The Immaturity of Testing

DRW TRADING GROUP

Jay Fields

developer testing is immature in almost every aspect

DRW TRADING GROUP

Jay Fields

External Tests

Unit Tests

Smoke Tests

Functional Tests

Integration Tests

Acceptance Tests


Finstown
Kirkwall
A966
Twatt
FAIL
failblog.org

second class citizens

attempt to replace exploratory testing

testing everything

DRW TRADING GROUP

Jay Fields

custom test cases

Jay Fields

slow

DRW TRADING GROUP

Jay Fields

100% test coverage is not the right goal

test the parts of the system that provide the most value

confidence to refactor

Jay Fields

shield you from regression bugs

remove pain from testing

Jay Fields

remove technical debt

Jay Fields

immaturity of selenium

DRW TRADING GROUP

Jay Fields

```ruby
def setup
   @browser = Selenium::SeleniumDriver.new("localhost", 4444,
     "*firefox", "http://www.google.com", 15000)
   @browser.start
 end

 def test_google
   @browser.open("http://www.google.com/webhp?hl=en")
   @browser.type("q", "hello world")
   @browser.click("btnG")
   @browser.wait_for_page_to_load("5000")
   assert_equal("hello world - Google Search",
     @browser.get_title())
end

def teardown
   @browser.stop
end
```


