

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

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

— Paul Watzlawick

Ich möchte heute über Scala reden. Eigentlich will ich aber nicht nur über Scala reden. Ich möchte nicht Scala verkaufen, sondern etwas völlig anderes. Scala ist spannend - für Nerds, die des Geldes wegen Java benutzen. Wir müssen aber auch über wichtigere Probleme reden. Und vielleicht passt Scala ja doch irgendwie.

In der IT wollen wir Probleme lösen und Möglichkeiten schaffen. Ich bin der festen Überzeugung, dass wir zu selten über den Kern unserer Probleme nachdenken und auch, dass sich diese Probleme gerade ändern und immer schneller ändern werden.

Ich glaube daher, dass wir unser Problem **und** unseren Werkzeugkasten kennen müssen. Ich glaube, dass wir zumindest wissen müssen, was im Baumarkt ausliegt und wir bei Bedarf einkaufen können. Daher kann es aus meiner Sicht nie schaden, sich neue Werkzeuge und Arbeitsweisen anzuschauen, damit man sie bei Bedarf zumindest im Hinterkopf hat. Sonst kann es passieren, dass wir das Problem gar nicht verstehen, weil wir kein passendes Werkzeug dafür besitzen und verwenden können.

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

Deswegen, möchte ich über zwei Dinge reden: Neue Herausforderungen und Scala.

Software Engineering

Software Engineering

2016-03-10

Scala

IT im Wandel

Software Engineering

Software Engineering. Passt der Begriff?

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

2016-03-10

Scala

IT im Wandel

Herausforderungen

Forderungen an IT

- Stabilität und Resilienz
- Wertbeitrag
- Businessstreiber

— Matthias Magnor – CEO Surface und Contract Logistics

Das Bewusstsein, dass ein Wandel stattfindet, ist im Business angekommen. Diese Forderungen stammen von

Matthias Magnor!

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)

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)

Es gibt Manifeste von Softwareentwicklern, die sehr Ähnliches aussagen. (Reaktives Manifest, agiles Manifest, Manifest der Software Craftmanship-Bewegung als Beispiele) Diese sind bekannt und – ich habe zumindest das Gefühl – auch anerkannt. Ich will auch nur sehr kurz darauf eingehen, es soll ja vor allem um Scala gehen.

Wo stehen wir?

Wo stehen wir im Wettbewerb?

2016-03-10

Scala

IT im Wandel

Manifeste

Wo stehen wir im Wettbewerb?

TODO: Was bedeutet das

Businesslogik?

1 // ToDo

2016-03-10

Scala

IT im Wandel

Manifeste

Businesslogik?

• // ToDo

Wo ist die Businesslogik versteckt?

Scala

Management Summary

Scalable Language

2016-03-10

Scala

Scala

Management Summary

Scalable Language

TODO: Was bedeutet das

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

Charakter

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

Passt in unseren Entwicklungsprozess. Leichte Änderungen an Deployment etc.

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)

2016-03-10

Scala

Scala

Management Summary

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 Java-Klasse

Eine Java-Klasse

```
1 public class Person {  
2     private final String firstName;  
3     private final String lastName;  
4     public Person(String firstName, String lastName) {  
5         this.firstName = firstName;  
6         this.lastName = lastName;  
7     }  
8     public String getFirstName() {  
9         return firstName;  
10    }  
11    public String getLastName() {  
12        return lastName;  
13    }
```


Eine Java-Klasse

```
1  @Override
2  public boolean equals(Object o) {
3      if (this == o) return true;
4      if (o == null || getClass() != o.getClass()) return false;
5      Person person = (Person) o;
6      if (firstName != null ?
7          !firstName.equals(person.firstName) :
8          person.firstName != null) return false;
9      if (lastName != null ?
10         !lastName.equals(person.lastName) :
11         person.lastName != null) return false;
12     return true;
13 }
```

Eine Java-Klasse

```
1  @Override
2  public int hashCode() {
3      int result = firstName != null ? firstName.hashCode() : 0;
4      result =
5          31 * result + (lastName != null ? lastName.hashCode() :
6              0);
7      return result;
8  }
```

Eine Scala-Klasse

Eine Scala-Klasse

```
1 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
-

2016-03-10

Scala

Scala

Spannendes

Akka

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

ScalaTest

ScalaTest

-

ScalaTest

-

-

ScalaTest

-
-
-

2016-03-10

Scala

Scala

Spannendes

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

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

2016-03-10

Scala

Scala

Mehr für Nerds

Mehr für Nerds

- [Sprecht mich an](#)
- [Eigener Termin bei Interesse](#)
- [Heiko Seeberger: „Durchstarten mit Scala. Tutorial für Einsteiger \(2. Aufl.\)“](#)

