

性能测试常见误区及如何避免无效压测

性能测试系列培训课程 2019/05/22 共享技术部-谢佳

背景及现状

性能测试常见误区

如何避免无效压测

背景及现状

- 没有专业的性能测试团队；
- 没有独立的性能测试环境；
- 很多都是临上线前开展性能测试，时间仓促，准备不足；
- 因为资源不足所以新手经常会被拉过来填坑性能测试任务；
- 需求方没有提出完整清晰性能要求，性能指标靠拍脑袋凑齐；
- 正向流程不容易记住，但是对于踩过的坑都会记忆犹新。

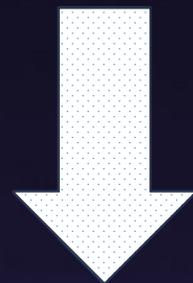
背景及现状

性能测试常见误区

如何避免无效压测

性能测试常见误区(1)

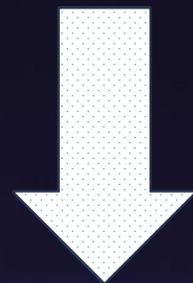
性能测试就是使用工具
模拟用户并发测试



工具依赖型

性能测试常见误区(2)

性能测试第一步是从脚本编写开始



不知为何而测

性能测试常见误区(3)

性能测试必须在功能测试之后开展



传统瀑布模式思维

性能测试常见误区(4)

性能测试要像功能测试
一样覆盖更多场景



不知道什么是重点

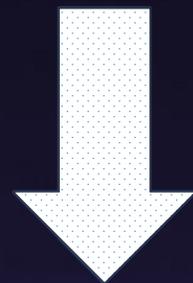
功能测试后简单压一压就行了，
并发量不大不需要进行压力测试



对性能问题带来的后果认识不足

性能测试常见误区(6)

性能测试是对后台的压测，所以
无需关注前端



对性能测试认识片面

提高硬件配置就能提高性能，处理能力随着硬件资源增加而变强



性能瓶颈不一定是硬件问题

不切实际的性能指标，过高或过低，很多时候性能指标靠拍脑袋



对性能指标来源没有任何质疑

性能测试常见误区(9)

客户或需求人员给出性能指标，
我们一定要想法设法达到



无形增添了沉重的压测负担

压测很随意，随便找服务器搭建
环境并压测



压测结果不具有可对比性

压测很随意，随便找一台办公用
电脑进行压测



客户端本身就是瓶颈！

性能测试常见误区(12)

使用JMeter可视化界面开展压力测试，简单又快捷



资源浪费在了图形处理上

性能测试常见误区(13)

JMeter脚本写的又全又健壮



资源浪费在了逻辑处理上

性能测试常见误区(14)

JMeter脚本日志打印的又多又全



磁盘IO瓶颈会影响并发量

为了尽快调试通过脚本，或者绕开技术难题，牺牲脚本业务逻辑



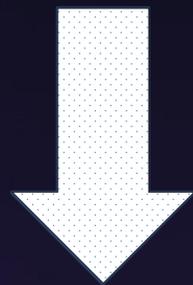
资源浪费在了逻辑处理上

为了简化请求过程，脚本使用相同数据，全程没有任何参数化



构思的场景简单但失真

脚本不加任何检查点或添加过多
检查点



到底是为了验证功能还是要提升并发量

所有脚本步骤增加集合点



集合点目的是什么？

性能测试常见误区(19)

