

# Scala

## Gedanken zu einer Programmiersprache

---

Sebastian Eidecker

16. März 2016

*Wer als Werkzeug nur einen Hammer hat,  
sieht in jedem Problem einen Nagel.*

*— Paul Watzlawick*

Inhalt

IT im Wandel

# Inhalt

IT im Wandel

Herausforderungen

# Inhalt

IT im Wandel

Herausforderungen

Manifeste

# Inhalt

IT im Wandel

Herausforderungen

Manifeste

Scala

---

# Inhalt

IT im Wandel

Herausforderungen

Manifeste

Scala

Management Summary

# Inhalt

IT im Wandel

Herausforderungen

Manifeste

Scala

Management Summary

Ein wenig Code



# Inhalt

IT im Wandel

Herausforderungen

Manifeste

Scala

Management Summary

Ein wenig Code

Spannendes

# Software Engineering

---

# Software Engineering

# IT im Wandel

---

## Herausforderungen

# Forderungen an IT

## Forderungen an IT

- Stabilität und Resilienz

## Forderungen an IT

- Stabilität und Resilienz
- Wertbeitrag

## Forderungen an IT

- Stabilität und Resilienz
- Wertbeitrag
- Businessstreiber



## Forderungen an IT

- Stabilität und Resilienz
- Wertbeitrag
- Businessstreiber

— Matthias Magnor – CEO Surface und Contract Logistics

# IT im Wandel

---

Manifeste

## Manifeste

- Antwortbereit, Widerstandsfähig, Elastisch, Nachrichtenorientiert (2013)

# Manifeste

- Gut gefertigt, Stets Mehrwert, Gemeinschaft aus Experten, Produktive Partnerschaften (2009)

# Manifeste

- Individuen und Interaktionen, Funktionierende Software, Zusammenarbeit mit dem Kunden, Reagieren auf Veränderung (2001)

## Manifeste

- Antwortbereit, Widerstandsfähig, Elastisch, Nachrichtenorientiert (2013)
- Gut gefertigt, Stets Mehrwert, Gemeinschaft aus Experten, Produktive Partnerschaften (2009)
- Individuen und Interaktionen, Funktionierende Software, Zusammenarbeit mit dem Kunden, Reagieren auf Veränderung (2001)

Wo stehen wir?

Wo stehen wir im Wettbewerb?

---



# Businesslogik?

|| // ToDo

# Scalable Language

# Scala

---

## Management Summary

Charakter

## Charakter

- Java 2.0

## Charakter

- Java 2.0
- Objektorientiert

## Charakter

- Java 2.0
- Objektorientiert
- Funktional

## Charakter

- Java 2.0
- Objektorientiert
- Funktional
- Java Virtual Machine



## Charakter

- Java 2.0
- Objektorientiert
- Funktional
- Java Virtual Machine
- Entwicklungsprozess wie gehabt

Vorteile

## Vorteile

- Modern

## Vorteile

- Modern
- Verständlich funktional

## Vorteile

- Modern
- Verständlich funktional
- Java-Ökosystem

## Vorteile

- Modern
- Verständlich funktional
- Java-Ökosystem
- Macht Spaß

## Vorteile

- Modern
- Verständlich funktional
- Java-Ökosystem
- Macht Spaß
- Statisch typisiert

## Vorteile

- Modern
- Verständlich funktional
- Java-Ökosystem
- Macht Spaß
- Statisch typisiert

## Nachteile



## Vorteile

- Modern
- Verständlich funktional
- Java-Ökosystem
- Macht Spaß
- Statisch typisiert

## Nachteile

- Komplex

## Vorteile

- Modern
- Verständlich funktional
- Java-Ökosystem
- Macht Spaß
- Statisch typisiert

## Nachteile

- Komplex
- Statisch typisiert  
(nicht mehr cool)

## Vorteile

- Modern
- Verständlich funktional
- Java-Ökosystem
- Macht Spaß
- Statisch typisiert

## Nachteile

- Komplex
- Statisch typisiert  
(nicht mehr cool)
- Java-Ökosystem  
(nicht mehr cool)

# Scala

---

Ein wenig Code

Eine Klasse

## Eine Klasse

```
|| case class Person(firstName:String, lastName:String)
```

# Scala

---

## Spannendes

---

Akka



## Akka

- Scalable real-time transaction processing

## Akka

- Scalable real-time transaction processing
- Will die aktuellen Probleme lösen

## Akka

- Scalable real-time transaction processing
- Will die aktuellen Probleme lösen
-

ScalaTest

# ScalaTest

-

## ScalaTest

- 

-

## ScalaTest

- 
- 
-

# Scala

---

Mehr für Nerds



Mehr für Nerds

## Mehr für Nerds

- Sprecht mich an

## Mehr für Nerds

- Sprecht mich an
- Eigener Termin bei Interesse

## Mehr für Nerds

- Sprech mich an
- Eigener Termin bei Interesse
- Heiko Seeberger: „Durchstarten mit Scala. Tutorial für Einsteiger (2. Aufl.)“





# Primzahlen

# Primzahlen

```
def swap(i: Int, j: Int) {  
    val t = xs(i); xs(i) = xs(j); xs(j) = t  
    ()  
}
```







Titel

Titel

.

# Titel

- 

-

# Titel

- 
- 
-

Titel

Titel

•



## Titel

- 
-

# Titel

- 
- 
-

Titel

- 
- 
- 

Titel

Titel

- 
- 
- 

Titel

-

Titel

- 
- 
- 

Titel

- 
-

Titel

- 
- 
- 

Titel

- 
- 
-