

state

design pattern **state**

Name

State / Objekte als Zustände

Klassifizierung

ein objektbasiertes Verhaltensmuster

design pattern **state**

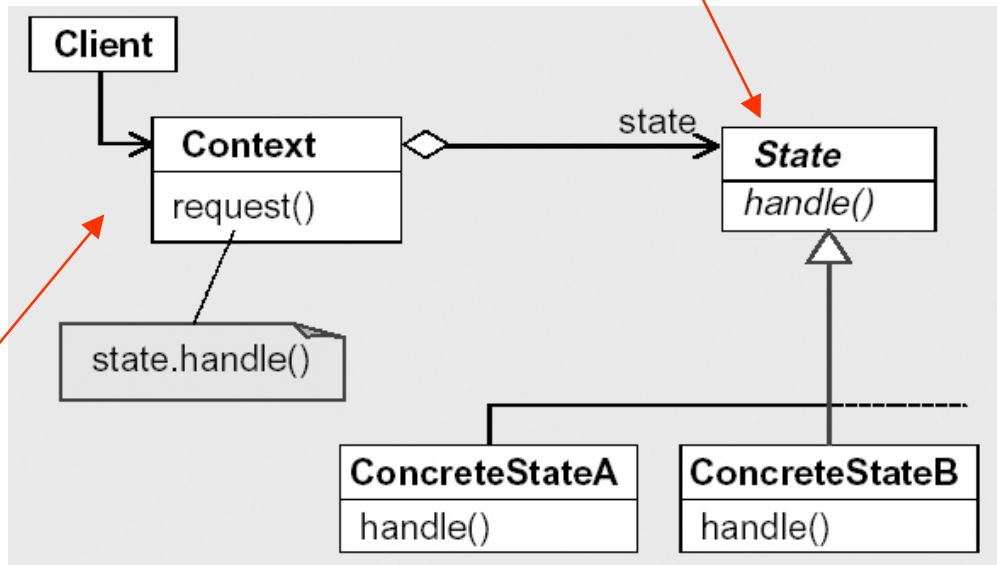
Problem

Erlaube einem Objekt, sein Verhalten zu ändern,
wenn sein interner Zustand sich ändert.

Es wird so aussehen als ob das Objekt seine Klasse
gewechselt hat.

design pattern **state**

Struktur - allgemein



Zustand

Definiert eine Schnittstelle für Klassen, die unterschiedliche operationale Zustände eines Objektes Kontext repräsentieren.

Kontext

Definiert die Schnittstelle für den Client.

Kontext verwaltet ein Exemplar einer KonkreterZustand-Unterklasse, welche den aktuellen Zustand definiert.

