

#### Softwaretechnik 1 - 3. Tutorium

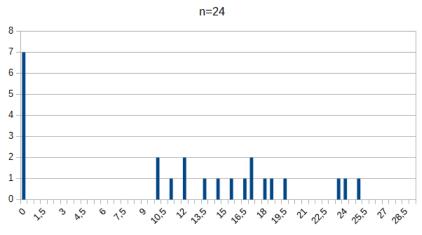
Tutorium 03

Felix Bachmann | 05.06.2018

KIT - INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION (IPD)

# 3. Übungsblatt Statistik





Ø 11,56 bzw. 16,32 von 28+1

Motivation

Orga



#### Allgemein

- verspätete Abgaben bekomme ich erst beim jeweils nächsten Tutorentreffen
  - ⇒ Rückgabe dauert länger; gibt keine Punkte, nur grobe Korrektur



### Aufgabe 1 (Plug-In-Architektur: PluginManager + JmjrstPlugin)

■ Plugins anhand Klassennamen vergleichen, nicht getName()

Orga Motivation

○○●○○○ ○○

Felix Bachmann – SWT1

Beobachter



#### Aufgabe 1 (Plug-In-Architektur: PluginManager + JmjrstPlugin)

- Plugins anhand Klassennamen vergleichen, nicht getName()
- Strings auslagern (Konstanten oder Datei)



#### Aufgabe 1 (Plug-In-Architektur: PluginManager + JmjrstPlugin)

- Plugins anhand Klassennamen vergleichen, nicht getName()
- Strings auslagern (Konstanten oder Datei)
- PluginManager gibt euch Iterable ⇒ nutzt Iterator
  - kein Casten, Kopieren in Liste



#### Aufgabe 1 (Plug-In-Architektur: PluginManager + JmjrstPlugin)

- Plugins anhand **Klassen**namen vergleichen, nicht getName()
- Strings auslagern (Konstanten oder Datei)
- PluginManager gibt euch Iterable ⇒ nutzt Iterator
  - kein Casten, Kopieren in Liste
- orientiert euch nicht am JMJRST-Stil



#### Aufgabe 2 (Plug-In)

■ Prüfen auf \*.png/\*.jpg sollte case insensitive sein

Entwurfsmuster

Beobachter

Vermittler

Tipps



#### Aufgabe 2 (Plug-In)

- Prüfen auf \*.png/\*.jpg sollte case insensitive sein
- Anm.: MetainfServices tut manchmal nicht richtig (oft hilft mvn clean package)

Orga Motivation

○○●○○○ ○○○

Felix Bachmann – SWT1



#### Aufgabe 2 (Plug-In)

- Prüfen auf \*.png/\*.jpg sollte case insensitive sein
- Anm.: MetainfServices tut manchmal nicht richtig (oft hilft mvn clean package)

#### Aufgabe 3 (iMage-Bundle)

keine:D



### Aufgabe 4 (Aktivitätsdiagramm)

Entwurfsmuster

Beobachter

Vermittler



### Aufgabe 4 (Aktivitätsdiagramm)

Entwurfsmuster

denkt an die Rauten!

Beobachter



### Aufgabe 4 (Aktivitätsdiagramm)

- denkt an die Rauten!
- [Bedingung]

Vermittler

Tipps



### Aufgabe 4 (Aktivitätsdiagramm)

- denkt an die Rauten!
- [Bedingung]
- verschachtelte Aktivitäten ⇒ irgendwo passender Kasten dazu

Orga Motivation

OOO OOO

Felix Bachmann – SWT1



#### Aufgabe 5 (Zustandsdiagramm)

Entwurfsmuster

Beobachter

Vermittler



#### Aufgabe 6 (Sequenzdiagramm)

Beobachter

Vermittler



#### Aufgabe 6 (Sequenzdiagramm)

bzgl. Konstruktor sind VL-Folien etwas blöd



8/40

#### Aufgabe 6 (Sequenzdiagramm)

- bzgl. Konstruktor sind VL-Folien etwas blöd
- asynchron vs. synchron (Pfeilspitzen sind wichtig!)



