Automated Testing with Erlang

“These go to eleven”

- **TTY**-who?
- Simple testing with **EUnit**
- Smarter testing with **QuickCheck**
- Heavy lifting with **Common Test**
- Up and running quickly with **Continuous Integration**

Ward Bekker

@wardbekker

TTY

2dehands.be

600k uniques p/day

eu1 the makers channel

Searcle

ADGOJI
MOBILE AUDIENCE TARGETTING

Djuke.it

TTY ❤ ERLANG

Ruby Developer

EUG-NL

ZOTONIC

Simple stuff that works

spilgames

Erlang Experts

maXimonster

INTERACTIVE THINGS

Channel.me

Erlang FACTORY LITE

Conference 16 October 2011

Amsterdam, Netherlands

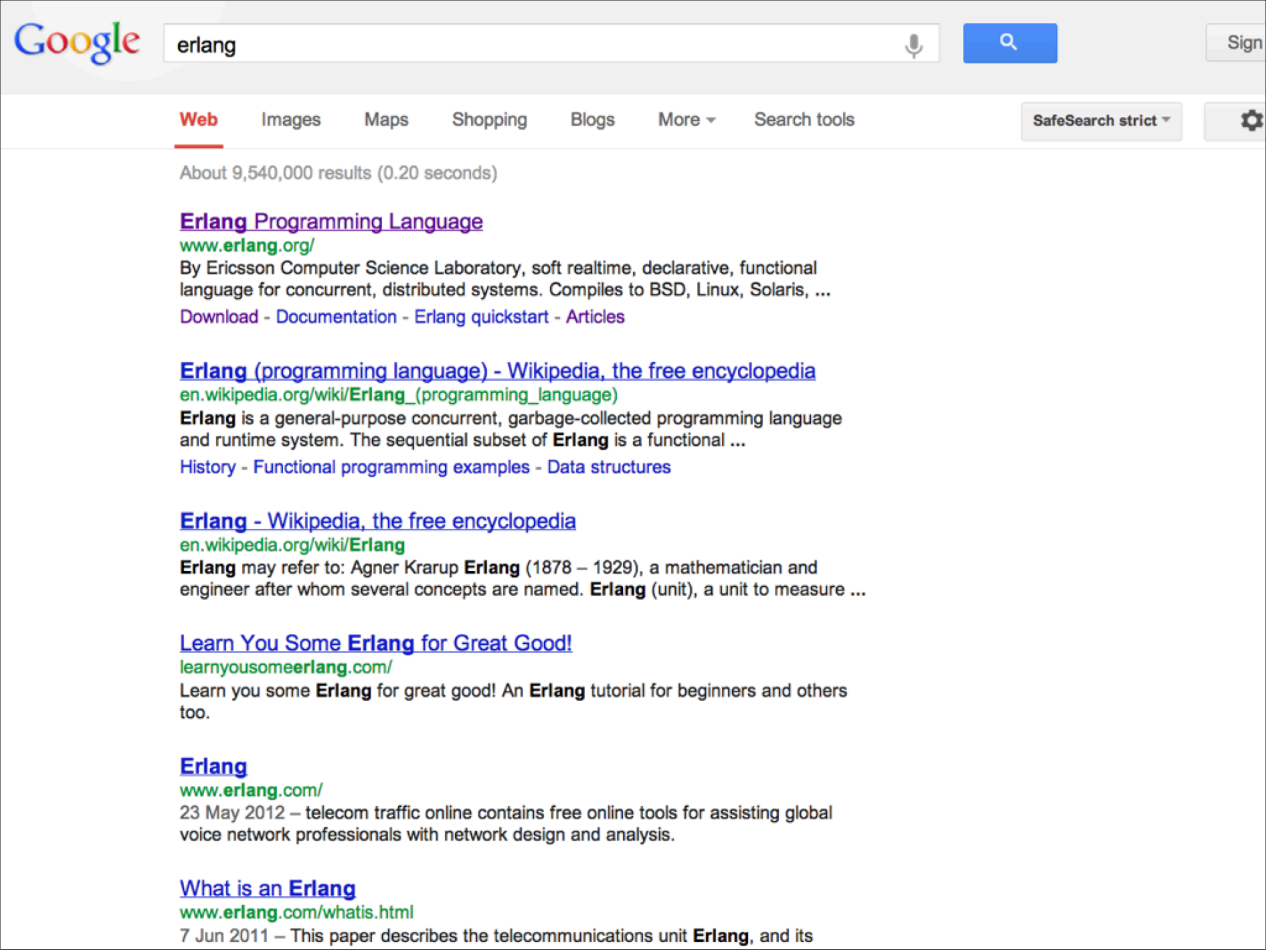
Travis

- **TTY**-who?
- Simple testing with **EUnit**
- Smarter testing with **QuickCheck**
- Heavy lifting with **Common Test**
- **Up and running quickly with Continuous Integration**

```
?assertEqual(
   Expected, Actual
  ).
```

```
?assertEqual(
   2, addition(1,1)
  ).
```
PASSED

```
?assertEqual(
   2, addition(1,2)
  ).
```
FAIL



About 9,540,000 results (0.20 seconds)

Erlang Programming Language
www.**erlang**.org/
By Ericsson Computer Science Laboratory, soft realtime, declarative, functional language for concurrent, distributed systems. Compiles to BSD, Linux, Solaris, ...
Download - Documentation - Erlang quickstart - Articles

Erlang (programming language) - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/**Erlang**_(programming_language)
Erlang is a general-purpose concurrent, garbage-collected programming language and runtime system. The sequential subset of **Erlang** is a functional ...
History - Functional programming examples - Data structures

Erlang - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/**Erlang**
Erlang may refer to: Agner Krarup **Erlang** (1878 – 1929), a mathematician and engineer after whom several concepts are named. **Erlang** (unit), a unit to measure ...

