

Softwaretechnik 1 - 3. Tutorium

Tutorium 03 Felix Bachmann | 12.06.2017

KIT - INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION (IPD



- Orga
- Motivation
- Entwurfsmuster
- Adapter
- Beobachter
- Iterator
- Stellvertreter
- Vermittler
- Klausuraufgabe
- Tipps

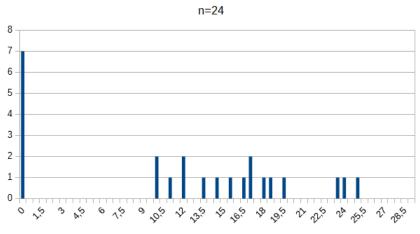
Orga

Entwurfsmuster

Vermittler

2. Übungsblatt Statistik





Ø 11,56 bzw. 16,32 von 28+1

•000000 Felix Bachmann - SWT1

Motivation

Orga

Entwurfsmuster

Adapter

Beobachter

Iterator

Stellvertreter

Vermittler

Klausuraufgabe 12.06.2017

Tipps 3/40



Allgemein

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

Adapter



Aufgabe 1 (Plug-In-Architektur): Ø 2,25 bzw. 3,86 von 5

Beobachter

Entwurfsmuster

Vermittler

Tipps



Aufgabe 1 (Plug-In-Architektur): Ø 2,25 bzw. 3,86 von 5

- keine riesigen switch-cases/if-else Anweisungen in compareTo() benutzen
 - ⇒ Erweiterbarkeit wird dadurch eingeschränkt



Aufgabe 1 (Plug-In-Architektur): Ø 2,25 bzw. 3,86 von 5

- keine riesigen switch-cases/if-else Anweisungen in compareTo() benutzen
 - ⇒ Erweiterbarkeit wird dadurch eingeschränkt
 - → Java vergleicht Enum-Elemente anhand ihrer Position in dem Enum



Aufgabe 1 (Plug-In-Architektur): Ø 2,25 bzw. 3,86 von 5

- keine riesigen switch-cases/if-else Anweisungen in compareTo() benutzen
 - ⇒ Erweiterbarkeit wird dadurch eingeschränkt
 - → Java vergleicht Enum-Elemente anhand ihrer Position in dem Enum
- orientiert euch nicht am JMJRST-Stil



Aufgabe 2 (Plug-In): Ø 1,79 bzw. 3,58 von 4

keine :D

Aufgabe 3 (iMage-Bundle): Ø 0,44 bzw. 1,75 von 2

• 6 Abgaben ...

Entwurfsmuster



Aufgabe 4 (Aktivitätsdiagramme (Geometrify)): \varnothing 4,83 bzw. 7,25 von 10

Entwurfsmuster



Aufgabe 4 (Aktivitätsdiagramme (Geometrify)): Ø 4,83 bzw. 7,25 von 10

denkt an die Rauten!

Entwurfsmuster



Aufgabe 4 (Aktivitätsdiagramme (Geometrify)): Ø 4,83 bzw. 7,25 von 10

- denkt an die Rauten!
- Geometrify sollte eigener Kasten mit Objektfluss sein



Aufgabe 4 (Aktivitätsdiagramme (Geometrify)): Ø 4,83 bzw. 7,25 von 10

- denkt an die Rauten!
- Geometrify sollte eigener Kasten mit Objektfluss sein
- [Bedingung]



Aufgabe 4 (Aktivitätsdiagramme (Geometrify)): Ø 4,83 bzw. 7,25 von 10

- denkt an die Rauten!
- Geometrify sollte eigener Kasten mit Objektfluss sein
- [Bedingung]
- verschachtelte Aktivitäten ⇒ irgendwo passender Kasten dazu

Orga Motivation

OOOOOOO

Felix Bachmann – SWT1



Aufgabe 5 (Sequenzdiagramm (main-Methode)): Ø 1,58 bzw. 2,71 von 5

Beobachter

Entwurfsmuster



Aufgabe 5 (Sequenzdiagramm (main-Methode)): Ø 1,58 bzw. 2,71 von 5

bzgl. Konstruktor sind VL-Folien etwas blöd

Motivation

Orga



Aufgabe 5 (Sequenzdiagramm (main-Methode)): Ø 1,58 bzw. 2,71 von 5

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



Aufgabe 5 (Seguenzdiagramm (main-Methode)): Ø 1,58 bzw. 2,71 von 5

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



Aufgabe 5 (Sequenzdiagramm (main-Methode)): Ø 1,58 bzw. 2,71 von 5

- 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 6 (Substitutionsprinzip): Ø 0,67 bzw. 1,23 von 3

