

Microservice Architecture and Design Principles

Service boundaries, communication, data and operational trade-offs

FA-0792 · 2 days

Software Development · DevOps, Cloud & Infrastructure

Introduction

Examine microservice design through theory, practical examples and case studies. Compare decomposition, communication, data management and operations to make informed architecture decisions. The two-day course is a design foundation rather than a promise of complete production implementation.

Learning outcomes

- Compare microservices and monoliths and identify benefits and trade-offs.
- Apply DDD concepts to proposed service boundaries and context maps.
- Evaluate synchronous/asynchronous integration and data-consistency choices.
- Outline security, observability, testing and CI/CD strategies.
- Analyse case studies and recognise common implementation pitfalls.

Prerequisites

- Basic understanding of software architecture concepts
- Familiarity with web development and APIs
- Knowledge of distributed systems and database concepts

Training outline

1. Session 1 — Architecture foundations

- Understanding the basics of microservice architecture
- Key characteristics and benefits of microservices
- Contrasting microservices with monolithic architecture
- Microservice ecosystem and components

2. Session 2 — Decomposition and DDD

- Identifying monolithic system challenges
- Approaches to decomposing monolithic systems into microservices
- Domain-driven design (DDD) principles
- Defining service boundaries and context mapping

3. Session 3 — Communication and integration

- Communication patterns between microservices
- Synchronous vs. asynchronous communication
- Event-driven architecture and message queues
- API gateways and service discovery

4. Session 4 — Data management

- Strategies for managing data in a microservice environment
- Database per service vs. shared database
- Event sourcing and CQRS (Command Query Responsibility Segregation)
- Distributed data consistency and eventual consistency

Database ownership, CQRS and event sourcing are evaluated as design choices with operational costs; they are not mandatory for every service.

5. Session 5 — Security and authentication

- Common security challenges in microservices
- Authentication and authorization strategies
- Implementing secure communication between services

- Best practices for securing microservices

6. Session 6 — Monitoring and logging

- Importance of monitoring and logging in microservices
- Monitoring key performance indicators (KPIs) and metrics
- Distributed tracing and logging techniques
- Using monitoring tools and platforms

7. Session 7 — Testing and delivery

- Strategies for testing microservices
- Unit testing vs. integration testing
- Implementing CI/CD pipelines for microservices
- Blue-green deployments and canary releases

8. Session 8 — Case studies and review

- Real-world case studies of successful microservice implementations
- Lessons learned and best practices from industry examples
- Challenges and pitfalls to avoid
- Q&A session and closing remarks