

# 性能测试

--计划测试

# 内容回顾

---

## ■ Configure SLA Rules

- 作用，使用方法

## ■ 集合点

- 定义
- 什么情况使用

# 内容回顾

---

## — 怎样使用

- 脚本中增加lr\_rendezvous(“事务名称”) ，加在被要求集合的请求前
- 在Controller中设置集合点规则
- 在Analysis中分析集合点生成图，分析并对比有无集合点响应时间、事务数等的变化
- 重点：理解集合点规则

# 内容回顾

---

## ■ 熟悉业务流程

## ■ 定义测试目标

### — 通过用户数据进行分析

- 用户给定的系统使用数据，分析性能测试指标：并发用户数、响应时间等

### — 通过系统日志进行分析

- 导出日志，使用分析工具进行分析

# 内容回顾

- 查看总的请求和平均请求量，日统计量分布，每小时统计量，找出峰值数据，针对峰值数据定义性能测试指标

## ■ 补充：

- $TPS = \text{并发数} / \text{响应时间}$
- $\text{并发数} = TPS * \text{响应时间}$

## 内容回顾

- 练习：某公司上班签到系统，早上8点上班，7点半到8点的30分钟的时间里用户会登录签到系统进行签到。公司员工为1000人，平均每个员工登录签到系统的时长为5分钟，请计算TPS，平均响应时间，并发数
  - $TPS = 1000 / (30 * 60)$  事务/秒
  - 平均响应时间为  $= 5 * 60$  秒
  - 并发数  $= TPS * 平均响应时间 = 1000 / (30 * 60) * (5 * 60) = 166.7$

# 目录

---

## ■ 定义性能测试指标

- 通过系统日志进行分析
- 通过同类业务进行分析
- 通过80/20原则进行分析

## ■ 编写性能测试计划

## ■ 编写性能测试方案

## ■ 编写性能测试用例

# 通过系统日志进行分析

- 系统负载是动态变化的
  - 公司内部C/S架构的系统，系统负载可控
  - 对外B/S架构系统，可能存在爆发式增长
- 如何应对用户量增长的问题？
  - 做容量规划



