

## Mit Design Patterns Entwicklungszeit einsparen

**Die wichtigsten Elemente für Entwurfsmuster, mit denen Sie standardisierte Software-Strukturen einfach wiederverwerten können.**

Bei der objektorientierten Entwicklung von Software-Architektur und Software-Design sowie der objektorientierten Programmierung (OOP) einer strukturierten Software-Entwicklungsmethode stößt man immer wieder auf ähnliche Angaben und Probleme. Der Einsatz von wiederverwendbaren Entwurfsmustern (praxisbewährten Lösungsschablonen) kann dabei einen sich wiederholenden Software-Programmieraufwand verhindern.

Mit dieser Methode lassen sich Entwicklungszeit einsparen und Fehler vermeiden, die bei einer doppelten Programmierung auftreten könnten.

### **Die wichtigsten Aspekte für den Einsatz von Entwurfsmustern (Design Patterns)**

- Höherer Grad der Wiederverwendung von Software
- Dokumentation von vorhandenen und erprobten modularen Lösungsansätzen
- Reduktion der Softwareentwurfs- und Programmierzeit
- Durch Anpassung lässt sich das Einsatzgebiet der Entwurfsmuster auf das konkrete Problem adaptieren
- Verbesserung der Software-Verständlichkeit

Die Beschreibung eines Entwurfsmusters sollte folgende Elemente enthalten:

#### **Name des Entwurfsmusters**

bestehend aus ein bis zwei Begriffen, die als Handler für die Beschreibung des Problem stehen.

#### **Beschreibung des Entwurfsproblems und dessen Konzept**

Enthält die Information, wann und wo dieses Entwurfsmuster eingesetzt werden sollte (Intention/Motivation/Applikation des Design Patterns). Es erklärt die Problemstellung und dessen Kontext, wie z.B. die Beschreibung des Designproblems und eine Liste der Bedingungen, die für eine sinnfällige Anwendung erfüllt sein müssen.

#### **Lösungsbeschreibung des Entwurfsmusters**

Enthält eine Auflistung der Elemente (z.B. Klassen und Objekte) und wie deren Anordnung aussehen kann, den Entwurf, die Beziehungen zu anderen Elementen, die Verantwortlichkeiten (z.B. bei einem Architekturentwurf) und die Zusammenarbeit. Diese Beschreibung sollte einen allgemein anwendbaren Ansatz des Entwurfsmusters darstellen.

## **Ergebnisbeschreibung und Bewertung (Konsequenz/en) der Anwendung des Entwurfsmusters**

Dient der Evaluierung von Kosten und Nutzen dieses Patterns. Sie informiert, z.B. über den Speicher- und Zeitaufwand und kann für eine Abschätzung der Ressourcennutzung von CPU-Zeit vs. Speicheraufwand verwendet werden.

Entwurfsmuster lassen sich zu einer Gruppe zusammenfassen, für die diese Methode bei der Erstellung der Software-Architektur und der objektorientierten Programmierung (OOP) eingesetzt werden können. Je nach Problemstellung ist das passende Softwaredesign aus der passenden Entwurfsfamilie auswählbar.

## **Typische Entwurfsfamilien**

**Erzeugendes Muster** (Creational Design Patterns)  
befasst sich mit der Objekt-Erzeugung

**Strukturelles Muster** (Structural Design Pattern)  
beschreibt die Komposition von Klassen und Objekten.

**Verhaltensmuster** (Behavioral Design Pattern)  
beschreibt die interaktive Kommunikation zwischen Objekten, das generische/individuelle Verhalten eines Objektes und die Verteilung der Verantwortlichkeiten.

Der Gestaltungsbereich des Entwurfsmusters definiert deren Anwendung:

- Klassen oder
- Objekte
- Modul

## **Zielplattform von Entwurfsmustern**

Entwurfsmuster können auf verschiedenen Plattformen angewendet werden:

- PC-basierende Systeme
- Embedded-Systeme

Bei eingebetteten Systemen (z.B. Prozess-Steuerungssoftware, die in Mikrocontrollern mit Realzeit-Anforderungen abgearbeitet wird) ist ein wichtiger Aspekt beim Einsatz der Entwurfsmuster der Ressourcen-Verbrauch in Form von CPU-Rechenzeit und dem dazu benötigten Speicherplatz (Programm- und Datenspeicher).

