# 性能测试方案

[性能测试方案 1](#_Toc10475)

[文档目的 1](#_Toc31563)

[测试目的 1](#_Toc2656)

[测试策略 1](#_Toc14133)

[第一轮测试： 1](#_Toc7270)

[第二轮测试 2](#_Toc31735)

[第三轮测试 2](#_Toc31049)

[第四轮测试 2](#_Toc4339)

[业务抽取 2](#_Toc12291)

[需要进行测试的主要业务： 2](#_Toc8664)

[用户行为模型及性能指标 3](#_Toc7100)

[监控方式 3](#_Toc19094)

[场景检查 3](#_Toc2148)

## 文档目的

描述BugFree性能测试的目的、策略、测试用例、流程等因素作为性能测试事实依据。

## 测试目的

评估BugFree在大负载下的稳定性

评估BugFree架构特点及处理能力

评估BugFree优劣

## 测试策略

### 第一轮测试：

首先进行64位平台下的BugFree测试执行，执行项目包括如下内容

Bug区：

1. 新建bug
2. 查询bug
3. 查看bug
4. 解决bug
5. 指派给某人的bug
6. 某人创建的bug
7. 标记的bug

Case区：

1. 新建case
2. 查询case
3. 查看case
4. 运行case

Result区：

1. 查询result
2. 查看result

需要搜集的关键数据：

各个测试执行行为的响应时间、注册完成数、吞吐量、带宽使用率、资源利用率

### 第二轮测试

获得相关数据后，针对该架构采取调整硬件配置的策略，进一步分析Phpwind的处理能力，容量扩展空间，性能瓶颈等，完成对该系统的负载测试。

第三轮测试

进行一次目标场景运行，确定BugFree是否满足客户需求（根据性能需求规格说明书指定目标）

### 第四轮测试

进行压力测试，确认系统能够在12h内稳定运行。

## 业务抽取

### 需要进行测试的主要业务：

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 业务名称 | 脚本编写要求 | 备注 |
| 1 | 建立bug报告 | 使用不同的bug数据提交bug报告，bug数据覆盖全部的业务规则，在并发的情况下，每次bug报告建立时间。 | 登陆后进行操作 |
| 2 | 查询bug报告 | 不同登录用户查询不同内容，获得查询返回所需要的时间。 | 登陆后进行操作 |
| 3 | 查看bug报告 | 在用户并发情况下，打开每个帖子所需要的时间。 | 登陆后进行操作 |
| 4 | 解决bug报告 | 使用不同的bug数据提交bug解决报告，bug数据覆盖全部的业务规则，在并发的情况下，每次bug报告解决时间。 | 登陆后进行操作 |
| 5 | 新建case | 使用不同的case数据，case数据覆盖全部的业务规则，在并发的情况下，每次case报告建立时间。 | 登陆后进行操作 |
| 6 | 查询case | 不同登录用户查询不同内容，获得查询返回所需要的时间。 | 登陆后进行操作 |
| 7 | 查看case | 在用户并发情况下，打开每个帖子所需要的时间。 | 登陆后进行操作 |
| 8 | 运行case | 填入所得运行结果，覆盖全部业务规则，在并发情况下，每个case运行的时间 | 登陆后进行操作 |
| 9 | 查询result | 不同登录用户查询不同内容，获得查询返回所需要的时间。 | 登陆后进行操作 |
| 10 | 查看result | 在用户并发情况下，打开每个帖子所需要的时间。 | 登陆后进行操作 |

### 用户行为模型及性能指标

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 测试场景描述 | 监控范围 | 备注 |
| 1 | 50个并发bug报告创建，并发率为10%  5个报告创建启动，每隔60s增加5个，持续5min，立即结束负载。 | 响应时间，服务器CPU，内存，磁盘开销 |  |
| 2 | 200个用户浏览  25个用户启动，每隔30s增加25个用户，持续5min，立即结束负载。 | 响应时间，服务器CPU，内存，磁盘开销 |  |
| 3 | 500个用户查询  10个用户启动，每隔30s增加25个用户，持续5min，立即结束负载 | 响应时间，服务器CPU，内存，磁盘开销 |  |
| 4 | 100个用户解决bug报告，解决并发率为10%。  10个用户启动，每隔30s增加10个用户，持续5min | 响应时间，服务器CPU，内存，磁盘开销 |  |
| 5 | 100个用户运行case，解决并发率为10%。  10个用户启动，每隔30s增加10个用户，持续5min | 响应时间，服务器CPU，内存，磁盘开销 |  |

## 监控方式

主要使用Rpc.rstatd服务将相关信息提交给Controller实现，对于详细分析时选择性使用Nmon及SpotLignt帮助进一步分析定位资源开销问题。

## 场景检查

日志关闭

集合点设置

IP虚拟关闭

地址屏蔽