Integration using Apache Camel and JBoss Fuse Course Contents(5 Days)

By Dr. Vishwanath Rao

Course Objectives

- Red Hat JBoss Fuse architecture.
- Understanding the basic standards and Terms associated with ESB and System Integartion.
- Understanding the various Tools available in the Market, and presentation of good and Bad practices of Integration.
- Learning integration methods and patterns of implementation in ESB.
- Acquisition of practical skills associated with JBoss Fuse / Fabric8.
- Fundamental OSGi concepts and usages.
- JBoss Fuse command line interface administration and management.
- JBoss Fuse security Practicing the Java Authentication and Authorization Service.

PRE-REQUISITES:

Knowledge of Java EE Application Server Administration
Basic Camel Knowledge
Familiarity with Tools Such as Maven

Course Curriculum

Red Hat JBoss Fuse

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)

DEVELOPMENT ENVIRONMENT
Introduction to Jboss Fuse
Fuse Ide
Installation Binaries
Apache Maven
JDK 1.6, Choice of Development Tool

DEVELOPMENT MODEL
Overview of Maven
Maven Archetypes
Java Code and Resources
Maven POM Files
Deployment Metadata
Dependency Injection Frameworks
Administrative Metadata

DEPENDENCY INJECTION FRAMEWORKS
Blueprint Or Spring?
Bean Registries
Springxml File Location
Spring Core annotation
Convert Spring Core component service
Expose Spring Component over Fuse ESB

Patterns to convert Spring components to services
Embedding Camel in a Spring application"\
Message Routing
Content-Based Routing

Filtering out unwanted messages
Wiretap sending a copy of the message elsewhere\
Multicast routing the same message to many endpoints"
Routing to Your Code

Managing routing with Camel
The endpoints
Create Java routes
Using message filter

Using multicasting

Data transformation in Camel

Introduction to data transformation
Transforming data with EIPs and Java
XML Transformation
Data format provided with Camel
Using Camel's CSV and JSON data format
Transforming with templates
Using Camel type converters

Beans and Camel

Beans invocation from Java
The Service Activator pattern
Using bean's registries
Camel's method-selection algorithm
Bean parameter binding

Error Handling

Understanding error handling
Error handlers in Camel

Using Camel's components

Overview of Camel components
File and FTP components
JMS component
CXF component for web services
MINA component for networking
JDBC and JPA components
Timer component

Using the Enterprise Integration Patterns

Aggregator
Splitter
Routing Slip
Dynamic Router
Load Balancer

Transactions in Camel

Transaction basics
The Transaction Client EIP
Configuring and using transactions

Concurrency and scalability

Using concurrency
Camel thread pool profiles
Using concurrency with EIPs
Synchronicity and threading
The concurrency client API
Using the asynchronous routing engine
Monitoring Camel
Checking health at the network level
Checking health at the JVM level
Using JMX with Camel
Verifying application activity
Managing Camel applications

CREATE A WEB SERVICES PROJECT

Build The Web Services Project
Check that the Bundle Has Started
Run the Ws Client
Deploy and Start the Ws Server
Create Project from the Command Line
Troubleshooting

Deployment

Creating Fat JAR
Docker deployment
Creating Docker compose project using 12 Factor Application Rule
Deploying Service Container to Kubernetes POD
Upgrading Component to Cloud Native support

CREATE A ROUTER PROJECT

Create Project From the Command Line
Disable the Test

Modify The Route
Build the Router Project
Add the Required Maven Dependency
Deploy and Start The Route
Test the Route with the Ws Client

Create tests for routes and error handling with Camel
Develop reliable routes by developing route tests and handling errors.

Route with Java beans
Create dynamic routes in Camel using Java beans.

Implement REST services
Enable REST support on Camel with Java REST APIs.

Deploy Camel routes
Package and deploy Camel applications for deployment with Red Hat Fuse.

Implement transactions
Provide data integrity in route processing by implementing transactions.

Implement parallel processing
Improve route processing throughput using Camel parallel processing mechanisms.

Create microservices with Red Hat Fuse
Create microservices from Camel routes.

TROUBLESHOOTING
Check the Status of a Deployed Bundle, Redeploying Bundles with Dev:Watch, Logging