Beobachter

Vermittler



Aufgabe 6 (Substitutionsprinzip): Ø 0,67 bzw. 1,23 von 3

die Methode wurde überladen

⇒ Java schaut sich nur die Signatur an



Aufgabe 6 (Substitutionsprinzip): Ø 0,67 bzw. 1,23 von 3

- die Methode wurde überladen
 - ⇒ Java schaut sich nur die Signatur an
- für Erfüllung des Substitutionsprinzips auch Verhalten wichtig



Aufgabe 6 (Substitutionsprinzip): Ø 0,67 bzw. 1,23 von 3

- die Methode wurde überladen
 - ⇒ Java schaut sich nur die Signatur an
- für Erfüllung des Substitutionsprinzips auch Verhalten wichtig
- Methoden sind dynamisch gebunden, Attribute sind statisch gebunden
 - merke: getter und setter benutzen
 - vermeidet mindfucks



die ersten 2 Phasen des Wasserfallmodells sind geschafft

Motivation

Entwurfsmuster

Klausuraufgabe

Tipps



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!

Beobachter

Iterator

Stellvertreter

Dokumente?

Motivation

Entwurfsmuster

Vermittler

10/40

Klausuraufgabe



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



10/40

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

Wozu Entwurf?



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



Softwarearchitektur ist Grundlage für Implementierung!

Abgrenzung Definition vs. Entwurf



Definition: Was ist zu implementieren?

Entwurfsmuster

Beobachter

Vermittler

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



Was sind Entwurfsmuster?



Entwurfsmuster

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

Was sind Entwurfsmuster?



Entwurfsmuster

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

schematische Klassendiagramme zur Lösung von häufig auftretenden Problemen

Was sind Entwurfsmuster?



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!

Was sind Entwurfsmuster?



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!



12.06.2017

Wozu Entwurfsmuster?



erleichtern Kommunikation

Entwurfsmuster

0000000

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.

12.06.2017

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 \implies 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

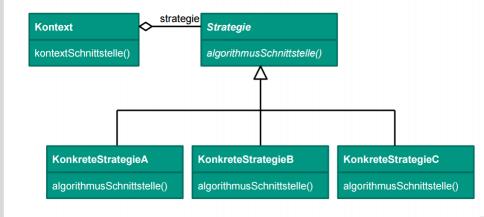


Motivation

Vorgriff: Entwurfsmuster Strategie



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



Entwurfsmuster

00000000

Motivation

Kategorien der Entwurfsmuster



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



Wahr oder falsch?

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



Wahr oder falsch?

 Das Entwurfsmuster Strategie bietet die Möglichkeit, eine Klasse mit einer von mehreren möglichen Verhaltensweisen zu konfigurieren.
 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 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.
- Das Muster Strategie kapselt austauschbares Verhalten und verwendet Delegierung, um zu entscheiden, welches Verhalten verwendet wird.

oooooo ooo Felix Bachmann – SWT1

12.06.2017



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

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 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

Entkopplungs-Muster



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

⇒ Teile austauschbar bzw. veränderbar

Adapter



21/40

Problem

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

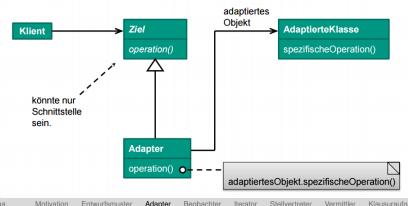
Orga Motivation Control Contro

Adapter



Problem

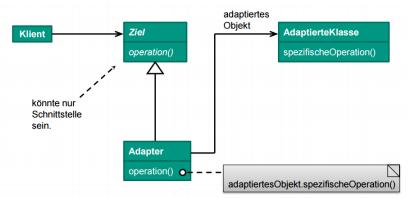
- Klassen mit inkompatiblen Schnittstellen, die wir aber zusammen benutzen wollen
- Schnittstellen nicht änderbar (z.B. externe Bibliotheken)



Felix Bachmann - SWT1

Adapter (Objektadapter)





Wir sind bei Entkopplung-Mustern, Preisfrage:

Wo ist hier die Entkopplung?

