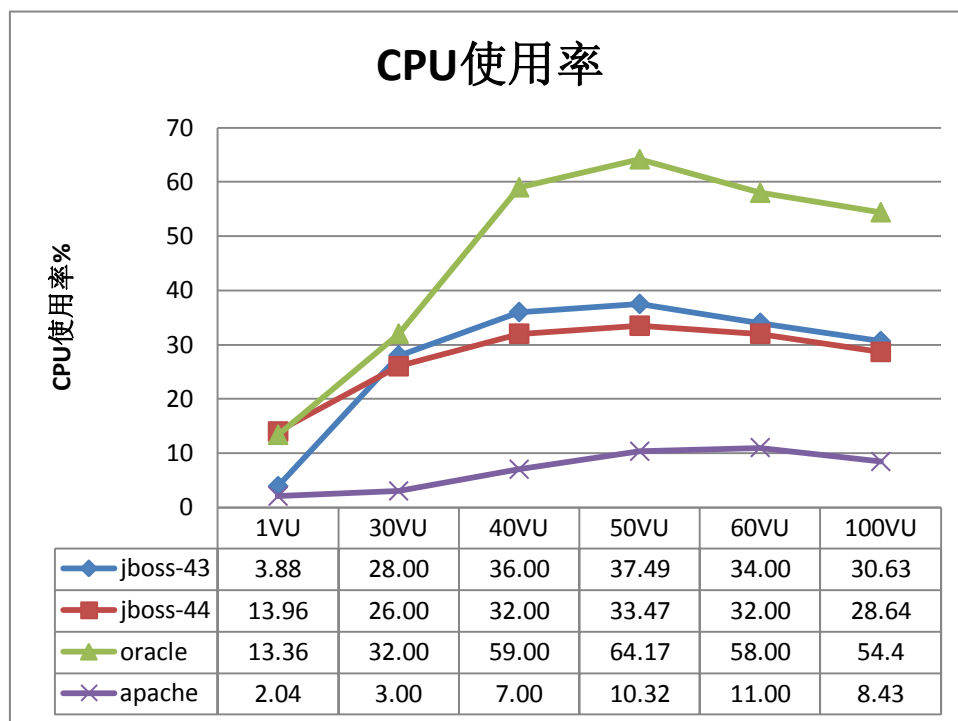
6.1 负载测试

6.1.1 场景一：新建融资项目

6.1.1.1 场景执行情况

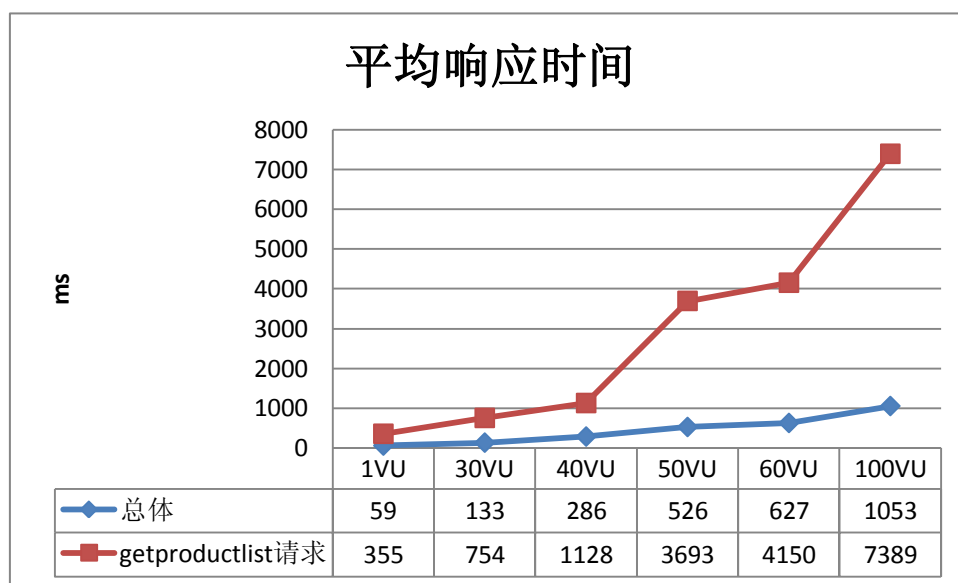
本场景共执行了 6 轮测试，思考时间设定为 0 秒，测得了系统最大吞吐率并确定了最适宜并发访问用户数。

6.1.1.2 测试结果



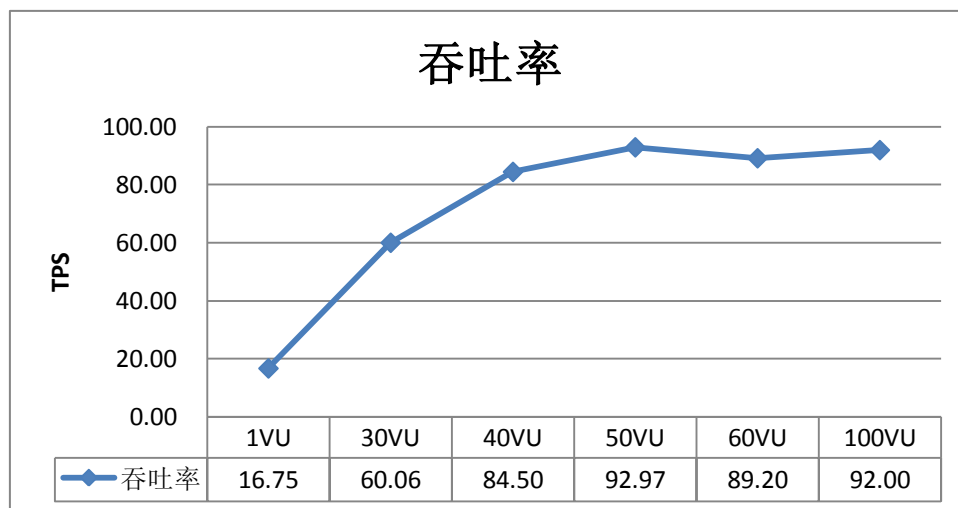
图表6.1 CPU使用率

由图表 6.1 可知,在 6 轮测试中,apache 和两台 jboss 服务器的 CPU 使用率均低于 40%,oracle 服务器的 CPU 使用率相对较高,但仍低于 70%,并未达到极限。