脚本不添加思考时间



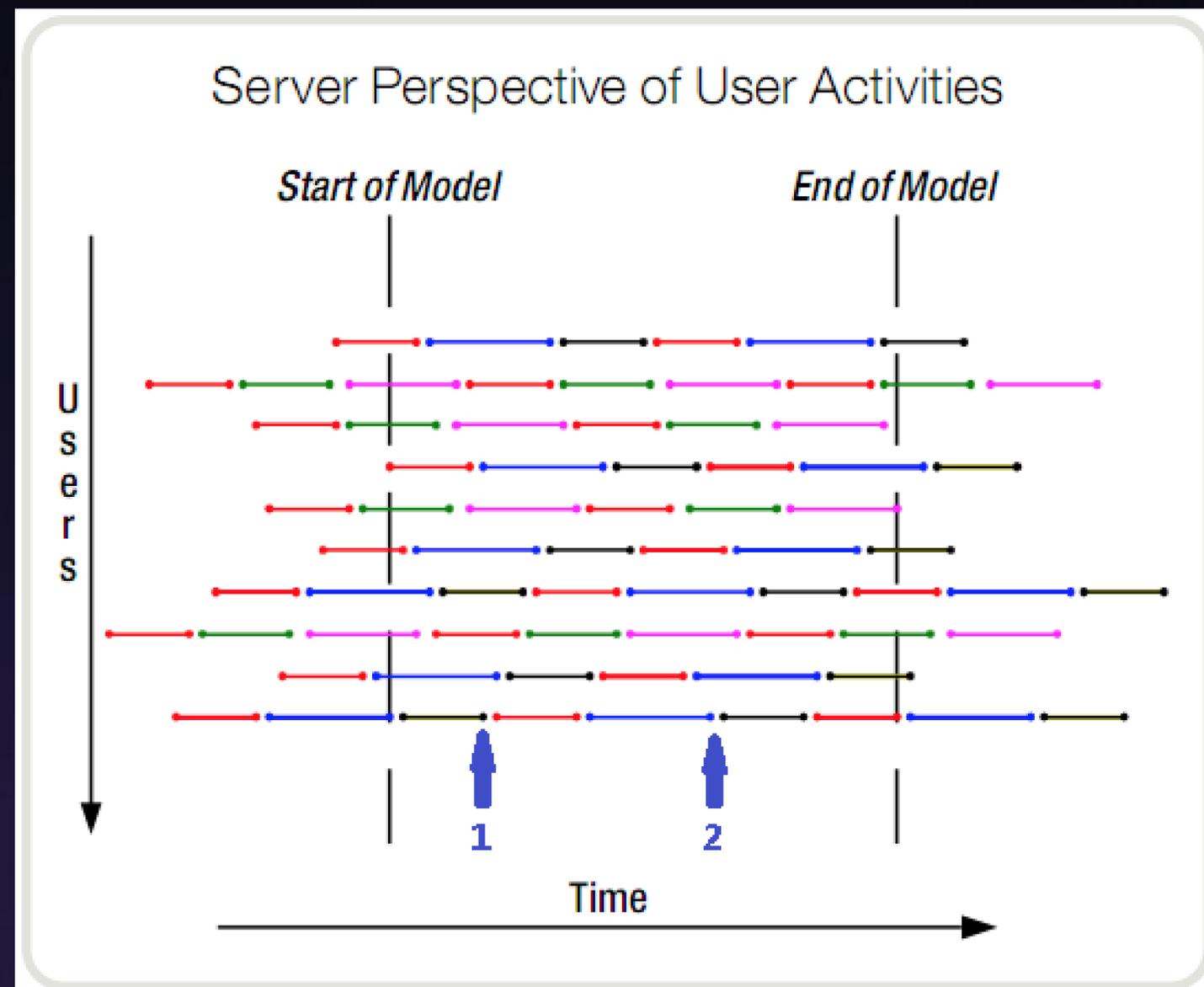
