

Harnessing Java with Scala

OSCON 2010

July 21, 2010

Thomas Lockney
@tlockney or thomas@lockney.net

Trenton Lipscomb
trentonl@amazon.com

Introduction

Understand the capabilities of sbt and Scala

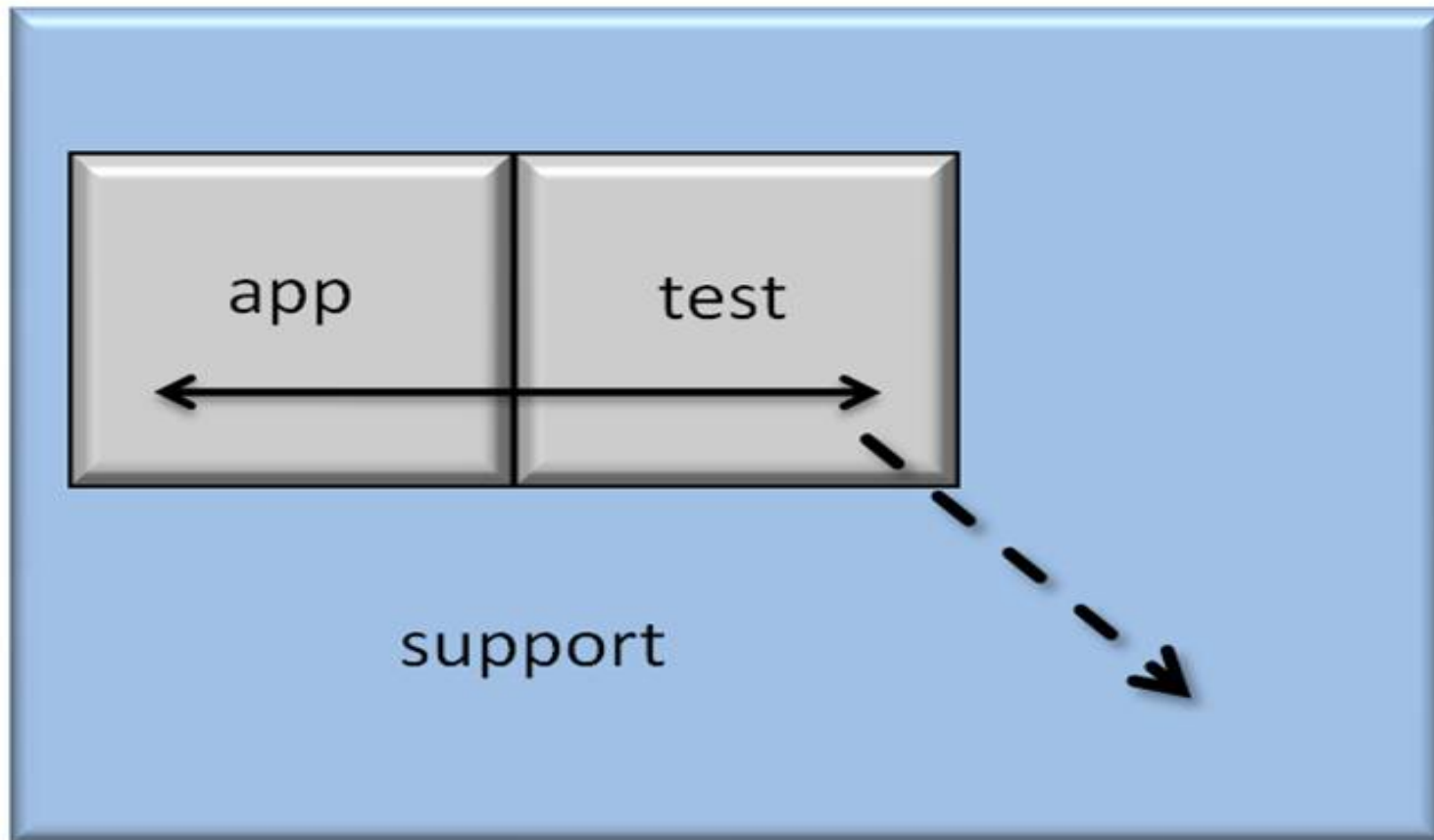
Demonstrate an outside-in adoption of Scala