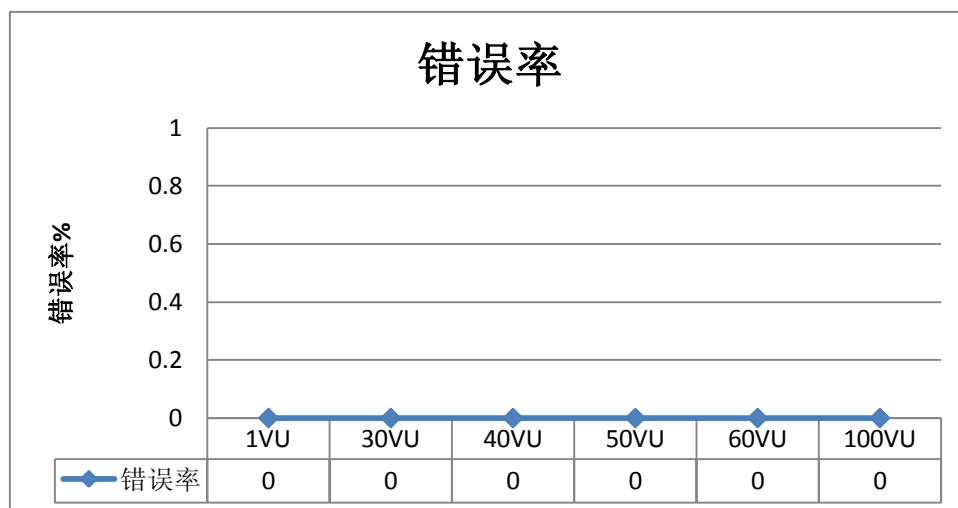
图表6.2 平均响应时间

由图表 6.2 可知，在 6 轮测试中，此场景总体的平均响应时间较快，在并发用户达到 100 人时，总体的平均响应时间为 1053ms，但其中的 getproductlist 请求的响应时间相对较长，在并发用户达到 100 人时的平均响应时间超过了 5s，用户体验较差，仍有优化空间。



图表6.3 吞吐率

由图表 6.3 可知，在 6 轮测试中，当并发用户达到 50 人时，系统吞吐量达到最大值 92.97TPS。



图表6.4 错误率

由图表 6.4 可知，在 6 轮测试中，系统的错误率始终为 0。因此，当并发用户小于 100 人时，系统在此场景下运行稳定。

6.1.1.3 结果分析

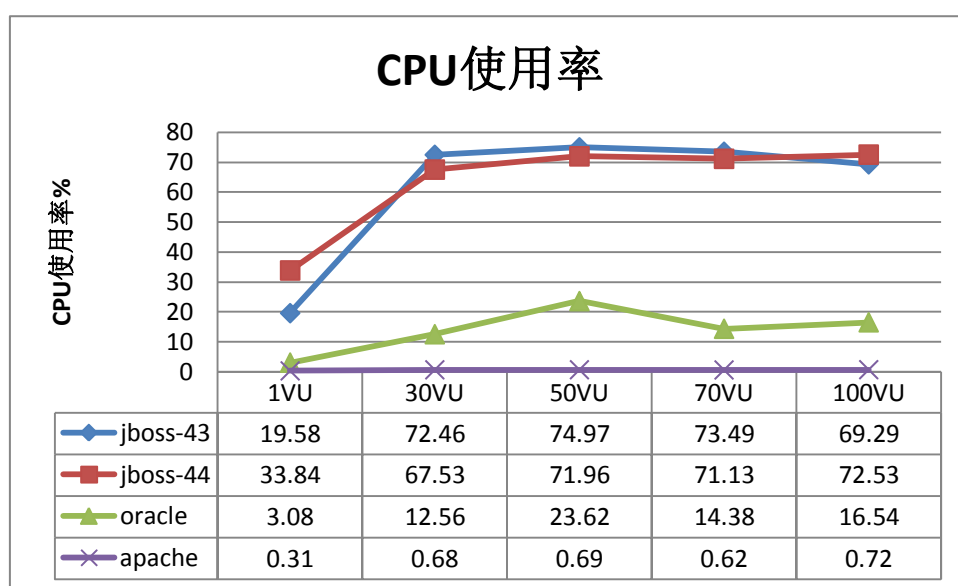
综合以上四图表，系统的最大吞吐率可以达到 92.97TPS，而结合 CPU 使用率，平均响应时间，错误率等影响客户体验的因素，此场景下系统的最佳并发用户数在 50 人左右。

6.1.2 场景二：投资人申购

6.1.2.1 场景执行情况

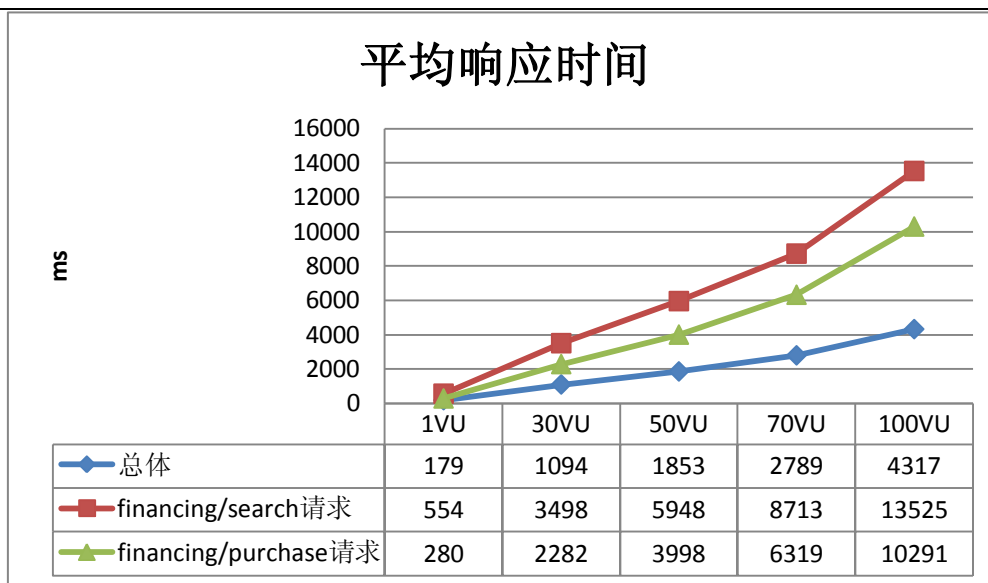
本场景共执行了 5 轮测试，思考时间设定为 0 秒，测得了系统最大吞吐率并确定了最适宜并发访问用户数。

6.1.2.2 测试结果



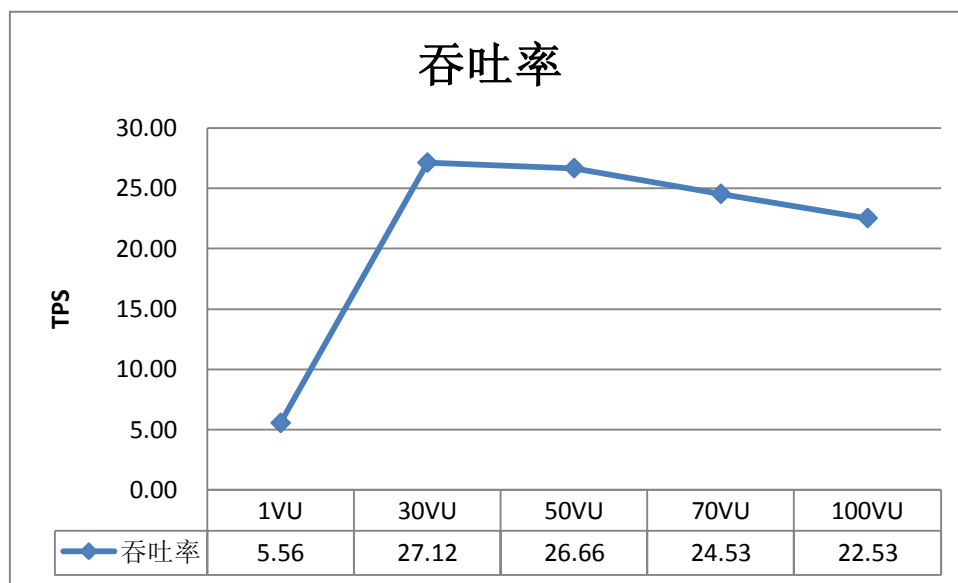
图表6.5 CPU使用率

由图表 6.5 可知，在 5 轮测试中，apache 服务器的 CPU 使用率均低于 1%，oracle 服务器的 CPU 使用率也未超过 30%，两台 jboss 服务器的 CPU 使用率相对较高，但仍低于 80%，并未达到极限。