#### Aufgabe 6 (Sequenzdiagramm)

- bzgl. Konstruktor sind VL-Folien etwas blöd
- asynchron vs. synchron (Pfeilspitzen sind wichtig!)
- nicht statische Instanzen unterstreichen



#### Aufgabe 6 (Sequenzdiagramm)

- bzgl. Konstruktor sind VL-Folien etwas blöd
- asynchron vs. synchron (Pfeilspitzen sind wichtig!)
- nicht statische Instanzen unterstreichen
- Instanz-Kästen erst dann hinzeichnen, wenn Instanz auch existiert

#### Aufgabe 7 (Testen mit Nachahmungen)



die ersten 2 Phasen des Wasserfallmodells sind geschafft

Entwurfsmuster

Beobachter

Iterator

Stellvertreter

Vermittler

Klausuraufgabe Tipps 05.06.2018



die ersten 2 Phasen des Wasserfallmodells sind geschafft

⇒ Welche waren das nochmal?



• die ersten 2 Phasen des Wasserfallmodells sind geschafft

⇒ Welche waren das nochmal? Planung, Definition!



- die ersten 2 Phasen des Wasserfallmodells sind geschafft
  - ⇒ Welche waren das nochmal? Planung, Definition!
  - ⇒ Dokumente?



- die ersten 2 Phasen des Wasserfallmodells sind geschafft
  - ⇒ Welche waren das nochmal? Planung, Definition!
  - ⇒ Dokumente? Lastenheft, Pflichtenheft (+ andere...)



- die ersten 2 Phasen des Wasserfallmodells sind geschafft
  - ⇒ Welche waren das nochmal? Planung, Definition!
  - ⇒ Dokumente? Lastenheft, Pflichtenheft (+ andere...)
- jetzt: Entwurf!

#### Wozu Entwurf?



10/40

- Pflichtenheft (einschl. Modelle)
- Konzept Benutzungsoberfläche
- · Benutzerhandbuch + Hilfekonzept



Softwarearchitektur ist Grundlage für Implementierung!

# **Abgrenzung Definition vs. Entwurf**



Definition: Was ist zu implementieren?

Beobachter

Entwurfsmuster

# **Abgrenzung Definition vs. Entwurf**



- Definition: **Was** ist zu implementieren?
- Entwurf: Wie ist das System zu implementieren?

# **Empfehlenswerte Literatur (wirklich!)**



knapp 700 Seiten

 $\implies$  als interaktives Nachschlagewerk, falls man bestimmte Muster nicht





#### Entwurfsmuster

Ein Software-Entwurfsmuster beschreibt eine Familie von Lösungen für ein Software-Entwurfsproblem.



#### Entwurfsmuster

Ein Software-Entwurfsmuster beschreibt eine Familie von Lösungen für ein Software-Entwurfsproblem.

 schematische Klassendiagramme zur Lösung von häufig auftretenden Problemen



#### Entwurfsmuster

Ein Software-Entwurfsmuster beschreibt eine Familie von Lösungen für ein Software-Entwurfsproblem.

- schematische Klassendiagramme zur Lösung von häufig auftretenden Problemen
- Wiederverwendung von Entwurfswissen ⇒ Rad nicht neu erfinden!

Adapter



#### Entwurfsmuster

Ein Software-Entwurfsmuster beschreibt eine Familie von Lösungen für ein Software-Entwurfsproblem.

- schematische Klassendiagramme zur Lösung von häufig auftretenden Problemen
- Wiederverwendung von Entwurfswissen ⇒ Rad nicht neu erfinden!