# 通过系统日志进行分析

## ■ 怎样做容量规划

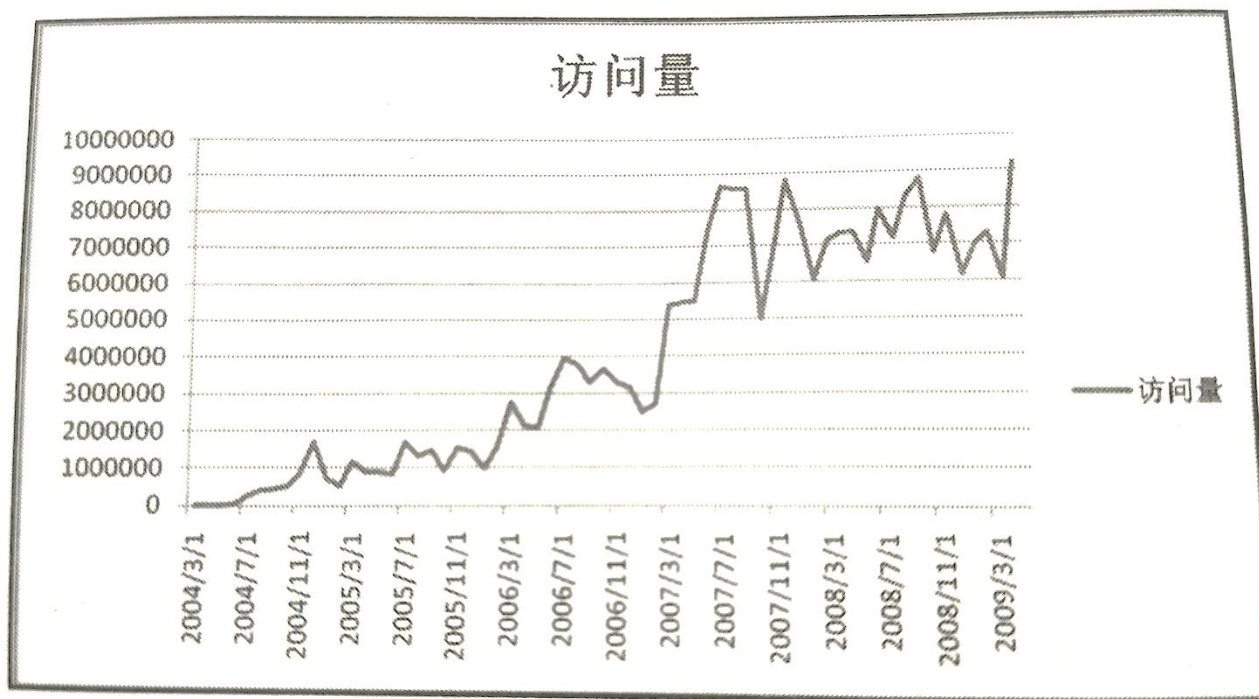
### — 容量：包含业务容量和处理容量

- 业务容量：业务的种类、业务的请求频率和业务的请求量
- 处理容量：最大处理能力，某个接口的处理吞吐量、处理响应时间和对资源的利用率
- 吞吐量：单位时间内CPU从存储设备读取、处理、存储信息的量（取决于CPU性能、时钟频率、周期数、指令条数等）