## **Liste von Entwurfsmustern (Design Patterns)**

### **Erzeugungsmuster**

- Entwurfsmuster Fabrikmethode (Factory Method Design Pattern)
- Entwurfsmuster Abstrakte Fabrik (Abstract Factory Design Pattern)
- Entwurfsmuster Prototyp (Prototype Design Pattern)
- Entwurfsmuster Singleton (Singleton Design Pattern)

## Strukturmuster

- Entwurfsmuster Adapter (Adapter Design Pattern)
- Entwurfsmuster Dekorierer (Decorator Design Pattern)
- Entwurfsmuster Proxy (Proxy Design Pattern)

## Verhaltensmuster

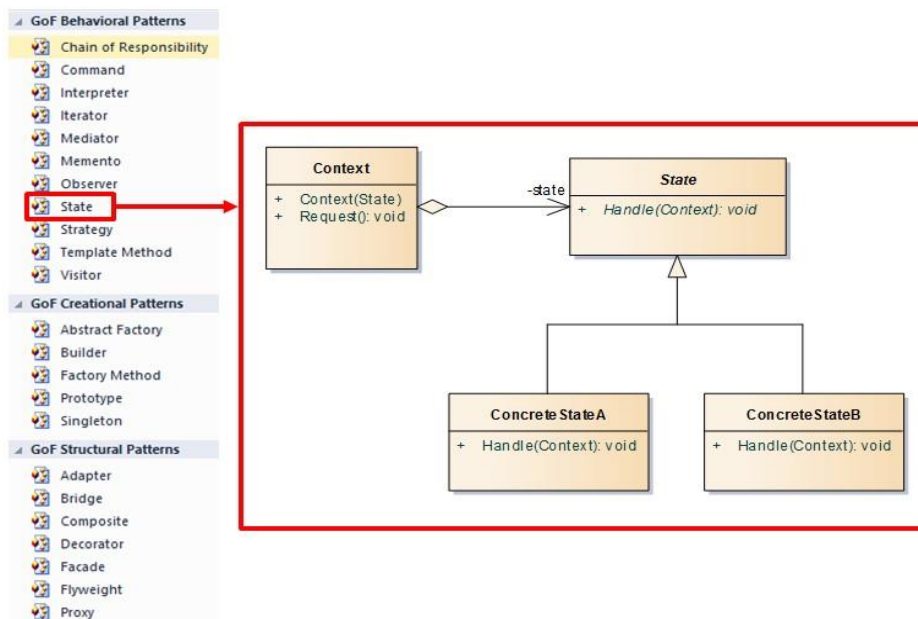
- Entwurfsmuster Beobachter (Observer Design Pattern)
- Entwurfsmuster Befehl (Command Design Pattern)
- Entwurfsmuster Zustand (State/Event Design Pattern)
- Entwurfsmuster Schablonenmethode (Template Method Design Pattern)

## Weitere Muster

- Entwurfsmuster Fassade (Facade Design Pattern)
- Entwurfsmuster Kompositum (Composite Design Pattern)
- Entwurfsmuster Memento (Memento Design Pattern)
- Entwurfsmuster Zuständigkeitskette (Chain of Responsibility Design Pattern)
- Entwurfsmuster Fliegengewicht (Flyweight Design Pattern)
- Entwurfsmuster Iterator (Iterator Design Pattern)
- Entwurfsmuster Vermittler (Mediator Design Pattern)

Viele UML- (Unified Modeling Language) Tools liefern bereits eine Auswahl bekannter Patterns mit, die direkt in der Software bzw. im Software-Modell verwendbar sind.

### MicroConsult - Experience Embedded Training and Coaching



*Bild 1: Design Patterns (Entwurfsmuster)*

## **Fazit**

Der Einsatz von wiederverwendbaren Entwurfsmustern kann zur Erstellung eines eigenen Pattern-Kataloges führen, um typische Missdeutungen zu vermeiden.

Vorteil: Das Thema Design Patterns wird im Zuge der Katalogerstellung im Team automatisch etabliert und kann seinen Vorzüge so voll ausspielen.

## **Autor: Thomas Batt**

Thomas Batt studierte nach seiner Ausbildung zum Radio- und Fernstechniker Nachrichtentechnik. Seit 1994 arbeitet er kontinuierlich in verschiedenen Branchen und Rollen im Bereich Embedded-/Realtime-Systementwicklung. 1999 wechselte Thomas Batt zur MicroConsult GmbH. Dort verantwortet er heute als zertifizierter Trainer und Coach die Themenbereiche Systems/Software Engineering für Embedded-/Realtime-Systeme sowie Entwicklungsprozess-Beratung.

**Wollen Sie sich auf den aktuellen Stand der Technik bringen?**

[MicroConsult Training & Coaching zum Thema Prozessmanagement](#)

[MicroConsult Training & Coaching zum Thema Requirements Engineering und Requirements Management](#)

[MicroConsult Fachwissen zum Thema Prozessmanagement](#)