



Software-Engineering für langlebige Systeme

Wie sieht eine Code-Analyse genauer aus? Welche Komponenten enthält sie? Sind diese so ähnlich wie in Aufgabenblatt 2?

- Kommt drauf an
- Es gibt verschiedene Arten
- Analyse auf Bufferoverflows
- Style-Analyse
-

Ist das Fühlen von Softwareerosion wirklich fast reine Erfahrung und Intuition?

Ja

- Vieles wird nicht explizit erfasst
- In verschiedenen Communities gilt verschiedenes als gut

Nein

- Metriken
- Smells
- Anti-Pattern
- Analyse-Tools
 - Buffer-Overflows
 - Style-Checker

Wie genau wird das Volumen des Programmes bei der Halstead-Metrik "gut" bestimmt? Raten? Scharfes Hinsehen? Trivial?

- Erfahrungswerte für verschiedene Systemtypen
- Es gibt verschiedene Meinungen welche Werte gut sind
- Meist für eine Firma oder ein Projekt festgelegt

Wird immer eine Teilmenge der Halstead-Metriken verwendet oder alle Metriken? Oder nur jeweils eine?

- Je nach Informationsbedürfnis
-

Was ist denn nun genauer die Bisimulation?

Seien $T_i = (Q_i, \rightarrow_i, q_{0,i})$, $i = 1, 2$ Transitionssysteme

Eine Relation $R \subseteq Q_1 \times Q_2$ heißt

starke Bisimulation zwischen T_1 und T_2 ,

falls $(q_{0,1}, q_{0,2}) \in R$ ist und für alle $(q_1, q_2) \in R$, $\alpha \in \text{Act}$ gilt:

1. $q_1 \xrightarrow{\alpha}_1 p_1 \Rightarrow \exists p_2 : q_2 \xrightarrow{\alpha}_2 p_2$ und $(p_1, p_2) \in R$,
2. $q_2 \xrightarrow{\alpha}_2 p_2 \Rightarrow \exists p_1 : q_1 \xrightarrow{\alpha}_1 p_1$ und $(p_1, p_2) \in R$.

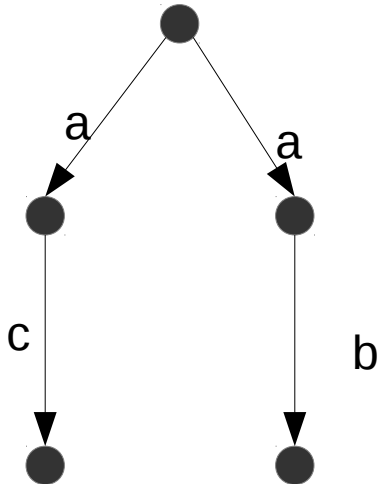
T_1 und T_2 heißen stark bisimulationsäquivalent ($T_1 \sim T_2$),
falls es eine starke Bisimulation zwischen T_1 und T_2 gibt.

Wie werden nichtdeterministische Events mit Traces dargestellt?

- Nicht durch einen Trace
- Nur durch die Menge der Traces
 - Für einen Prefix sind mehrere Tails möglich

Können Transitionssysteme überhaupt nichtdeterministisch sein? (bzgl. Transitionen ungleich der Tau-Transitionen)

- Eingeschränkt ja:



Sind Evolutionsfilter Homomorphismen?

- Nicht zwingend
- Sie sind Äquivalenzrelationen

Was genau ist die NAC bei Graphtransformationssystemen definiert?

- Syntaktisch
 - Als Graph mit Identitätsvariablen
- Semantisch
 - Als Teilgraph der nicht im Graphen enthalten sein darf

Warum nimmt man beim Anbinden von Altsystemen als Wrappersprache keine Sprache mit mehreren Programmierparadigmen? Ist der Grund dafür etwa der, dass man sonst mehrere Programmierparadigmen zu sehr mischt?

