

Peter  
Fichtner

Fiducia & GAD IT AG



Tilmann  
Glaser

it-economics GmbH

Wieso...

Regeln für schlechte  
Tests?

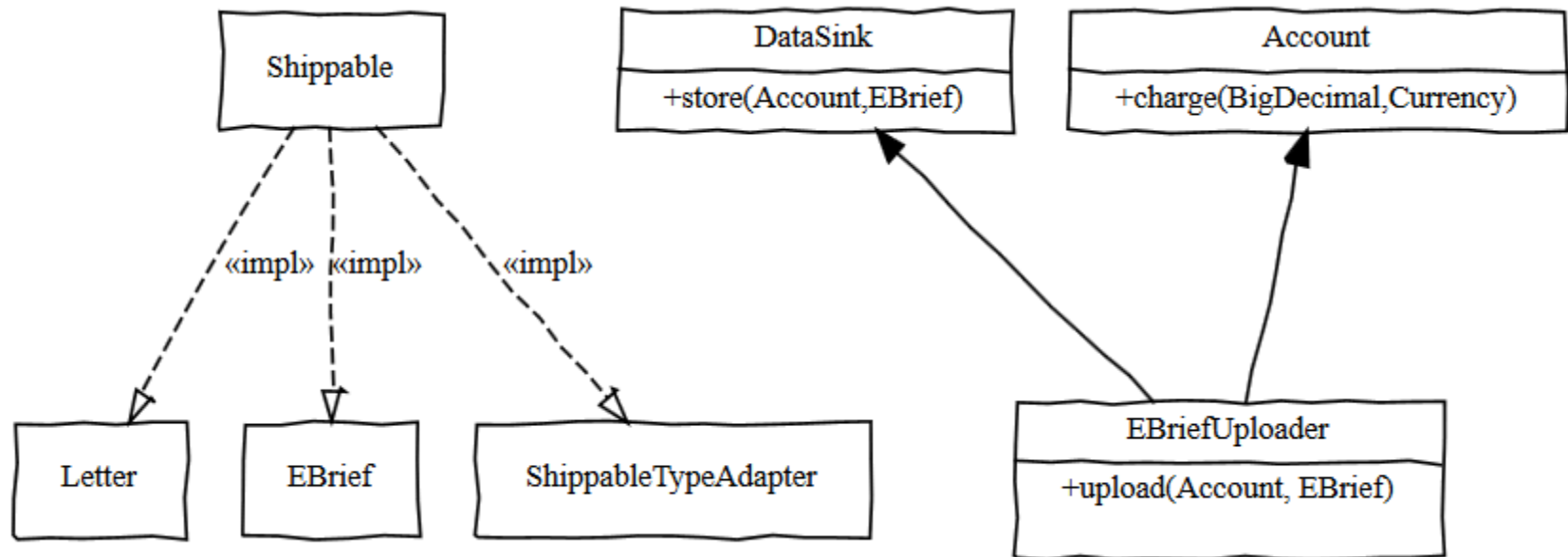
## Regeln für schlechte Tests!

- Schlechte Tests anhand von Merkmalen erkennbar
  - Das Fehlen der Merkmale heißt aber nicht, dass ein Test gut ist
- Je mehr unserer 10 goldenen Regeln auf einen Test zutreffen, desto schlechter ist der Test
- Regeln lassen sich grob Gruppieren in
  - Zu klein
  - Zu groß
  - Zu fokussiert
  - Zu genügsam

