

Tomcat – Administration and Architecture

Course code: JTOMCAT

This course will guide you through the architecture of Apache Tomcat, the most popular Java server of all time (according to recent surveys, up to 60% of applications run on top of Tomcat). The course focuses also on Java servlet/web applications and their deployment and monitoring, especially Spring Framework and/or JPA/Hibernate based. Basic as well as advanced Java EE developers and administrators will gain knowledge of many shadowy facts and will be able to utilize Tomcat components more efficiently than what would normally be the default set up.

Who is the course for

The course is targeted at Apache Tomcat administrators and Java EE developers who use Tomcat for application development and/or deployment

What we teach you

- Requirements and internals of web applications (especially Spring Framework based)
- Tomcat architecture
- Detailed Tomcat configuration and tuning
- Logging, roll-over, maintenance
- Apache HTTPD integration
- Load balancing and clustering

Required skills

Mid-level Java web programming or common Tomcat use.

Course Outline

Introduction

- Overview of Java SE, EE platforms
- History, JServ, Tomcat 3, 4, 5, 5.5, 6, 7
- Overview of the features and functionality provided by Tomcat

Examining the Tomcat installation directories

- bin
- conf
- lib
- logs
- temp
- webapps
- work

Java EE application internals

- WAR file structure
- web.xml
- JSP
- Servlet
- URL mapping rules
- HttpSession
- AJAX, Asynchronous servlets and Comet
- JavaServer Faces a FacesServlet
- Flash Scope
- Remote debugging (jdwp)
- Dynamic JSP reloading
- Dynamic class reloading

GOPAS Praha

Kodářská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno

Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava

Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved

Tomcat – Administration and Architecture

- JRebel - reloading classes without re-deploy

Configuring Tomcat

- TOMCAT/conf/context.xml
- TOMCAT/conf/web.xml
- CATALINA_HOME and CATALINA_BASE
- Default Servlet
- Static file (applies
-)
- Welcome file (uses
-)
- 404 error page
- Folder listing
- JSP Servlet

JNDI and DataSources

- Classic JDBC approach
- Better approach: JNDI resources
- JNDI ObjectFactory
- Connection pooling
- Apache DBCP
- Tomcat 7 JDBC Pool
- C3P0
- MySQL DataSource Pooling
- Oracle Thin Driver Pooling

Tomcat architecture

- server.xml (detailed walkthrough)
- Shared web hosting for multiple domains
- Connectors
- BIO (HTTP/1.1), NIO, APR, AJP
- timeouts
- keep-alive
- Valve for disabling web browser cache
- SSL
- JSSE and Keytool
- OpenSSL and APR Connector

ClassLoading

- Java SE ClassLoader
- Tomcat ClassLoader hierarchy
- Order of class loading
- Priority order for duplicate classes
- Endorsed dir

Logging

- Configuring Tomcat logging (JULI)
- catalina.out
- Configuring application logging
- Log4J
- java.util.logging
- LogBack

GOPAS Praha

Kodaňská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno

Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava

Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved

Tomcat – Administration and Architecture

- Commons Logging
- SLF4J
- Log rotation, Formatters, Appenders

Security

- File system security
- Java security manager
- Web application authentication (BASIC, FORM)
- Form login screen, `j_security_check`
- Realms
- Using the `UserDatabaseRealm`
- Using the `JDBC` realm
- Writing your own Realm
- JAAS - Java Authentication and Authorization Service
- Windows Integrated Authentication in Windows Domain (DC)

Memory management

- Understanding Java garbage collection
- Using `JAVA_OPTS`, `JMX` and `JConsole` to monitor and tune Tomcat memory usage
- Sizing Tomcat's JVM memory heap
- Persistent sessions
- Using `JMX` and `JConsole` to configure Tomcat via Tomcat's MBeans
- Hyperic monitoring tool (optional)

Performance tuning strategies

- Additional JVM tuning tips
- Enabling parallel garbage collection
- Building native connectors
- Disabling/removing unneeded applications
- Tuning incoming connections and database connection pools
- Turning off Jasper development mode
- Precompiling JSPs
- Preloading servlets

Apache HTTPD server integration with Tomcat

- Introducing the `mod_jk` module
- Introducing the `mod_proxy` module
- Introducing JBoss `mod_cluster` module
- Understanding Tomcat connectors
- Understanding Tomcat workers
- Connecting Tomcat with Apache HTTPD
- Sticky session

Apache HTTPD and OpenSSL

- Setting up certificates
- Configuring SSL
- Tomcat load balancing with Apache

Tomcat clustering

- Clustering model
- Load balancing
- Session sharing
- Working with Tomcat clustering

GOPAS Praha

Kodářská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno

Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava

Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved

Tomcat – Administration and Architecture

- Session management
- Using
 - to replicate sessions across Tomcat instances
- Configuring a cluster
- Testing the cluster

GOPAS Praha
Kodaňská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno
Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava
Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved