徐雯蕾 杨政达 侯峂欣 孙雨晶

**华东师范大学 计算机科学与软件工程学院**

摘要

报告将从项目简介，性能测试策略，测试案例设计，性能测试标准等多个方面详细介绍面向Timeline项目的性能测试计划。

性能测试计划

面向Timeline系统

目录

[1 项目概要介绍 3](#_Toc534662018)

[1.1 项目简介 3](#_Toc534662019)

[1.2 项目成员 3](#_Toc534662020)

[1.3 测试范围 3](#_Toc534662021)

[1.3.1 范围内 3](#_Toc534662022)

[1.3.2 范围外 3](#_Toc534662023)

[1.4 前提假设 3](#_Toc534662024)

[1.5 测试目标 3](#_Toc534662025)

[2 性能测试策略 5](#_Toc534662026)

[2.1 性能测试模型 5](#_Toc534662027)

[2.2 性能测试场景 6](#_Toc534662028)

[2.3 重点测试策略 6](#_Toc534662029)

[2.3.1 重点测试原则 6](#_Toc534662030)

[2.3.2 重点测试交易 7](#_Toc534662031)

[3 测试案例设计 8](#_Toc534662032)

[3.1 生产压力分析 8](#_Toc534662033)

[3.2 场景通过标准 8](#_Toc534662034)

[3.3 测试场景设计 8](#_Toc534662035)

[3.3.1 独立场景 8](#_Toc534662036)

[3.3.2 混合场景 9](#_Toc534662037)

[4 测试实施安排 10](#_Toc534662038)

[4.1 测试进度 10](#_Toc534662039)

[4.2 测试流程 10](#_Toc534662040)

[5 性能测试标准 11](#_Toc534662041)

[5.1 启动标准 11](#_Toc534662042)

[5.2 中止标准 11](#_Toc534662043)

[5.3 通过标准 11](#_Toc534662044)

[6 测试环境规划 12](#_Toc534662045)

[6.1 部署环境 12](#_Toc534662046)

[6.2 执行环境 12](#_Toc534662047)

[7 测试风险分析 12](#_Toc534662048)

表目录

[Table 1性能测试类型 5](#_Toc501358463)

[Table 2 场景通过标准 7](#_Toc501358464)

[Table 3 独立场景设计 7](#_Toc501358465)

[Table 4 混合场景设计 8](#_Toc501358466)

[Table 5 峰值场景设计1 8](#_Toc501358467)

[Table 6 峰值测试设计2 8](#_Toc501358468)

[Table 7 容量测试场景设计 9](#_Toc501358469)

[Table 8 疲劳测试场景设计 10](#_Toc501358470)

[Table 9 测试报告需求 10](#_Toc501358471)

# 项目概要介绍

## 项目简介

本项目实现的Timeline是社交软件中按时间显示动态功能的简化版，分为桌面应用和Web应用两种形式。用户可以在Timeline上发布新内容，且发布的内容支持文字和图片。此外，用户还可以获取最新动态以及查询更早信息。

## 项目成员

徐雯蕾 10165101112 杨政达 10165101224

侯峂欣 10165101105 孙雨晶 10165101144

## 测试范围

### 范围内

由于本次项目中桌面版和Web版的Timeline共用后端代码且桌面版前端是由web前端通过工具electron转换得到的，因此本次性能测试主要包括：

1. 利用Chrome开发者工具进行Web前端性能测试；
2. 利用JMeter对不同的功能接口进行独立场景测试，包括“注册”，“登录”，“浏览动态”，“发布动态”；
3. 利用JMeter对整个系统进行混合场景测试；
4. 根据JMeter的性能测试分析报告对系统性能进行总结分析。

### 范围外

## 前提假设

考虑该系统在校园内上线推广后，未来1-3年内的用户量可达到5000左右，一般情况下，系统在线用户数为系统总用户数的20%，因此，未来1-3年内Timeline的在线用户数约为1000。根据经验公式C = n/10，Timeline的最大并发用户数约为100。