Have more fun with Java

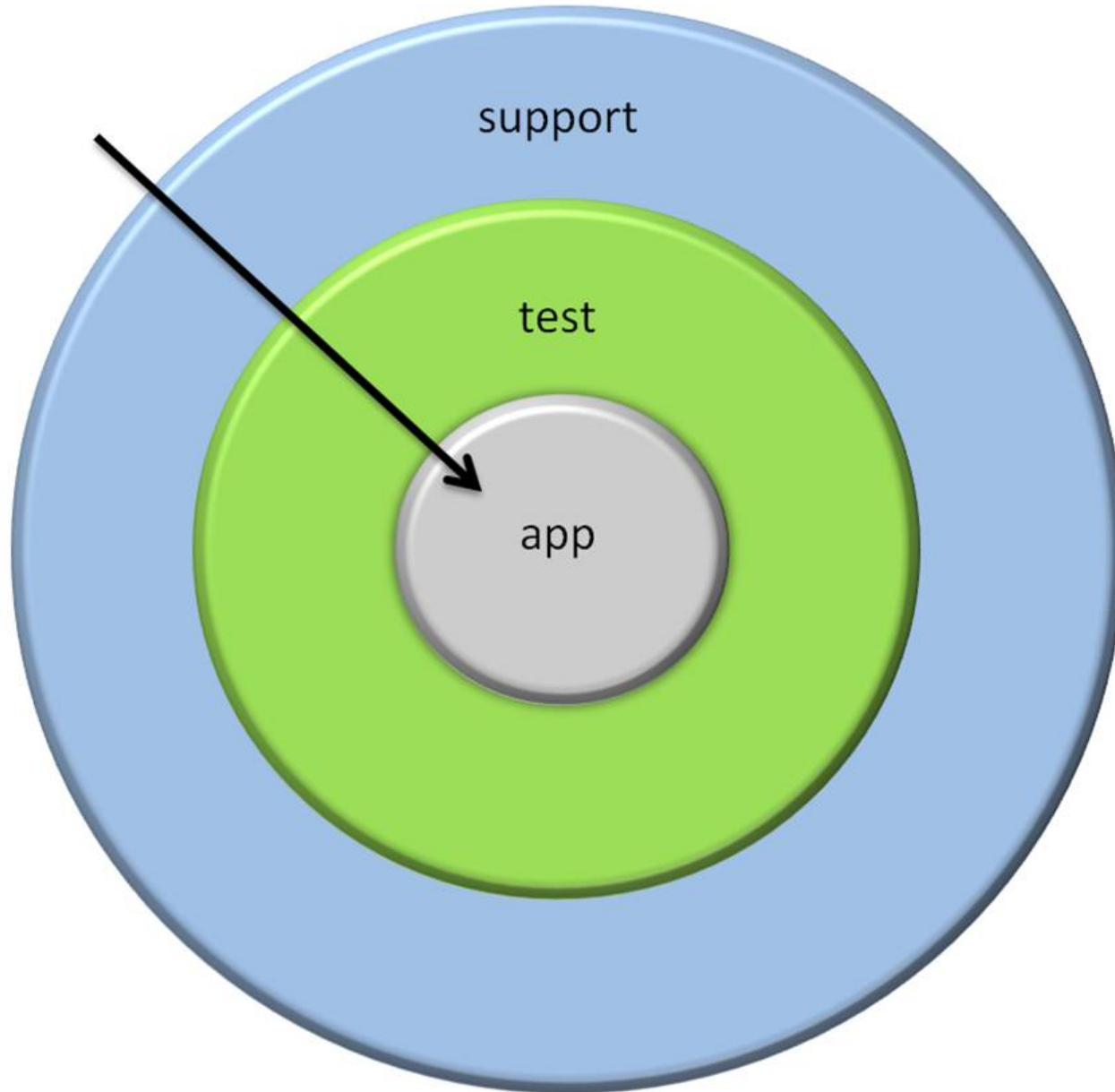
Typical adoption of a new language

Learn app & test code at the same time

Possible use later in a support/one-off code



Outside-in adoption



Scala? Why another language?

Functional / imperative hybrid, with advanced static typing, immutable data structures, implicit conversions

A complete language, with specification

Complies to Java bytecode. Runs on any JVM.

Native interoperability with Java

Scala? Why another language?

