软件性能测试

**课程适合人群：**

软件性能测试工程师

**本课程重点解答&解决：**

* 介绍性能测试的基本概念
* 介绍高级性能测试技术
* 介绍性能测试工具LoadRunner
* 介绍性能测试工具JMeter
* 介绍操作系统、JMV、数据库等层面的监控
* 简单介绍性能的优化

**本课程亮点+核心价值介绍：**

* 性能测试的高级技术
* 从各个层面分析性能测试
* 详细介绍JMeter的使用
* 详细介绍操作系统、JMV、数据库等层面的监控

**教学时间**

4天

**教学内容**

软件性能测试知识

软件性能测试基本知识

身边的性能

由于性能测试不足带来的Bug

什么是性能

性能在ISO25010 2016软件质量模型中的位置

性能在敏捷测试的位置

软件性能问题的定位

衡量软件性能的四个维度

参加性能需求分析的干系人

性能测试工具

性能测试的作用

软件性能测试类型及测试方法

性能测试类型

普通性能测试

前端性能测试

负载测试（Load Testing）

并发测试（ Concurrency Testing）

容量测试（Volume Testing）

负载测试寻找拐点的方法

 逐步递增法

 二分法

 黄金分割法

 疲劳性测试（Stress Testing）

 强度测试（Strength Testing）

 配置测试（Configuration Testing）

 基准测试（Benchmark Testing）

软件性能测试指标

性能指标

响应时间（Response Time）

前端响应时间

吞吐率

资源利用率

思考时间（Think Time）

在线用户数与并发用户数

性能计数器

理发师模型

响应时间、吞吐率、资源利用率关系

软件性能测试需求分析

分析用户行为

性能测试的难点

需求层面

技术层面

解决思路

需求分析典型案例

需求的来源

性能测试通过标准

性能测试用例设计及分析

性能测试场景设计完整过程

测试环境、工具、数据准备

测试脚本录制、编写与调试

场景制定

测试执行

结果分析与定位问题

测试报告与测试评估

性能测试方案

构建性能测试方案

软件结构演进

性能测试数据的创建方法

性能测试数据的创建时机

线下性能测试环境

性能测试场景设计

高级性能测试

全链路压测

蓝绿联盟

金丝雀发布

数据回放技术

DevperfOps

HTTP协议及其抓包工具使用

HTTP消息包

请求包

响应包

HTTP协议特性

Fiddler4使用

LoadRunner

虚拟用户

Vuser Script录制流程

认识VuGen的录制原理

录制选项设置

Run-time Setting

参数化

参数化目的

参数类型（Parmeter type）

参数列表的配置

参数取值方式

如何打印参数化变量

参数化步骤

事务、集合点和思考时间

响应时间

响应时间组成

添加事务

事务时间组成

检查点

web\_find()函数

web\_reg\_find()函数

界面操作

图片检查点

集合点

 集合点策略

关联

为什么要设置关联

录制中建立关联

用户自定义关联

录制后建立关联

手动建立关联

场景

场景用途

设计场景流程

启动场景流程

两种场景模式

设计场景

场景控制

查看场景

IP欺骗

为什么要使用IP欺骗

IP欺骗技术设置

查看设置是否正确

负载均衡

负载均衡原理

主机器配置

副机器配置

Controller中配置

场景运行时间设置

为每个场景设置运行时间

为所有场景设置一个运行时间

测试报告分析

基本分析方法

默认报告

总体报告

运行的虚拟用户变化

每秒点击率

 吞吐量

 交易概述

平均交易反馈时间

生成标准各种格式的报告

分析图合并

 设置

 叠加

 平铺

关联

 分析图关联技术

 分析图钻取技术

JMeter

安装与环境变量配置

JMeter 简介

录制

 BadBoy录制

 用JMeter录制

JMeter基本元件介绍

测试计划

线程组

HTTP Cookie 管理

HTTP头信息管理

HTTP Request

查看结果树

Debug Sample

 参数化方法

CSV参数化

函数助手

JDBC参数化

断言

响应断言

BeanShell断言

Size Assertion断言

断言持续时间

 其他元件

正则表达式提取器

边界提取器

设置Cookies

网速模拟

Throughput Controller

事务控制器

报告

Summary Report

Aggregate Graph

Response Time Graph

时间响应图

非GUI下的运行方式

基本命令行运行

通过Python Taurus运行

通过Java ant运行

监控

JMeter集群+Influxdb存储+Grafana实时数据

Influxdb安装及使用

Grafana安装及使用

性能监控与分析

资源信息分析图

性能监控

 Exporter + Prometheus + Grafana进行监控

 Exporter安装及使用

 Prometheus安装及使用

 Grafana使用

 对MYSQL的监控

 全链路监控

微服务架构

全链路监控选择依据

APM

Skywalking安装及使用

Nmon性能监控

 用于实时监控

 用于测试过程中监控

 Windows性能监控

 监控搭建

 启动监控

 监控报告

 Tomcat监控工具Probe

线程池

正在运行的线程

内存监控

MySQL监控工具Monyog

性能分析

性能瓶颈分析方法

基于Windows的性能分析

Process

Memory

DISK

Processor

网络

基于Linux的性能分析(简约版)

 CPU

 top

 top -Hp 进程号

 jstack

useful-scripts

Memory

 free -m

Disk

 iostat -x

iotop

 Network

netstat

 基于Linux的性能分析(详细版)

Linux架构

 CPU

 常用命令-uptime

 常用命令-top

 CPU使用率

 平均负载

 不可中断的睡眠态进程

 僵尸进程

 CPU状态转换

 中断

 节拍率

 上下文切换

 显示10个消耗CPU最多的进程

 常用命令-perf top、 perf record

 常用命令- execsnoop

 在多CPU的系统里，可以查看所有CPU的信息

 内存

 内核空间、用户空间

 内核映射

 虚拟内存空间分布

 内存的分配与回收

 查看剩余内存

 cache和buffer

 获得系统内存摘要信息

 详细显示某个进程所用的内存情况

 文件和磁盘

 文件与磁盘概念

 虚拟文件系统 VFS（Virtual File System）

 文件系统分类

 通用块

 Linux存储堆栈图

 性能指标

 常用命令- /proc/slabinfo

 查看所有目录项和各种文件系统索引节点的缓存情况

 查看I/O状态

 网络

 基本概念

 TCP 协议通信的网络包

 Linux 网络栈

 接收、发送流程

 性能指标

 常用命令-ifcon

 常用命令-ip

 网络吞吐和 PPS

 连通性和延时

 网络性能测试

 分析工具

JVM监控与调优

JVM调优与监控

 JVM的六种状态

 JVM结构

 垃圾回收机制

 为什么要区分新生代和老年代？

 如何回收

 垃圾回收算法

 垃圾收集器类别

 通用的GC算法百分比

 垃圾收集器组合

 命令

Tomcat调优与监控

Tomcat计数器

 MYSQL监控与调优

 一次 SQL 的查询过程

影响性能的关键点

MYSQL计数器

最大连接数

实际连接数

innodb\_buffer\_pool\_size

慢SQL定位技术

SQL语句的执行计划

show full processlist

索引覆盖

联合索引

索引失效的场景

性能调优

前端优化

CPU调优

性能调优方法

性能分析流程

系统性能关注点

程序优化

配置优化

线程优化

DB优化

业务流程优化

结构优化

实战

负载测试

疲劳性测试