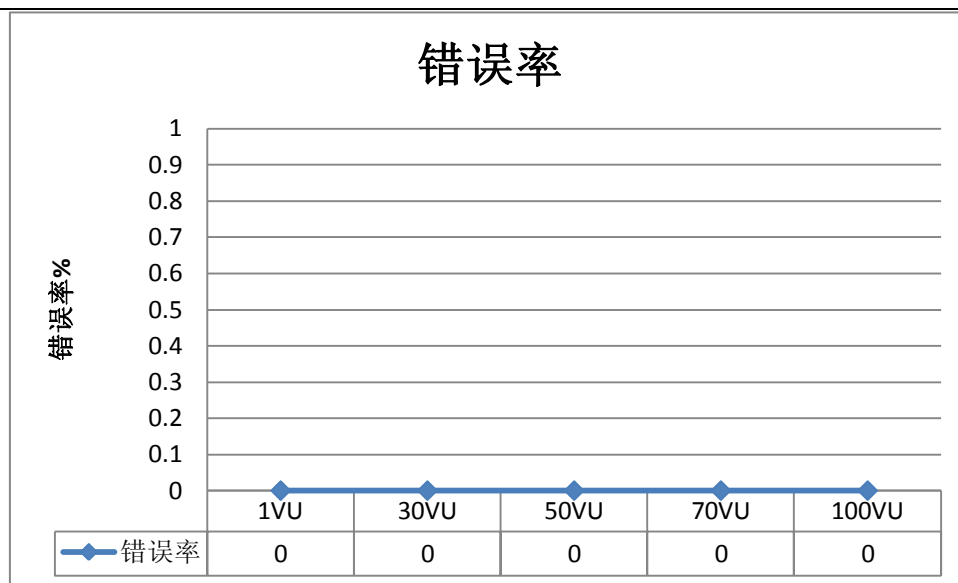
图表6.6 平均响应时间

由图表 6.6 可知，在 5 轮测试中，此场景总体的平均响应时间较快，在并发用户达到 100 人时，总体的平均响应时间为 4317ms，但其中的 financing/search 请求和 financing/purchase 请求的响应时间相对较长，在并发用户达到 50 人时 financing/search 请求的平均响应时间超过了 5s，用户体验较差，仍有优化空间。



图表6.7 吞吐率

由图表 6.7 可知，在 5 轮测试中，当并发用户达到 30 人时，系统吞吐量达到最大值 27.12TPS。



图表6.8 错误率

由图表 6.8 可知，在 5 轮测试中，系统的错误率始终为 0。因此，当并发用户小于 100 人时，系统在此场景下运行稳定。

6.1.2.3 结果分析

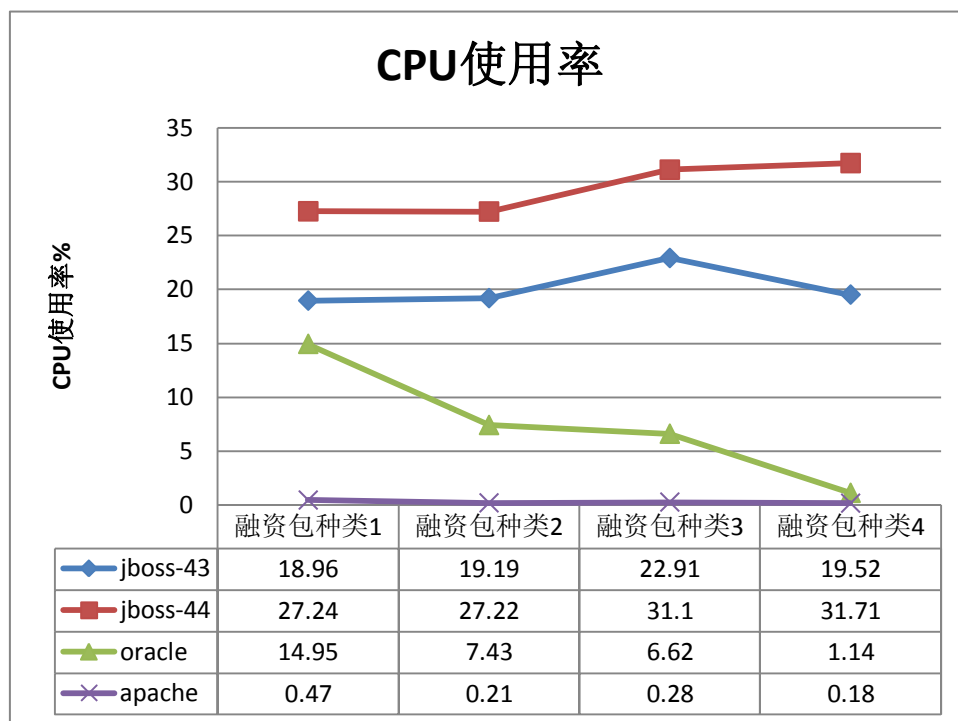
综合以上四图表，系统的最大吞吐率可以达到 27.12TPS，而结合 CPU 使用率，平均响应时间，错误率等影响客户体验的因素，此场景下系统的最佳并发用户数在 30 人左右。

6.1.3 场景三：自动申购

6.1.3.1 场景执行情况

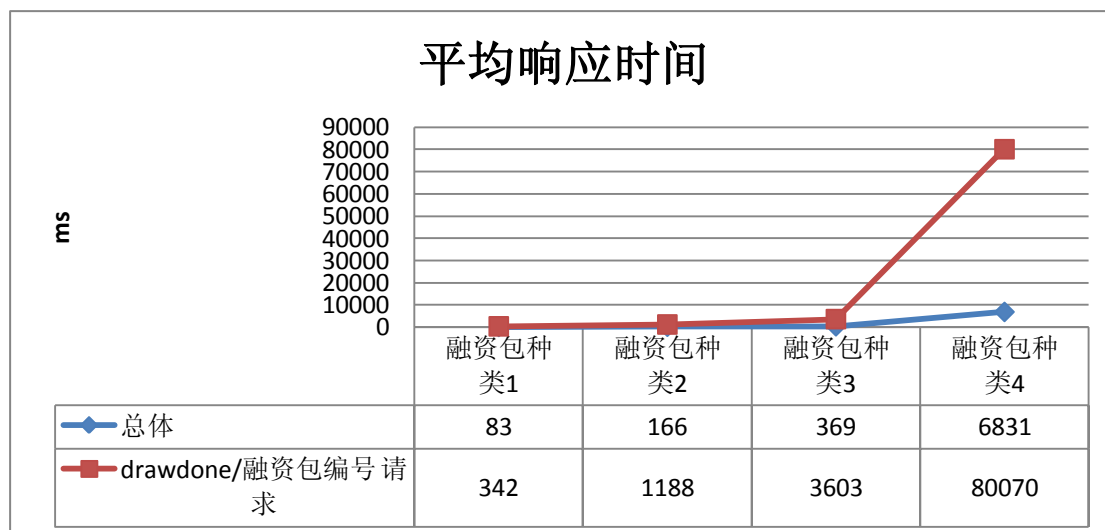
本场景共执行了 4 轮测试，思考时间设定为 0 秒，测得了不同类型融资包在 2 万投资用户下进行自动申购时系统的吞吐率等参数。4 轮测试对应的融资包类型分别为：1. 以日计，3 天，到期一次还本付息，A 本金担保，10 优质；2. 以月计；3 个月，按月等额还息，到期一次还本 A 本金担保，10 优质；3. 以月计，5 个月，按月等本等息，到期一次还本，10 优质；4. 以月计，5 个月，按月等额本息，A 本金担保，10 优质。

