

# Architectural Design Patterns

## Audience Course Architectural Design Patterns

The course Architectural Design Patterns is aimed at experienced developers and software architects who want to apply design patterns for the architecture of systems.

## Prerequisites Course Architectural Design Patterns

Knowledge of an object-oriented programming language like C++, C# or Java and experience with object oriented analysis and design with UML is desired.

## Realization Training Architectural Design Patterns

The course Architectural Design Patterns has a hands-on character. The theory is covered on the basis of presentation slides and is illustrated with demos of architectural patterns. There are exercises in design problems where architectural patterns can be applied. The course material is in English. The course times are from 9.30 up and to 16.30.

## Certification Architectural Design Patterns


Participants receive an official certificate Architectural Design Patterns after successful completion of the course.

## Duration: 2 days

## Price: € 1299

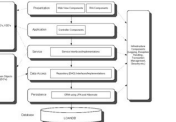
# Open Schedule

---



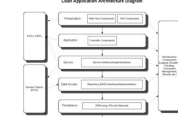
**Architectural Design Patterns**

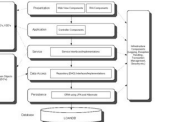
Learn Application Architectural Design




**Architectural Design Patterns**

Learn Application Architectural Design







## Content Course Architectural Design Patterns

The course Architectural Design Patterns discusses the importance and principles of architectural modeling and the application of Architectural Design Patterns. Architectural Design patterns are concerned with the large-scale organization of systems and the cooperation of their components and layers and not to the internal architecture of the individual software components.

## Intro Architectural Design

The course starts with a review of the basic principles of software architecture, software design patterns and the terminology used.

## Architectural Patterns

Subsequently attention is paid to the role of design patterns in the architecture of software systems. The main categories of architectural patterns are discussed.

## Call and Return Patterns

For example Call and Return Patterns such as the Layers pattern and the Client Server pattern are covered. The pros and cons are discussed.

## Data Flow Patterns

Data Flow patterns are treated as well. Examples are the Pipes pattern as used in the Linux operating system and the Filter pattern that is known from the Java Servlet implementation. Shared Memory patterns such as the Blackboard pattern are also on the program of the course.

## Distributed Systems Patterns

Distributed systems patterns such as the Proxy and Broker pattern are also discussed. Practical implementations of each pattern are reviewed and the pros and cons are explained.

## Concurrency Patterns

Finally advanced patterns about concurrency and threads are covered and a number of patterns that do not belong to a particular category are treated. Participants will practice with the patterns and discuss the application of the patterns to practical problems in software architecture.

## Modules Course Architectural Design Patterns

Module 1 : Software Architecture	Module 2 : Architectural Patterns	Module 3 : Call and Return Patterns
What is Architecture? Software Architecture Layered Architecture Software Characteristics Analogy to Building Role of Architect Software Architecture Elements Architecture Context Architectural Viewpoints Logical Architecture Non Functional Requirements Physical Architecture Early Load Testing	Architectural Modeling Model Characteristics Architectural Viewpoints Pattern Terminology Gang of Four Design Patterns Architectural Patterns Architectural Pattern Categories Batch Sequential Pattern Pipe and Filter Pattern Blackboard Pattern Publish and Subscribe Pattern Peer to Peer Pattern Model View Controller Pattern	Object Oriented Pattern OO Benefits and Drawbacks Object Oriented Architecture Layers Pattern Layers Problem Layers Solution Network Architecture Layers Benefits and Drawbacks Layers Pattern Variant Client Server Pattern Client Server Architecture Three Tier Pattern Three Tier Architecture
Module 4 : Data Flow Patterns	Module 5 : Shared Memory Patterns	Module 6 : Distributed Systems
Data Flow Architecture Batch Sequential Pattern Data Flow Pattern Problems Batch Sequential Pipes and Filter Pattern Pipes and Filter Forces Pipes and Filter Patterns Servlet Filters Web Service Handlers Call Chain of Handlers Benefits and Drawbacks Pipes and Filter Variants	Data Centered View Shared Repository Pattern Shared Repository Architecture Active Repository Pattern BlackBoard Pattern BlackBoard Architecture BlackBoard Context Speech Recognition BlackBoard Solution BlackBoard Variants BlackBoard Know Uses Benefits and Drawbacks	Proxy Pattern Types of Proxies Copy on Write Proxy Remote Proxy RMI Proxy Architecture Broker Pattern Broker Forces Broker Solution Bridge Component Broker Variations Benefits and Drawbacks Broker Class Diagram
Module 7 : Interactive Systems	Module 8 : Implicit Invocation	Module 9 : Concurrency Patterns
MVC Pattern MVC Architecture MVC Model MVC View MVC Controller Multiple Views Same Data Known Uses MVC Benefits and Drawbacks PAC Pattern PAC Structure PAC Solution PAC Benefits and Drawbacks	Communication Patterns RPC Pattern Publish Subscribe Pattern Queue versus Publish Subscribe Topics and Queues Data Distribution Pattern Request Reply Pattern Request Reply Correlation Multiple Replies Scalable Request Reply Guaranteed Delivery Scenario Guaranteed Delivery	Reactor Pattern Server Socket Loop Reactor Pattern Forces Event Driven Design Single Threaded Reactor Non Blocking IO Thread Pool Reactor Reactor Known Uses Reactor Benefits and Drawbacks Active Object Pattern Active Object Method Scheduling Active Object Method Dispatch
Module 10 : Other Patterns		
MicroKernel Pattern Microkernel Structure State Machine Pattern State Machine Structure Reflection Pattern Reflection Structure Process Control Pattern Process Control Structure Master and Slave Pattern Master and Slave Structure		