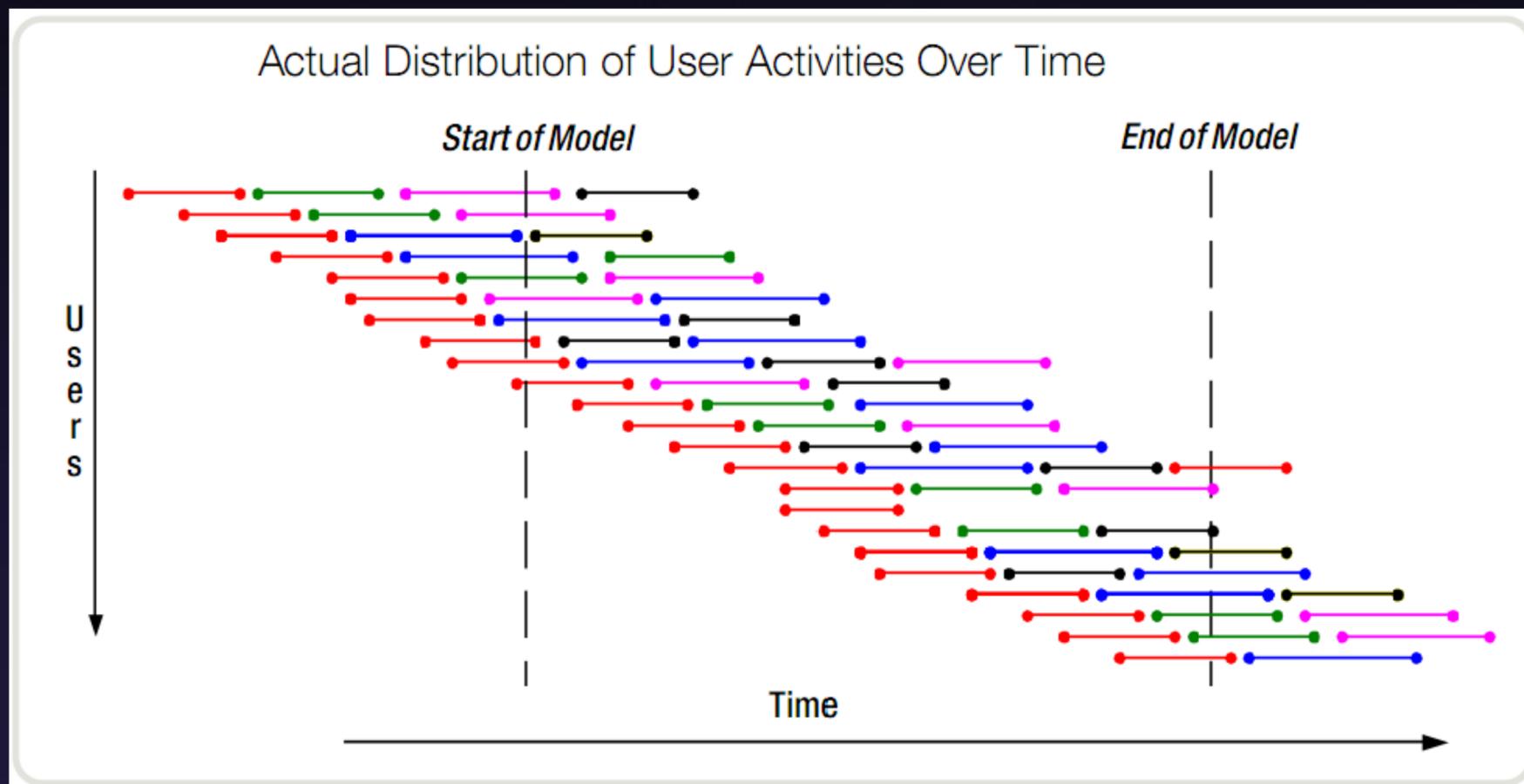
服务器额外承担了N倍隐形压力

