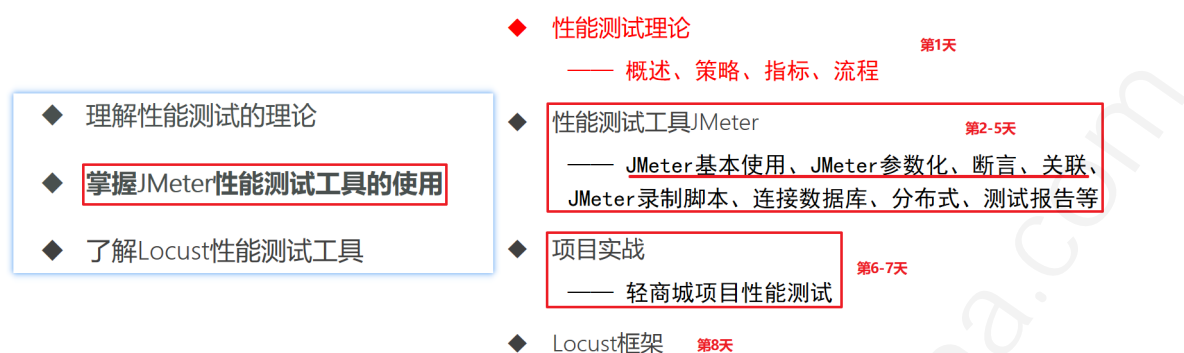


## 性能测试课程总体目标:



## 性能测试第一天课堂笔记

### 当日学习要点

- 理解性能测试的定义和目的
- 理解性能测试中常见的测试策略
- 理解性能测试中常见的性能指标
- 理解性能测试的流程
- 能够对比说出Loadrunner和jmeter的优缺点

### 性能测试概述

#### 为什么要进行性能测试?

- 满足真实场景的业务需求
- 招聘需要

#### 性能的概念:

### (1) 什么是性能？

- **时间**：系统处理用户请求的响应时间
- **资源**：系统运行过程中，系统资源的消耗情况

### (2) 什么是性能测试？

使用**自动化工具**，模拟**不同的场景**，对**软件各项性能指标**进行测试和评估的过程。

### (3) 什么是性能测试的目的？

- 评估**当前系统能力**
- 寻找性能瓶颈，**优化性能**
- 评估软件是否能够满足**未来的需要**

## 性能测试和功能测试：

### (1) 功能测试和性能测试有什么不同？

- 功能测试：验证系统的功能需求规格。焦点：功能 (**正向、逆向**)
- 性能测试：验证系统的业务需求场景。焦点：**时间、资源**

### (2) 功能测试和性能测试有什么关系？

- 一般项目中，**先功能测试**通过后，**后进行性能测试**

## 性能测试分类：

---

### 基准测试：

## (1) 什么是基准测试？

- 狭义上讲：就是**单用户测试**。（单用户循环多次得到的数据）
- 广义上讲：建立**基准线**，当系统的**软硬件环境发生变化**之后再进行一次基准测试以**确定变化对性能的影响**。

