



TDC2013

the developer's conference



**Lições do Futuro:
O que eu queria saber há alguns anos atrás
sobre como manter uma suíte de testes**



 **@ederign**



 **@flsusp**



dextra

CRAFTING
SOFTWARE

transforming

BUSINESS

Confidence

Câmbio

***Maior grupo financeiro de
câmbio do Brasil***



+120 agências
adm de cartões e banco



+1.000.000
clientes por ano


13 em cada 100
dolares que saíram do Brasil

...Confidence

busca empresa para desenvolver sua
nova **plataforma operacional**

*coding your **business***



A person wearing a dark jacket and jeans is holding a large white sign with both hands. The sign features a motivational message in a bold, black, sans-serif font. The background consists of a red brick wall and a dark grey or black door or wall section.

**IF YOU WORK
REALLY HARD AND
ARE KIND, AMAZING
THINGS WILL HAPPEN.**



quinta-feira, 11 de julho de 2013

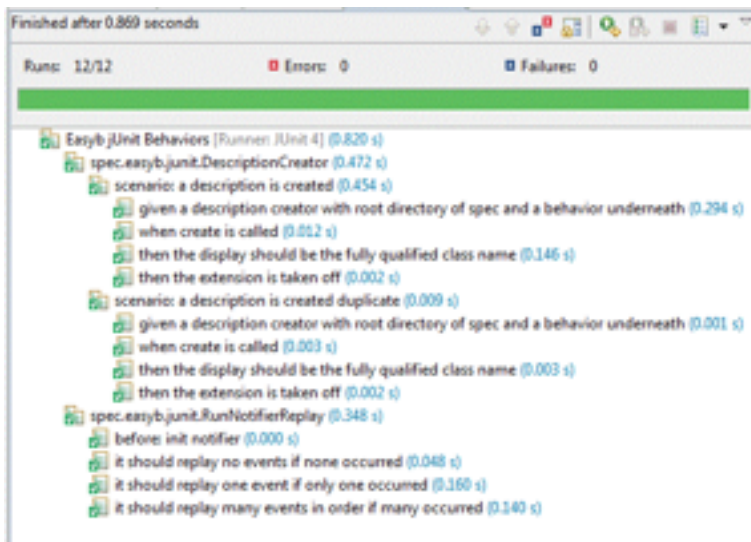
Qualidade não é negociável!!!

Você tem o direito de fazer certo!



Testes Unitários

Testes de Integração



BDD

Desenvolvimento de Framework Próprio



Testes Funcionais



Abandono Testes Integração



BDD + Testes Funcionais

```
Finished after 0.000 seconds
Runs: 12/12      Errors: 0      Failures: 0

jruby_junit Behaviors (Runned: 3/14) (0.000 s)
  spec ruby_junit.DescriptionCreator (0.472 s)
    scenario: a description is created (0.434 s)
      given a description creator with root directory of spec and a behavior underneath (0.204 s)
      when create is called (0.012 s)
      then the display should be the fully qualified class name (0.140 s)
      then the extension is taken off (0.062 s)
    scenario: a description is created duplicate (0.000 s)
      given a description creator with root directory of spec and a behavior underneath (0.000 s)
      when create is called (0.003 s)
      then the display should be the fully qualified class name (0.003 s)
      then the extension is taken off (0.002 s)
  spec ruby_junit.RunNotifiedReply (0.340 s)
    before: init notifier (0.000 s)
    it should reply no events if none occurred (0.040 s)
    it should reply one event if only one occurred (0.140 s)
    it should reply many events in order if many occurred (0.140 s)
```

+





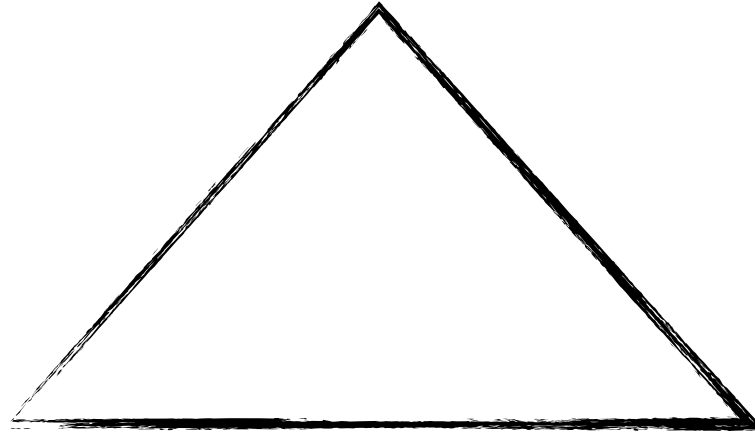
Alta taxa de Cobertura

Cliente Feliz

4 anos de projeto (~12 pessoas)