Learn You Some **Erlang** for Great Good!
learnyousome**erlang**.com/
Learn you some **Erlang** for great good! An **Erlang** tutorial for beginners and others too.

Erlang
www.**erlang**.com/
23 May 2012 – telecom traffic online contains free online tools for assisting global voice network professionals with network design and analysis.

What is an **Erlang**
www.**erlang**.com/whatis.html
7 Jun 2011 – This paper describes the telecommunications unit **Erlang**, and its

- Fishing
- Fished
- Fisher
- Fish

abattement	abatt
abbe	abb
abbess	abbess
abbey	abbei
abbeys	abbei
abbominable	abbomin
abbot	abbot
abbots	abbot
abbreviated	abbrevi

```
dict:map(
  fun(Word, Stem) ->
          ?assertEqual(Stem,stem(Word))
  end,
  Vocabulary
 ).
```

```
105..|          m_repl(0, Drow, Mets, <<"suo">>);
     |      step2(Drow = <<"noitazi",Mets/binary>>) ->
   1..|          m_repl(0, Drow, Mets, <<"ezi">>);
     |      step2(Drow = <<"noita",Mets/binary>>) ->
 374..|          m_repl(0, Drow, Mets, <<"eta">>);
     |      step2(Drow = <<"rota",Mets/binary>>) ->
  24..|          m_repl(0, Drow, Mets, <<"eta">>);
     |      step2(Drow = <<"msila",Mets/binary>>) ->
   0..|          m_repl(0, Drow, Mets, <<"la">>);
     |      step2(Drow = <<"ssenevi",Mets/binary>>) ->
   2..|          m_repl(0, Drow, Mets, <<"evi">>);
     |      step2(Drow = <<"ssenluf",Mets/binary>>) ->
  13..|          m_repl(0, Drow, Mets, <<"luf">>);
     |      step2(Drow = <<"ssensuo",Mets/binary>>) ->
  17..|          m_repl(0, Drow, Mets, <<"suo">>);
     |      step2(Drow = <<"itila",Mets/binary>>) ->
  30..|          m_repl(0, Drow, Mets, <<"la">>);
```

Who tests the tester?

```
addition(A, B) ->
    A + B.

?assertEqual(
    4, addition(2,2)
  ).
```