#### Wozu Entwurfsmuster?



erleichtern Kommunikation

Beobachter

### Wozu Entwurfsmuster?



- erleichtern Kommunikation
- erleichtern "gute" Entwürfe und das Schreiben von wartbarem/erweiterbarem Code

## Geheimnisprinzip



## Geheimnis- / Kapselungsprinzip

Jedes Modul verbirgt eine wichtige Entwurfsentscheidung hinter einer wohldefinierten Schnittstelle, die sich bei einer Änderung der Entscheidung nicht mit ändert.

## Geheimnisprinzip



## Geheimnis- / Kapselungsprinzip

Jedes Modul verbirgt eine wichtige Entwurfsentscheidung hinter einer wohldefinierten Schnittstelle, die sich bei einer Änderung der Entscheidung nicht mit ändert.

## Warum eigentlich?

lokale Änderungen sollen sich nicht auf andere Teile auswirken ⇒ weniger Fehler und Arbeit

Beispiel?

## Geheimnisprinzip



## Geheimnis- / Kapselungsprinzip

Jedes Modul verbirgt eine wichtige Entwurfsentscheidung hinter einer wohldefinierten Schnittstelle, die sich bei einer Änderung der Entscheidung nicht mit ändert.

### Warum eigentlich?

lokale Änderungen sollen sich nicht auf andere Teile auswirken weniger Fehler und Arbeit

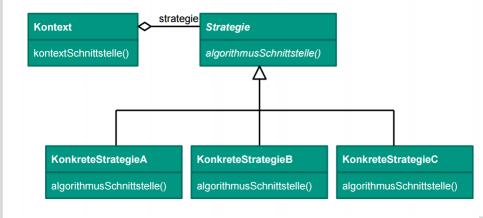
Beispiel?  $\implies$  private Attribute mit get()- und set()-Methoden



# Vorgriff: Entwurfsmuster Strategie



- Ziel: Algorithmen kapseln, austauschbar machen
- wird in vielen Entwurfsmustern verwendet



Entwurfsmuster

00000000



#### Wahr oder falsch?

■ Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren.

ooooo ooo
Felix Bachmann – SWT1



#### Wahr oder falsch?

 Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren.
 wahr



05.06.2018



#### Wahr oder falsch?

- Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren.
   wahr
- Das Strategiemuster erfüllt das Geheimnisprinzip, indem es Datenstrukturen, die in einer konkreten Strategie enthalten sind, vor dem Klienten verbirgt.



#### Wahr oder falsch?

- Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren.
   wahr
- Das Strategiemuster erfüllt das Geheimnisprinzip, indem es
   Datenstrukturen, die in einer konkreten Strategie enthalten sind, vor dem Klienten verbirgt.

Adapter



#### Wahr oder falsch?

- Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren. wahr
- Das Strategiemuster erfüllt das Geheimnisprinzip, indem es Datenstrukturen, die in einer konkreten Strategie enthalten sind, vor dem Klienten verbirgt. wahr
- Das Muster Strategie kapselt austauschbares Verhalten und verwendet Delegierung, um zu entscheiden, welches Verhalten verwendet wird.

Felix Bachmann - SWT1

05.06.2018



#### Wahr oder falsch?

- Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren. wahr
- Das Strategiemuster erfüllt das Geheimnisprinzip, indem es Datenstrukturen, die in einer konkreten Strategie enthalten sind, vor dem Klienten verbirgt. wahr
- Das Muster Strategie kapselt austauschbares Verhalten und verwendet Delegierung, um zu entscheiden, welches Verhalten verwendet wird. wahr



#### Wahr oder falsch?

- Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren. wahr
- Das Strategiemuster erfüllt das Geheimnisprinzip, indem es Datenstrukturen, die in einer konkreten Strategie enthalten sind, vor dem Klienten verbirgt. wahr
- Das Muster Strategie kapselt austauschbares Verhalten und verwendet Delegierung, um zu entscheiden, welches Verhalten verwendet wird. wahr
- Das Hinzufügen einer neuen konkreten Strategie erfordert keine Änderung existierender konkreter Strategien.