Parceria TI e Business



**Planejamento
focado na
mitigação de
riscos**

**Arquitetura
Sólida e
escalável**

Deploy ~1x/Semana Plataforma Bancária



“Continuous Delivery”

#osDevPira



memegenerator.net

Números do Projeto

~1 Release por Semana

1.6k Testes de Aceitação

160k Operações/mês

400k Requests/dia

1k Usuários em dias normais

17 Nós no Cluster

Explosão na Quantidade de Testes Funcionais



Lentidão na execução da suíte de testes



Time Triste



Onde erramos?

Como saber se minha suíte de testes não está indo para o buraco?



Bugs em Produção

Seus testes não passam?

Você não tem testes?



Bugs em Produção



Seus testes passam?

*Você não está testando a
funcionalidade certa!*

*Você não está testando o
sistema correto*

Teste quebra



Em produção funciona?

Você não está testando o sistema certo!

Funcionalidade é alterada

Seu teste passa sem precisar de alteração?

Você não está testando a funcionalidade correta!



Você corrige um defeito

Seu teste passa a quebrar?

*Você não está focando seus testes na funcionalidade
e sim no sistema*



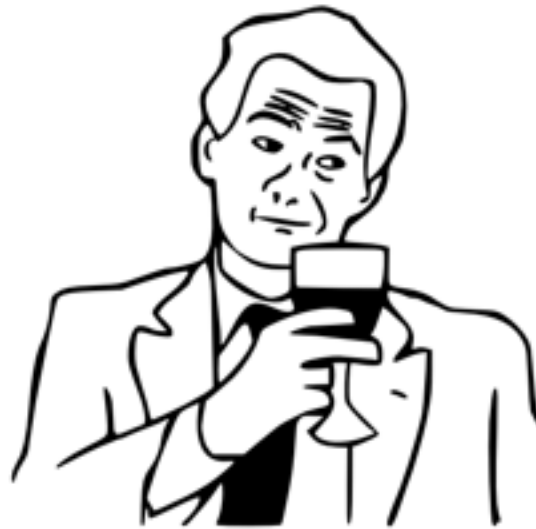
Você altera o sistema

Não tem feedback rápido dos testes?



*Você tem um problema arquitetural nos
seus testes!*

*Você não vai rodar a sua suíte!
(sob pressão)*



TRUE STORY

We know stress

Você faz uma pequena alteração no sistema

*É difícil identificar um teste que possa ser rodado
agora?*

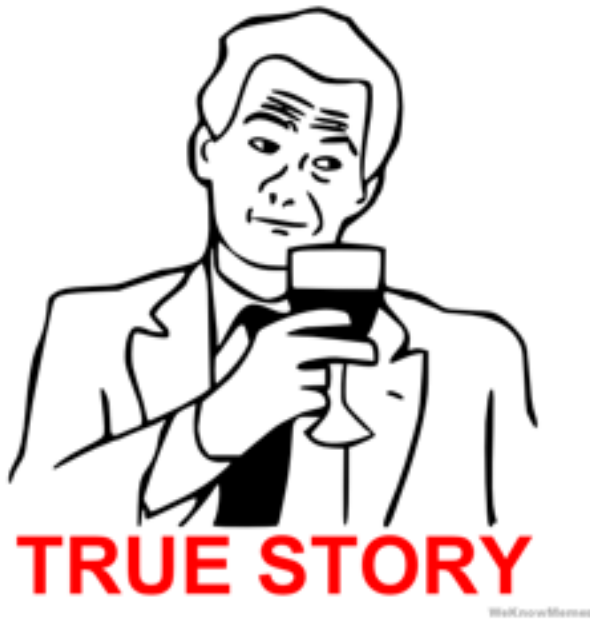
É difícil identificar qual teste foi afetado?