PASSED

```
?assertNotEqual(
    3, addition(1,1)
  ).
```

PASSED

```
addition(_A, _B) ->
    4.

?assertEqual(
   4, addition(2,2)
  ).
```

PASSED

```
?assertNotEqual(
   3, addition(1,1)
  ).
```

PASSED

∞

```
?assertEqual(0, addition(0,0)),
?assertEqual(1, addition(0,1)),
?assertEqual(1, addition(1,0)),
?assertEqual(2, addition(1,1)),
?assertEqual(3, addition(1,2)),
?assertEqual(3, addition(2,1)),
?assertEqual(4, addition(2,2)),
?assertEqual(5, addition(3,2)),
?assertEqual(5, addition(2,3)),
?assertEqual(6, addition(3,3)),
?assertEqual(7, addition(4,3)),
addition(A, B) when A > 999999 ->
    0,
addition(A, B) ->
    A + B.

?assertEqual(
   1000000, addition(999999,1)
  ).
```

FAIL

Who tests the tester?

- **TTY**-who?
- Simple testing with **EUnit**
- Smarter testing with **QuickCheck**
- Heavy lifting with **Common Test**
- Up and running quickly with **Continuous Integration**

Write a program that generates test cases!

Quickcheck

a + b = b + a

```
?FORALL(
   {A, B},
   {int(), int()},
   addition(A,B) == addition(B,A)
  ).
```

$$a + (b + c) = (a + b) + c$$

```
?FORALL(
    {A, B, C},
    {int(), int(), int()},
    addition(A,addition(B,C)) ==
        addition(addition(A,B),C)
  ).
```

$a + 0 = a$

```
?FORALL(
    A,
    int(),
    addition(A,0) == A
  ).
```

-45737,463

-45515,-58059

-44556,74371

-44200,35406

-44088,48783

-43823,2764

-43200,28630

-42844,-51039

-42583,-16465

- Number of tests cases - default 100
- Size

0,216,265,82,157,458,119,28

[1,2,3]