Felix Bachmann - SWT1



#### Wahr oder falsch?

- Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren. wahr
- Das Strategiemuster erfüllt das Geheimnisprinzip, indem es Datenstrukturen, die in einer konkreten Strategie enthalten sind, vor dem Klienten verbirgt. wahr
- Das Muster Strategie kapselt austauschbares Verhalten und verwendet Delegierung, um zu entscheiden, welches Verhalten verwendet wird. wahr
- Das Hinzufügen einer neuen konkreten Strategie erfordert keine Änderung existierender konkreter Strategien. wahr

## Kategorien der Entwurfsmuster



- Entkopplungs-Muster
  - Adapter
  - Beobachter
  - Iterator
  - Stellvertreter
  - Vermittler
  - Brücke
- Varianten-Muster
- Zustandshandhabungs-Muster
- Steuerungs-Muster
- Bequemlichkeits-Muster

## **Entkopplungs-Muster**



 übergeordnetes Ziel: System in Teile aufspalten, die unabhängig voneinander sind

⇒ Teile austauschbar bzw. veränderbar

## Adapter



### **Problem**

- Klassen mit inkompatiblen Schnittstellen, die wir aber zusammen benutzen wollen
- Schnittstellen nicht änderbar (z.B. externe Bibliotheken)

oooooo ooo Felix Bachmann – SWT1

Motivation

Adapter

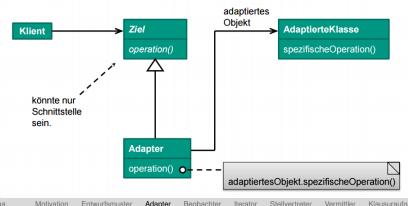
•00

## **Adapter**



### **Problem**

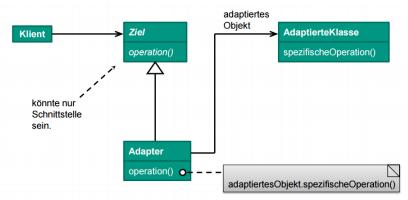
- Klassen mit inkompatiblen Schnittstellen, die wir aber zusammen benutzen wollen
- Schnittstellen nicht änderbar (z.B. externe Bibliotheken)



Felix Bachmann - SWT1

# Adapter (Objektadapter)





## Wir sind bei Entkopplung-Mustern, Preisfrage:

Wo ist hier die Entkopplung?

Motivation

Entwurfsmuster

Adapter 000

Beobachter

Iterator

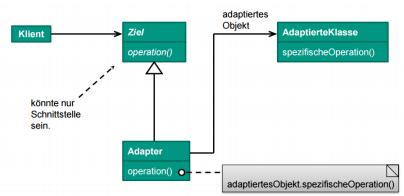
Stellvertreter

Vermittler

Klausuraufgabe

# Adapter (Objektadapter)





## Wir sind bei Entkopplung-Mustern, Preisfrage:

Wo ist hier die Entkopplung?

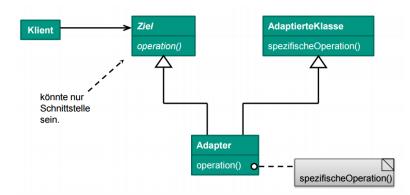
der Klient ist von der adaptierten Klasse entkoppelt ⇒ austauschbar

Felix Bachmann – SWT1

Motivation

## Adapter - Alternative (Klassenadapter)



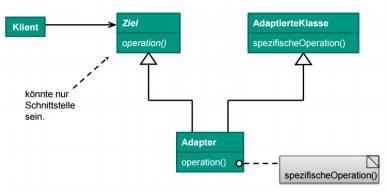


Felix Bachmann – SWT1

Motivation

# Adapter - Alternative (Klassenadapter)





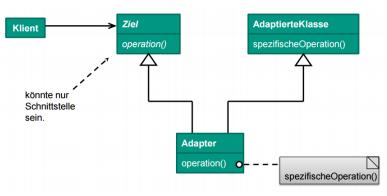
Was für ein Problem bekommt ihr, wenn ihr das auf einem ÜB implementieren müsst?

oooooo ooo Felix Bachmann – SWT1

05.06.2018

## Adapter - Alternative (Klassenadapter)





Was für ein Problem bekommt ihr, wenn ihr das auf einem ÜB implementieren müsst?