*Você não vai testar a funcionalidade e nem rodar a
suíte completa porque ela é lenta*

Você vai implantar com a suíte quebrada



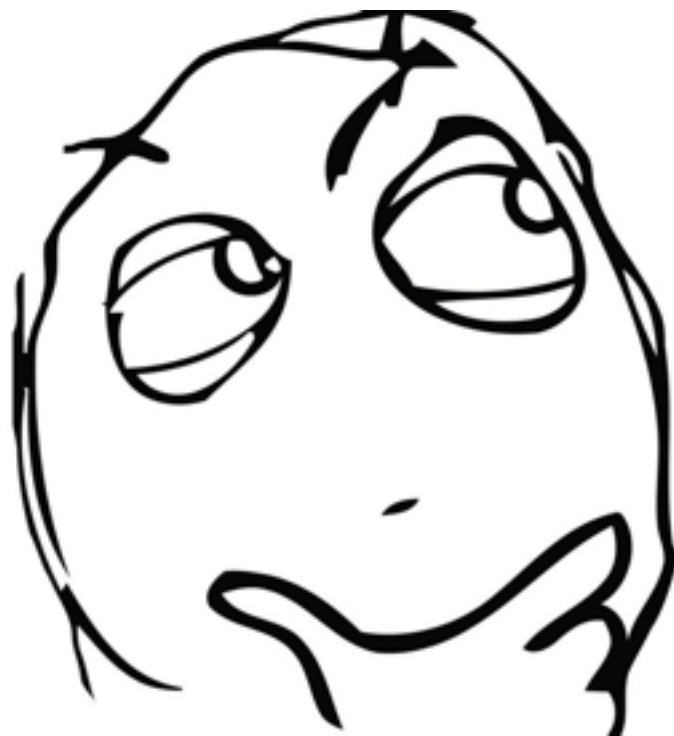
Pequena alteração, muitos testes quebram?



*Você vai demorar para
identificar o que quebrou*

*Você vai implantar em
produção com a sua suíte
quebrada*

Mas quais são as características de uma boa suíte de testes?



Uma boa suíte de testes é qualquer suíte de testes que não tenha os problemas apresentados!



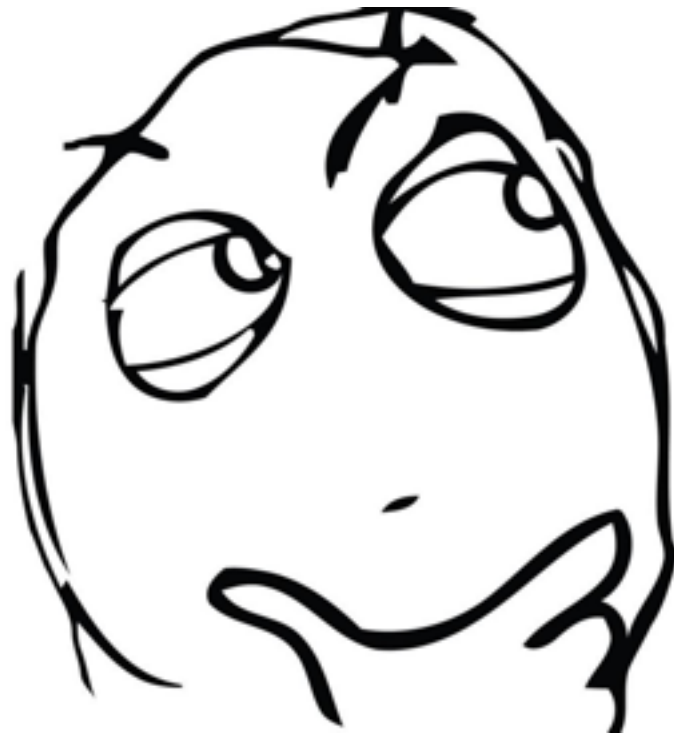


Mas e se eu tivesse uma máquina do tempo?