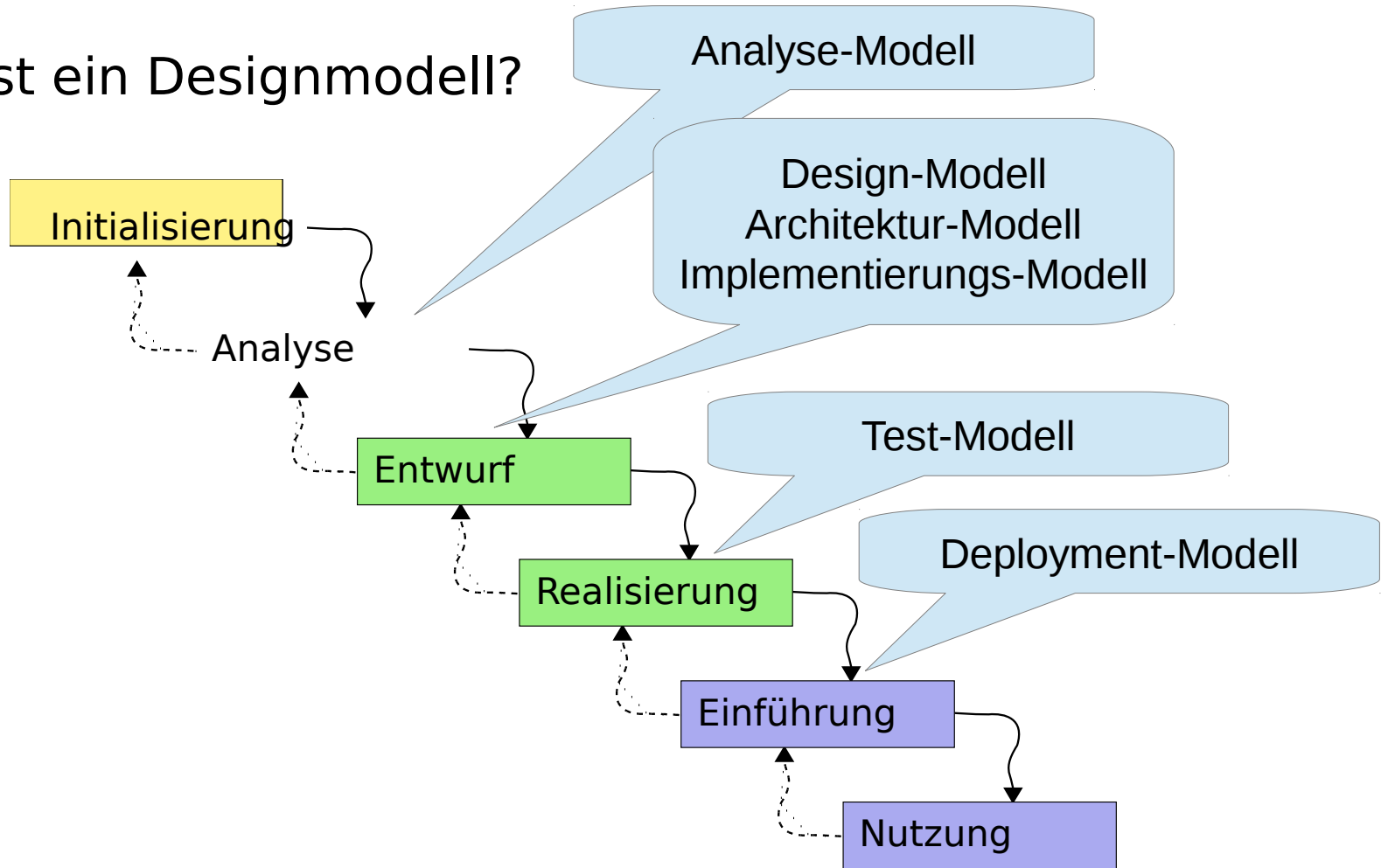
- Wie soll so eine Sprache aussehen?
 - Integration von zwei Ansätzen möglich
 - Nicht alle Paradigmen lassen sich ordentlich mischen
 - Teilweise sehr unterschiedliche Umsetzung von Paradigmen
 - Mehrfachvererbung vs. Einfachvererbung
 - Explizites Speichermanagement vs. Garbage-Collection
- Welche Paradigmen müssen dann enthalten sein?
 - OO
 - Imperativ
 - Modular
 - Funktional
 - Deklarative

Was sind die Vorteile von Nassi-Schneiderman, BPMN und Jackson-Diagramme? Wie sehen diese Diagramme semantisch aus?

- Diskussion und Tafel

Sind bei Sachverständnissen (siehe Kapitel 10) die Einschätzungen eher objektiv oder subjektiv?

Was ist ein Designmodell?



Wie ist der FLOW bezüglich Wasseradern und Feng Shui im Büro eingestellt?

- Supercalifragelisticexpialotisch
- Flow ist ein Effekt der wissenschaftlich beschrieben ist
 - FLOW ist nicht esoterisch
 - Es wird ein Zustand beschrieben
- Der Effekt von Wasseradern ist nicht wissenschaftlich belegt
 - Arbeiten die einen positiven Effekt schildern, haben nachweisbare methodische Mängel
- Feng Shui
 - Mischung aus Esoterik und Erfahrung

Erfahrungswerte, die nachweisbar sind, aber gerne esoterisch begründet

- Stellung des Schreibtisches im Raum
 - Sichtbarkeit der Tür
 - Wasser im Raum
 - Höhere Luftfeuchtigkeit → Bessere Gesundheit
 - Türen nicht in einer Flucht
 - Besserer Luftaustausch

Wie sehen Metamodelle zu Modellen aus? (Diese Metafrage musste sein)

- An der Tafel
- Meta-Meta-Model
- Meta-Model
- Model
- Instance

Ich hätte gerne noch einmal den Zusammenhang zwischen Refinement-Failure, Refinement-Trace und der Bisimulation eines Transitions-Systems erklärt. Vielleicht noch einmal mit einem Hinweis auf häufige Fehler die gemacht werden und auf häufige Fehlannahmen.

- Tafel

~~Außerdem würde ich mich sehr über ein weiteres~~
Beispiel eines Input/Output Trace Transitions System
freuen (T4.1). Vielleicht mit einer mehr formalen
Definition der Vor- respektive Nachbedingungen.