⇒ keine Mehrfachvererbung in Java!

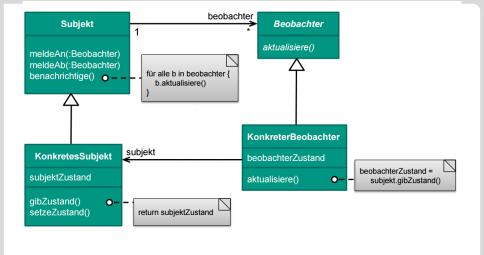
## Beobachter/Observer: abstrakt



### **Problem**

- ein Subjekt, viele Beobachter
- Subjekt ändert Zustand ⇒ Beobachter machen "irgendwas"

Beobachter



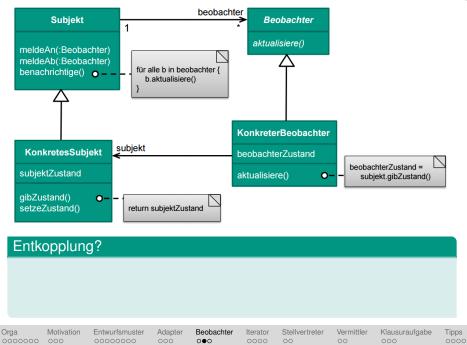
Motivation

Orga

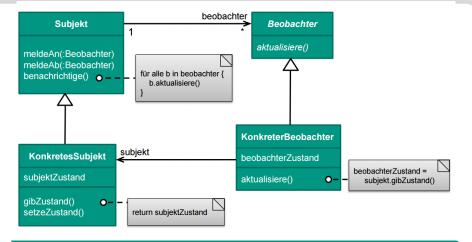
Entwurfsmuster

Beobachter

000



Felix Bachmann - SWT1 05.06.2018 24/40



### Entkopplung?

jeder Beobachter definiert, was bei Benachrichtigung passiert,
 Subjekt kriegt davon nichts mit

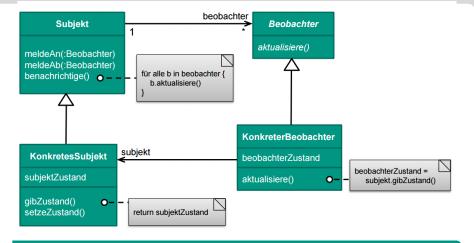
oooooo ooo Felix Bachmann – SWT1 Beobachter Iterator

○●○ ○○○○

Iterator Stellvertreter

Vermittler 00

ler Klausuraufgabe



### Entkopplung?

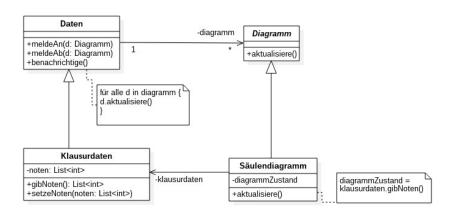
- jeder Beobachter definiert, was bei Benachrichtigung passiert,
   Subjekt kriegt davon nichts mit
- zur Laufzeit änderbar: Anzahl der Beobachter

Felix Bachmann – SWT1

Motivation

# Beobachter/Observer: am Beispiel





Orga Motivation Entwurfsmuster Adapter Beobachter Iterator Stellvertreter Vermittler Klausuraufgabe Tipps
0000000 000 000 000 000 000 000
000 000 000 000
000 000 000 000
000 000 000 000
000 000 000 000



### **Problem**

wollen über Datenstruktur iterieren + Operationen ausführen

⇒ Hinzufügen, Löschen...