# 通过系统日志进行分析

## ■ 怎样做容量规划

- 可以参考日志的变化规律，通过一个模型来预估未来的容量

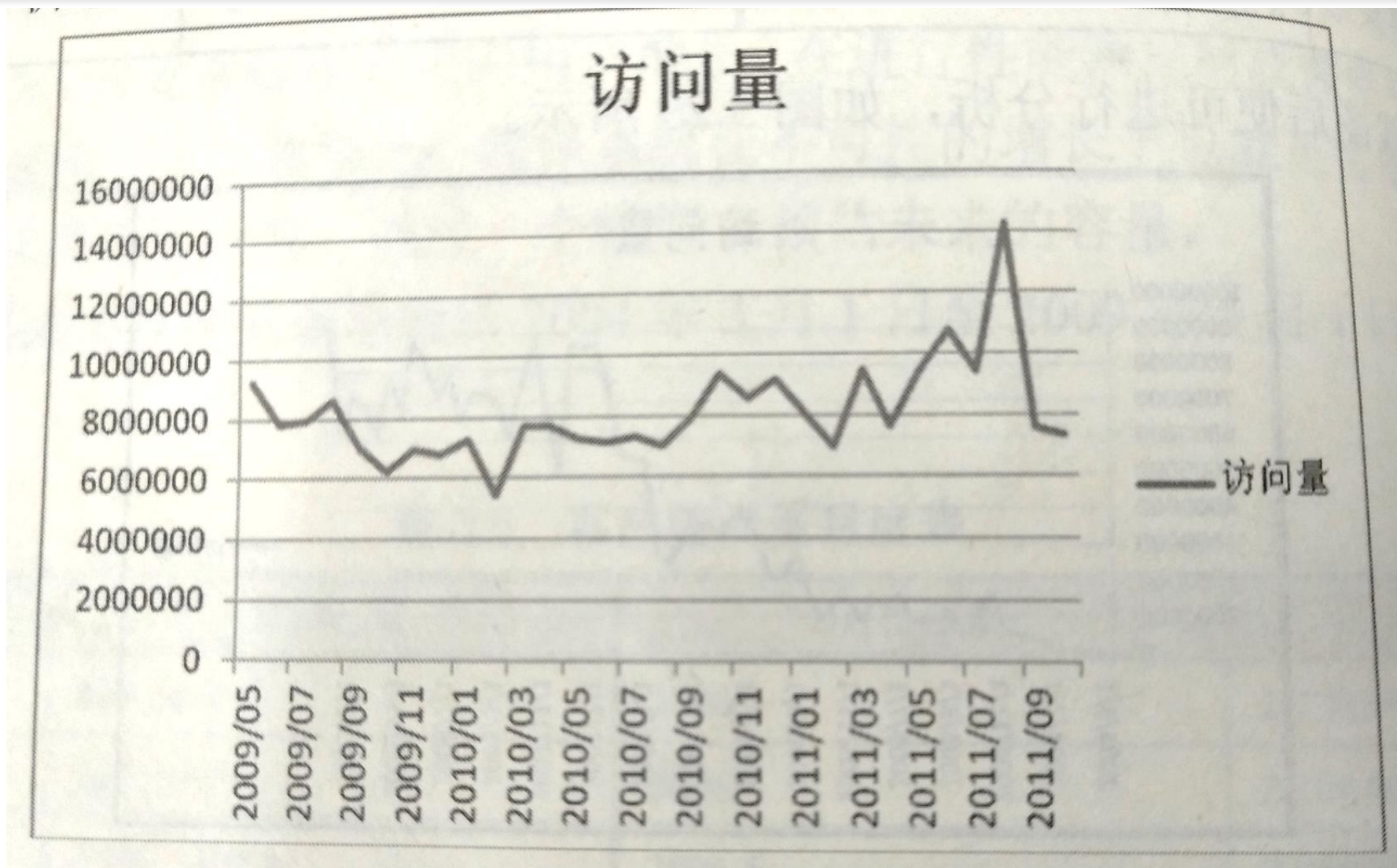


# 通过系统日志进行分析

## ■ 分析日志中的数据：

- 1 2007年3月—7月出现爆发式增长
- 2 随后两年内访问量比较稳定（1000万以内）
- 3 最高容量预估每月1100万
- 4 以上数据没有超过1100万的

# 通过系统日志进行分析



# 通过系统日志进行分析

## ■ 继续分析之后的日志，可以发现：

1 2009年5月—2011年7月没有超出之前的用户容量1100万

2 2011年8月出现了1400万访问量，超出了之前的预估

## ■ 预估是不是有问题？

1 2011年最高峰在8月份，1400万

2 按照前面20%的访问量评估策略， $1100 * 120\%$ ，1400万的访问量也基本在需求范围内

# 通过系统日志进行分析

## ■ 对于爆发式增长，通过什么方式来解决？

- 容量扩展规划和压力测试来解决

## ■ 容量规划

- 进行容量规划时，必须对访问量和访问规律有比较清晰的分析，才能确保推导容量时尽量合理
- 容量规划又有时效性，过分的容量规划会大大提高成本

# 通过系统日志进行分析—容量规划

## ■ 举例：

- 普及千兆网络入户的小区，从长远来看有必要，避免以后不做百兆升千兆的调整
- 但是这些年百兆网络已经绰绰有余
- 现在布置千兆网的设备与将来调整成本差别不大，所以这种一步到位的想法不太合理
- 更多的时候考虑扩展性



# 通过系统日志进行分析—容量规划

## ■ 举例：磁盘寻址方式的容量规划

- 20实际80年代初，系统管理员为了方便使用不同文件系统存储数据发明了主引导记录（MBR），当时支持最多32位信息，也就是 $2^{32}$ 字节，大约2.2T，当时来看是天文数字（当时最大的民用硬盘不过5MB，价格超过1500美元）
- 90年代时，人们已经意识到2.2TB容量限制问题，为此多家公司联合制定了一种弹性分区机制，就是GUID分区表，简称GPT，GPT最多支持64位信息，理论上最大容量是9.4ZB



