Quick Notes

Table of Contents

[1 Java Performance Tuning 2](#_Toc17235035)

[1.1 Java Internal 2](#_Toc17235036)

[1.1.1 Code Compilation and JIT 2](#_Toc17235037)

[1.2 Java Profiling 3](#_Toc17235038)

[1.2.1 Sampling 3](#_Toc17235039)

[1.2.2 Instrumentation 3](#_Toc17235040)

[1.3 Efficient – Collections 3](#_Toc17235041)

[1.4 Performance Analysis Tools 3](#_Toc17235042)

[1.4.1 JProfiler 3](#_Toc17235043)

[1.4.2 Eclipse MAT 3](#_Toc17235044)

[1.4.3 JVisualVM 3](#_Toc17235045)

[1.5 Best Practices 3](#_Toc17235046)

[2 SOAP Based Web Service 3](#_Toc17235047)

[2.1 Building web service 3](#_Toc17235048)

[2.2 Building web service Client 3](#_Toc17235049)

[2.3 Securing web service 3](#_Toc17235050)

[2.4 Transaction Management 3](#_Toc17235051)

[2.5 Session Management 3](#_Toc17235052)

[2.6 Web Service Testing 3](#_Toc17235053)

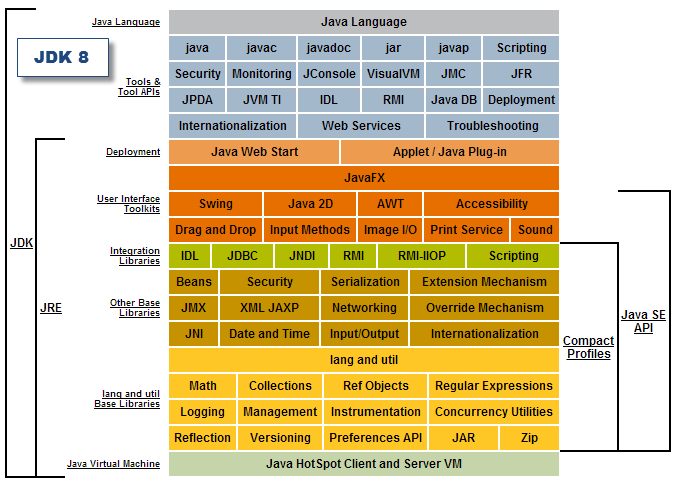
[2.7 Performance Testing 3](#_Toc17235054)

[2.8 Load/Stress Testing 3](#_Toc17235055)

[2.9 Development Strategy 3](#_Toc17235056)

# Java Performance Tuning

## Java Internal

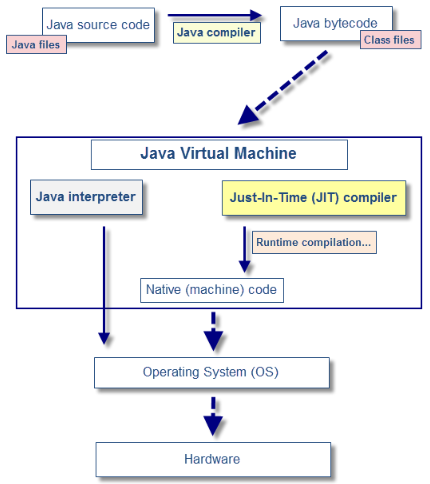


### Code Compilation and JIT

Java byte code interpretation is clearly not as fast as native code executed directly from the host. In order to improve performance, the Hotspot JVM looks for the busiest areas of byte code and compiles these into native, more efficient, machine code (adaptive optimization). Such native code is then stored in the code cache in non-heap memory.

Note: most JVM implementations offer ways to disable the JIT compiler (Djava.compiler=NONE). You should only consider disabling such crucial optimization in the event of unexpected JIT problems such as JVM crashes.

The following diagram illustrates the Java source code, just-in-time compilation processes and life cycle.



## Java Profiling

### Sampling

### Instrumentation

## Efficient – Collections

## Performance Analysis Tools

### JProfiler

### Eclipse MAT

### JVisualVM

### JConsole

## Best Practices

# SOAP Based Web Service

## Building web service

## Building web service Client

## Securing web service

## Transaction Management

## Session Management

## Web Service Testing

## Performance Testing

## Load/Stress Testing

## Development Strategy