+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 25/06/2025

Sign up

Apache karaf training

3 days (21 hours)

Overview

Our Apache Karaf training course will enable you to master a modular, lightweight and extensible runtime platform, ideal for deploying dynamic Java applications in a spirit of agility and resilience. You'll learn how to structure your services in OSGi bundles, manage their lifecycle in real time, and orchestrate their configuration in a secure, highly customizable environment.

You'll discover the fundamentals of Apache Karaf: its installation, interactive console, file structure and fine-tuned management of hot modules. Modularity becomes an asset, not a constraint.

You'll learn how to integrate frameworks such as Apache Camel for orchestration, CXF for REST/SOAP services, or Blueprint for dependency injection, into a clear, scalable service-oriented architecture.

The course also covers advanced aspects: monitoring via JMX/Jolokia, custom feature packaging, integration in a CI/CD chain and supervision with Prometheus or Grafana.

As with all our training courses, this one uses the latest version of Apache Karaf.

Objectives

- Install and configure an Apache Karaf instance in a development or production environment.
- Understand OSGi modular architecture and the bundle lifecycle in Karaf
- Master the deployment, management and dynamic supervision of hot Java services
- Create and integrate business services via Blueprint, Apache Camel or CXF in a Karaf container
- Configure security, access roles and logging with Pax Logging and ConfigAdmin

- Supervise and automate the deployment of Karaf bundles in a CI/CD chain using Maven and Jolokia

Target audience

- Developers
- DevOps engineers
- Technical architects

Prerequisites

- Notions of integration architecture
- Basic knowledge of Java

Apache Karaf training program

Introduction to Apache Karaf

- Definition of an OSGi containerPositioning of Apache Karaf in the Java ecosystem
- Comparison with other runtimes: Spring Boot, JBoss, etc.
- Integration platform
- Deployment of REST/SOAP APIs
- Dynamic modular environments

Installing Apache Karaf

- Prerequisites...
- Downloading Karaf
- Starting and stopping an instance
- Role of folders: bin/, etc/, deploy/, lib/, data/
- Deployment logic via deploy/
- Essential commands
- Navigation and help
- Auto-completion and session management

OSGi bundle management

- What is a bundle?
- Lifecycle
- Classloader isolation and dependency resolution
- Via console
- Via deploy/ folder
- Via XML features
- Detection of startup errors

- Resolution of missing dependencies
- Using diag and resolve

Runtime configuration and management

- File structure in etc/
- Using the ConfigAdmin service
- Hot modifications to .cfg files
- Users.properties file
- Defining roles and authorizations
- Protecting the console and JMX interfaces
- Supported formats
- Configuration via etc/org.ops4j.pax.logging.cfg
- Redirection to files, syslog, console

Features, services and integrations

- XML feature format
- Installation and uninstallation via feature:install
- Creating custom features with Maven
- Add camel-core, camel-http features, etc.
- Deploying XML or Java DSL routes
- Case study: reading files ? transforming ? logging
- Defining REST or SOAP services
- Exposure via HTTP
- Security with WS-Security
- Dependency injection with Blueprint
- Deploying blueprint.xml files
- Comparison with Spring XML

Supervision and clustering

- Local and remote JMX access
- Integration with Jolokia for HTTP/JSON exposure
- Prometheus / Grafana integration
- History of Karaf Cellar
- Clustering concepts with Hazelcast / Zookeeper
- High availability considerations

Packaging, CI/CD and testing

- Configuring karaf-maven-plugin
- Creating features from Maven projects
- Inclusion of dependencies in bundles
- Build pipeline for OSGi bundles
- Automated publishing and deployment

- Classic unit testing with JUnit
- OSGi tests with Pax Exam
- Checking service registration

Hands-on workshops

- Starting up Karaf
- Navigating the console
- Deploying a simple bundle
- Maven OSGi project
- XML feature file
- Installation in Karaf
- Deploying a simple route in Karaf
- Transformation test and log
- Creation of a JAX-RS endpoint
- Calling via Postman
- Exposure of Karaf metrics
- Import into a Prometheus dashboard

Conclusion and best practices

- Karaf's strengths
- Limits and alternatives depending on context
- Modular bundle organization
- Centralized configuration management
- Clear separation between business and technical features
- Apache Karaf documentation
- Pax Exam, Karaf Cookbook
- GitHub community repositories

Further information

Companies concerned

This training course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the forthcoming course, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical course: 60% Practical, 40% Theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire verifies the correct acquisition of skills.

Certification

A certificate will be awarded to each trainee who has completed the entire course.