### **Problem**

- wollen über Datenstruktur iterieren + Operationen ausführen
   Hinzufügen, Löschen...
- das Ganze ohne Kentniss des internen Aufbaus der Datenstruktur

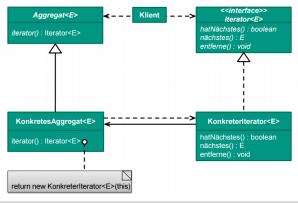
Orga Motivation Entwurfsmuster Adapter Beobachter Iterator Stellvertreter Vermittler Klausuraufgabe Tipps
000000 000 000 000 000 000
Felix Bachmann - SWT1

05.06.2018 26/40



#### **Problem**

- wollen über Datenstruktur iterieren + Operationen ausführen
   Hinzufügen, Löschen...
- das Ganze ohne Kentniss des internen Aufbaus der Datenstruktur

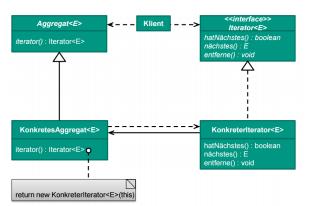


COCCOCC COC

Motivation

Orga

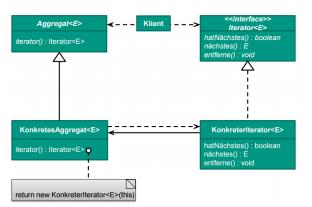




# Entkopplung?

Orga Motivation





## Entkopplung?

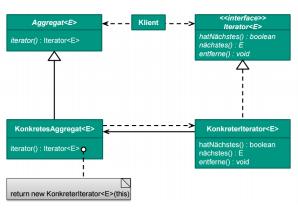
 Klient benutzt nur Methoden der Schnittstelle auf dem konkreten Iterator

⇒ Implementierung austauschbar

COCCOCC COC

Motivation





Beispiel in Java: list.iterator();



#### Wahr oder falsch?

Klienten können mithilfe des Iterator-Musters Sammlungen von Objekten und einzelne Objekte auf die gleiche Weise behandeln.

000000 000 0000000 000 000 000 00 000 00 000 00 0

Beobachter

Iterator

Stellvertreter

Vermittler

Klausuraufgabe

29/40

Adapter

Entwurfsmuster



#### Wahr oder falsch?

 Klienten können mithilfe des Iterator-Musters Sammlungen von Objekten und einzelne Objekte auf die gleiche Weise behandeln.



Beobachter

Iterator

Stellvertreter

Vermittler

Klausuraufgabe

29/40

Adapter

Entwurfsmuster



#### Wahr oder falsch?

- Klienten können mithilfe des Iterator-Musters Sammlungen von Objekten und einzelne Objekte auf die gleiche Weise behandeln. falsch
- Das Entwurfsmuster Iterator ist den Variantenmustern zuzuordnen.

Felix Bachmann - SWT1

Entwurfsmuster

Adapter

Beobachter Iterator 000

Stellvertreter

Vermittler

Klausuraufgabe 05.06.2018

# Quiz (Ankreuzaufgaben aus Klausuren)



#### Wahr oder falsch?

 Klienten können mithilfe des Iterator-Musters Sammlungen von Objekten und einzelne Objekte auf die gleiche Weise behandeln.

falsch

Das Entwurfsmuster Iterator ist den Variantenmustern zuzuordnen.

falsch

Felix Bachmann - SWT1

Entwurfsmuster

Adapter Beobachter Iterator 000

Stellvertreter

Vermittler

Klausuraufgabe 05.06.2018



#### **Problem**

 wollen Zugriff auf ein Objekt kontrollieren, ohne seine Klasse zu ändern

Orga

Motivation



#### **Problem**

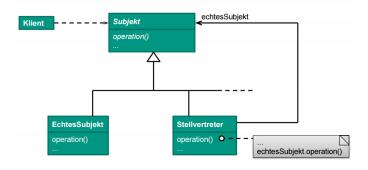
- wollen Zugriff auf ein Objekt kontrollieren, ohne seine Klasse zu ändern
  - ⇒ Stellvertreter macht Zugriffskontrolle