Motivation

Entwurfsmuster

Adapter 000

Beobachter

Iterator

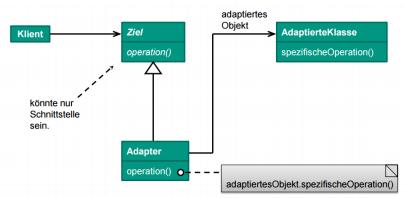
Stellvertreter

Vermittler

Klausuraufgabe

Adapter (Objektadapter)





Wir sind bei Entkopplung-Mustern, Preisfrage:

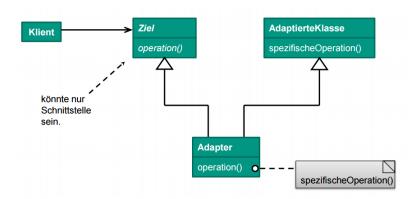
Wo ist hier die Entkopplung?

der Klient ist von der adaptierten Klasse entkoppelt ⇒ austauschbar

oooooo ooo Felix Bachmann – SWT1

Adapter - Alternative (Klassenadapter)

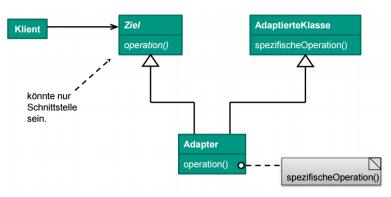




Adapter - Alternative (Klassenadapter)



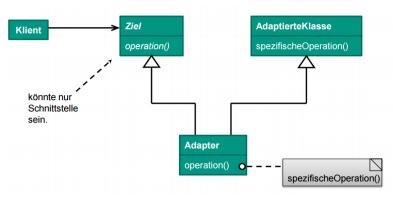
23/40



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

Adapter - Alternative (Klassenadapter)





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

⇒ keine Mehrfachvererbung in Java!

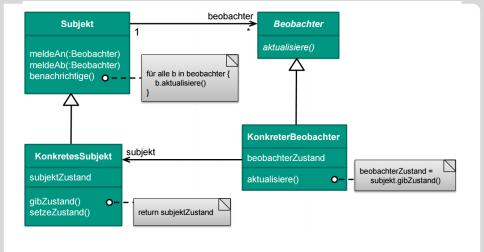
Beobachter/Observer: abstrakt



Problem

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

• 0 0



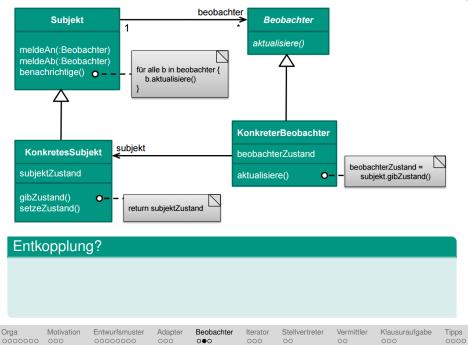
Motivation

Orga

000

Entwurfsmuster

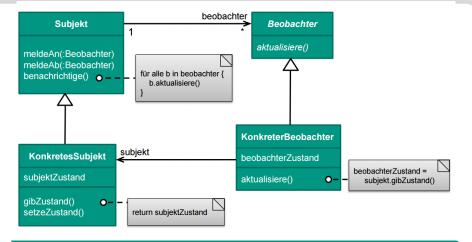
12.06.2017



12.06.2017

25/40

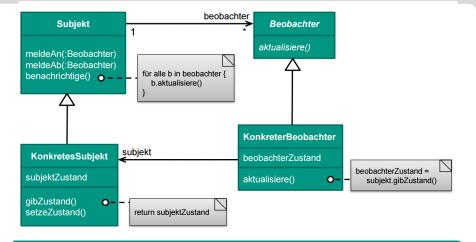
Felix Bachmann - SWT1



Entkopplung?

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

25/40



Entkopplung?