## 测试目标

在上述的测试范围和前提假设下，我们希望系统可以达到以下目标：

1) 系统在单步操作响应时间方面，控制在4s以内；

2) 系统的批量作业运行稳定，处理能力能满足并发用户数达到100时的需求；

3) 在业务处理能力方面，可以满足在线用户数达到1000的需求；

4) 发现并解决宕机，内存泄漏等严重问题，使系统具备良好的稳定性、健壮性；

5) 系统在资源使用方面比较合理，各项资源平均利用率在30%左右。

# 性能测试策略

与其它测试类型一样，性能测试周期分为：性能测试需求分析、性能测试设计、性能测试实现和性能测试报告四个存在迭代的阶段。其中，性能测试需求和性能测试设计由测试人员手工完成，而性能测试实现需借助性能测试工具达成，通过对性能测试报告的分析判定系统是否存在性能缺陷。

## 性能测试模型

性能测试模型定义了性能测试涉及的各个待测方面，是性能测试设计的指南。



Figure 1 性能测试模型

根据Figure 1所示的性能测试模型，可以设计如Table 1所示的性能测试类型

Table 1性能测试类型

|  |  |  |  |
| --- | --- | --- | --- |
| **测试种类** | **测试方法** | **测试内容** | **场景类型** |
| 性能测试 | 以系统最大并发用户数100作为上限对“发布和浏览动态”进行梯度加压测试，在压力时间内通过的交易量应接近峰值时段的交易量，甚至超过系统全天的交易量 | 浏览动态功能  发布动态功能 | 独立场景 |
| 压力测试 | 以系统预期最大并发用户数100作为上限对常用/关键交易进行混合场景梯度加压测试，在压力时间内通过的交易量应接近峰值时段的交易量，甚至超过系统全天的交易量 | 浏览动态功能  发布动态功能 | 混合场景 |
| 峰值测试 | 按照系统峰值时预期最大并发用户数100的2倍对峰值时段涉及的核心交易进行的混合场景梯度加压测试，在测试时间内通过的交易数量应接近或超过系统全天的交易量 | 浏览动态功能  发布动态功能 | 峰值场景 |
| 容量测试 | 通过阶段性不断增加系统并发用户的梯度加压方式，测试出系统对核心交易的最大并发处理能力以及系统的最大业务处理能力，测试时最大并发用户数为系统预期最大并发用户数100的3倍 | 浏览动态功能  发布动态功能 | 容量场景 |
| 疲劳测试 | 以系统预期最大并发用户数100的1倍作为上限对关键交易进行持续加压的稳定性与健壮性测试，在压力时间内通过的交易量应为全天交易量3倍以上 | 浏览动态功能  发布动态功能 | 疲劳场景 |

## 性能测试场景

根据Timeline项目的实际需求，确定本次性能测试场景为：

独立场景

混合场景

峰值场景

容量场景

疲劳场景

## 重点测试策略

### 重点测试原则

为了全面评估系统性能，本次重点测试策略安排如下：

1. 对关键交易开展全面的独立场景测试；
2. 对关键交易开展全面的峰值、容量测试；
3. 多渠道发起以关键/常用交易为主的疲劳测试；
4. 在集成性能测试方面，对交易量大、容易出现性能瓶颈的模块，重点进行测试并安排专人在系统发布前一直跟进开发组的需求变更以便进行及时的回归性能测试。
5. 在渠道性能测试方面，对渠道场景按照业务量及重要性来划分优先级，按照优先级先后来执行，保证系统上线后关键业务的稳定性。

### 重点测试交易

1. 用户注册
2. 用户登录
3. 浏览动态
4. 发布动态

# 测试案例设计

## 生产压力分析

Timeline项目的生产压力主要来源于对核心交易“发布动态”和“浏览动态”的并发访问，当并发用户数达到一定数量时，可能会导致信息加载过慢，系统卡顿等问题。因此我们拟通过独立场景和混合场景的性能测试来探索系统性能能否满足未来1-3年的运行要求，以及如果不满足，系统性能的瓶颈可能会出现在哪些地方。

## 场景通过标准

Table 2 场景通过标准

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **场景类型** | **单步操作响应时间** | **加压时间** | **操作账户数据** | **处理业务笔数** | **事务成功率** | **各后台利用率** | **并发/在线上限** | **在线思考时间设置** |
| 独立场景 | <5秒 | 30分钟 | 100 | 500 | 100% | <50% | 100 | 无 |
| 混合场景 | <5秒 | 1小时 | 100 | 500 | 100% | <80% | 200 | 无 |
| 峰值场景 | <5秒 | 1小时 | 100 | 500 | 100% | <80% | 200 | 无 |
| 容量场景 | <5秒 | 4小时 | 100 | 500 | 100% | <80% | 300 | 无 |
| 疲劳场景 | <5秒 | 12小时 | 100 | 500 | 100% | <50% | 100 | 无 |