## (2) 基准测试数据的用途？

- 基准测试不会单独存在
- 为多用户并发测试和综合场景测试等提供参考依据
- 为系统/环境配置、系统优化前后的性能提升/下降提供参考指标

## 负载测试

### 1、概念：

通过**逐步增加系统负载**，确定在**满足系统的性能指标（如响应时间等）**情况下，找出系统所能够承受的**最大负载量的测试**。

### 2、作用

系统**最大负载量**达到**用户要求**时，系统才能正式上线使用。

电梯行业规范：电梯从1楼到5楼（15m）的运行时间不超过24s

进行负载测试：

case1： 1人乘坐电梯，从1楼到5楼，运行时间为20s

case2： 7人乘坐电梯，从1楼到5楼，运行时间为20s

case3： 13人乘坐电梯，从1楼到5楼，运行时间为20s **最大负载量**

case4： 16人乘坐电梯，从1楼到5楼，运行时间为25s

Case5： 19人乘坐电梯，从1楼到5楼，运行时间为28s

Case6： 21人乘坐电梯，从1楼到5楼，运行过程中绳子断了。。。

注意：

- 通过负载测试，可以确定系统的最大负载量和极限负载量
- 系统对外宣称的最大负载量
- 负载测试的时间一般为1-2小时

## 稳定性测试：

### 1、概念：

在服务器**稳定运行（用户正常的业务负载下）**的情况下进行**长时间测试（1天-1周等）**，并最终保证服务器能满足线上业务需求。

### 2、作用

系统**在用户要求的业务负载下**运行达到**规定的时间**时，系统才能正式上线使用。

## 其他分类：

### 压力测试：

#### 1、概念：

在强负载下的测试，查看系统在**峰值情况下**是否功能隐患、系统是否具有**良好的容错能力和可恢复能力**。

#### 2、测试场景

- **极限负载**情况下的**破坏性压力测试**
- **高负载**下的长时间的**稳定性压力测试**

分类：

- 高负载下的长时间稳定性压力测试（如：B-C区间内进行24/3\*24小时长时间测试）
- 极限负载下的破坏性压力测试（如：C-D区间内进行测试）

### 并发测试：

#### 1、概念：

并发测试（绝对并发）：是指在**极短的时间内**，发送**多个请求**，来验证服务器对并发的处理能力。

#### 2、应用场景

特定活动场景，如：抢红包、秒杀、抢购等。

## 生活中的案例：

悬赏任务： 做菜 —— 西红柿炒鸡蛋（但是只有一个鸡蛋和一个西红柿）



与负载测试对比：

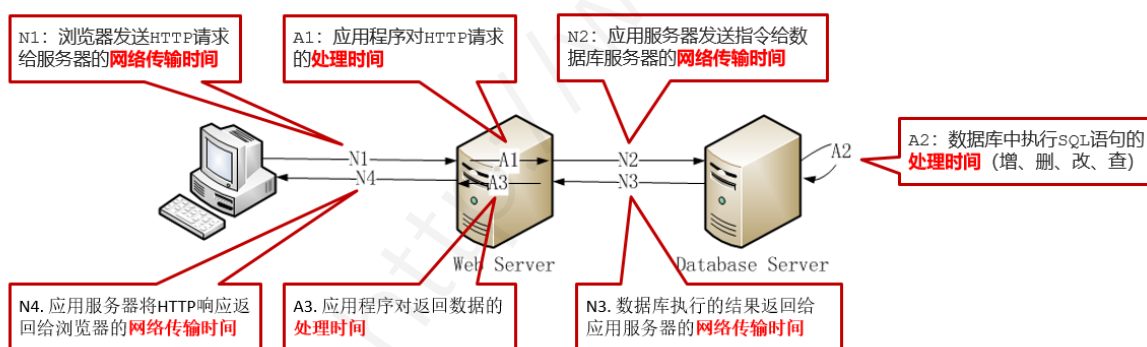
- 负载测试：主要目的是测试高负载情况下，对系统资源的消耗，是否会耗尽的问题（双11活动）
- 并发测试：主要目的是测试极短时间内，并发请求时，系统资源争抢的问题（抢红包、秒杀）