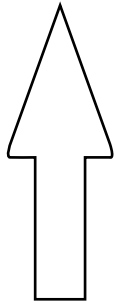
"... if someone had shown me the *Programming in Scala* book ... back in 2003, I'd probably have never created Groovy." James Strachan, July 2009

Designed by Martin Odersky, creator of Pizza (the forerunner of Java generics), author of GJ compiler

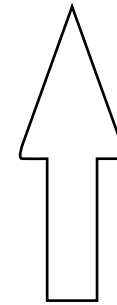
Under active development at Ecole Polytechnique Fédérale de Lausanne (EPFL) and in the open source community

Scala is simple

```
class StringUtils {  
  val vowels = Set('a', 'e', 'i', 'o', 'u')  
  
  def countVowels(in: String): Int = {  
    in.filter(vowels).length  
  }  
}
```



the return value is
the last evaluated
statement



return types are
generally optional

sbt? Why another build system?

"[sbt] has the best dependency management of any build system that works with Scala... plus the continuous compile/test mode. With sbt, it's possible to use emacs to develop Scala code and get quick feedback when changes don't compile." David Pollak, July 2010

Written in Scala, great at building Scala and Java projects

Does NOT require Scala to be installed; will bootstrap

Under active development

Uses Ivy under-the-hood, follows Maven project layout standard

Simpler sbt

Single-command bootstrapping - <http://lockney.net/sbt>

```
wget http://lockney.net/sbt  
less sbt          # if you'd like to  
chmod +x sbt  
./sbt
```

Harnessing a new library

Use case: evaluating a 3rd party library

Build a new project

Get the library

Play with it.

Our examples are a Google search and a Twitter feed pull.

What did we just do?

Iterate through a collection and print

```
resp.getAllHeaders.foreach { println _ }
```

Scala supports XML directly in code

```
val xml = <xml><a><b/></a></xml>  
xml \\ "b"  
res6: scala.xml.NodeSeq = <b></b>
```

What did we just do?

Implicits defs allow type conversions

```
implicit def tuple2NVP(t:Tuple2[String,String]):  
  BasicNameValuePair =  
    new BasicNameValuePair(t._1, t._2)
```

Gives recipe for converting from given type to required type
i.e., Tuple2 -> BasicNameValuePair

Why is this better?

Get going fast: drop jars into a directory and off you go

Incremental compiling

Dynamic classpath generation -- no IDE needed

Use the tools you prefer (Vim, Emacs, Notepad!)

Harnessing a Maven project

Use case: run a maven project

Turn a Maven project into sbt

Run things from sbt console

Why is this better?

Run a class in a project without lots of classpath wrangling

Easily convert Maven projects to sbt

Harnessing an existing project

Use case: one-off use of your code

Application with Hibernate mapped classes and DAOs

Assumption is an existing Hibernate project

Find customers and orders using HQL interactively

What did we just do?

Reuse of large chunks of existing code

```
val emf = Persistence.createEntityManagerFactory(  
    "hibernateExample")  
val em = emf.createEntityManager
```

sbt gives you interactive access to everything

```
// incomplete; see readme for how to exec query...  
val customers = em.createQuery(  
    "FROM Customer c where c.lastOrder > '2010-07-01'")...  
  
printCustomers(customers)
```

Why is this better?

DRY-out your entity query language; why support SQL and HQL?

Think of sbt as a Java/Scala shell: it has command line completion, history

Whole team can use it: developers, QA, analysts

Contribute code back to team, rather than hoard a one-off; copy from console into permanent test case

Team more likely to maintain code in a project, than a one-off in a wiki

Where to go from here?

sbt can build your stuff easily
sbt and Scala get along really well

Scala and Java get along really well

You can adopt Scala or sbt by starting with harnesses, one-off programs, and build systems

No one will see you do it. Everyone will be happier for it

Questions & Answers

S-99: Ninety-nine Scala Problems

<http://aperiodic.net/phil/scala/s-99/>

euler bench

a multi-language implementation of Project Euler

github.com/notbenh/euler_bench

Scala for Java Refugees

<http://www.codecommit.com/blog/scala/roundup-scala-for-java-refugees>

Appreciation & plugs

EPFL

PDXScala

Amazon Web Services

Portland's tech community

Attendees like you

Thomas Lockney <thomas@lockney.net> @tlockney

Trenton Lipscomb <trentonl@amazon.com>

This is really the end.

The Power of Scala

Use case: showboating sbt + Scala

Use HttpClient + the example app

Twilio fancy pants