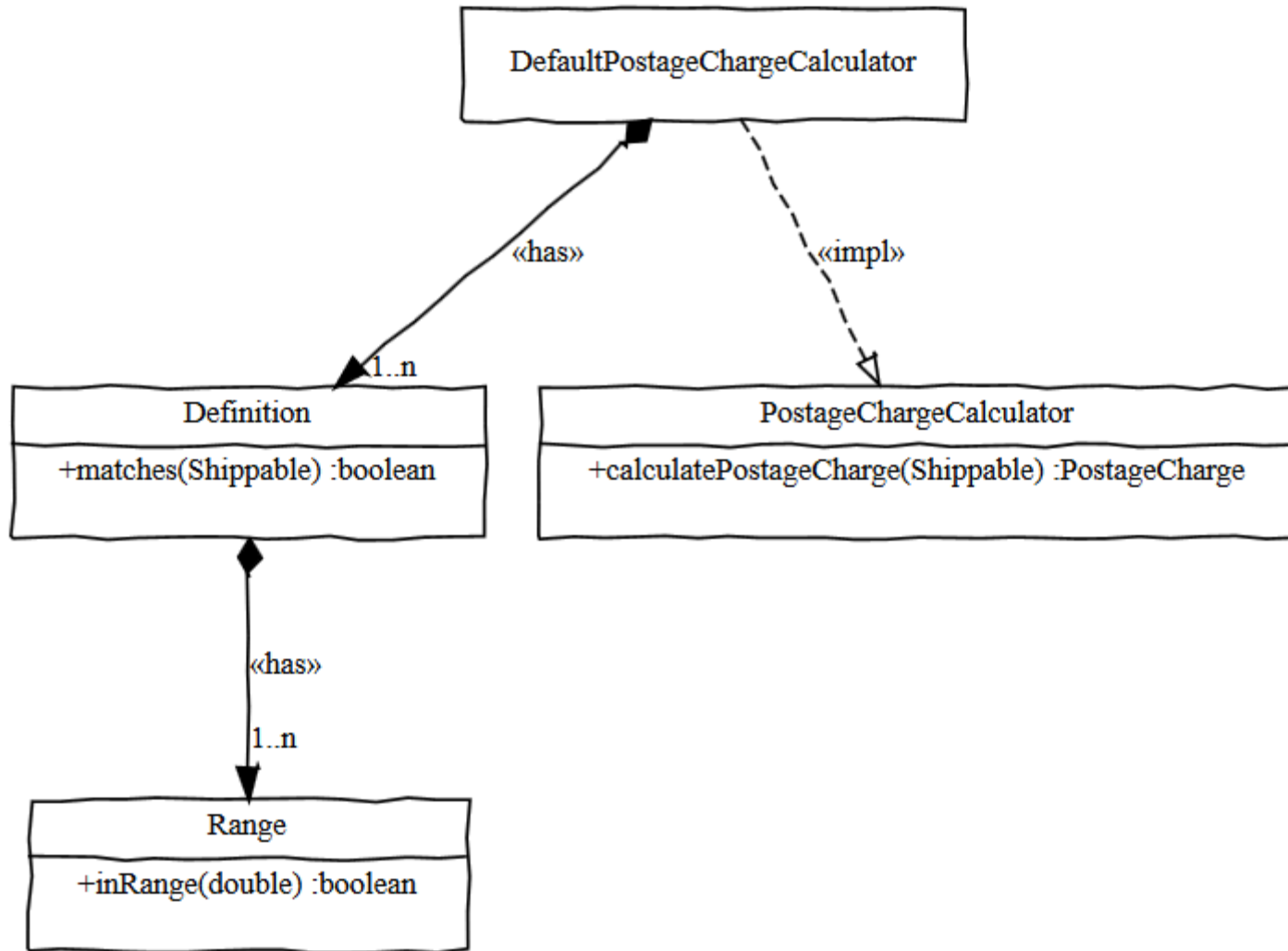
# Das „System under Test“



# Das „System under Test“



# Das „System under Test“



# Das „System under Test“

## ➤ Code-Basis:

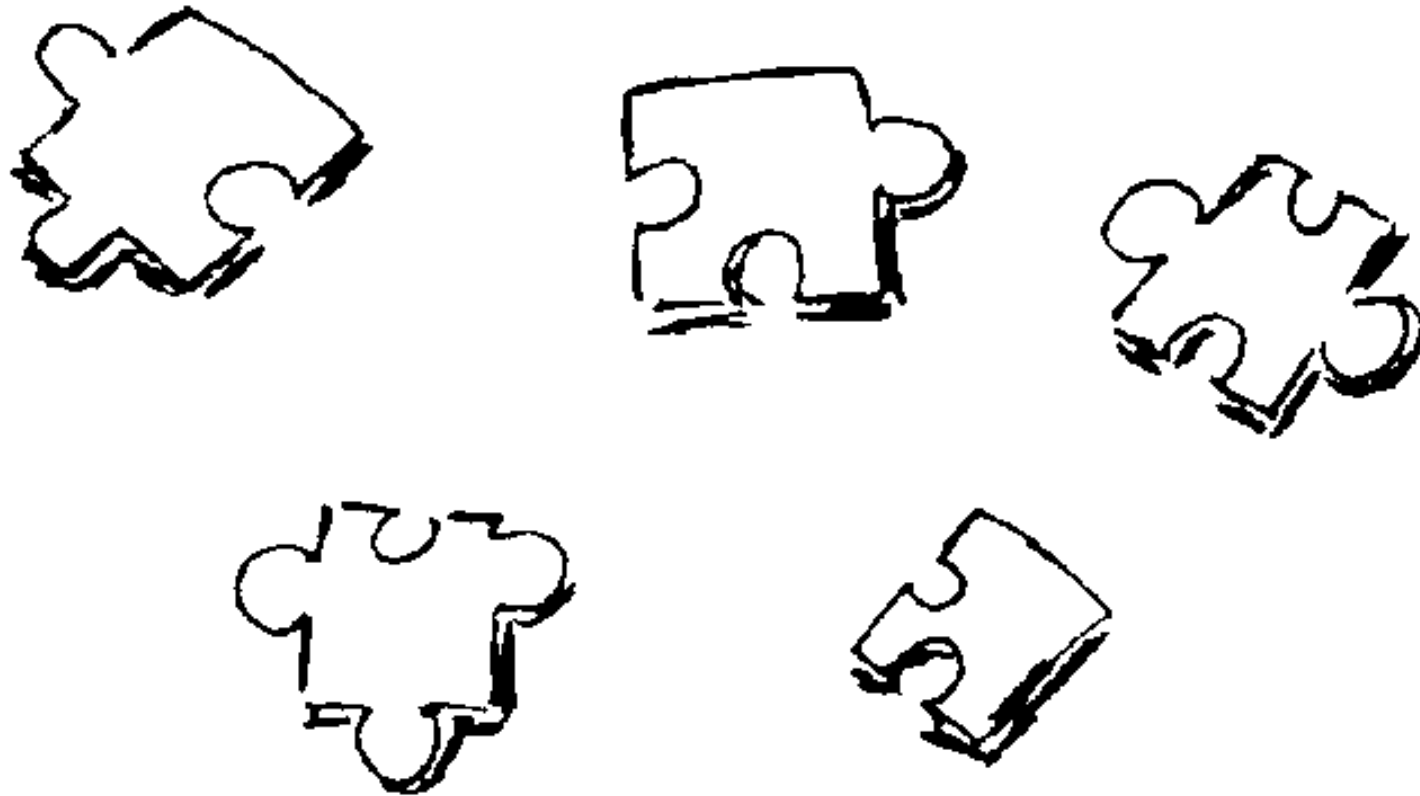
<https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests>

## ➤ Gute Tests:

<https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/blob/master/test/de/fiduciagad/ase/postagecharge/PostageChargeCalculatorTest.java>