00000000 00000001, 00000000 00000010, 00000000 00000011

1 2 3

Elias γ Encoding

00000000 00000001

00000000 00000010

00000000 00000011

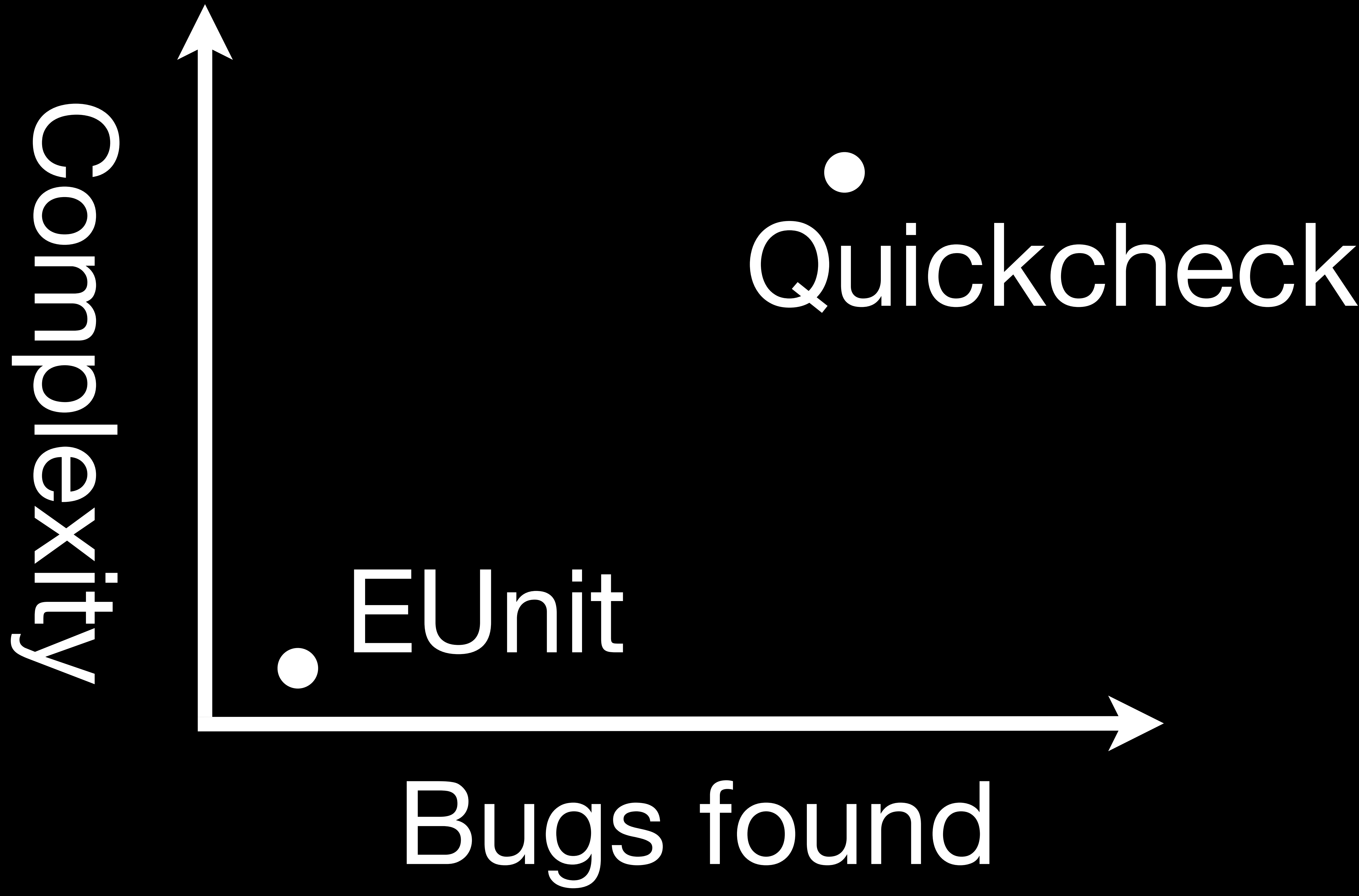
11011

1 010 011

4 = 00100

9 = 0001001

```
?FORALL(
    Xs, list(pos_integer()),
    Xs =:= decode_gamma(encode_gamma(Xs))
   ).
```


Complexity
Quickcheck
EUnit
Bugs found

- **TTY**-who?
- Simple testing with **EUnit**
- Smarter testing with **QuickCheck**
- Heavy lifting with **Common Test**
- Up and running quickly with **Continuous Integration**