6.1.3.2 测试结果



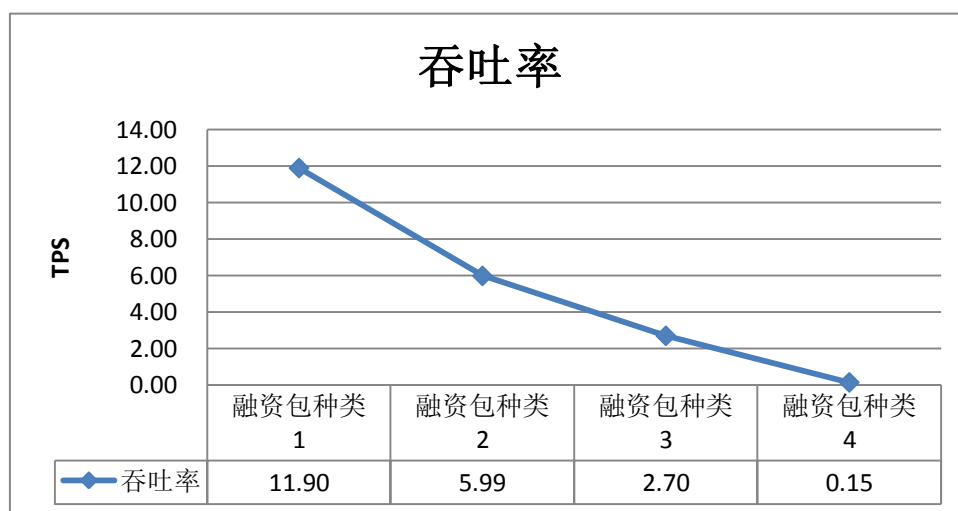
图表6.9 CPU使用率

由图表 6.9 可知，在 4 轮测试中，apache 服务器的 CPU 使用率均低于 1%，oracle 服务器和两台 jboss 服务器的 CPU 使用率相对较高，但仍低于 35%，并未达到极限。



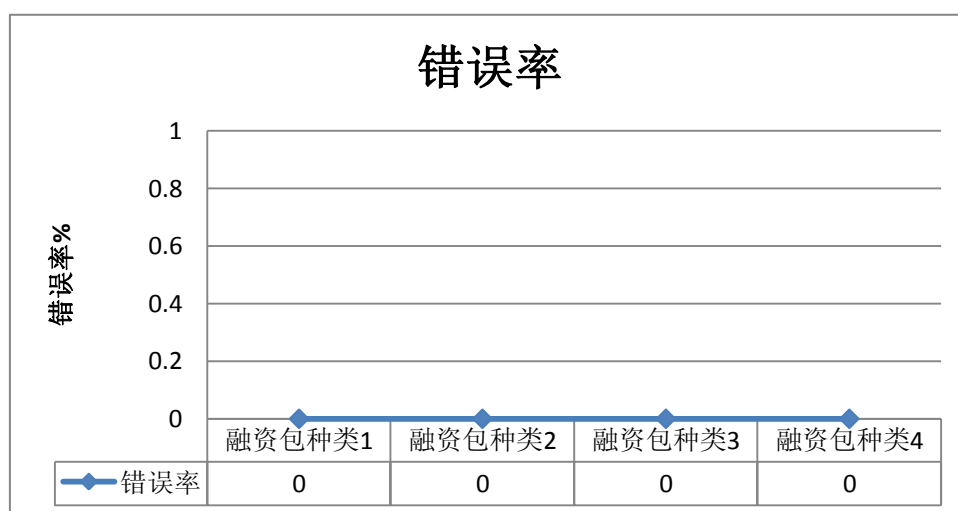
图表6.10 平均响应时间

由图表 6.10 可知，在 4 轮测试中，第 1、2、3 种融资包的平均响应时间较快，总体的响应时间没有超过 500ms，第 4 种融资包的平均响应时间较慢，其中 drawdone/融资包编号 请求的响应时间超过了 80s，仍有优化空间。



图表6.11 吞吐率

由图表 6.11 可知，在 4 轮测试中，融资包种类 1 的吞吐率最大，可达到 11.90TPS。



图表6.12 错误率

由图表 6.12 可知，在 4 轮测试中，系统的错误率始终为 0。因此，4 种不同类型的融资包在并发用户数为 1 人的情况下，系统运行稳定。

6.1.3.3 结果分析

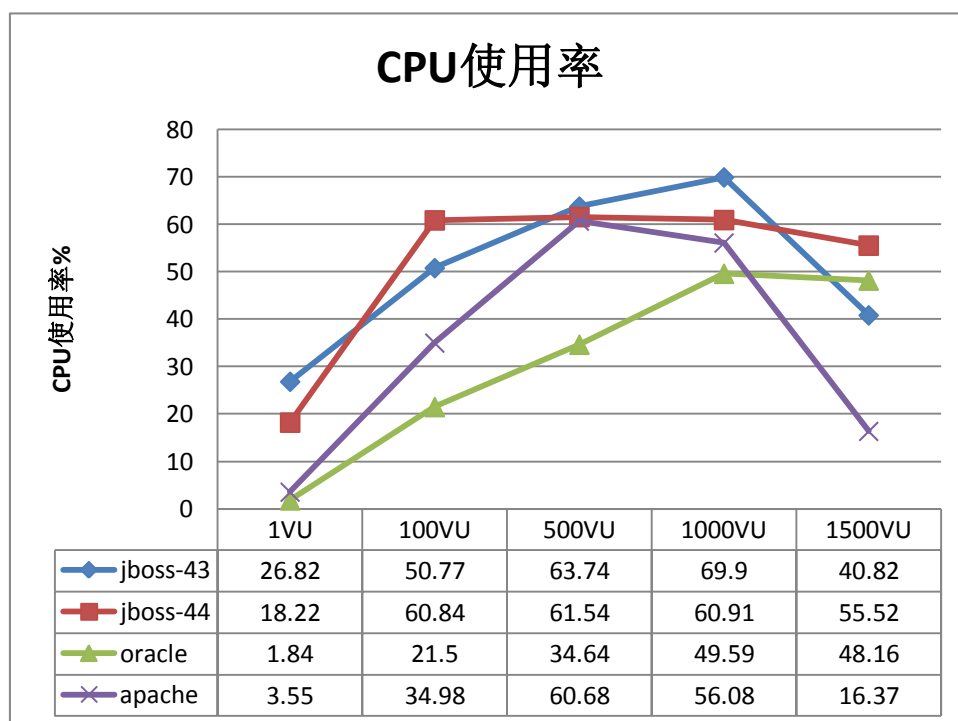
综合以上四图表，系统的最大吞吐率可以达到 11.90TPS，而结合 CPU 使用率，平均响应时间，错误率等影响客户体验的因素，此场景下 4 种类型的融资包最佳的为第 1 种，最差的为第 4 种。