性能测试常见误区(19-2)

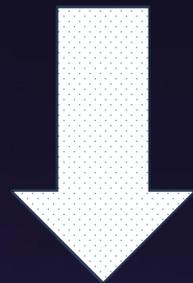
思考时间对并发执行影响

相对并发用户数
(用户视角)

绝对并发用户数
(服务器视角)



工具的测试结果就是我的测试结论和测试报告



统计从来不排除错误和偏差样本

为了节省时间压测五分钟最多不超过十分钟，所有样本取平均值



从不在乎统计结果样本质量和数量

性能测试常见误区-总结

压测要尽可能的贴近或等同真实场景

压测的请求本质是产生流量，工具或脚本只是流量生成器

流量要考虑数据组成、数据权重、数据路径、流量分布等

性能结果收集就是数据采样，需要对采样结果进行多维度分析

背景及现状

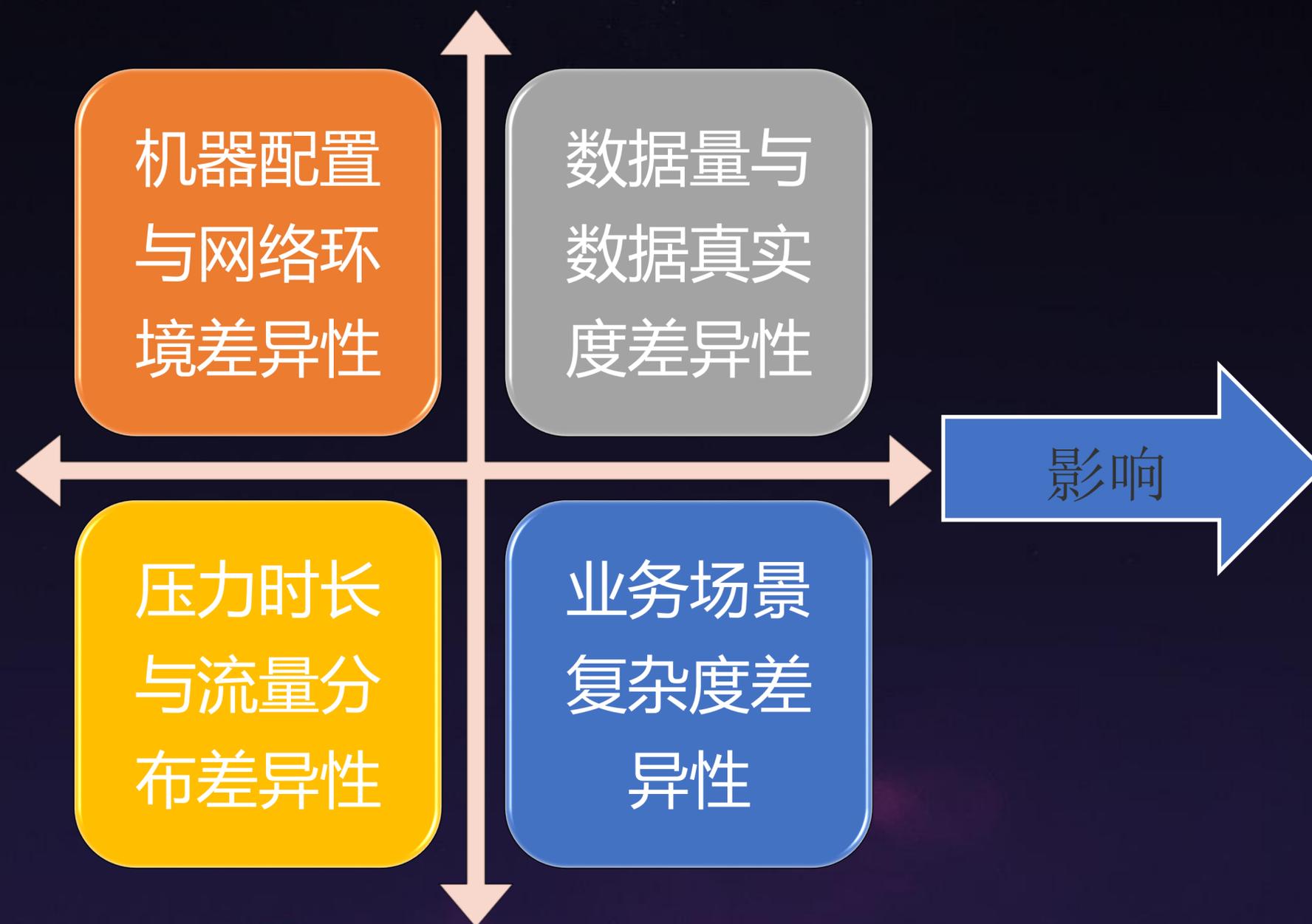
性能测试常见误区

如何避免无效压测

做好充分的性能需求调研



充分评估线上线下差异性因素及影响



非等同环境或等比例缩小环境，无法评估或无法折算成线上环境真实处理能力

检查硬件、网络等初始化配置



OS、系统软件、应用软件初始化配置



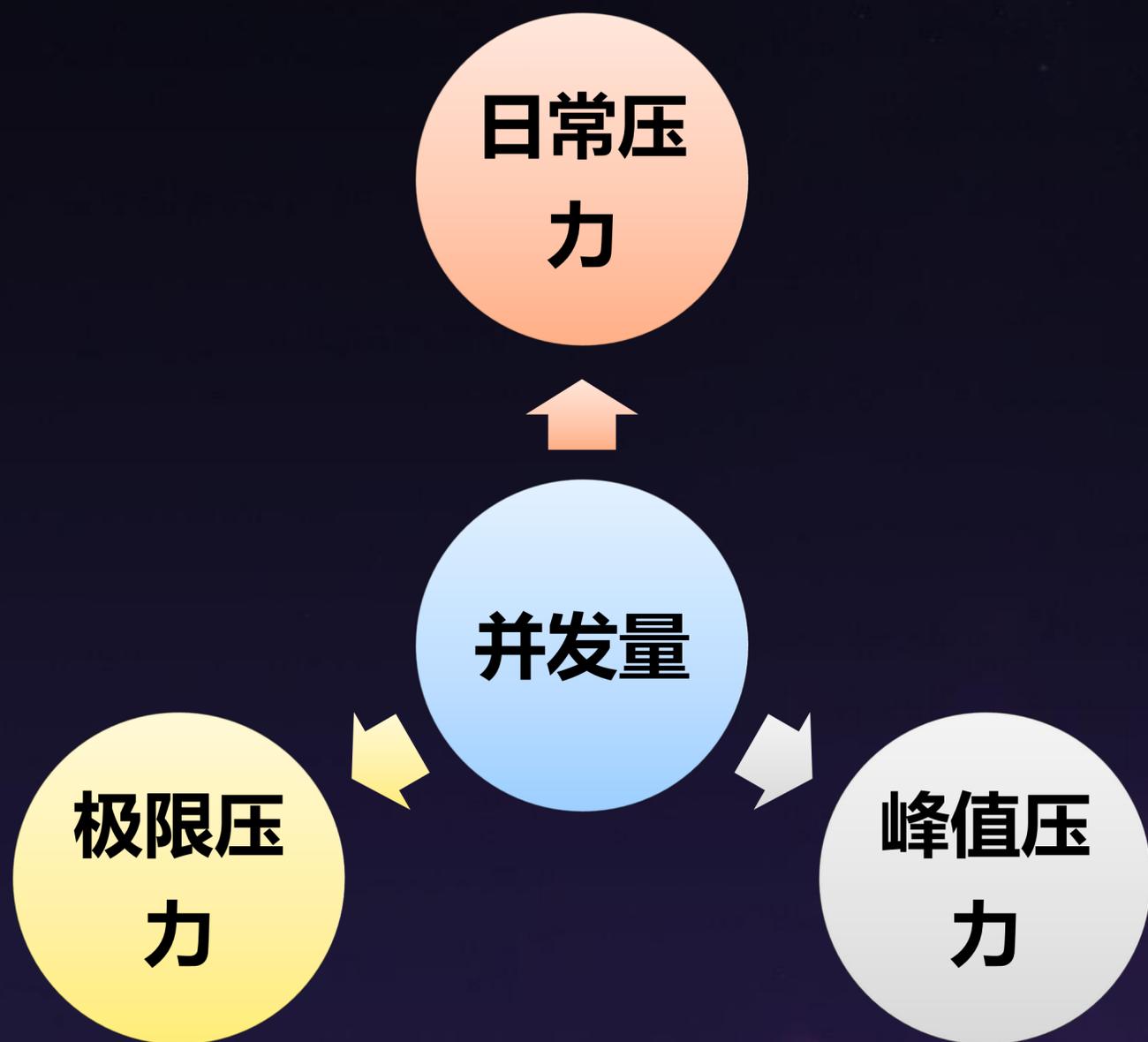
做好业务模型分析



做好数据模型分析



做好压力模型分析



用户加载策略

阶梯式

线性式

集中式

脉冲状

浪涌式

做好压测时长设置

压测执行时长

单场景时长 (10分钟)

混合场景时长 (30分钟)

稳定性测试 (8小时+)

大数据测试 (100W次)