<https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/ebrief>

Zu klein





Jede Methode im  
Produktivcode hat  
genau eine  
Testmethode

Durchbreche die  
Schnittstellen  
deines Systems

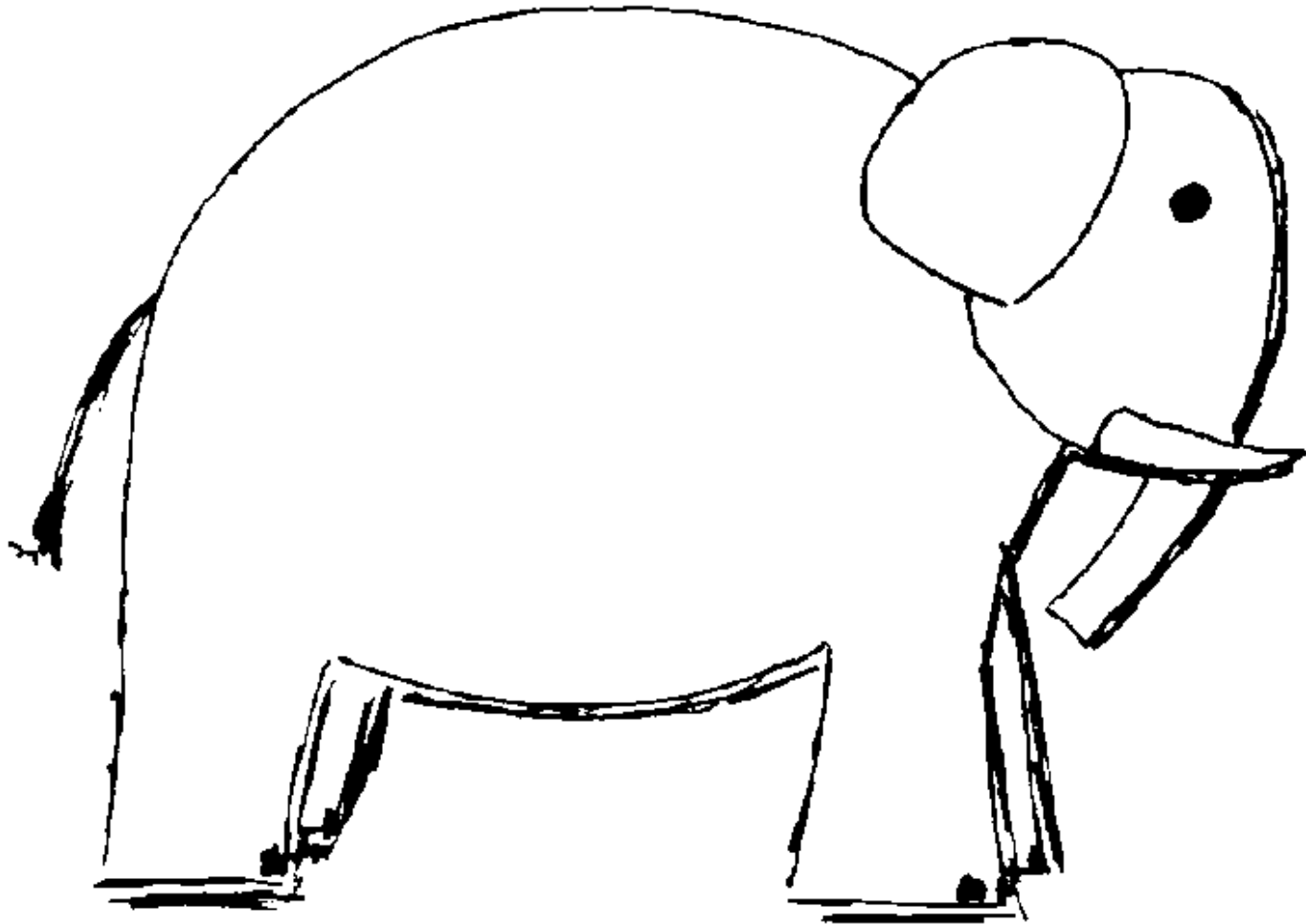
Treffe Architektur-  
entscheidungen möglichst  
früh. Zementiere diese  
Entscheidungen über Tests

Zu klein - Codebeispiele

➤ Codebeispiele:

[https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad\\_small](https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad_small)

Zu groß



Mache dir keine  
Gedanken über den  
Schnitt deiner  
Software

Schreibe Tests immer  
erst nachdem du den  
Produktivcode  
geschrieben hast

Nutze dein einmal  
konfiguriertes System  
für möglichst viele  
Tests



## Zu groß - Codebeispiele

### ➤ Codebeispiele:

[https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad\\_integrativ](https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad_integrativ)

[https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad\\_noarchitecture](https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad_noarchitecture)