6.1.4 场景四：综合场景

6.1.4.1 场景执行情况

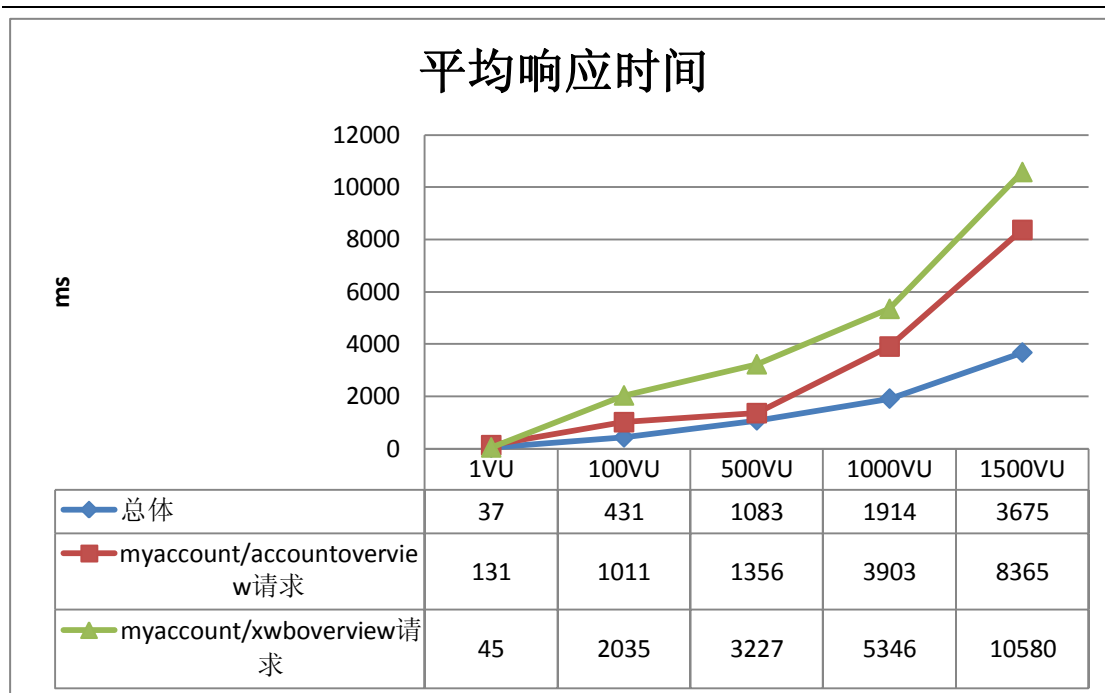
本场景共执行了 5 轮测试，思考时间设定为 0 秒，测得了系统最大吞吐率并确定了最适宜并发访问用户数。

6.1.4.2 测试结果



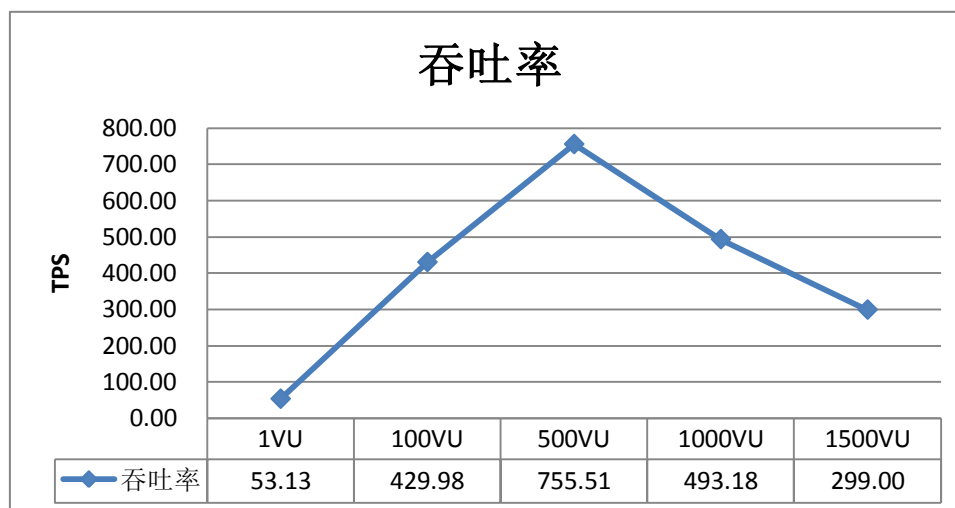
图表6. 13 CPU使用率

由图表 6. 13 可知，在 5 轮测试中，4 台服务器的 CPU 使用率均低于 70%，并未达到极限。



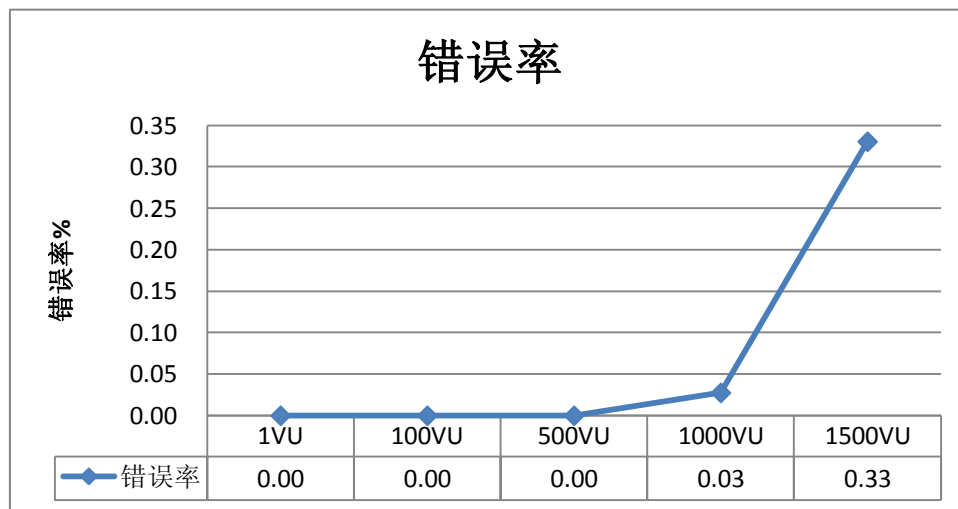
图表6.14 平均响应时间

由图表 6.14 可知，在 5 轮测试中，此场景总体的平均响应时间较快，在并发用户达到 1500 人时，总体的平均响应时间为 3675ms，但其中的 myaccount/accountoverview 请求和 myaccount/xwbooverview 请求的响应时间相对较长，在并发用户达到 1500 人时两个请求的平均响应时间均超过了 5s，用户体验较差。



图表6.15 吞吐率

由图表 6.15 可知，在 5 轮测试中，当并发用户达到 500 人时，系统吞吐量达到最大值 755.51TPS。



图表6.16 错误率

由图表 6.16 可知,在 5 轮测试中,系统的错误率在并发用户数为 1500 人时达到 0.33%,但仍在可接受范围内。因此,当并发用户小于 1500 人时,系统在此场景下运行稳定。