## 通过系统日志进行分析—容量规划

---

- 如果将来9.4ZB不够用了，可以将信息的容量升级到128位，整体升级的成本和复杂性大大降低

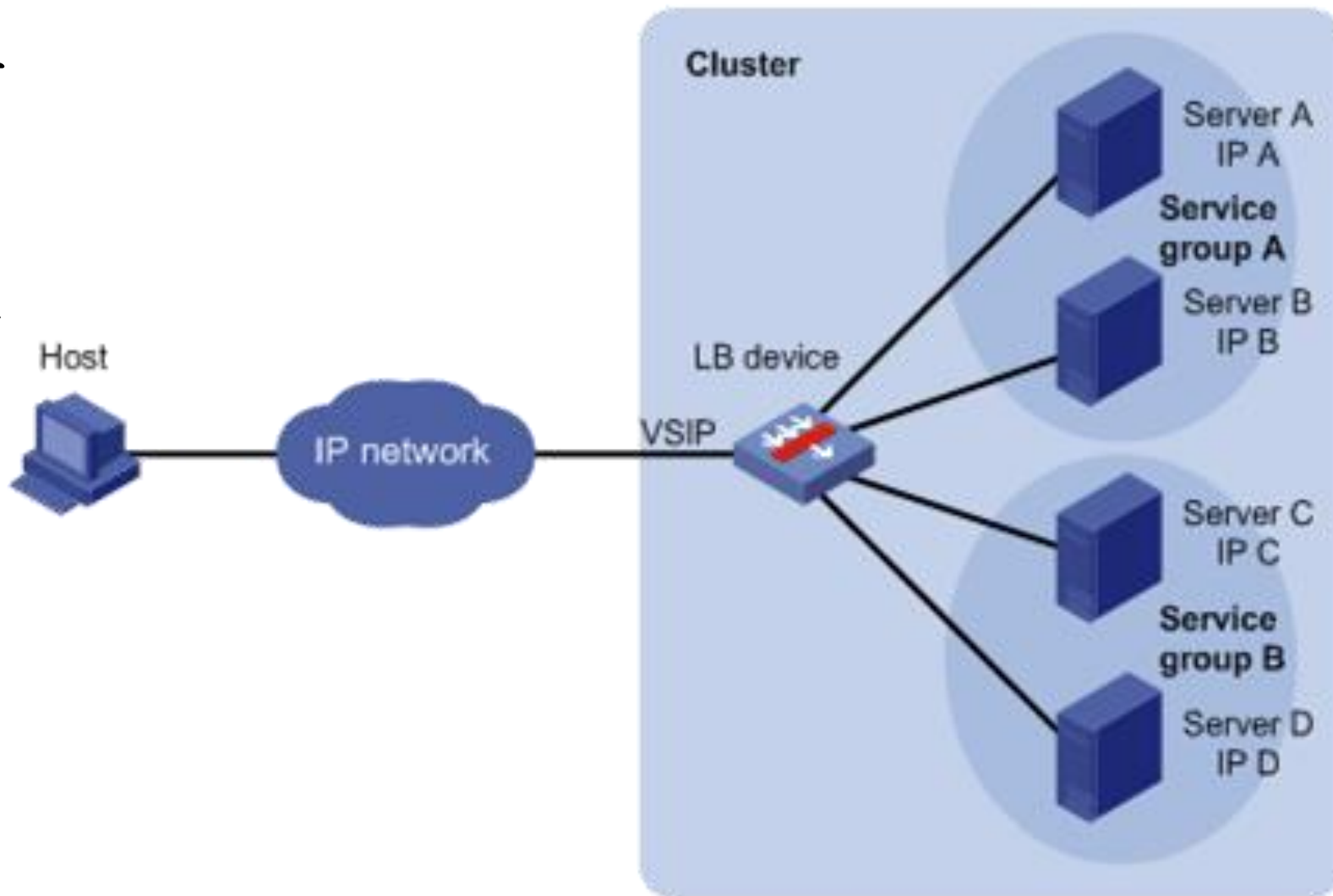
## 通过系统日志进行分析—容量规划

- 处理容量规划主要是针对业务容量规划，评估当前处理能力是不是能满足需求中对当前及未来的业务需要，如果不满足是否可以通过**易扩展的策略**来完成升级操作
  - 例如：当前每分钟30笔业务，未来两年中有可能提升到300笔/分钟，需要系统能够满足当前及未来两年中的容量需求，甚至有所富余

# 通过系统日志进行分析—容量规划

## ■ 提升容量的方法

- 硬件提升
- 负载均衡策略



# 通过系统日志进行分析—容量规划

- 思考：如果一台服务器可以处理2000个TPS，20000个TPS是不是需要10台服务器
  - 不是
  - 服务器越多，处理能力和同步能力就越难
- 验证当前系统处理能力对硬件的依赖性，确认是否能够按照硬件提升来获得线性处理能力的提升
  - 通过性能测试发现和定位随着业务容量上升中的拐点，确定系统扩展能力

## 通过系统日志进行分析—容量规划

- 举例：1小时内，系统承受10万用户在线，页面刷新96万次、点击量640万次、带宽吞吐量382G，在该负载下CPU及内存资源利用率低于80%，无明显磁盘网络瓶颈
  - 通过硬件评测，一般每台服务器支撑动态请求1000个/(台.s)，每台服务器支撑静态流量400MB/(台.s)，每个页面包括1-3个动态请求，问需要多少台服务器完成动态请求
    - 每秒并发请求： $(960000 * 3) / (1 * 3600) = 800$ 个/s
    - 峰值并发10倍计算： $(800 * 10) / 1000 = 8$  台

## 通过系统日志进行分析—容量规划（修改）

- 预计每个页面包含img/js/css等50个静态文件，预计每个文件5Kb，平均并发是多少？平均带宽是多少？根据带宽来看，需要多少台服务器？
  - $960000/(1*3600) * 50 = 13\ 333/s$
  - 峰值并发约 140 000
  - $(960000*50*5/8)/3600 = 8333\ KB$  (约8M)（平均带宽）
  - 峰值带宽约80MB/s
  - 从带宽角度计算，只需要一台服务器

