

java.next

stuart halloway http://thinkrelevance.com

Copyright 2007-2009 Relevance, Inc. This presentation is licensed under a Creative Commons Attribution-Noncommercial-Share Alike 3.0 United States License. See http://creativecommons.org/licenses/by-nc-sa/3.0/us/

java can also be part of the solution

most java code is bad

repetitive, bureaucratic code untested/untestable stuck on an old version of miles and miles of crap









java, circa 2003

```
1 public ActionForward edit(ActionMapping mapping,
                             ActionForm form,
 3
                             HttpServletRequest request,
                             HttpServletResponse response)
       throws Exception {
     PersonForm personForm = (PersonForm) form;
     if (personForm.getId() != null) {
       PersonManager mgr =
9
         (PersonManager) getBean("personManager");
10
       Person person = mgr.getPerson(personForm.getId());
11
       personForm = (PersonForm) convert(person);
12
       updateFormBean(mapping, request, personForm);
13
14
     return mapping.findForward("edit");
15 }
```

5

java.next: 2005-?

```
def edit
   @person = Person.find(params[:id])
end

def new
   @person = Person.new
end
```

evolving style

convention over configuration

reasonable defaults

no checked exceptions

YAGNI

domain nouns, not language nouns

DSLs

how java.next can help



Photo credit: http://www.flickr.com/photos/uhduh/92437037/sizes/l/

common java.next features

everything is an object

easy beans

higher-order functions

unchecked exceptions

open APIs

DSLs

11

everything is an object

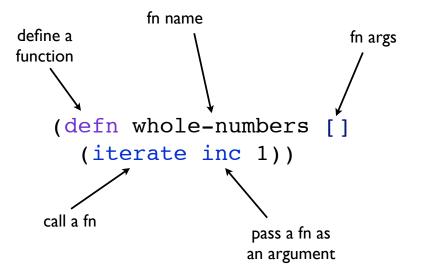
```
1 ; is java enterprise-ready?
2 2000000 * 2000000
3 -> 1385447424

1 # ruby is
2 2_000_000 * 2_000_000
3 => 400000000000
```

easy beans

13

higher-order functions





Ceremony is: checked exceptions. Photo credit: http://www.flickr.com/photos/marinegirl/2036373729

closed APIs

closed

17

```
1 // Java (from the Jakarta Commons)
2 public class StringUtils {
3    public static boolean isBlank(String str) {
4    int strLen;
5    if (str == null || (strLen = str.length()) == 0) {
6       return true;
7    }
8    for (int i = 0; i < strLen; i++) {
9       if ((Character.isWhitespace(str.charAt(i)) == false)) {
10         return false;
11    }
12    }
13 }
14</pre>
```

open APIs

groovy metaclass

```
1 // Groovy
2 String.metaClass.isBlank = {
3 length() == 0 || every{ Character.isWhitespace(it.charAt(0)) }
4 }
```

21

22

ruby open class

```
1 # Ruby (from Rails)
2 class String
3  def blank?
4  empty? || strip.empty?
5  end
6 end
```

scala implicit coercion

```
1 // Scala
2 class CharWrapper(ch: Char) {
3   def isWhitespace = Character.isWhitespace(ch)
4 }
5 implicit def charWrapper(ch: Character) = new CharWrapper(ch)
6 class BlankWrapper(s: String) {
7   def isBlank = s.isEmpty || s.forall(ch => ch.isWhitespace)
8 }
9 implicit def stringWrapper(s: String) = new BlankWrapper(s)
```

clojure generic dispatch

DSLs

```
(defn blank? [s]
  (every? #(Character/isWhitespace %) s))
```

ruby (from rails)

```
methods feel
like keywords

class Coach < ActiveRecord::Base
  belongs_to :team
  has_many :sponsors, :as => :spokesperson
  validates_presence_of :first_name, :last_name
end

flexible punctuation
```

groovy

25

```
//from EasyB, www.easyb.org
given "an invalid zip code", {
   invalidzipcode = "22101"

   and

given "the zipcodevalidator is initialized", {
   zipvalidate = new ZipCodeValidator()
}

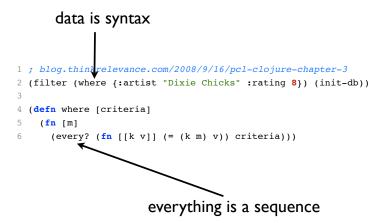
when "validate is invoked", {
   value = zipvalidate.validate(invalidzipcode)
}

then "the validator should return false", {
   ensure(!value)
}

flexible punctuation
```

scala

clojure



29

which language should I learn?

every time you start a green-field Java project, God kills a kitten

clojure considerations

- + functional
- + multimethods
- + concurrency (STM et al)
- + lisp
- + a la carte
- youngest java.next language

groovy considerations

- + easiest to learn
- + easiest bi-di interop
- ? more committed to reusing Java libs
- worst Java baggage
- no concurrency/multicore story

3

ruby considerations

- + biggest community
- + commercial support: Sun Oracle Engine Yard
- + Rails
- + multiple platforms
- no concurrency/multicore story

scala considerations

- + functional
- + high performance
- + pattern matching
- + actor model
- ? hybrid object/functional
- hardest to learn

contact stu

Email: stu@thinkrelevance.com

Office: 919-442-3030

Twitter: twitter.com/stuarthalloway

Talks: http://blog.thinkrelevance.com/talks

Blog: http://blog.thinkrelevance.com

Java.next: http://blog.thinkrelevance.com/2008/9/24/java-next-overview Book: http://www.pragprog.com/titles/shcloj/programming-clojure