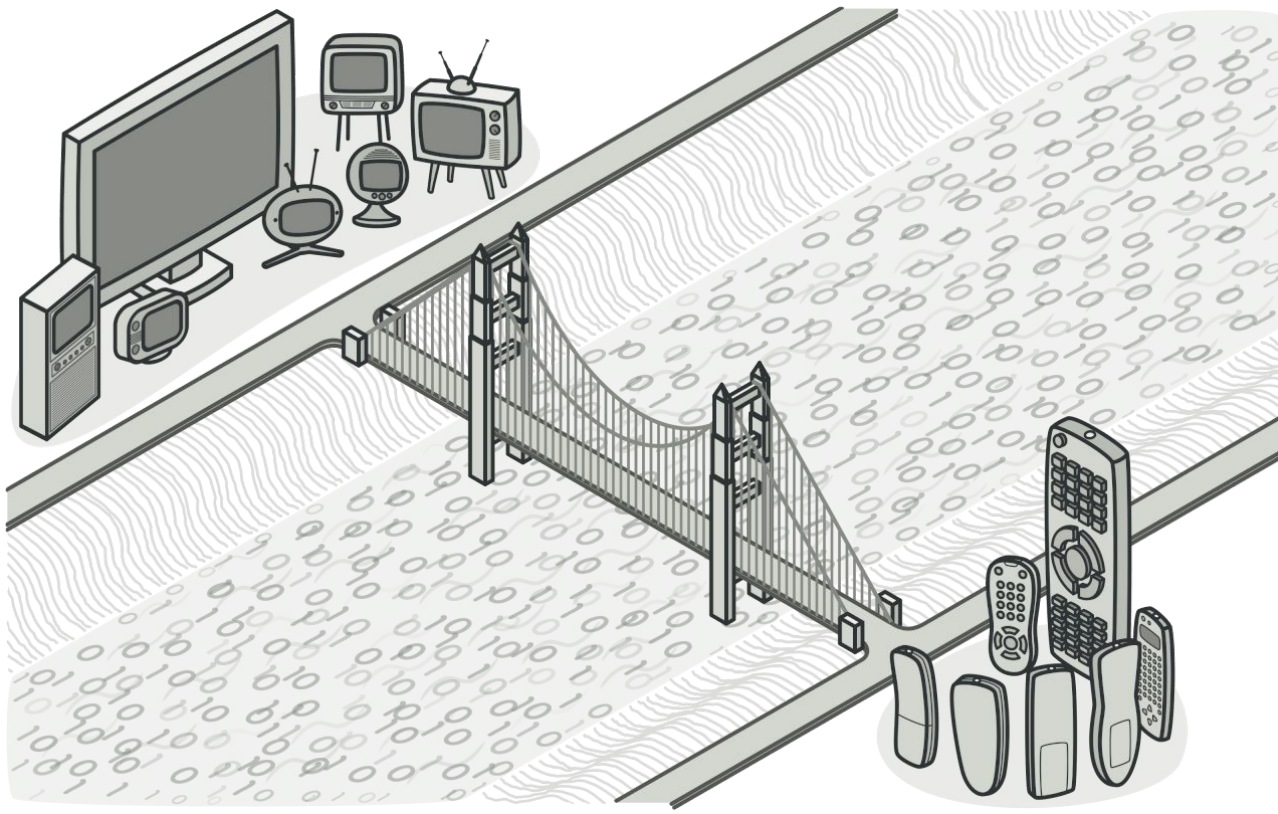




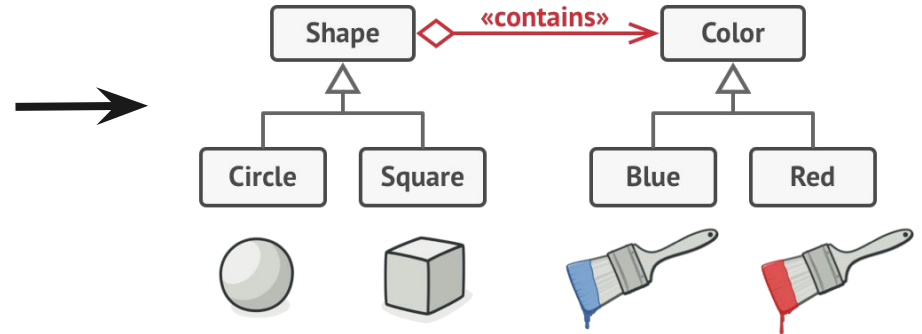
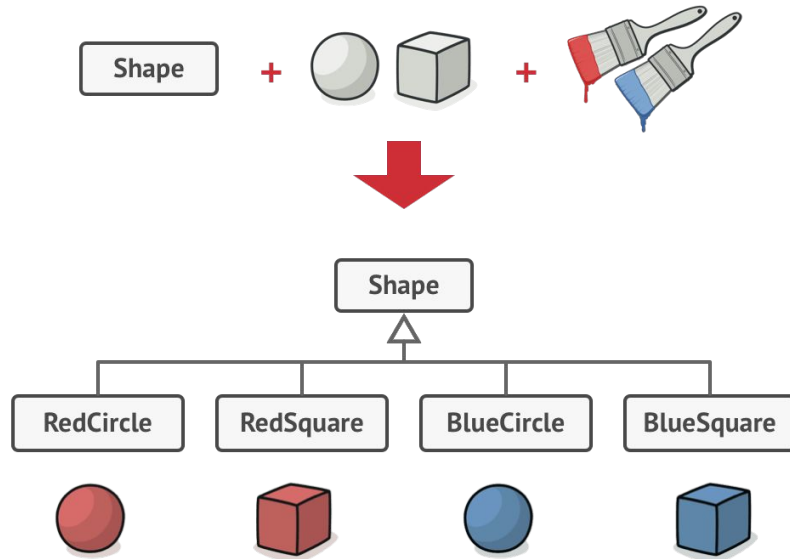
# Bridge Pattern

## Gruppe 4:

Marco Neumair, Bettina Schalthöfer, Kay Wessel, Kai Crecelius und Falk Bachmann



# Ein einfaches Beispiel

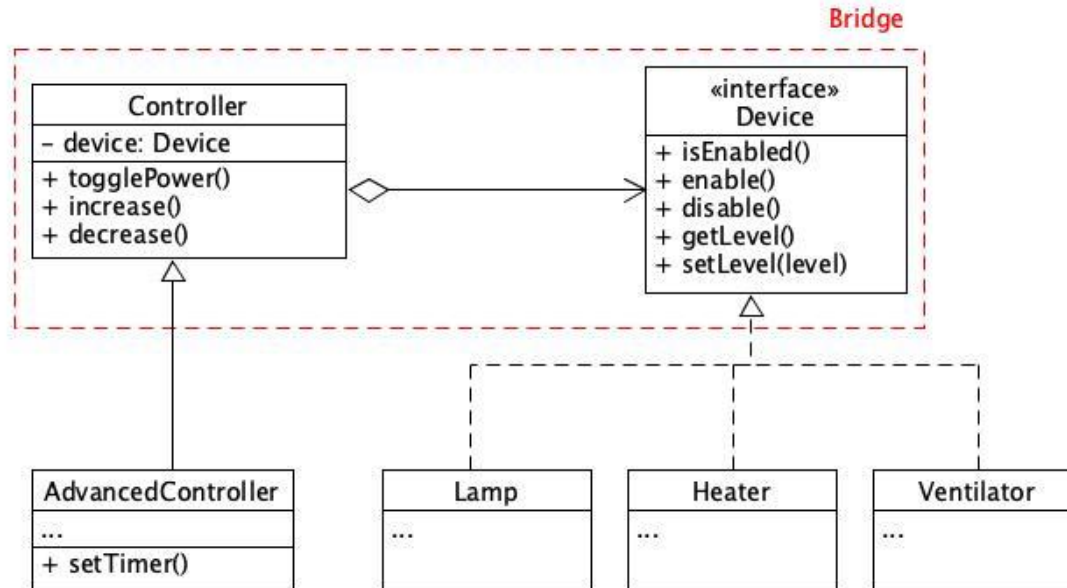




# Eigenschaften des Bridge Patterns

- gehört zu den **Strukturmustern (Structural Patterns)**
- Trennung der **Implementierung** von ihrer **Abstraktion** bzw. der **Schnittstelle**
- **Vorteile:**
  - ◆ Abstraktion und Implementierung werden **entkoppelt**
  - ◆ **zur Laufzeit** kann die Implementierung dynamisch verändert werden
  - ◆ ermöglicht **plattformunabhängige** Entwicklung
  - ◆ Client interagiert mit der Schnittstelle, während die low-level Implementierung vor ihm **verborgen** bleiben kann
- **Nachteile:**
  - ◆ Anwendung auf eine bereits stark kohärente Klasse führt zu kompliziertem Code

# UML - Beispiel



---

# Python Code Beispiel

**Vielen Dank für eure  
Aufmerksamkeit**





# Quellen

- <https://refactoring.guru/design-patterns/bridge>
- [https://de.wikipedia.org/wiki/Brücke\\_\(Entwurfsmuster\)](https://de.wikipedia.org/wiki/Brücke_(Entwurfsmuster))
- <https://www.geeksforgeeks.org/bridge-design-pattern/>