CALL US ON +44 (0)207 107 2620 OR EMAIL INFO@SKILLSMATTER.COM

Core Terracotta: Scaling your Java Application with Terracotta (TERRACOTTA-01-(

CODE: TERRACOTTA-01-02 DURATION: 2 DAY/S

Learn how to use Terracotta's JVM-level clustering technology with your application. Leverage Terracotta to make your Java Applications highly available and scalable, with minimal effort.

We'll discuss usage of Terracotta technology in System Implementations. We'll highlight real-world use-cases where Terracotta was employed and as compared to alternatives, provided better Scale and HA with semantic correctness, whilst simultaneously providing low time-to-market and in-depth visibility into the Java Cluster. We will cover the following use-cases: Http sessions, Distributed Caches, clustering POJOs (e.g. Publisher-Consumer, Batch Processing etc.), Spring singleton-Beans/Context Events, JVM Co-ordination and clustering other development frameworks.

LEARN HOW TO:

- Apply Common Use Cases
- Use Session Clustering Apply POJO Clustering
- Write a TIM
- Co-ordinate Clusters
- Manage Clusters
- Configure Distributed Cache
- Tune for performance
- Roll-out for Production

PROGRAMME

DAY-1:
Introduction

- Terracotta the Company
- Terracotta Services
- Terracotta Flag-ship Technology: Distributed Shared Objects:
  - Philosophy
  - Concepts
  - Architecture Overview
- Terracotta Installation
- Terracotta Samples and Examples

Common Use-Cases – Introduction, Real-life Examples and what Terracotta provides as compared with alternatives:

- Session Clustering
- POJO Clustering
- The De-Container.
- JVM Co-ordination
- Cluster Management.
- Distributed Cache (Various topologies):
  - Cache Aside
  - Write through Cache
  - Write behind Cache
  - Partitioned Data-Grid
  - Other configurations

Session Clustering:

- Sample Application/Lab (Clustering a sample app)
- Scale, HA, Correctness, characteristics

POJO Clustering:

- Concepts and Considerations
- Sample Application/Lab (Clustering a LinkedBlockingQueue)

The De-Container:

- TIMs Concept
- Terracotta TIMs for EHCache, Hibernate 2L Cache, Wicket, Struts, Quartz, Lucene, Compass.
- Sample Applications/Labs (Quartz, Compass)
- Writing a TIM.

Cluster Co-ordination:

- Distributed Wait-Notify
- Distributed Method Invocation

NEXT SESSION

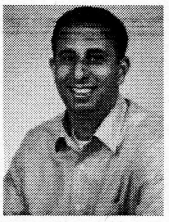
The next course will be delivered at Skills Bhasin.

COURSE DATES & REGISTRATION

Date	City	Cost	Offer
20-07-09	London	£1095.00	-
27-07-09	Paris	~~€1165.00~~	€1050.00 if book
14-09-09	Aarhus	~~kr8675.00~~	kr7795.00 if book
14-09-09	Oslo	~~11000.00~~	9900.00 if booke
21-09-09	London	~~£1095.00~~	£985.00 if booke

Book Online or Call Sales on +44 (0)207 107 2

ABOUT THE AUTHOR



Ari Zilka is the founder and CT
More about Ari Zilka

IN PARTNERSHIP WITH TERRACOTTA

TERRACOTTA

IS THIS COURSE FOR YOU?

If you are a senior JEE/J2SE developer or architect, Java Applications and solve this problem generically, test, deploy and manage Terracotta's technology with you!

COURSE PREREQUISITES

- Intermediate knowledge of Core Java
- Appreciation of non-functional concerns of a Manageability, Maintainability etc.
- Reading this book, prior to the class, might al Cluster The JVM For Spring, Hibernate And H Hibernate and POJO Scalability (Expert's Voi

COURSE LABS & EXERCISES

30 % Labs, 70% Presentation

COURSE DATES & REGISTRATION

Date	City	Cost	Offer
20-07-09	London	£1095.00	-
27-07-09	Paris	~~€1165.00~~	€1050.00 if book
14-09-09	Aarhus	~~kr8675.00~~	kr7795.00 if book
14-09-09	Oslo	~~11000.00~~	9900.00 if booke
21-09-09	London	~~£1095.00~~	£985.00 if booke
09-11-09	Aarhus	~~kr8675.00~~	kr7795.00 if book
14-12-09	London	~~£1095.00~~	£985.00 if booke

Book Online or Call Sales on +44 (0)207 107 2

- Master/Worker

Cluster Management:

- JMX Events
- Sample Application

DAY-2:
Distributed Cache & Cache Configurations

- Distributed Cache:
- Cache configurations
  - Read-Only
  - Write Through
  - Write Behind
  - Cache Aside
  - Partitioned Data Grid
- Scale, HA, Correctness, Simplicity characteristics of each configuration.
- Sample Application/Lab (Clustering a ConcurrentHashMap, EHCache)

Performance Tuning

- Memory Tuning:
  - Garbage Collection
  - Virtual Memory Manager
  - Distributed Garbage Collection
- Lock Tuning:
  - Striping
  - Lock Granularity
  - Lock Types
  - Lock Pessimism/Optimism
- Instrumentation Scope
- Other Tc-config.xml , tc.properties tuning
- Tools:
  - Admin Console (Lock Profiler, DGC Tab etc.)
  - Statistics Visualization Tab
  - Debug options (tc-config.xml)
- LAB: Improve Throughput/Latency of a poorly performing app

Production Roll-out

- Deployment Options
- Failure Analysis
- Monitoring the cluster in production
- Upgrades with minimal downtime
- Other post-production best practices

Summary & Conclusions

© Copyright 2003-2008, Skills Matter Ltd

Web development by You In Control