1-2-3-Fail
failblog.org

15 different ways to drive selenium

Jay Fields

15 different languages to write tests in

Jay Fields

so..... painfully..... slow

large suites become unmaintainable

buggy

poor defect localization

DRW TRADING GROUP

Jay Fields

cross browser verification

DRW TRADING GROUP

Jay Fields

runs against full stack

easy to use

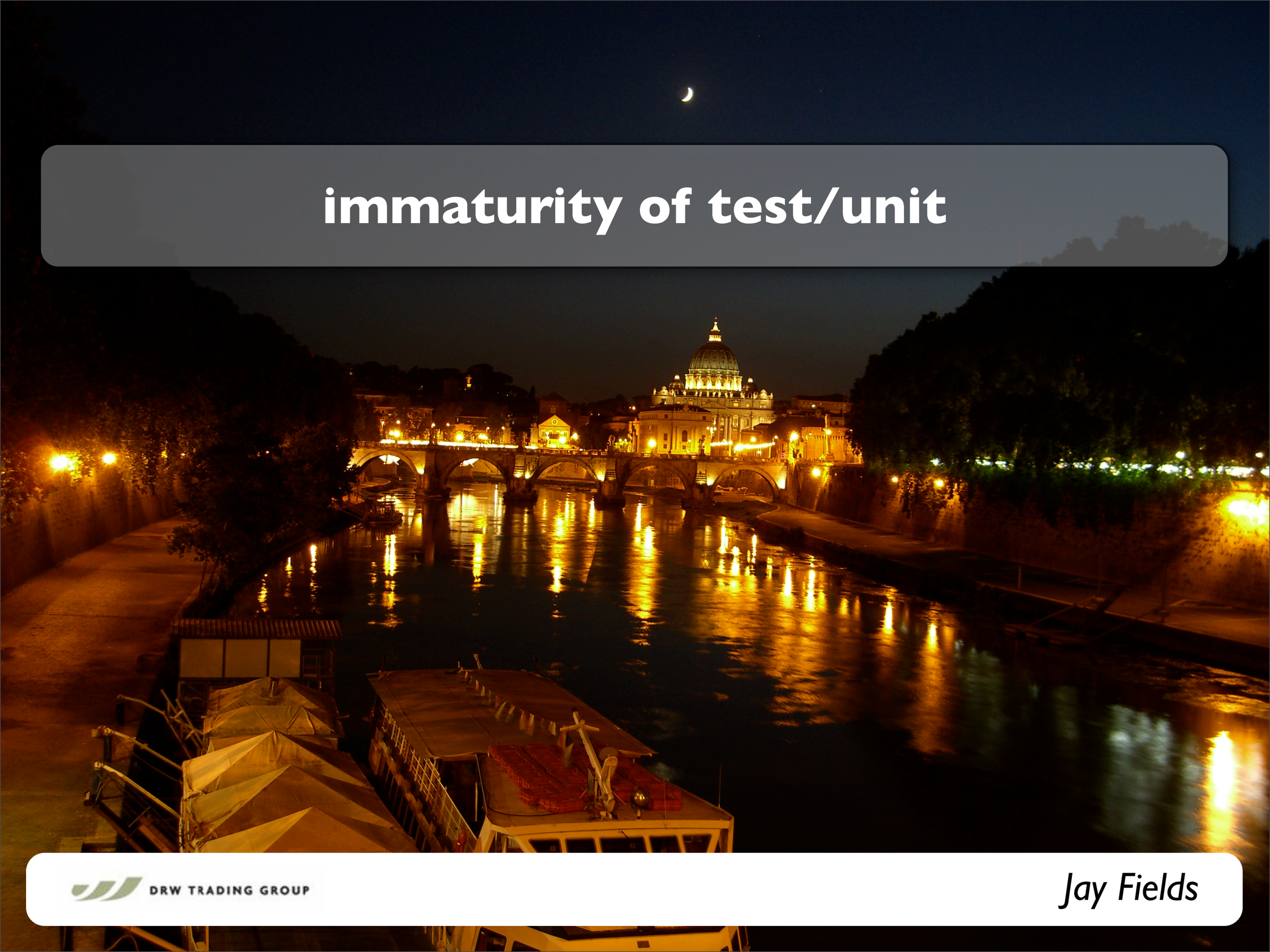
has a recorder

tester friendly

possibly the best solution for smoke testing

Jay Fields

immaturity of test/unit



DRW TRADING GROUP

Jay Fields

```ruby
require 'test/unit'

class BowlingTests < Test::Unit::TestCase
  def setup
    @bowling = Bowling.new
  end

  def test_score_0_for_gutter_game
    20.times { @bowling.hit(0) }
    assert_equal 0, @bowling.score
  end
end
```


HIDING PLACE
FAIL!
failblog.org

tester unfriendly

object oriented thus creating a paradigm mismatch

too granular, no integration verification

syntax is ugly

no mocking framework built in

DRW TRADING GROUP

Jay Fields

very developer friendly

granular defect localization

Jay Fields

easy to write

runs quickly

immaturity of RSpec

DRW TRADING GROUP

Jay Fields

```ruby
describe Bowling do
  before(:each) do
    @bowling = Bowling.new
  end

  it "should score 0 for gutter game" do
    20.times { @bowling.hit(0) }
    @bowling.score.should == 0
  end
end
```


FAIL
VIOLATORS
WILL BE TOWED
AND FIND $50.
failblog.org

too much magic

too based on english

consistent to hamcrest, not pols

not easily extendable

DRW TRADING GROUP

Jay Fields

encourages behavior testing

free documentation generation

Jay Fields

built in mocking

Jay Fields

immaturity of Synthesis

DRW TRADING GROUP

Jay Fields

```ruby
class PostOffice
  def send(object)
    # do stuff
  end

class Merchant
  def sell(object)
    # other stuff
    post_office.send(object)
  end
```

```ruby
class PostOfficeTests
  def test_send
    # setup stuff
    post_office.send(object)
  end

class MerchantTests
  def test_sell
    # setup stuff
    post_office.expects(:send).with(object)
  end
```


FAIL
fail blog.org

runs the test suite twice

requires too much specification / unnecessary tests

largely unknown

DRW TRADING GROUP

Jay Fields

fast

encourages decoupled objects

confidence without integration testing

Jay Fields

immaturity of Expectations

DRW TRADING GROUP

Jay Fields

```ruby
expect 2 do
  1 + 1
end

expect stub.to.receive(:dial).with("2125551212") do |phone|
  phone.dial("2125551212")
end

expect 1..5 do
  3
end

expect Process.new.to.have.finished do |process|
  process.finished = true
end
```


Keep to
Right
Direction FAIL
failblog.org

displays how, but not necessarily why

no test monikers

no free documentation generation

DRW TRADING GROUP

Jay Fields

very concise

encourages desired duplication

encourages granular testing

promotes literals as expected values

universal assertion syntax

Jay Fields

encourages expressive software design

Jay Fields

questions?

what do I do?

DRW TRADING GROUP

Jay Fields

unit testing: expectations

Jay Fields

functional testing: rspec

Jay Fields

smoke testing: selenium

DRW TRADING GROUP

Jay Fields

experiment. do what works well for you.

Jay Fields


obrigado,
jay@
jayfields
.com
DRW TRADING GROUP
Jay Fields