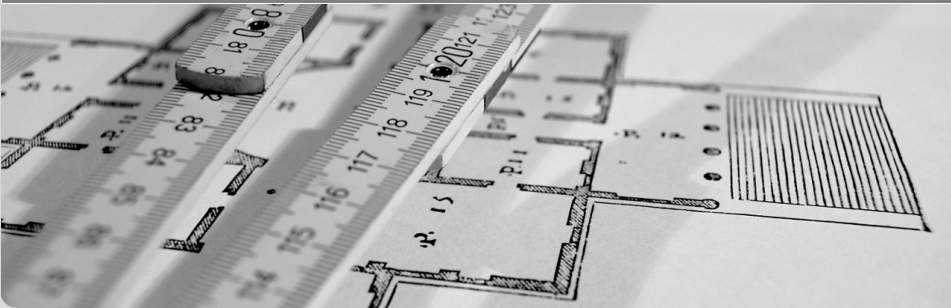


Softwaretechnik 1 - 3. Tutorium

Tutorium 03

Felix Bachmann | 12.06.2017

KIT - INSTITUT FÜR PROGRAMMSTRUKTUREN UND DATENORGANISATION (IPD)



- 1 Orga
- 2 Motivation
- 3 Entwurfsmuster
- 4 Adapter
- 5 Beobachter
- 6 Iterator
- 7 Stellvertreter
- 8 Vermittler
- 9 Tipps

2. Übungsblatt Statistik

Allgemein



Aufgabe 1 (Plug-In-Architektur): Ø von 4 (+ 1)



Aufgabe 2 (Plug-In): Ø von 4



Aufgabe 3 (iMage-Bundle): Ø von 2



Aufgabe 4 (Aktivitätsdiagramme (Geometrify)): Ø von 10



Aufgabe 5 (Sequenzdiagramm (main-Methode)): Ø von 5



Aufgabe 6 (Substitutionsprinzip): Ø von 3



- die ersten 2 Phasen des Wasserfallmodells sind geschafft

- 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!

- Pflichtenheft (einschl. Modelle)
- Konzept Benutzungsoberfläche
- Benutzerhandbuch + Hilfekonzzept



Entwurfsprozess



Softwarearchitektur

Softwarearchitektur ist Grundlage für Implementierung!

- Definition: **Was** ist zu implementieren?

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

Empfehlenswerte Literatur (wirklich!)

knapp 700 Seiten

⇒ als interaktives Nachschlagewerk, falls man bestimmte Muster nicht versteht



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 \implies Rad nicht neu erfinden!

Entwurfsmuster

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

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



Geheimnis- / Kapselungsprinzip

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

Geheimnis- / Kapselungsprinzip

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

Sinn?

Geheimnis- / Kapselungsprinzip

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

Sinn? \implies Änderungen ohne Risiko durchführen

Geheimnis- / Kapselungsprinzip

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

Sinn? \implies Änderungen ohne Risiko durchführen
Beispiel?

Geheimnis- / Kapselungsprinzip

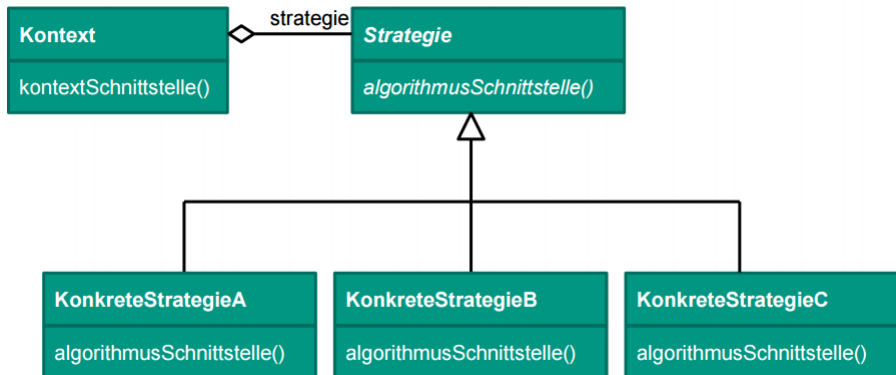
Jedes Modul verbirgt eine wichtige Entwurfsentscheidung hinter einer wohldefinierten Schnittstelle, die sich bei einer Änderung der Entscheidung nicht mit ändert.

Sinn? \implies Änderungen ohne Risiko durchführen

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

Vorgriff: Entwurfsmuster Strategie

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



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

- übergeordnetes Ziel: System in Teile aufspalten, die unabhängig voneinander sind
⇒ Teile austauschbar bzw. veränderbar

Inhalt...

Inhalt...

Inhalt...

Inhalt...

Inhalt...

Aufgabe



Aufgabe



Aufgabe



Aufgabe



Aufgabe



Aufgabe



Abgabe

- Deadline am 21.6 um 12:00

Bis dann! (dann := 26.06.17)