Konkrete Zustände

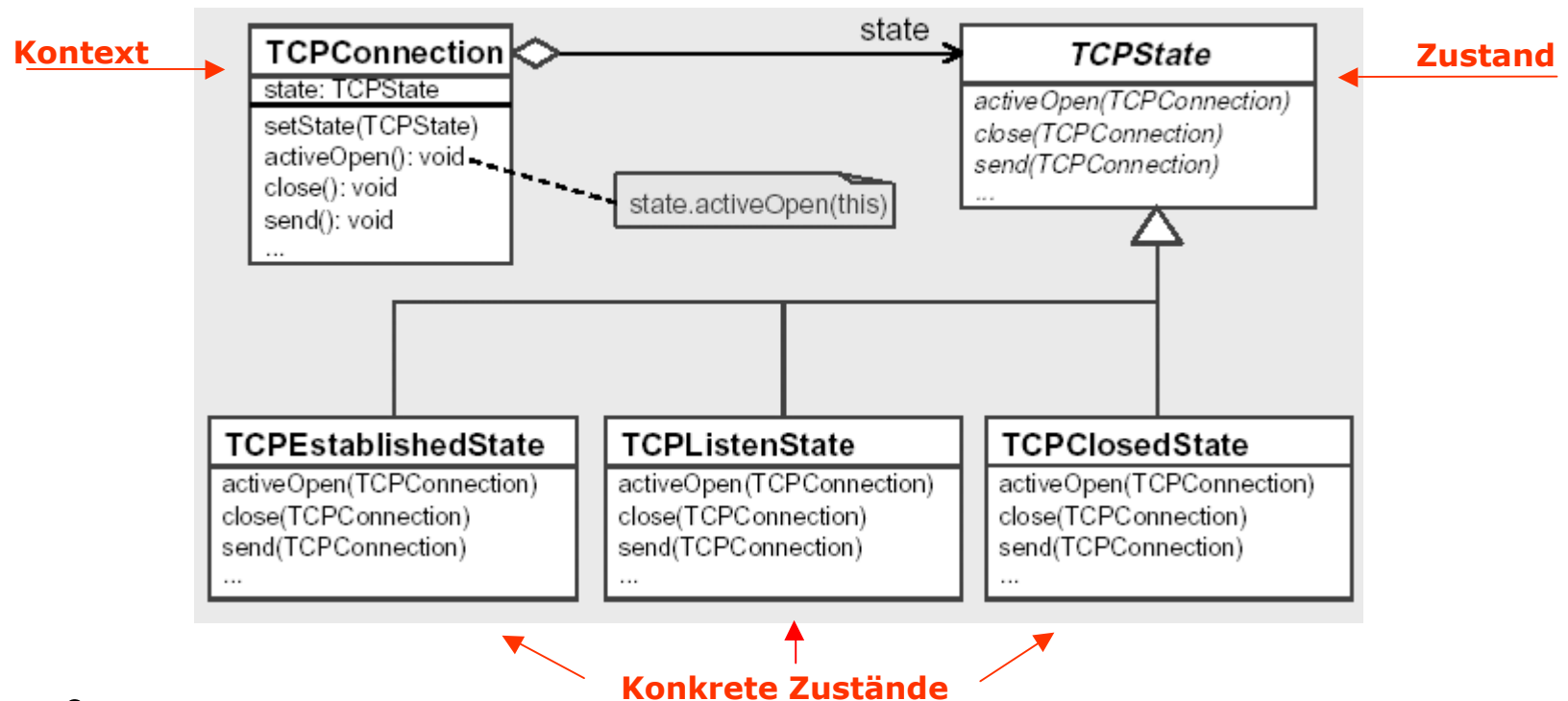
Implementierung der konkreten Zustände. Ein konkreter Zustand kapselt das mit einem bestimmten Zustand des Kontextobjekts verbundene Verhalten.

design pattern **state**

Beispiel TCP

Struktur

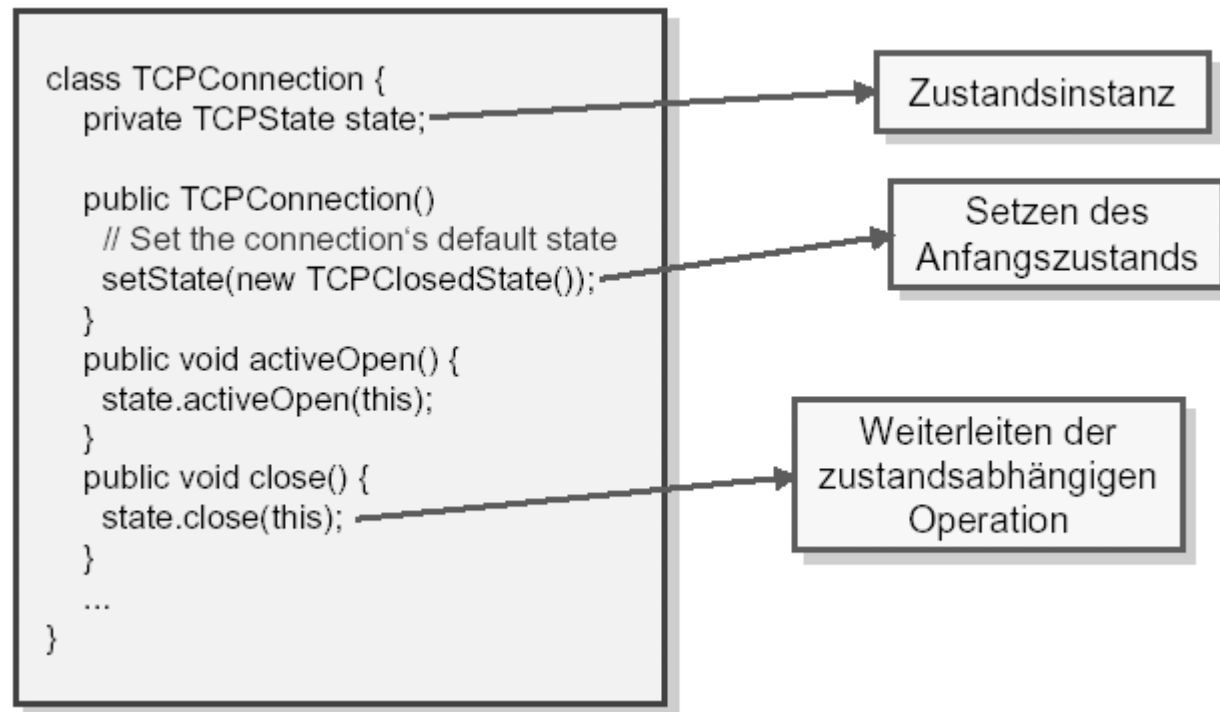
Eine Objekt einer Klasse TCP-Connection reagiert z. B. auf ein Aufruf von "open" in Abhängigkeit seines derzeitigen Zustands (z. B.: *Listen*, *Closed*).
Hierfür werden die Aufrufe an ein Statusobjekt delegiert.