## 性能测试的指标

### 响应时间：

-指从客户端发起请求开始，到客户端接收到结果的总时间

-包括：服务器处理时间 + 网络传输时间



### 并发用户数：

-某一时刻同时向服务器发送请求的用户数

淘宝系统案例 —— 哪个是并发数？

场景1：淘宝商城，注册用户数有5亿。

系统用户数（系统注册的总用户数）  
—— 数据库用户表的数据条数，对性能无影响

场景2：当前登录了淘宝商城的用户数为2000万。

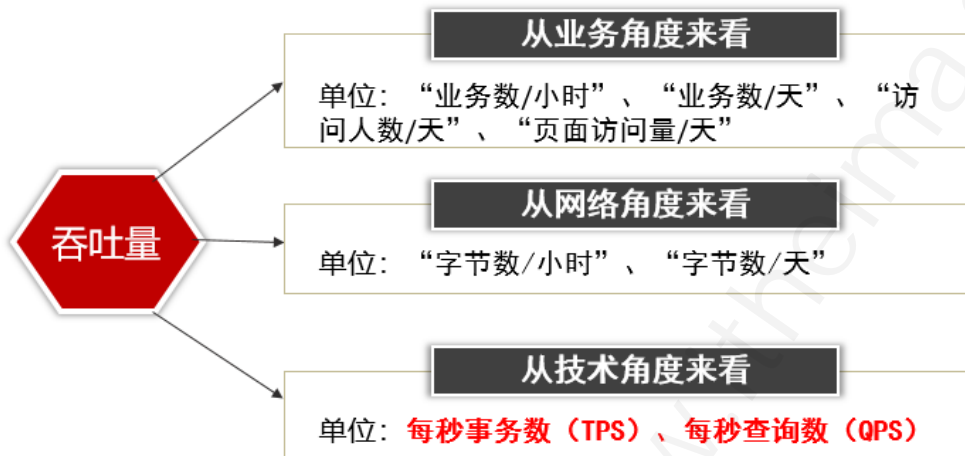
在线用户数（某段时间内访问过系统的用户）  
—— 这些不一定同时向系统在提交请求

场景3：目前正在刷淘宝的用户数有500万。

并发用户数（某段时间内同时向系统提交请求）  
—— 请求就会产生负载

## 吞吐量:

吞吐量 (Throughput) : 指的是**单位时间内**处理的客户端**请求数量**, 直接体现软件系统的性能承载能力。



## QPS:

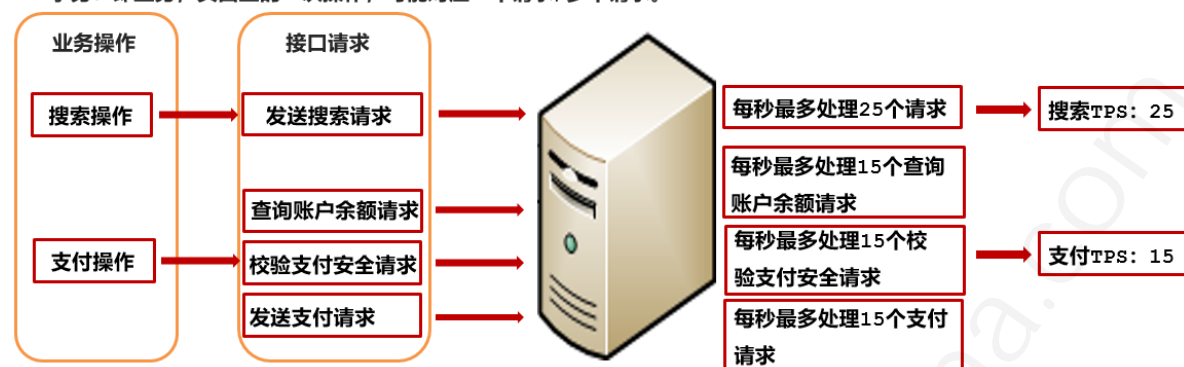
QPS (Query Per Second) 每秒查询数: 即控制服务器**每秒**处理的指定**请求**的数量