6.1.4.3 结果分析

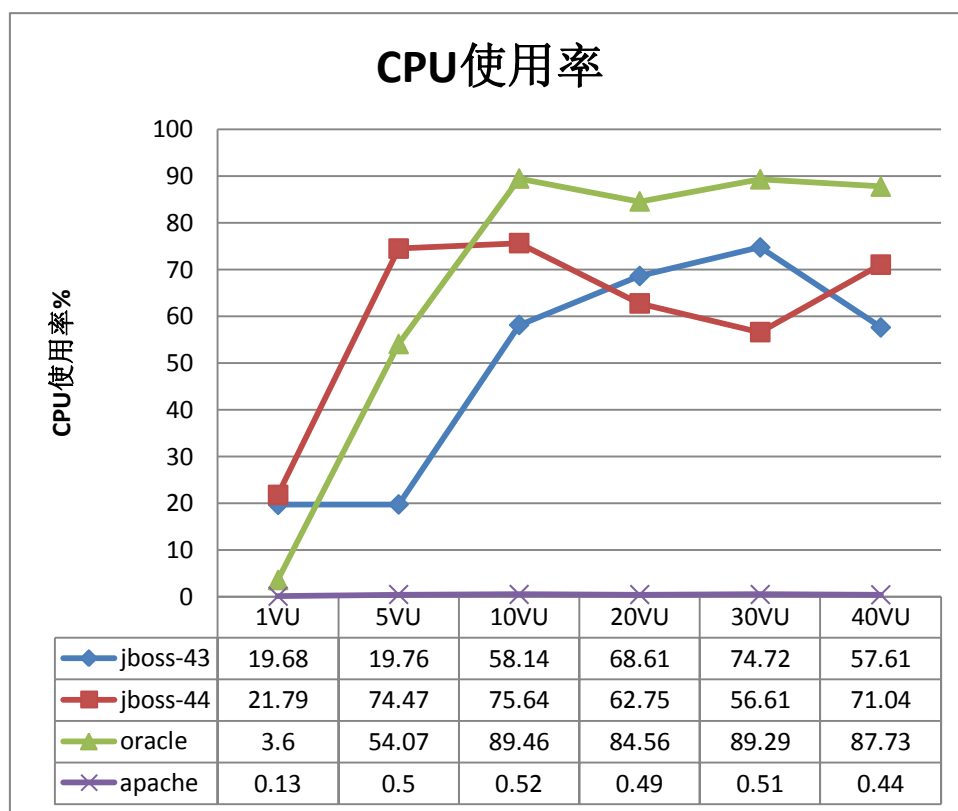
综合以上四图表,系统的最大吞吐率可以达到 755.51TPS,而结合 CPU 使用率,平均响应时间,错误率等影响客户体验的因素,此场景下系统的最佳并发用户数在 1000 人左右。

6.1.5 场景五：登陆

6.1.5.1 场景执行情况

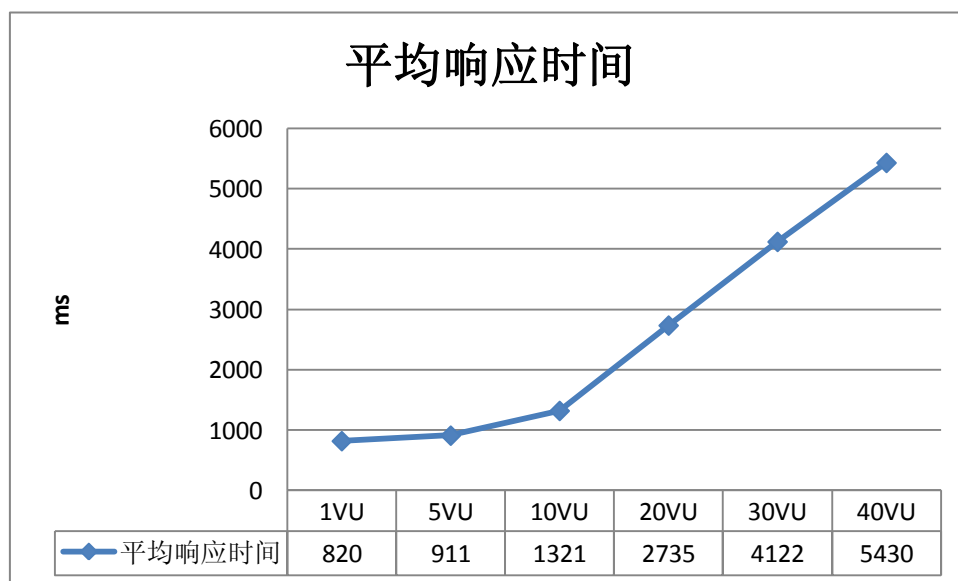
本场景共执行了 6 轮测试,思考时间设定为 0 秒,测得了系统最大吞吐率并确定了最适宜并发访问用户数。

6.1.5.2 测试结果



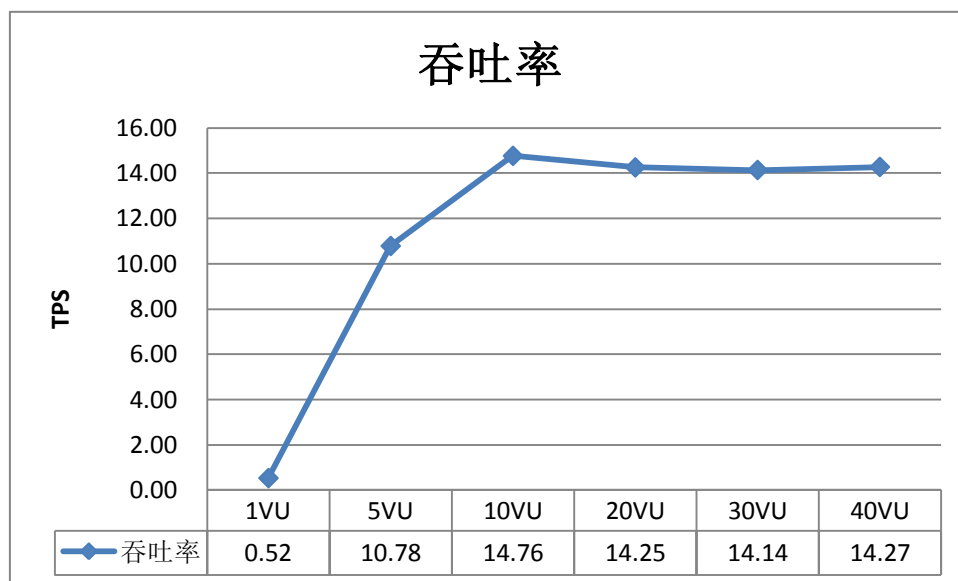
图表6.17 CPU使用率

由图表 6.17 可知，在 6 轮测试中，apache 服务器的 CPU 使用率均低于 1%，两台 jboss 服务器的 CPU 使用率低于 80%，oracle 服务器的 CPU 使用率相对较高，最高时接近 90%，并未达到极限。