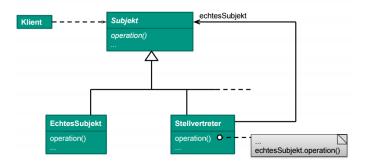
30/40

#### Problem

- wollen Zugriff auf ein Objekt kontrollieren, ohne seine Klasse zu ändern
  - ⇒ Stellvertreter macht Zugriffskontrolle







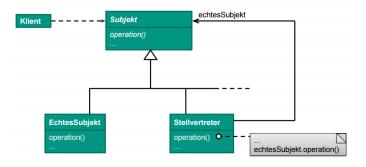
## Entkopplung?

Orga Motivation

OOOOOO OOO

Felix Bachmann – SWT1





### Entkopplung?

Klient hat keinen direkten Zugriff auf das echte Subjekt

oooooo ooo Felix Bachmann – SWT1

05.06.2018



### Problem

mehrere voneinander abhängige Objekte

Beobachter

Tipps



#### **Problem**

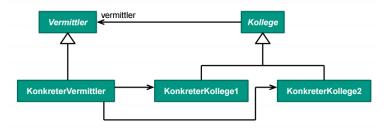
- mehrere voneinander abhängige Objekte
  - ⇒ Zustände der Objekte von anderen Zuständen abhängig

Beobachter



#### **Problem**

- mehrere voneinander abhängige Objekte
  - ⇒ Zustände der Objekte von anderen Zuständen abhängig



Felix Bachmann - SWT1

Motivation

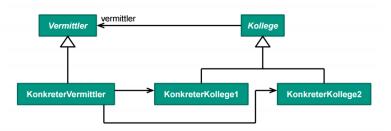
Entwurfsmuster Adapter Beobachter

Stellvertreter

Vermittler

Klausuraufgabe 05.06.2018





## Entkopplung?

Orga Motivation Felix Bachmann - SWT1

Entwurfsmuster

Adapter

Beobachter

Iterator

Stellvertreter

Vermittler 0

Klausuraufgabe

05.06.2018

Tipps 33/40





# Entkopplung?

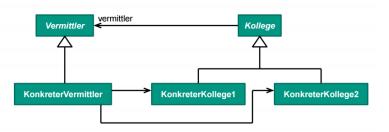
Kollegen kennen sich nicht direkt

Orga Motivation

OOOOOO OOO

Felix Bachmann – SWT1





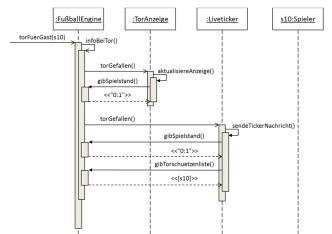
## Entkopplung?

- Kollegen kennen sich nicht direkt
  - ⇒ Hinzufügen eines Kollegen erfordert keine Änderung der alten Kollegen

oooooo ooo Felix Bachmann – SWT1 Beobachter 000

# Klausuraufgabe (Hauptklausur SS 2012)



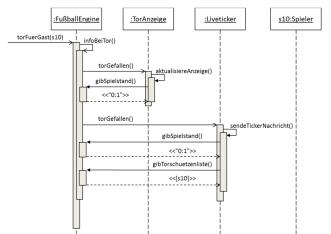


### Aufgabe 1

Welches Entwurfsmuster erkennen Sie in diesem Diagramm?