## 通过系统日志进行分析—容量规划

- 预计每个页面包含img/js/css等50个静态文件，预计每个文件5Kb，平均并发是多少？平均带宽是多少？根据带宽来看，需要多少台服务器？
  - $960000 / (1 * 3600) * 50 = 13\ 333/s$
  - 峰值并发约 14 000
  - $960000 / (3600 * 50 * 5/8) = 0.521GB/s$ （平均带宽）
  - 峰值带宽约5—5.5GB/s
  - $(5-5.5) * 1000 / 400 = 13—14$ 台服务器

# 定义性能指标—通过日志进行分析

## ■ 通过日志分析得到

- 数据变化规律
- 峰值数据的计算
- 容量规划等等



# 目录

---

## ■ 定义性能测试指标

- 通过系统日志进行分析
- 通过同类业务进行分析
- 通过80/20原则进行分析

## ■ 编写性能测试计划

## ■ 编写性能测试方案

## ■ 编写性能测试用例

# 定义性能指标—通过同类业务进行分析

- 如果新建一个项目，没有日志，怎样定义性能指标
  - 分析同类型项目
  - 奥运会售票系统
    - 参考国外奥运会售票系统
    - 预上线尝试
  - 比如：要做一个类似CSDN规模的技术论坛，怎样定义性能测试指标
    - 打电话了解访问量和相关业务

# 目录

---

## ■ 定义性能测试指标

- 通过系统日志进行分析
- 通过同类业务进行分析
- 通过80/20原则进行分析

## ■ 编写性能测试计划

## ■ 编写性能测试方案

## ■ 编写性能测试用例

## 定义性能指标—通过80/20原则进行分析

---

- 80%的用户只使用20%的功能，80%的业务只有20%的人使用
  - 对于性能测试来说，应该将精力集中在用户最常使用的功能上
- 通过80/20规则结合前面的日志或同类项目进行功能划分，找到性能测试的需求

# 定义性能指标——生活中的例子

## ■ 地铁站换乘大厅的楼梯应该修建多宽？

- 假设发车频率是2分44秒，每节车厢设计容量320人（高峰期挤入400人），假设地铁8节车厢，大概能容纳3200人，地铁站是双向的，高峰期发车频率平均1分22秒会有1列地铁到站下车。假设到达该站有65%的乘客需要下车换乘，也就是1分22秒将有 $3200 * 0.65 = 2080$ 名乘客从月台上疏散，对乘客的疏散能力做一个限制，将1分22秒，限制在1分10秒（提供12秒的富余空间）

## 定义性能指标—生活中的例子

- 70秒如何疏散2080名乘客？假设有4个楼梯，每个楼梯平均疏散多少名乘客？每个楼梯每秒种需要疏散多少名乘客？需要设计同时有多少名乘客上楼的楼梯？（大约多少米宽）
  - 每个楼梯乘客数： $2080/4 = 520$ 名
  - 每个楼梯每秒种需要疏散的乘客数： $520/70 = 8$ （约等于）
  - 预计宽度：6米
- 如果同时考虑有下楼梯的乘客，宽度应该设计多少合适？

# 定义性能指标

## ■ 性能测试的关键

- 知道用户会做什么
- 知道用户怎么做这件事
- 知道多少用户做这件事情
- 知道对应的性能指标
- 清楚每个指标的估算方法

## 练习

- 如何在PC环境上执行性能测试去评估线上环境（假设线上环境在阿里云），如果现有系统每秒2000笔交易，测试环境每秒500笔交易，是不是如果有4台一样的测试环境做负载均衡就能处理2000笔业务？
  - 不是，原则上用5台
  - 原因：不一定都均衡



# 练习

## ■ 由业务需求，确定TPS

- 某银行每天有100W笔存款，平均6小时完成，问：TPS并发是多少？
  - $100W / (6 * 3600)$
- 某客户管理系统，现有用户50000，每天新增100个用户，求TPS？
  - $100 / (24 * 3600)$  ?
  - 经过与客户沟通，其新增用户业务主要集中在每天上班前两小时和下班前两小时
  - $100 / (4 * 3600) = 0.0069$
  - 不需要做性能测试