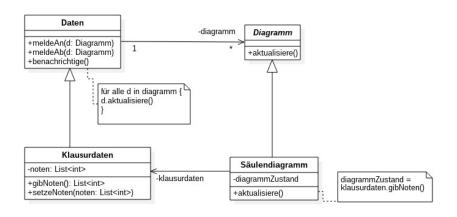
- jeder Beobachter definiert, was bei Benachrichtigung passiert,
 Subjekt kriegt davon nichts mit
- zur Laufzeit änderbar: Anzahl der Beobachter

Felix Bachmann – SWT1

Motivation

Beobachter/Observer: am Beispiel







Problem

• wollen über Datenstruktur iterieren + Operationen ausführen

→ Hinzufügen, Löschen...



Tipps



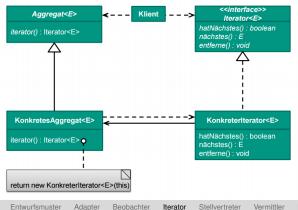
Problem

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



Problem

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

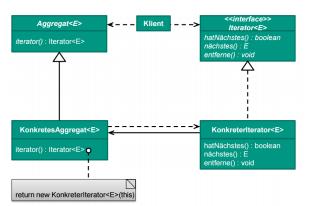


Felix Bachmann – SWT1

Motivation

Orga

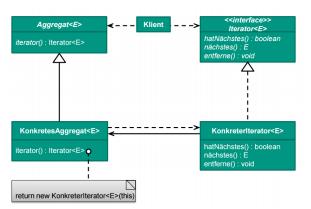




Entkopplung?

Orga Motivation





Entkopplung?

 Klient benutzt nur Methoden der Schnittstelle auf dem konkreten Iterator

⇒ Implementierung austauschbar

COCCOCC COC

Motivation



Wahr oder falsch?

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

Quiz (Ankreuzaufgaben aus Klausuren)



29/40

Wahr oder falsch?

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



Entwurfsmuster

Stellvertreter Beobachter Klausuraufgabe 000 Felix Bachmann - SWT1 12.06.2017

Iterator

Vermittler

Adapter

Quiz (Ankreuzaufgaben aus Klausuren)



Wahr oder falsch?

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

Felix Bachmann – SWT1

Entwurfsmuster

Stellvertreter

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

COCCOCC

ooo

Entwurfsmuster 0000000 Adapter Beobachter

Iterator ○○● Stellvertreter 00 Vermittler 00 Klausuraufgabe



Problem

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

Orga

Motivation



Klausuraufgabe

30/40

Vermittler

12.06.2017

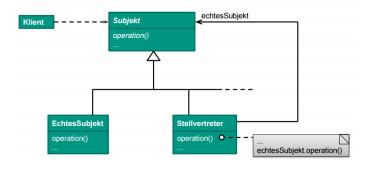
Problem

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

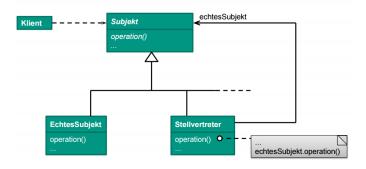


Problem

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







Entkopplung?

Orga Motivation Felix Bachmann - SWT1

Entwurfsmuster

Adapter

Beobachter

Iterator

Stellvertreter

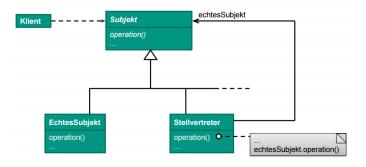
Vermittler

Klausuraufgabe

31/40

Tipps





Entkopplung?

Klient hat keinen direkten Zugriff auf das echte Subjekt

oooooo ooo Felix Bachmann – SWT1

Motivation Entwu

lterator 000 Vermittler



Problem

mehrere voneinander abhängige Objekte

Beobachter

Vermittler

Klausuraufgabe



Problem

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

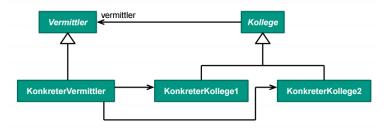
Stellvertreter

Beobachter



Problem

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



0000000





Entkopplung?