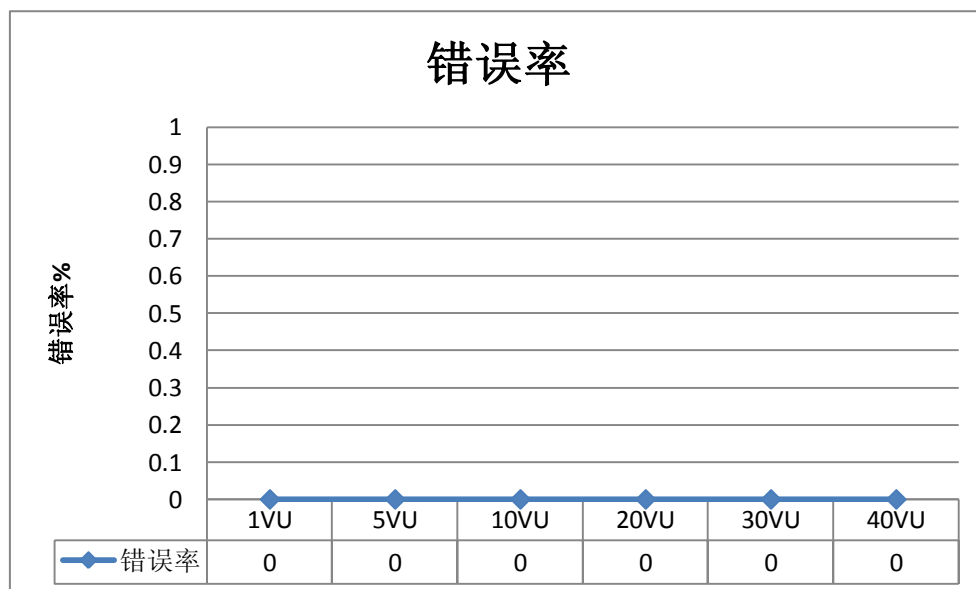
图表6.18 平均响应时间

由图表 6.18 可知，在 6 轮测试中，当并发用户达到 40 人时，系统的平均响应时间超过 5s，用户体验较差。



图表6.19 吞吐率

由图表 6.19 可知，在 6 轮测试中，当并发用户达到 10 人时，系统吞吐量达到最大值 14.76TPS。



图表6.20 错误率

由图表 6.20 可知，在 6 轮测试中，系统的错误率始终为 0。因此，当并发用户小于 40 人时，系统在此场景下运行稳定。

6.1.5.3 结果分析

综合以上四图表，系统的最大吞吐率可以达到 14.76TPS，而结合 CPU 使用率，平均响应时间，错误率等影响客户体验的因素，此场景下系统的最佳并发用户数在 30 人到 40 人之间。

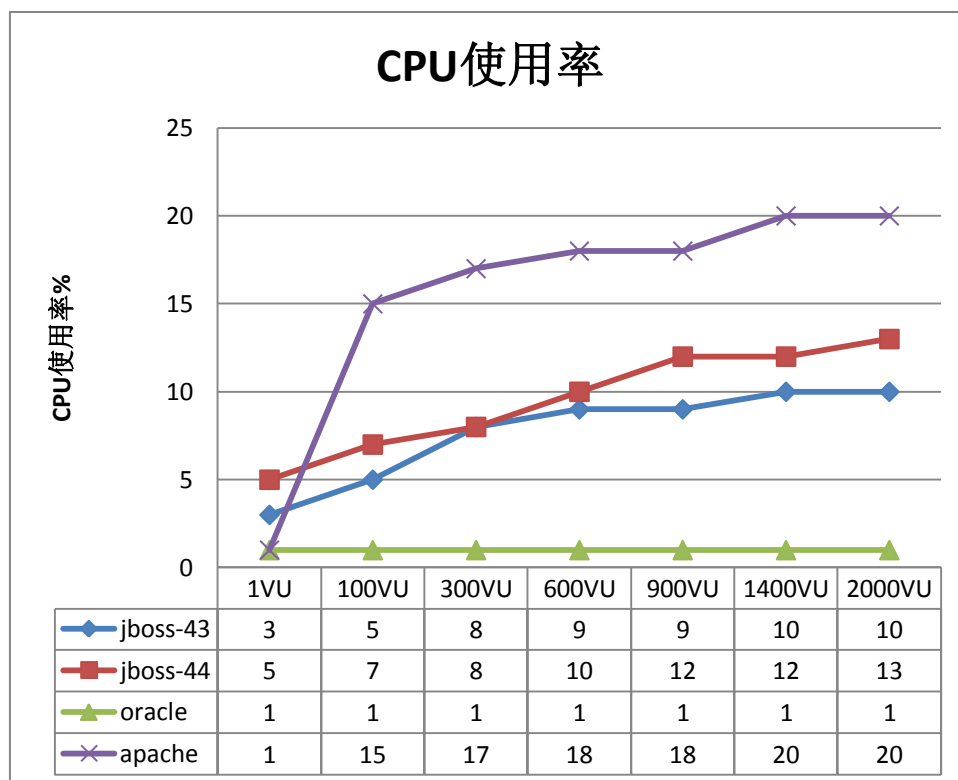
6.2 尖峰冲击式测试

6.2.1 场景一：首页加载

6.2.1.1 场景执行情况

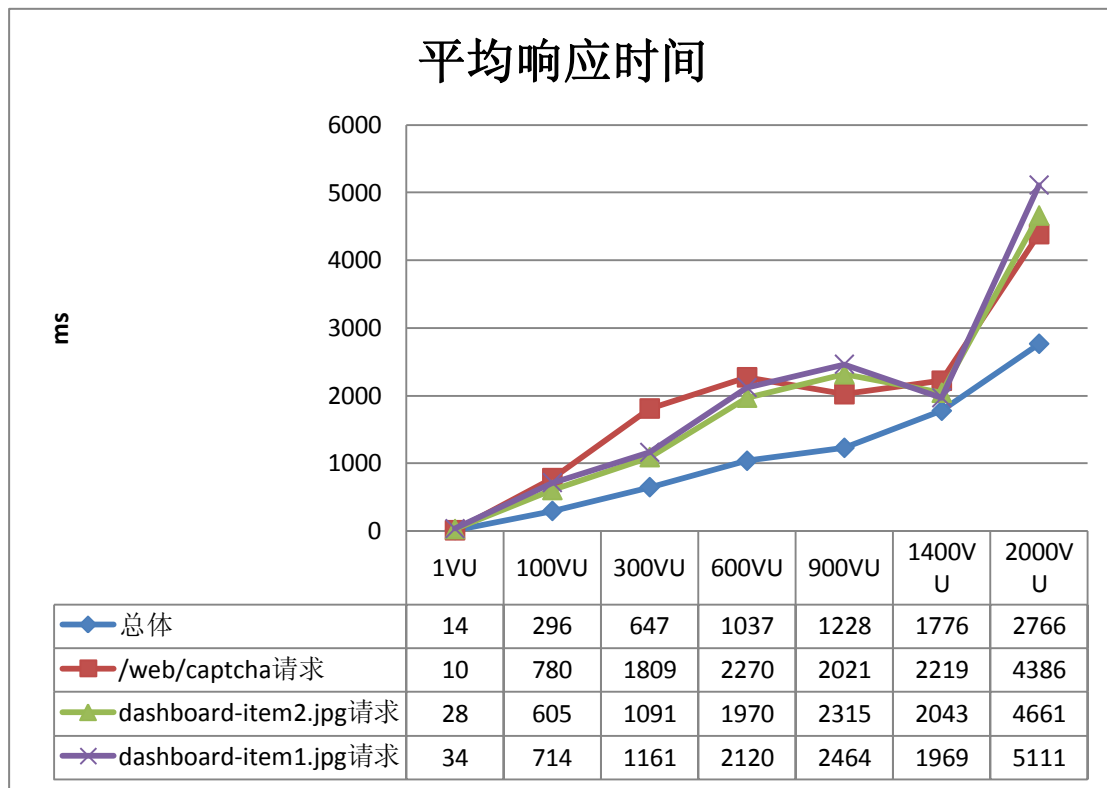
本场景共执行了 7 轮测试，思考时间设定为 0 秒，测得了系统最大吞吐率并确定了最适宜并发访问用户数。