Erlang/OTP CT > 6 hours

test_add_delete_index

test_add_delete_document

test_queries

test_redundancy

test_replication

test_redundancy

test_replication

test_queries

test_add_delete_document

test_add_delete_index

test_redundancy

test_add_delete_document

test_replication

test_add_delete_index

test_queries

test_add_delete_index

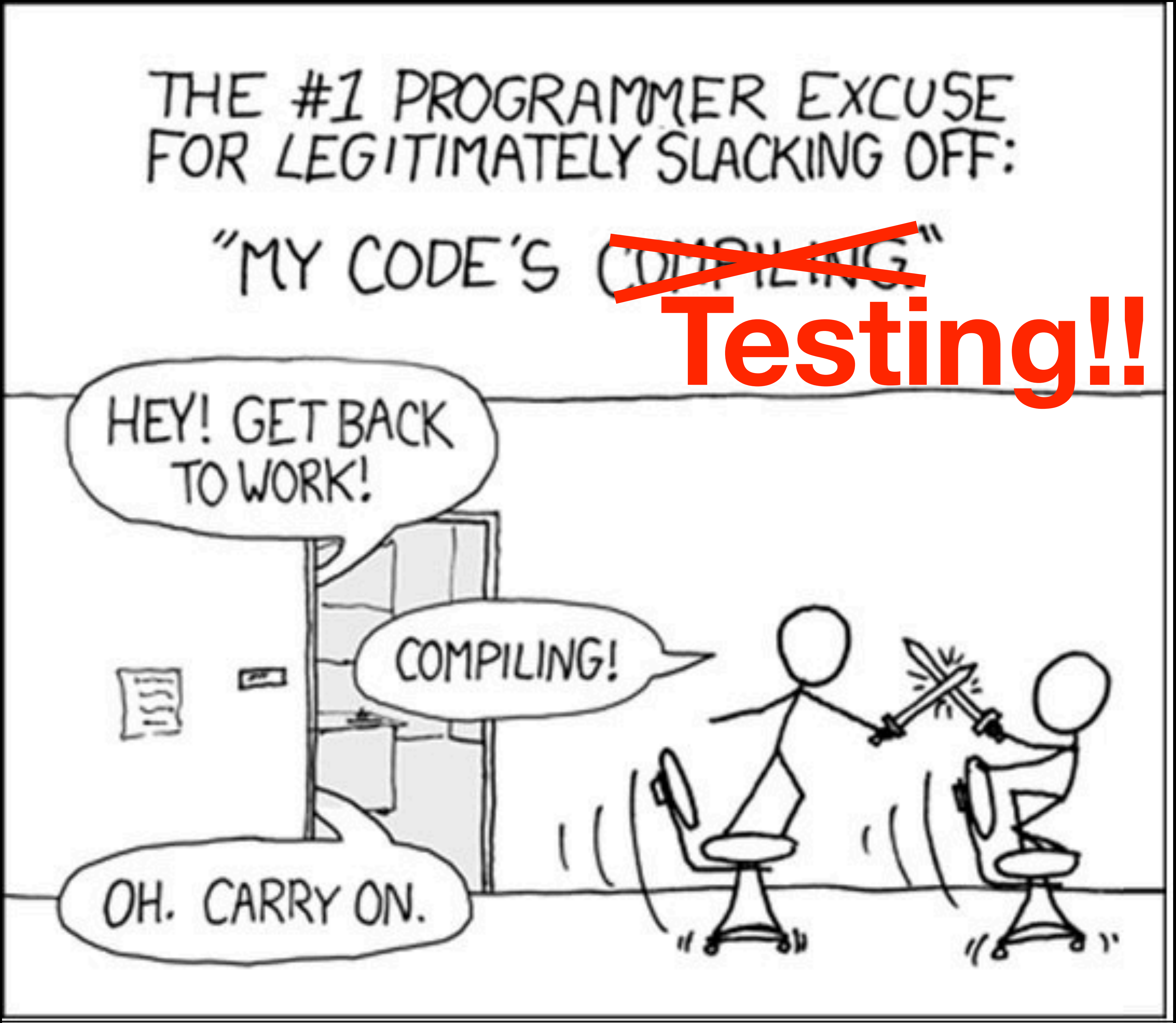
test_add_delete_document

test_queries

test_redundancy

test_replication

- **TTY**-who?
- Simple testing with **EUnit**
- Smarter testing with **QuickCheck**
- Heavy lifting with **Common Test**
- Up and running quickly with **Continuous Integration**


THE #1 PROGRAMMER EXCUSE FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S ~~COMPILING~~"
Testing!!
HEY! GET BACK TO WORK!
COMPILING!
OH. CARRY ON.