Zu fokussiert



Nutze Mocks wo immer  
es geht. Nutze alle  
Features der Mocking  
Frameworks

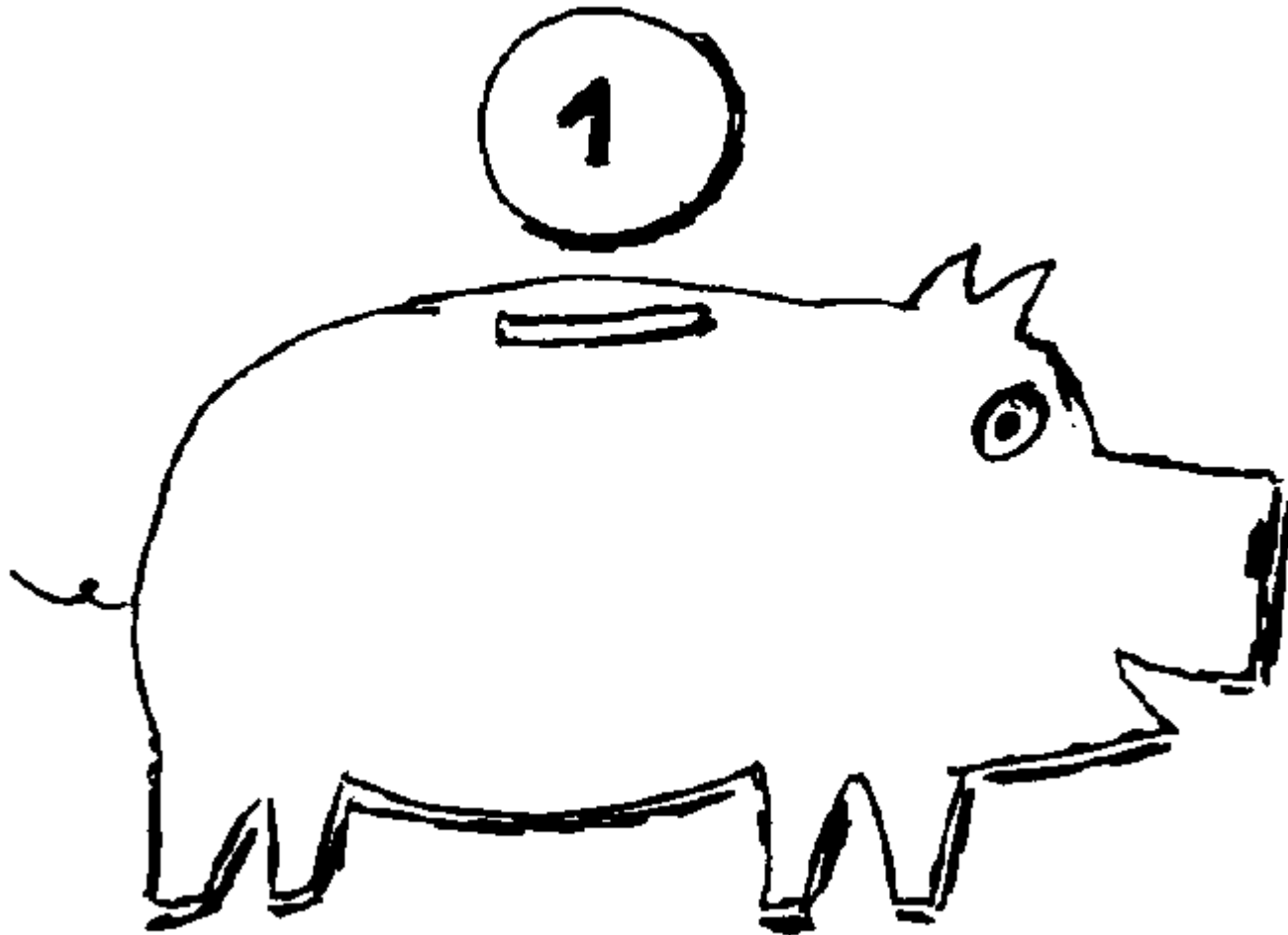
Teste deine  
Macks

Zu fokussiert - Codebeispiele

➤ Codebeispiele:

[https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad\\_test\\_your\\_mock](https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad_test_your_mock)

Zu genügsam



Formuliere Tests  
immer  
vollumfänglich  
technisch

Entscheide dich für  
genau eine Ebene der  
Testpyramide



Zu genügend - Codebeispiele

➤ Codebeispiele:

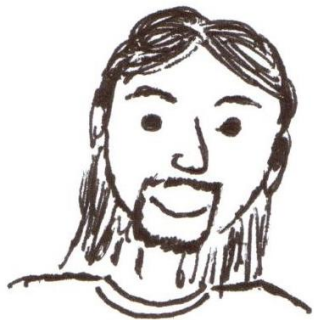
[https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad\\_technical\\_not\\_professional](https://github.com/fiduciagad/die10goldenenRegelnFuerSchlechteTests/tree/master/test/de/fiduciagad/ase/postagecharge/bad_technical_not_professional)

## Fazit

- Schlechte Tests
  - sind manchmal leicht, oft schwer zu erkennen
  - machen keinen Spaß
    - weder dem, der sie schreibt
    - noch dem, der sie lesen und warten muss
  - sind verprasstes Budget
  - sind immer noch (ein bisschen) besser als keine Tests
- Habt Mut, eure Tests zu verbessern.
  - Denkt über den Tellerrand hinaus
- Entscheidungen werden uns Entwicklern nicht abgenommen. Sie sind von uns zu treffen

## Fazit

- Fragen & Anregungen?
  - Meldet euch gerne
    - [peter.fichtner@fiduciagad.de](mailto:peter.fichtner@fiduciagad.de)
    - [tglaser@it-economics.de](mailto:tglaser@it-economics.de)



Peter  
Fichtner

Fiducia & GAD IT AG



# Die 10 goldenen Regeln für schlechte Tests



Tilmann  
Glaser

it-economics GmbH