Red Hat JBoss Fuse Accelerated

Description

Red Hat JBoss Fuse Rapid Track is a 5-day course that gives Java™ developers, architects, and Red Hat JBoss Fuse administrators an understanding of Red Hat JBoss Fuse, Red Hat JBoss A-MQ (Java Messaging Service broker), and Apache Camel.

Learn the skills you need to develop and implement enterprise integration patterns (EIP) using Camel, and see how to deploy EIPs in Red Hat JBoss Fuse—an enterprise service buse (ESB) built on Apache ServiceMix. In addition, learn how to use Red Hat JBoss A-MQ—a messaging platform based on Apache Camel.

Delegates will learn

- Core concepts of OSGi-based deployment and design
- How to rapidly implement, test, and debug EIPs
- Overview of advanced EIPs
- Fundamentals of message-oriented middleware (MOM), JMS concepts, and enhancements to JMS by Red Hat JBoss A-MQ
- Numerous broker topologies
- Performance-tuning techniques
- Configuration management using Fabric

Audience

Java developers and architects who need to learn how to implement EIPs using Camel, as well as how to use the messaging capabilities of Red Hat JBoss A-MQ (ActiveMQ). Integration system managers who need to configure and maintain networks of brokers and enterprise service bus (ESB) instances using features of Fuse Fabric.

Outline

Introduction

- JBoss Fuse product family

Red Hat JBoss Fuse

- Introduction
- Install and customize
- Management
- Implement of OSGi in production

Red Hat JBoss A-MQ

- Install and configurate
- Customize and integrate of network of brokers
- Increase availability and reliability
- Increase performance
- Connect using Java Message Services (JMS)

Camel

- Introduction
- Develop integration solutions
- Perform basic data transformations
- Perform advanced data transformations
- Perform basic components
- Perform network-based components
- Test applications
- Manage error handling
- Threading and transactions
- Integrate Camel and Fabric

Prerequisites

Understanding of the Spring framework and Java, including Java developments tools (e.g., Ant, Maven, and Eclipse). OSGi basic concepts.