

Terracotta Ehcache
Performance at Any Scale:
From a Single Node to Full Data Virtualization for Cloud Deployments
Application Scale for
Cloud and Virtualized Architectures
2 DAY Course Curriculum
(Subject to change without notice)

Get the most out of Terracotta Ehcache.

Ehcache is a free, open-source caching product suite for building highly scalable, highly available java applications.

In this intensive 2-day course, you will learn how to build scalable applications using Terracotta Ehcache, the most widely deployed open source java caching framework. Through one simple API you will be able to address all of your data scale needs from single instance caches to replicated caches to distributed, coherent, highly reliable caches delivering massive scale. Learn how to take full advantage of Ehcache and all of the available functionality to deliver high application performance and scale.

Learn How to:

- ✓ Identify what data & how much data to cache
- ✓ Understand common uses for Ehcache
- ✓ Use the various Ehcache editions and leverage each to meet your HA, performance and scale needs.
- ✓ Use the Ehcache console and developer tools to maximize your productivity
- ✓ Configure, manage, and tune your caches for optimal performance, reliability, availability and scale
- ✓ Instantly boost application scale and performance through Hibernate interface
- ✓ Deploy Ehcache in enterprise architectures and virtualized architectures
- ✓ Learn about the advanced capabilities of Terracotta Ehcache and Terracotta Core Platform

Curriculum:

Introduction to Terracotta Ehcache:

- The Company
- The Products and Editions
- Services

Terracotta Ehcache:

- Why Caching
- Why Distributed Caching
- Why Terracotta Ehcache - the Caching Continuum
- Ehcache Core: Introduction
 - Ehcache Architecture & Concepts
 - Ehcache common use cases
 - Hibernate Caching Provider
 - General Object Caching
 - Web Caching
- Ehcache Standalone
 - Pros and Cons
 -

Terracotta Ehcache
Performance at Any Scale:
From a Single Node to Full Data Virtualization for Cloud Deployments
Application Scale for
Cloud and Virtualized Architectures
2 DAY Course Curriculum
(Subject to change without notice)

- Ehcache Distributed - Typical Features and how Terracotta Ehcache delivers them:
 - Ehcache DX - Replication
 - Pros and Cons
 - Use Cases
 - Ehcache EX & FX
 - Express Mode and Custom Mode
 - Pros and cons
 - Use Cases
 -
- Developer Concerns:
 - Developer API
 - Cache Correctness |Coherence:
 - Intra/Inter JVM
 - JVM to DB
 - Cache Data Liveness
 - Large Cache Sizes
 - Disk Spillover
 - Size/Memory Based Size Specifications
 - Terracotta Server Array
 - Application Partitioning
 - Eventing
 - Bulk-Loading
 - Heterogeneous Client Support
 - Cache Exception Handling
 - Cache Extensibility
 - Asynchronous Write Behind to DB
 - Class Loader Best Practices
 - Additional API's
 - JCache(JSR 107)
 - RESTful API
 - OpenJPA
 - Google Appengine
- Operational Concerns:
 - Scale
 - Latency
 - High Availability
 - Security
- Visibility and Management Tools
 - Terracotta Ehcache DX Monitor
 - Terracotta Developer Console EX |FX

Terracotta Ehcache
Performance at Any Scale:
From a Single Node to Full Data Virtualization for Cloud Deployments
Application Scale for
Cloud and Virtualized Architectures
2 DAY Course Curriculum
(Subject to change without notice)

- Ehcache Panel
- Terracotta Operations Center
- JMX Events & JMX API for Monitoring and Alerting
-
- Lab1: INSTALL Install and INTEGRATE Integrate Terracotta Ehcache in a Sample App.
 - Install Ehcache Express & Custom
 - Setup FX Server Array Striping

Terracotta Ehcache Scale/Performance Tuning

- Methodology:
 - Architectural and Deployment Choice Impact on Performance/Scale
 - Memory Tuning (Cache-Sizes /Virtual Memory, Garbage Collection, Eviction Tuning)
 - Configuration Tuning (ehcache.xml, tc-config.xml, tc.properties)

Terracotta Ehcache for Hibernate Integration (1 hour 45 min):

- Role of the ORM
- Common Performance/Scale Issues with Hibernate.
- Role of 2nd Level Cache
 - Query Cache
 - When to use it
 - Entity Cache
 - When to use it
- Terracotta as 2nd Level Cache Provider.
- Integration How-To.
 - Install Express & Custom
- Terracotta Developer Console EX |FX
 - How to use - Hibernate 2nd Level Cache Panel
- Lab2: Integrate Terracotta as 2nd Level Cache provider in a Sample App.
- Lab3: Improve Performance and Scale of a poorly performing 2nd Level Cache.

Deployment and Post Deployment Concerns:

- Deployment Options
 - Standard Enterprise & Virtualized Deployment
 - Capacity Planning
 - Provisioning guidelines
 - Considerations as per cache size (data volume, object payload)
- Failure Analysis & Recovery
- Monitoring Ehcache in production
- Upgrades with minimal downtime (Planned Downtime)
- Other post-production best practices

Summary and Conclusions:

Terracotta Ehcache
Performance at Any Scale:
From a Single Node to Full Data Virtualization for Cloud Deployments
Application Scale for
Cloud and Virtualized Architectures
2 DAY Course Curriculum
(Subject to change without notice)