## TPS:

TPS (Transactions Per Second) 每秒事务数：即控制服务器**每秒**处理的**事务请求**的数量

事务：即业务，页面上的一次操作，可能对应一个请求/多个请求。



## 点击数：

-所有的**页面元素**（如：图片、链接、框架等）的**请求总数\*\*量\*\***

-注意：

-点击数是请求数，不是页面上的一次点击

## 错误率：

-指系统在**负载情况**下，失败业务的概率

-注意：

-错误率是性能指标，是高负载下的失败业务的概率

-随机bug是功能bug，先解决随机bug才能进行性能测试

## 资源利用率：

### （1）什么是资源利用率？

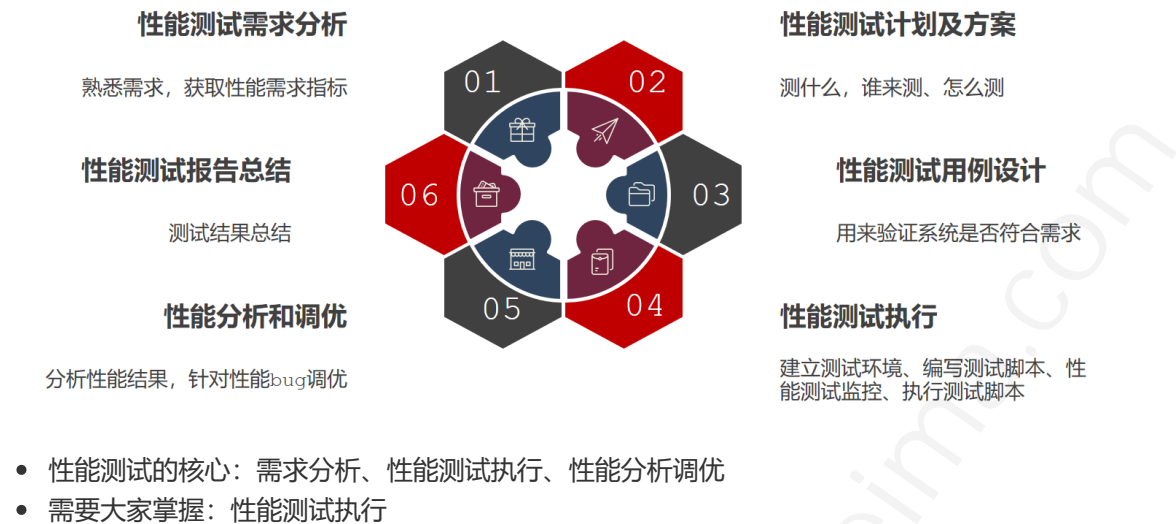
- 系统各种资源的使用情况， **“资源的使用量/总的资源可用量×100%”**

### （2）常见资源指标有哪些？

- CPU使用率：不高于75%-85%
- 内存（大小）使用率：不高于80%
- 磁盘IO（速率）：不高于90%
- 网络（速率）：不高于80%



# 性能测试的流程：



## 需求分析：



## 性能测试计划：



## 01 测什么

- 项目背景
- 测试目的
- 测试范围

## 02 谁来测

- 进度与分工
- 交付清单

## 03 怎么测

- 测试策略

性能测试用例：

用例名称	获取首页数据 标题			
用例编号	Index002 编号			
用例描述	TPS达到100的情况下，进入首页的时间不超过5s 描述			
前置条件	准备10万商品数据 预置条件			
用例步骤	动作	期望的性能		
1	进入商城首页	<5s	步骤和预期结果	
2				实际结果
并发用户数与事务响应				
并发用户数	事务平均响应时间	事务最大响应时间	平均每秒处理事务数 (TPS)	事务成功率
5				
10				
30			系统事务的性能指标	
50				
100				
并发用户数与服务器性能				
并发用户数	CPU利用率	内存利用率	磁盘IO情况	其他参数
5				
10				
30			应用服务器的各项资源使用情况	
50				
100				
并发用户数与数据库性能				
并发用户数	CPU利用率	内存利用率	磁盘IO情况	其他参数
5				
10				
30			数据库服务器的各项资源使用情况	
50				
100				

## 性能测试执行：

<b>建立测试环境</b>	<ul style="list-style-type: none"> <li>- 搭建性能测试环境，包括硬件环境、软件环境、网络环境</li> <li>- 提示：一般情况下可以要求运维和开发工程师协助完成</li> </ul>
<b>编写测试脚本</b>	<ul style="list-style-type: none"> <li>- 按照性能测试用例的需要，使用性能测试工具进行编写测试脚本</li> <li>- 提示：脚本可以自己编写，也可以使用工具来录制</li> </ul>
<b>性能测试监控</b>	<ul style="list-style-type: none"> <li>- 在脚本执行前，配置各项性能的监控指标。</li> <li>- 如：响应时间、TPS、错误率、资源使用率（CPU、内存、磁盘等）</li> </ul>
<b>执行测试脚本</b>	<ul style="list-style-type: none"> <li>- 设置性能运行场景，执行性能测试，并同步收集各项性能指标</li> <li>- 提示：执行性能测试脚本前，保证脚本都调试通过</li> </ul>

## 性能测试分析和调优：

说明：性能测试分析人员经过对结果的分析以后，如果不符合性能需求，则会提出性能bug，然后由开发人员进行后续的调优。

## 性能测试报告:

- 测试工作的经过回顾
- 缺陷分析和调优
- 风险评估
- 性能测试结果
- 测试工作总结与改进