Jenkins

Travis

Travis

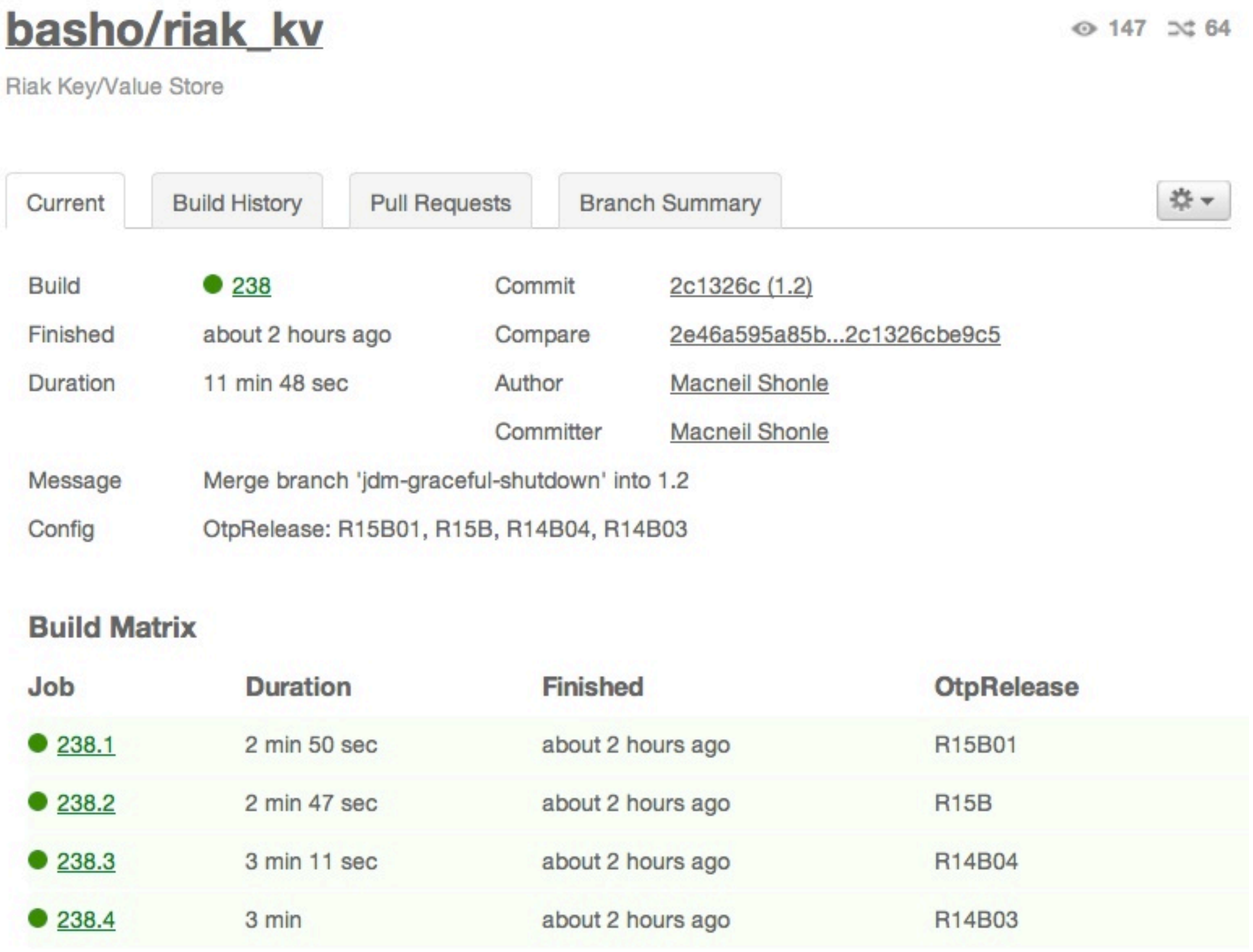
basho

mochi

Apache CouchDB

.travis.yml

```yaml
language: erlang
notifications:
  email: ward@tty.nl
otp_release:
  - R15B01
  - R15B
  - R14B04
  - R14B03
```



Riak Key/Value Store

Build	● 238	Commit	2c1326c (1.2)
Finished	about 2 hours ago	Compare	2e46a595a85b...2c1326cbe9c5
Duration	11 min 48 sec	Author	Macneil Shonle
		Committer	Macneil Shonle
Message	Merge branch 'jdm-graceful-shutdown' into 1.2		
Config	OtpRelease: R15B01, R15B, R14B04, R14B03		

Build Matrix

Job	Duration	Finished	OtpRelease
● 238.1	2 min 50 sec	about 2 hours ago	R15B01
● 238.2	2 min 47 sec	about 2 hours ago	R15B
● 238.3	3 min 11 sec	about 2 hours ago	R14B04
● 238.4	3 min	about 2 hours ago	R14B03

Build Environment

- 32-bit Ubuntu Linux 12.04
- R14B02 upto R15B02 with Kerl
- Rebar
- Riak, MongoDB, MySQL, PostgreSQL
- RabbitMQ, Memcached, Cassandra etc

- Simple testing with **EUnit**
- Smarter testing with **QuickCheck**
- Heavy lifting with **Common Test**
- Up and running quickly with **Continuous Integration**