## 练习

- 客服使用的电话系统，1000个客服，每个客服每天打120个电话，TPS是多少？
  - $1000 * 120 / (8 * 3600) = 4$

# 性能指标计算总结

- $\text{TPS} = \text{事务总数} / \text{时间}$
- Hits（用户点击量）：一个页面有多少个请求，就等于多少个hits（抓包可以得到）
- 带宽：整个页面的大小就是所需要的带宽数
- 并发用户数：总带宽/每个用户所需要的带宽

# 定义性能指标

■ 定义性能指标主要为在一定负载下：

- 响应时间
- 服务器资源占用率
- 不同业务的最大吞吐量
- 系统能支持的并发用户数

# 目录

---

## ■ 定义性能测试指标

- 通过系统日志进行分析
- 通过同类业务进行分析
- 通过80/20原则进行分析

## ■ 编写性能测试计划

## ■ 编写性能测试方案

## ■ 编写性能测试用例

# 编写性能测试计划

## ■ 为什么编写性能测试计划

- 为整个性能测试阶段的管理工作和技术提供指南

## ■ 测试计划需要包含的内容：

- 在测试计划中需要明确测试的内容和范围，为评价系统提供依据
- 对设备及人员资源的需求
- 对测试结果的评价指标

## ■ 如果被测系统比较简单，在计划中明确测试方法及对象；如果比较大的项目，除了测试计划还需要性能测试方案

# 性能测试计划书写要点

## ■ 文档目的

- 描述性能测试流程、范围、环境、风险等因素作为性能测试实施的依据

## ■ 项目背景介绍

- 介绍该项目的项目背景，主要功能、业务场景等

## ■ 术语及缩写

- 专有名词，如：负载测试、压力测试、配置测试、基准测试以及用到的软件比如LAMP、LNMP、LANMP

# 性能测试计划

## ■ 参考文档

- 《项目文档》、《性能需求规格说明书》、《系统架构设计文档》、《系统测试计划》、入口标准等

## ■ 系统运行环境

- 网络运行拓扑图
- 软硬件配置



# 性能测试计划

## ■ 测试内容

- 浏览、发帖、注册及查询进行性能测试，评估LAMP、LNMP、LANMP三大主流架构在CentOS系统32位及64位下的运行表现，确认哪种架构更适合此论坛系统，进一步得到各个功能在一定平台下的最大数据处理能力.....

# 性能测试计划

## ■ 非测试内容

- 由于以下功能在真实情况中使用比较少，并对响应时间无明确需求，故不进行测试
  - 用户间短消息功能
  - 帖子移动管理功能
  - 论坛后台管理功能

## 性能测试计划—角色和职责

角色	资源数量	职责	备注
测试经理	1	跟踪监督性能测试项目进度 审核性能测试报告	
高级性能测试工程师	1	撰写性能测试计划 分析性能需求，制定性能测试方案，测试用例 设计性能测试场景及性能测试平台 辅助开发人员修改性能缺陷 撰写测试报告	
性能测试工程师	1	开发性能测试脚本 执行性能测试场景 搭建测试环境	

## 性能测试计划—工具列表

测试工具	版本	许可	用途	备注
LoadRunner	V11	Web 10000 Vuser	性能测试	
SVN	V1.6.5		配置管理	
HttpWatch	V7.0		HTTP分析	
Word	V2016		撰写测试报告	
Visio	V2016		绘制流程图	
Project	V2016		跟踪项目进度	
Spotlight			CentOS及MySQL监控	
Nmon			CentOS监控	
Nmon Analysis			Nmon日志分析	
RPC.rstatd			CentOS监控	

# 性能测试计划

## ■ 进度安排

任务名称	起始时间	结束时间	工作日	资源
测试计划	2019-3-5	2019-3-7	2	测试经理、高级性能测试工程师
测试方案				
测试用例及脚本开发				
测试环境搭建				
测试执行				
测试报告总结				

# 性能测试计划

## ■ 出口标准

- 获得各平台下的测试数据，评估得到最佳运行平台
- 评估得到最佳平台上处理能力峰值
- 压力测试连续12小时无故障
- 测试结果达到预期目标，系统满足用户处理能力及稳定性需求
- .....

