

22.07.01

Java and WebLogic Performance tuning

Duration: 5 days

This performance tuning course guides the student through a process of improving the various performance quality of services (general performance, throughput, latency, scalability etc)

Who can benefit?

Students should be familiar with Java, Java EE and general knowledge of Oracle WebLogic

Workshop overview

This is an overview of the course. For a detailed description with objectives per topic, we refer to the detailed workshop overview, listed further below in this document.

Performance defined

Introduction
QoS in a process

Technical Architecture in relation to QoS

Technical Architecture/Design and performance
Architecture
Technologies

Testing and profiling

Testing requirements during development
The Java Virtual Machine
Coding Java Performance

Oracle WebLogic and Performance

Overview
Runtime execution
QoS Lab environment

Detailed overview

Performance defined

Introduction

- Define performance
- List different QoS (Quality of Service) requirements that influence performance
- Preparing: Quality of Service requirements in a Software Development process
- Process of requirements Modelling
- Process of requirements Management
- Requirements of well defined Quality of Service requirements (e.g, SMARTT)
- Using PLanguage for specifying QoS requirements
- Managing conflicting requirements (you can't have one's cake and eat it too)

QoS in a process

- Measuring performance
- Setting up a test environment
- Benchmarking process
- Establishing a baseline

Technical Architecture in relation to QoS

Technical Architecture/Design and performance

- Deciding on technologies
- Architectural areas where most likely performance issues will appear
- Architectural areas their performance considerations

Architecture

- Planning for meeting QoS requirements
- Addressing scalability and availability
- Architectural topologies
- Clustering concepts
- Load Balancing
- Fail over and replication
- Performance issues in distributed environments
- Role of the Application Server
- The C10K problem
- Using vitalisation to address these kind of requirements

Technologies

- Networking technologies (TCP, UDP, MultiCasts, etc)
- Distributed technologies (EJB/CORBA/IIOP,JMS, Web Services, REST)
- Web technologies (HTTP, Ajax, JavaScript)
- Supporting technologies (JSON, XML, XSD, XSL etc)
- Persistence (database, JPA etc)
- The database itself
- Other technologies
- Other frameworks and services
- The Operation System

Testing and profiling

Testing requirements during development

- Test-driven development and Continues integration
- Testing frameworks
- Position of resting QoS requirements in this process
- Designing tests
- Performance and load test environments (grinder, jmeter))
- Profiling as a process
- Profiling tools and techniques



The Java Virtual Machine

- Understand the virtual machine
- Understand the garbage collector and its role on performance
- Understand the heap generations (eden space, tenured and perm)
- different VMs (Sun, JRockit etc)
- Real-time VMs and deterministic performance
- Monitoring heap sizes (JMX, Consoles, VisualVM)

Coding Java Performance

- Performance tips and tricks
- Caching in Java
- Common programming errors
- List of coding techniques

Oracle WebLogic and Performance

Overview

- Monitoring performance with WebLogic
- JMX integration with other tools (e.g, Nagios)
- WebLogic facilities for performance
- WebLogic clusters
- WebLogic QoS capabilities
- Things to look out for in the WebLogic

Runtime execution

- Setting QoS alerts in WebLogic
- Defining QoS rules in WebLogic
- Capacity sharing on Weblogic
- Workmanagers and Thread pools

QoS Lab environment

- Setting up a load and performance test environment
- Choosing infrastructure
- Planning and execution
- Analyse results