6.2.1.2 测试结果



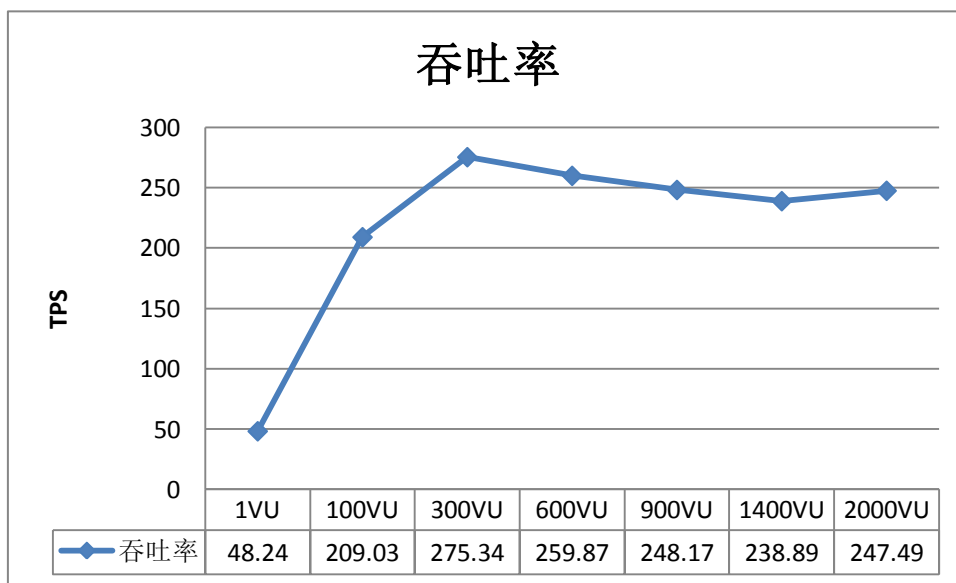
图表6.21 CPU使用率

由图表 6.21 可知，在 7 轮测试中，oracle 服务器的 CPU 使用率均低于 1%，两台 jboss 服务器的 CPU 使用率均低于 15%，apache 服务器的 CPU 使用率均低于 20%，并未达到极限。



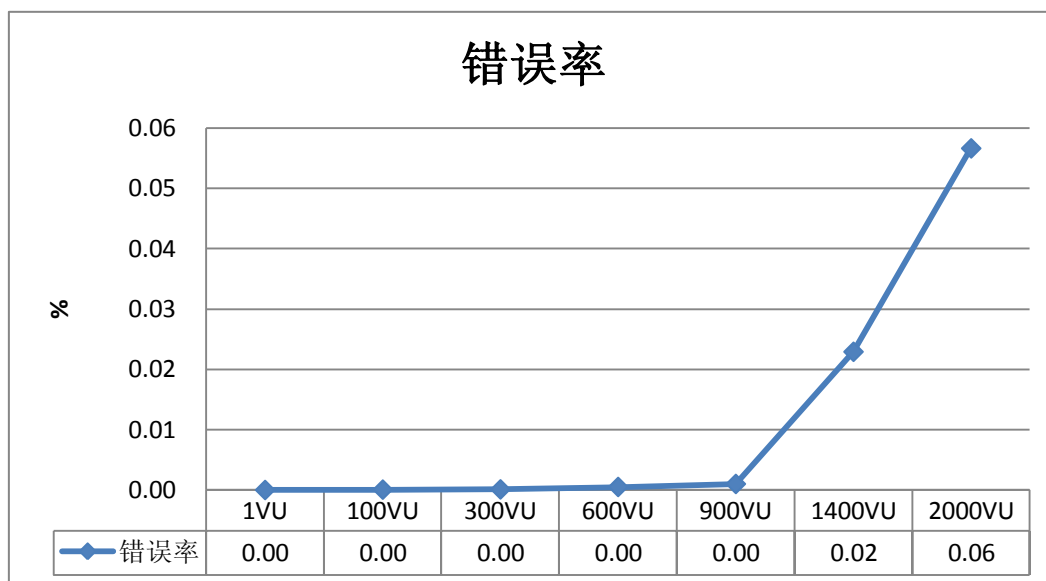
图表6.22 平均响应时间

由图表 6.22 可知，在 7 轮测试中，当并发用户达到 1500 人时，dashboard-item1.jpg 的平均响应时间超过 5s，用户体验较差，同时 web/captcha 和 dashboard-item2.jpg 的平均响应时间也比较高。



图表6.23 吞吐率

由图表 6.23 可知, 在 7 轮测试中, 当并发用户达到 300 人时, 系统吞吐量达到最大值 275.34TPS。



图表6.24 错误率

由图表 6.24 可知, 在 7 轮测试中, 系统的错误率在并发用户数为 2000 人时达到 0.06%, 但仍在可接受范围内。因此, 当并发用户小于 2000 人时, 系统在此场景下运行稳定。

6.2.1.3 结果分析

综合以上四图表, 系统的最大吞吐率可以达到 275.34TPS, 而结合 CPU 使用率, 平均响应时间, 错误率等影响客户体验的因素, 此场景下系统的最佳并发用户数在 2000 人左右。

6.3 耐久度测试

6.3.1 场景执行情况

本场景执行了 2 天, 在测试期间, 系统日夜不停的以 500 并发用户且无思考时间的方式跑综合场景, 检查系统的稳定性, 各项性能表现良好。

6.3.2 测试结果

经过 2 天的测试, 系统一直保持较稳定的状态, CPU 使用率基本维持在 60%以下, 没有出现大的波动

6.3.3 结果分析

由以上结果可以看出，系统可以在高压下持续稳定较好的完成工作，不会出现崩溃等极限情形，可以长时间支持正常的商务运作。

6.4 其他数据分析和折算

6.4.1 内存泄露分析

通过长时期对系统运行情况的监控以及开发及时修改，系统内存并没有随着长时间的使用而逐渐耗尽，在执行垃圾回收后都能完美返回最初状态，系统不存在内存泄露问题。

6.4.2 数据库锁争用分析

通过多种监控工具的综合测试监控，系统在达到极限之前都没有发生数据库锁争用现象。

6.4.3 死锁死循环分析

通过多种监控工具的综合测试监控，系统不会出现 CPU 被吃光且居高不下的情况。

7 系统评价

经过六个场景的多轮性能测试，系统整体表现稳定，错误率较低，能够满足目前的业务需求。

服务器的 CPU 等资源在部分场景中并未充分利用，部分请求的响应时间和 sql 语句的执行时间较长，系统仍有较大的优化空间。