执行等待时间

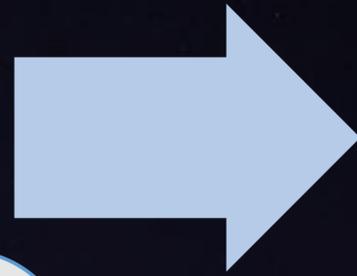
步骤之间等待时间

脚本迭代间隔时间

压测时同步观察业务指标变化情况

性能指标（点）

- 平均值
- 最小值
- 最大值
- 中位数
- 90%Line
- 95%Line
- 99%Line
- 每秒发送字节数
- 每秒接收字节数



性能图表（面）

- 用户数随时间变化图
- 响应时间随时间变化图
- TPS随时间变化图
- code(返回码)随时间变化图
- 响应时间随用户变化图
- TPS随用户变化图
- 响应时间区间分布图
- 响应时间百分比图

压测时同步做好服务器资源监控

系统资源

- 1、CPU内核数、CPU使用率 (sy、us、wa、id)、CPU平均负载 (1分钟、5分钟、15分钟)
- 2、Mem (总大小、已使用、空闲、可用、cache占用、buffer占用)、Swap (总大小、已使用)
- 3、磁盘IO (繁忙度、读写速度、队列数、平均服务时间、平均等待时间、空间利用率)
- 4、网络 (上行、下行)

JVM

- 1、GC : GC频率、Full GC频率、Full GC平均时长、GC总时间
- 2、Eden区使用大小、Eden使用率、Old区使用大小、Old使用率
- 3、堆初始化值、堆最大值、堆已使用、Metaspace初始化值、Metaspace最大值、Metaspace已使用

SQL

- 慢SQL、死锁
- 连接数 (打开连接数、激活连接数)、当前用户连接数、session
- Sql语句TPS (select、insert、update、delete、begin、commit、rollback)
- 每秒读/写字节数

集团自动化测试平台性能监控模块



学习参考地址-百度云盘学习视频分享

百度云盘JMeter学习视频分享

链接：https://pan.baidu.com/s/1_9lvQ5jXOGM4dSfux30_bQ

提取码：4qv5



学习参考地址-公司wiki学习地址

公司confluence学习地址

<http://confluence.midea.com/pages/viewpage.action?pageId=21466592>

<http://confluence.midea.com/pages/viewpage.action?pageId=17865945>

性能测试

- 性能测试体系
- 性能测试工具
 - JMeter
 - JMeter使用规范标准**
 - JMeter学习路线图
 - 【JMeter学习路线图】之【JM...
 - 脚本模板

接口测试

流程管理规范

测试周报

线上问题跟踪列表

空间管理

页面 /... / JMeter

JMeter使用规范标准

创建：谢佳，最新修改：陈国涛 于一月 15, 2019

本文档主要以附件形式体现，包含了以下5个附件：

- JMeter工程与脚本命名规范_V1.0_R1_201805191800.xlsx
- JMeter脚本执行过程记录表_V1.0_R1_201805191800.xlsx
- JMeter命令行参数详解及范例_V1.0_R1_201805191800.xls
- 数据准备-被测系统数据库铺底数据说明.xlsx
- 数据准备-测试脚本参数化数据说明.xlsx

远程站点: /home/jmeter/jmeterProject/systemName

- jmeter
 - apache-jmeter-3.1
 - jmeterProject
 - systemName (系统名称)
 - subSystemName (子系统名称)
 - data (数据及配置文件夹)
 - result (测试报告)
 - html (测试日志: 执行日志和统计日志)
 - jitl
 - script (测试脚本)
 - zjcgpt

性能测试工程命名规范说明:

1. 压力机的性能测试根路径“/home/jmeter/” 此路径下有2个子目录:

页面 /... / 培训

HTTP协议详解 (20180718培训ppt材料)

创建：谢佳，最新修改：匿名用户 于七月 23, 2018

MIP问题整改计划

安全管理工作

性能测试

接口测试

培训

- HTTP协议详解 (20180718培训ppt材料)**
- JMeter-http协议接口测试 (20180721)
- JMeter-使用命令行生成html报告 (20180801)
- JMeter-命令行执行详解 (20180830)
- JMeter常用功能介绍 (20180801培训)
- JMeter录脚本&过滤脚本(20180919培训)
- JMeter-聚合报告详解 (20180823培训)
- JMeter脚本工程目录结构及命名规范
- 接口测试前为什么要学http协议?

PDF

5人赞了它

问答时刻



THANKS