## 测试场景设计

### 独立场景

对于“用户注册”场景，“用户登录”场景，“发布动态”场景和“浏览动态”场景，将分别对其采用渐进式加压的方式来进行独立场景测试。每个场景测试多组并发、并发数从1逐步增加到100。

Table 3 独立场景设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **并发数** | **响应时间要求** | **TPS要求** | **加压时间** | **加压方式** |
| 1 | <5秒 | N/A | 3分钟 | 直接加压 |
| 5 | <5秒 | N/A | 5分钟 | 直接加压 |
| 10 | <5秒 | N/A | 10分钟 | 直接加压 |
| 20 | <5秒 | N/A | 10分钟 | 直接加压 |
| 50 | <5秒 | >100笔/秒 | 20分钟 | 直接加压 |
| 100 | <5秒 | >100笔/秒 | 20分钟 | 直接加压 |

### 混合场景

将整个系统除注册以外的所有场景集成为为混合场景进行性能测试，以测试这些关联交易是否存在性能问题。每个场景测试多组并发，并发数从20逐步增加到200。对于具体的场景，测试几组并发依据案例来执行（案例主要依据业务量来设计）。对于使用频度较低的场景，可以采用独立/混合场景的方式进行测试，即在背景压力下以小并发的方式来进行长时间疲劳测试。

Table 4 混合场景设计

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **并发数** | **响应时间要求** | **TPS要求** | **加压时间** | **加压方式** |
| 20 | <5秒 | N/A | 10分钟 | 1个用户/5秒 |
| 50 | <5秒 | >100笔/秒 | 20分钟 | 1个用户/5秒 |
| 100 | <5秒 | >100笔/秒 | 20分钟 | 1个用户/5秒 |
| 200 | <5秒 | >100笔/秒 | 10分钟 | 1个用户/1秒 |

# 

# 测试实施安排

## 测试进度

|  |  |  |
| --- | --- | --- |
| **时间** | **测试安排** | **操作者** |
| 12.24-12.28 | 学习性能测试基本知识，了解性能测试工具 | 孙雨晶 |
| 12.29-12.31 | 熟练掌握工具的使用方法，并分别录制脚本 | 孙雨晶 |
| 1.1-1.5 | 将测试脚本应用到不同的场景进行性能测试并对测试结果进行分析 | 孙雨晶 |
| 1.6-1.7 | 完善性能测试的相关报告 | 孙雨晶 |

## 测试流程

* 针对本项目进行特定的性能测试需求分析
* 制定性能测试计划
* 录制测试脚本
* 将测试脚本分别应用到不同的场景进行性能测试
* 对性能测试的结果进行分析
* 如果发现性能问题及时将其反馈给开发人员

# 性能测试标准

## 启动标准

1. 测试环境满足计划需求
2. 基准参数配置完成校验
3. 关键交易通过冒烟测试

## 中止标准

1. 测试环境或关键系统不可用
2. 测试环境距生产标准差距太大
3. 缺陷周转周期不符合规定的时间
4. 出现宕机、不响应等严重的性能问题
5. 系统的交易成功率低于95%

## 通过标准

系统上线至少满足下面标准：

1. 系统无宕机、不响应类的严重性能问题
2. 系统响应时间80%达到系统的期望值
3. 系统的业务吞吐量达到预期目标，即当前生产需求的3倍以上
4. 柜台类系统要求通过7\*12小时以上的疲劳强度测试
5. 电子渠道类系统要求通过7\*24小时以上的疲劳强度测试

# 测试环境规划

## 部署环境

服务器操作系统ubuntu16.04 64位 单核2G 带宽1Mbps

node v10.14.2

数据库 MairaDB 10.1.35

## 执行环境

Chrome 开发者工具

JMeter 5.0

# 测试风险分析

1. 可能会出现由于测试机器的性能问题导致在渐进式加压时最大用户并发数无法到达预期数量。
2. 可能会由于时间不足导致测试不够全面