Orga Motivation Felix Bachmann - SWT1

Entwurfsmuster

Adapter

Beobachter

Iterator

Stellvertreter

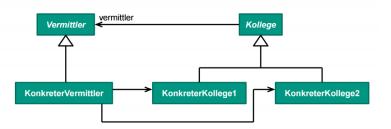
Vermittler 0

Klausuraufgabe

12.06.2017

Tipps 33/40





Entkopplung?

Kollegen kennen sich nicht direkt

Orga Motivation

OOOOOO OOO

Felix Bachmann – SWT1

Entwurfsmuster 00000000 Adapter 000 Beobachter 000 lterator 000

rator Stellvertreter

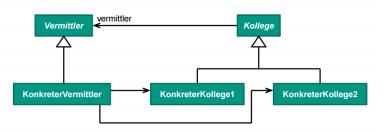
reter **V**e

Vermittler ○●

ler Klausuraufgabe 000 12.06.2017

Tipps 0000





Entkopplung?

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

Orga Motivation

Entwurfsmuster

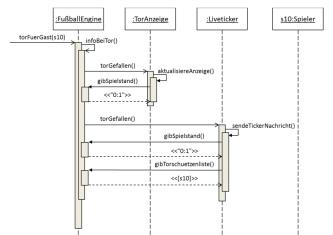
Adapter 000 Beobachter 000 lterator 000 Stellvertreter

Vermittler ○●

ler Klausuraufgabe 000 12.06.2017

Klausuraufgabe (Hauptklausur SS 2012)





Aufgabe 1

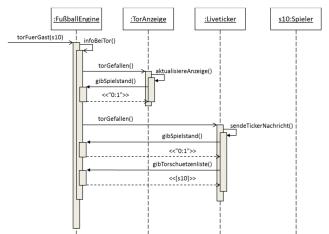
Welches Entwurfsmuster erkennen Sie in diesem Diagramm?

 Orga
 Motivation 000000
 Entwurfsmuster 000000
 Adapter 000000
 Beobachter 0000
 Iterator 0000
 Stellvertreter 000
 Vermittler 000
 Klausuraufgabe 000
 Tipps 000

 Felix Bachmann − SWT1
 34/40

Klausuraufgabe (Hauptklausur SS 2012)



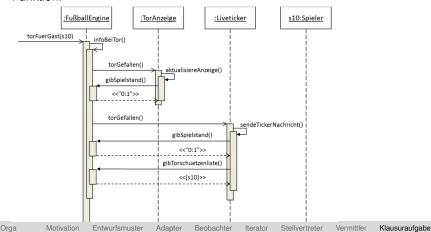


Aufgabe 1

Welches Entwurfsmuster erkennen Sie in diesem Diagramm? Beobachter.

Orga	Motivation	Entwurfsmuster
0000000	000	00000000
Felix Bachmann - SWT1		

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.



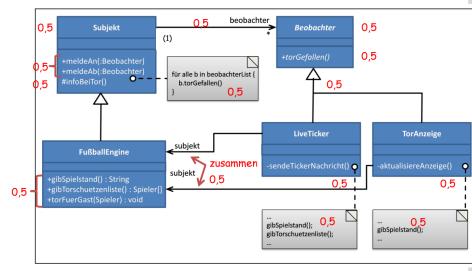
Felix Bachmann - SWT1

000

Musterlösung



36/40





Aufgabe 1: Zustandsdiagramm (LEZ)

nochmal Syntax anschauen

Entwurfsmuster

⇒ Was darf in Zustandsdiagramm, was nicht? (laut VL)

Beobachter

12.06.2017



Aufgabe 1: Zustandsdiagramm (LEZ)

- nochmal Syntax anschauen
 - ⇒ Was darf in Zustandsdiagramm, was nicht? (laut VL)

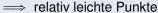
Aufgabe 2: Die Abbottsche Methode

- back to Deutsch-Unterricht
 - ⇒ prinzipiell nicht schwierig



Aufgabe 3: iMage-GUI

macht die "kleinen" Bonusaufgaben



Entwurfsmuster

Beobachter



Aufgabe 3: 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



Aufgabe 3: 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 4: Geheimnisprinzip

- leichte Punkte
- Attribute sollten ?? sein?
 - ⇒ Und warum nochmal?

Denkt dran!



Abgabe

- Deadline am 21.6 um 12:00
- A{1,2,4} handschriftlich!

Beobachter

Bis dann! (dann := 26.06.17)