# 性能测试计划

## 交付物

序号	交付件名称	交付件编号	备注
1	《性能测试计划》		
2	《性能测试方案》		
3	《性能测试用例》		
4	《LoadRunner性能测试脚本》		LoadRunner格式
5	《LoadRunner脚本业务报告》		LoadRunner格式
6	《LoadRunner性能测试场景及测试结果》		LoadRunner格式
7	《性能测试报告》		

# 性能测试计划

## ■ 风险

- 受环境限制，无法在测试环境中模拟系统真实上线情况下LVS负载均衡及CDN环境，测试环境服务器配置与在线架构有一定误差，故该结构无法完全代表在线运行情况



# 目录

---

## ■ 定义性能测试指标

- 通过系统日志进行分析
- 通过同类业务进行分析
- 通过80/20原则进行分析

## ■ 编写性能测试计划

## ■ 编写性能测试方案

## ■ 编写性能测试用例

# 编写性能测试方案

---

- 文档目的
- 测试目的
- 测试策略
- 业务抽取（测试脚本）
  - 需要进行测试的主要业务
  - 用户行为模型及性能指标（测试场景）
- 监控方式
- 场景检查

# 性能测试方案

## Phpwind 系统性能方案

### 文档目的

描述 Phpwind 性能测试目的、策略、测试用例、流程等因素作为性能测试实施依据。

### 测试目的

- 评估 LAMP、LNMP、LANMP 三大架构优劣。
- 评估 CentOS 5.5 32 位系统及 CentOS 5.6 64 位系统优劣。
- 评估 Phpwind 85 架构特点及处理能力。
- 评估 Phpwind 85 及 Discuz X2 优劣。
- 评估 Phpwind 85 在大负载下的稳定性。

### 测试策略

第一轮测试：

首先进行 32 位平台下 LAMP、LNMP、LANMP 三种架构下的 Phpwind 85 论坛测试执行，执行项目包括如下内容。

- 多用户注册。
- 多用户在线发帖。
- 大容量多用户浏览帖子。
- 大容量多用户查询帖子。



得到相关数据后，进行 64 位平台下的同样项目架构测试，搜集其中关键性能数据：

- 注册响应时间、注册完成数、吞吐量、带宽使用率、资源利用率。
- 发帖响应时间、发帖完成数、吞吐量、带宽使用率、资源利用率。
- 帖子浏览响应时间、浏览完成数、吞吐量、带宽使用率、资源利用率。
- 查询响应时间、查询完成数、吞吐量、带宽使用率、资源利用率。

通过对比这样的两大系统、3 种架构、4 项测试所获得的数据，确定何种平台架构最适合 Phpwind 85 论坛运行。

#### 第二轮测试：

在获得该平台架构数据后，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：

在最优平台下进行 Discuz X2 对比性能测试，使用同样的负载策略及用户操作模式，横向对比 Discuz X2 和 Phpwind 85 两个论坛的优劣。

#### 第四轮测试：

进行一次目标场景运行，确定 Phpwind 85 是否满足用户需求（根据性能需求规格书指定目标）。

#### 第五轮测试：

进行压力测试，确认系统能够在 12 小时内稳定运行。



平台对比测试策略如下。

每轮测试均执行3次,环境安装完毕后通过 VMWare 制作一个 Snapshot 为测试环境回归提供基础。测试结果取最大值,用来完成平台对比测试(如果数据偏差过大超过5%,则需根据情况补充测试执行)。

每轮测试前均需要重载 Snapshot 确保每次测试前环境相同,并且对将要执行的功能先手工执行一次,以减小误差。

性能测试准备流程如下。

在虚拟机下安装 CentOS 5.6 32 位和 64 位系统,分别安装对应的 LAMP、LNMP、LANMP,安装步骤细节参考《环境搭建手册》。

LANMP2.1 包含的组件版本如下。

- httpd-2.2.17。
- nginx-0.8.54。
- php-5.2.17。
- mysql-5.1.56。
- phpmyadmin-3.3.7。
- zend-3.3.3。
- eAccelerator-0.9.6.1。
- pure-ftpd-1.0.32。

性能测试执行流程:





- 通过 Snapshot 还原被测系统。
- 在场景中运行指定脚本。
- 为每一个脚本执行设置规范的目录名称（脚本编号+被测环境名称）。
- 执行完成保存场景。
- 打包场景数据记录场景执行信息。