design pattern **state**

Beispiel TCP

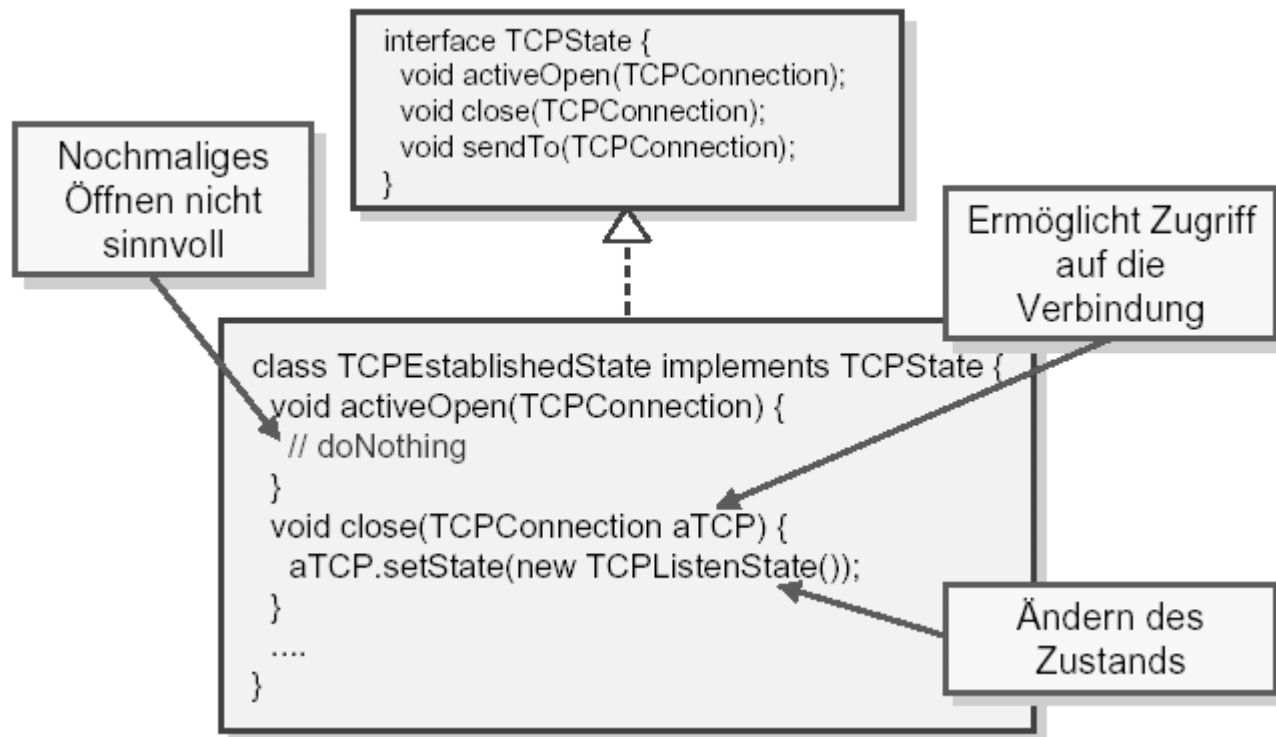
Kontext



design pattern **state**

Beispiel TCP

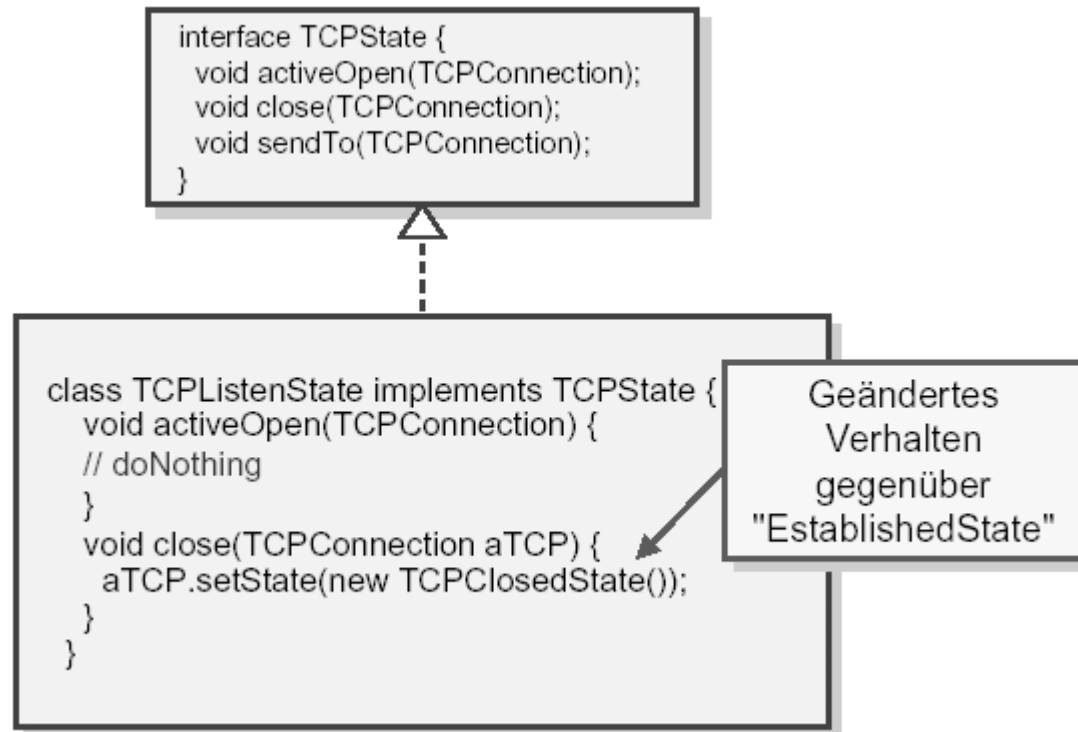
Konkreter Zustand (Established)



design pattern **state**

Beispiel TCP

Konkreter Zustand (Listen)



Konsequenzen

Neue Zustände und Übergänge können einfach als Unterklassen definiert werden. Jedoch geht die Kompaktheit verloren.

Zustände werden durch das Zustandsobjekt explizit. Inkonsistente Zustände, hervorgerufen durch falsch gesetzte (Zustands-)Variablen, werden vermieden. Zustandsübergänge sind atomar.

Die Alternative bestünde darin, den Zustand durch Aufzählungskonstanten des Kontextobjektes zu repräsentieren; mit dem Zustandsmuster lässt sich jedoch Verhalten unabhängig vom Kontextobjekt variieren.

design pattern **state**

Implementierungen

Werden Zustandsobjekte gemeinsam genutzt, handelt es sich um Fliegengewichte (Flyweights).

Zustandsobjekt selbst sind oft Singletons.

design pattern **state**

Bekannte Verwendungen

Interaktive Zeichenprogramme.

Tools die gewählt werden können, stellen die Zustandsobjekte dar.

design pattern **state**

Literatur

Entwurfsmuster

Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides

<http://genesis.iitb.fhg.de/>