# Klausuraufgabe (Hauptklausur SS 2012)





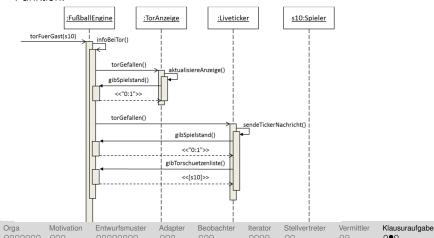
## Aufgabe 1

Motivation

Welches Entwurfsmuster erkennen Sie in diesem Diagramm? Beobachter.

Org	ga	Motivation	Entwurfsmuster	Adapter	Beobachter	Iterator	Stellvertreter	Vermittler	Klausuraufgabe	Tipp
00	00000	000	00000000	000	000	0000	00	00	●00	00
Fel	Felix Bachmann - SWT1						05.06.2018 3		34/40	

Entwerfen Sie das folgende Klassendiagramm passend zu dem Seguenzdiagramm; es soll alle verwendeten Klassen und Methoden enthalten. Kennzeichnen Sie die Zugreifbarkeiten der Methoden mit den Symbolen +, -, #; seien Sie dabei möglichst restriktiv. Verzichten Sie auf die Modellierung von Attributen, Kennzeichnen Sie die Elemente des Entwurfsmusters und deren Funktion.



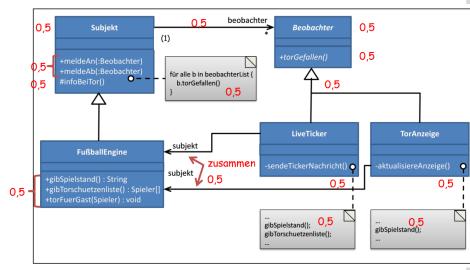
Felix Bachmann - SWT1

05.06.2018

## Musterlösung



36/40



Orga Motivation Entwurfsmuster Adapter Beobachter Iterator Stellvertreter Vermittler Klausuraufgabe
000000 000 000 000 000 000

Felix Bachmann − SWT1

05.06.2018



### Aufgabe 1: iMage-GUI

macht die "kleinen" Bonusaufgaben

⇒ relativ leichte Punkte





### Aufgabe 1: iMage-GUI

- macht die "kleinen" Bonusaufgaben
  - ⇒ relativ leichte Punkte
- schaut euch die verschiedenen LayoutManager aus Java Swing an
  - ⇒ verschiedene LayoutManager möglich (via mehrerer Container,
  - z.B. JPanel)

Felix Bachmann - SWT1

Tipps **-0**00



### Aufgabe 1: iMage-GUI

- macht die "kleinen" Bonusaufgaben
  - ⇒ relativ leichte Punkte
- schaut euch die verschiedenen LayoutManager aus Java Swing an
  - → verschiedene LayoutManager möglich (via mehrerer Container,
  - z.B. JPanel)

### Aufgabe 2: Zustandsdiagramm (LEZ)

- nochmal Syntax anschauen
  - → Was darf in Zustandsdiagramm, was nicht? (laut VL)
- von Hand!





## Aufgabe 3: Git

- echo "hallo">> test.txt schreibt hallo in test.txt
- git-Dokumentation anschauen



Beobachter



### Aufgabe 3: Git

- echo "hallo">> test.txt schreibt hallo in test.txt
- git-Dokumentation anschauen

#### Aufgabe 4: Architekturstile

- Jmjrst (gedanklich) umbauen
- Zusammenhang der Klassen anschauen (z.B. Main-Klasse)

Beobachter

oooooo ooo Felix Bachmann – SWT1

## Denkt dran!



## Abgabe

Deadline am 13.6 um 12:00

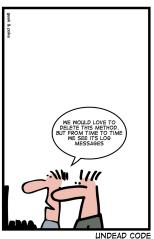
Entwurfsmuster

A2-4 handschriftlich!

Beobachter

## Bis dann! (dann := 19.06.17)







Orga Motivation Entwurfsmuster

Adapter

Beobachter

Stellvertreter

Vermittler

Klausuraufgabe 05.06.2018

**Tipps** 0000 40/40