### 业务抽取（测试脚本）

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	注册用户	不同的用户使用不同的注册数据，注册数据覆盖全部的业务规则，在用户并发情况下，每个用户注册时间	
2	浏览帖子	游客浏览不同板块下的首页帖子，在用户并发情况下，打开每个帖子所需要的时间	用户会随机选择板块进行浏览，这里的数据需要使用参数化
3	查询	不同登录用户查询不同内容，获得查询返回所需要的时间	查询内容需要随机
4	回帖	不同登录用户在板块中回帖，回帖提交所需要的响应时间和登录所需要花费的时间	用户随机选择板块进行发帖



用户行为模型及性能指标 (测试场景)

序 号	测试场景描述	监控范围	备 注
1	50 个并发注册用户, 并发率为 10% 5 个用户启动, 每隔 60 秒增加 5 个用户, 持续 5 分钟, 立即结束负载	注册响应时间 服务器 CPU、内存、磁盘开销	系统容量为: 1 个论坛板块, 共 500 万条帖子及 3 万名会员 带宽设置均为 512kb/s
2	500 个游客浏览帖子 25 个用户启动, 每隔 30 秒增加 25 个用户, 持续 5 分钟, 立即结束负载	浏览响应时间 服务器 CPU、内存、磁盘开销	同上
3	500 个用户查询 10 个用户启动, 每隔 30 秒增加 25 个用户, 持续 5 分钟, 立即结束负载	查询响应时间 服务器 CPU、内存、磁盘开销	同上
4	100 个用户登录后发帖, 回帖并发率为 10%, 登录不设置并发 10 个用户启动, 每隔 30 秒增加 10 个用户, 持续 5 分钟, 立即结束负载	登录、发帖响应时间 服务器 CPU、内存、磁盘开销	同上





5	目标场景： 测试目标为系统支持发帖 20 TPS	发帖响应时间 服务器 CPU、内存、磁盘开销	同上
6	压力测试场景 脚本比例为： (1) 25% (2) 25% (3) 25% (4) 25% 总用户数为 1 800，50 个 用户启动，每隔 30 秒增加 10 个用户，持续 7 小时	各事务响应时间 服务器 CPU、内存、磁盘开销	同上 出于模拟项目实施考虑，脚本并不做复杂的逻辑设计并且使用简单的脚本均分比例

### 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 Spotlight 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。



# 目录

---

## ■ 定义性能测试指标

- 通过系统日志进行分析
- 通过同类业务进行分析
- 通过80/20原则进行分析

## ■ 编写性能测试计划

## ■ 编写性能测试方案

## ■ 编写性能测试用例

# 编写性能测试用例

用例编号	1
用例名称	注册用户
用例步骤	<ol style="list-style-type: none"><li>1. 访问首页</li><li>2. 点击注册</li><li>3. 输入用户名及密码</li></ol> 用户名使用随机模式（虚拟用户编号+时间），密码使用统一密码 51testing <ol style="list-style-type: none"><li>4. 点击注册，完成注册</li></ol>
备注	在注册功能上添加手工事务检查是否出现注册成功关键字，注册前添加集合点，前添加思考时间 10 秒
用例编号	2
用例名称	浏览帖子
用例步骤	<ol style="list-style-type: none"><li>1. 访问首页</li><li>2. 访问板块</li><li>3. 随机选择一个帖子</li></ol> 帖子由于是使用 DF 生成的 500 万条记录，所以只要使用参数化随机帖子 ID 即可完成访问，不考虑翻页
备注	在打开帖子操作上添加事务，无须手工验证，无集合点，选择帖子前 5 秒思考时间
用例编号	3



用例编号	3
用例名称	登录查询帖子
用例步骤	论坛首页 使用随机用户名登录 由于用户是使用 DF 生成的，密码相同，所以只需要使用参数化随机即可在查询条件中输入 5 位长度随机英文内容，点击查询
备注	查询前 5 秒思考时间
用例编号	4
用例名称	登录回复帖子
用例步骤	论坛首页 使用随机用户名登录 随机选择一个帖子 由于用户是使用 DF 生成的，密码相同，所以只需要使用参数化随机即可回复帖子内容为随机 255 个字符内容
备注	对回复帖子做手工事务，添加集合点，登录添加事务，回复前 10 秒响应时间

# 内容总结

---

## ■ 定义性能测试指标

- 通过系统日志进行分析
- 通过同类业务进行分析
- 通过80/20原则进行分析

## ■ 编写性能测试计划

## ■ 编写性能测试方案

## ■ 编写性能测试用例



# Question

---