Não existe bala de prata



***Mas o que talvez
(provavelmente)
funcione?***



3 F's - Corey Haines

Fast

Ciclo TDD - < 1 min

Ciclo Integração - rápido o suficiente

Suíte Lenta == Builds Quebradas

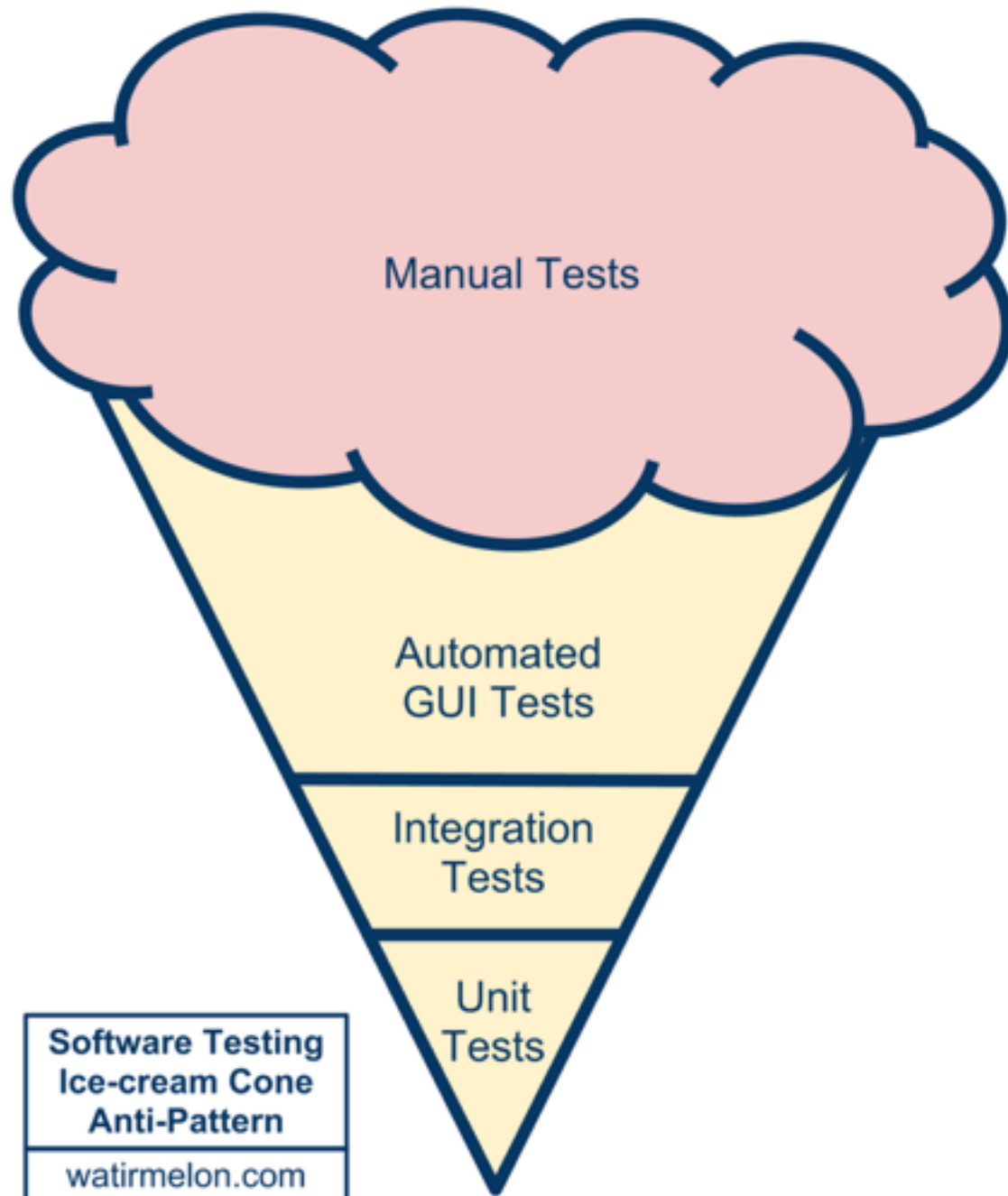
Focused

Um Teste = Uma Razão pra Mudança

Feedback Rápido

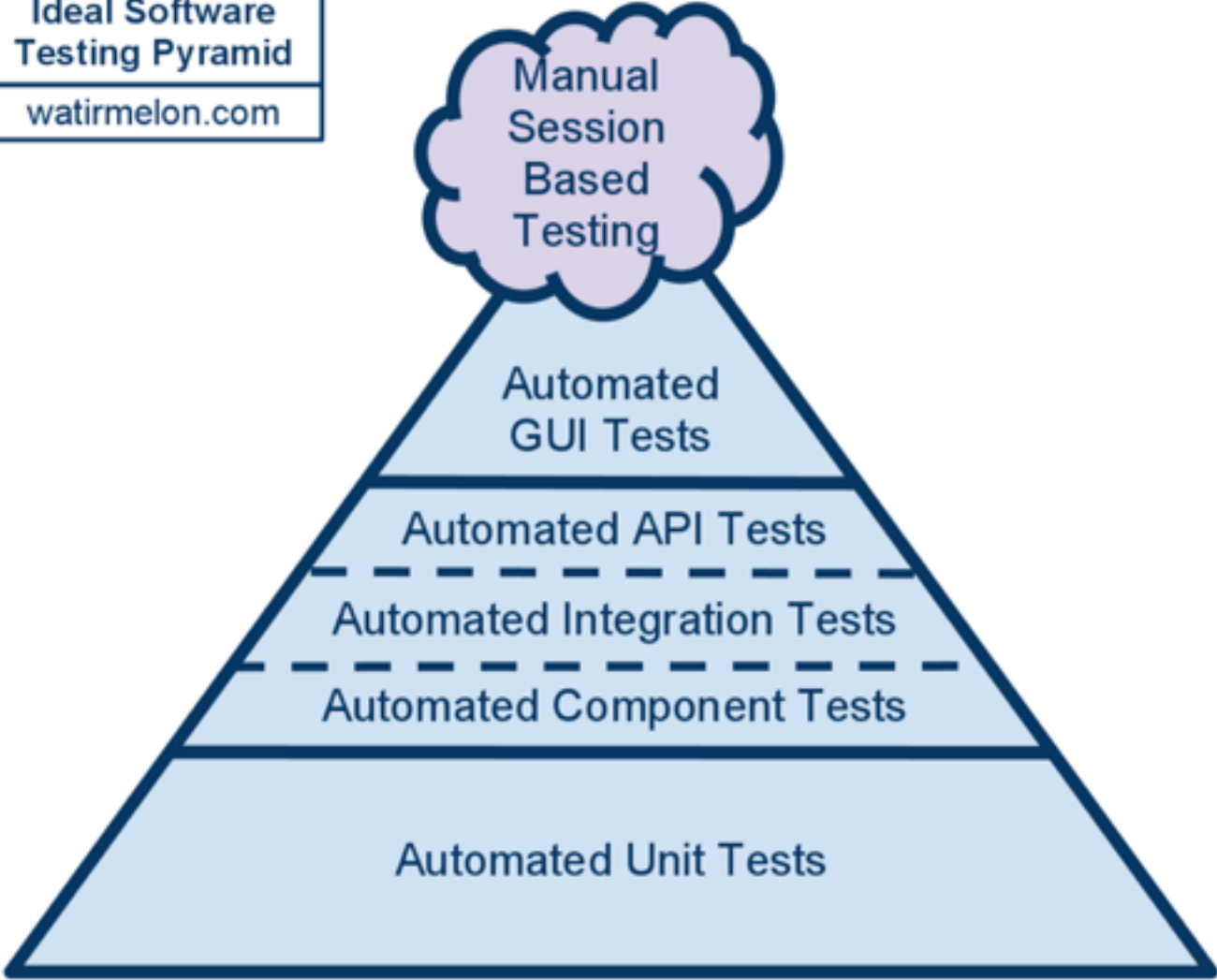
Full

Cobrir o maior % possível do sistema



Software Testing
Ice-cream Cone
Anti-Pattern
watirmelon.com

Ideal Software
Testing Pyramid
watirmelon.com



Código de teste é tão importante quanto código de produção

Testes bem escritos e em grupos pequenos no mesmo contexto

Toda a equipe deve escrever a suíte de testes



Consistente e Confiável

A sua suíte de testes deve conquistar a confiança da sua equipe, do seu gerente e do cliente



“Isolada”

Um teste não pode influenciar o resultado de outro

Feedback Rápido



Duráveis

Mantém seu valor durante toda a vida do produto



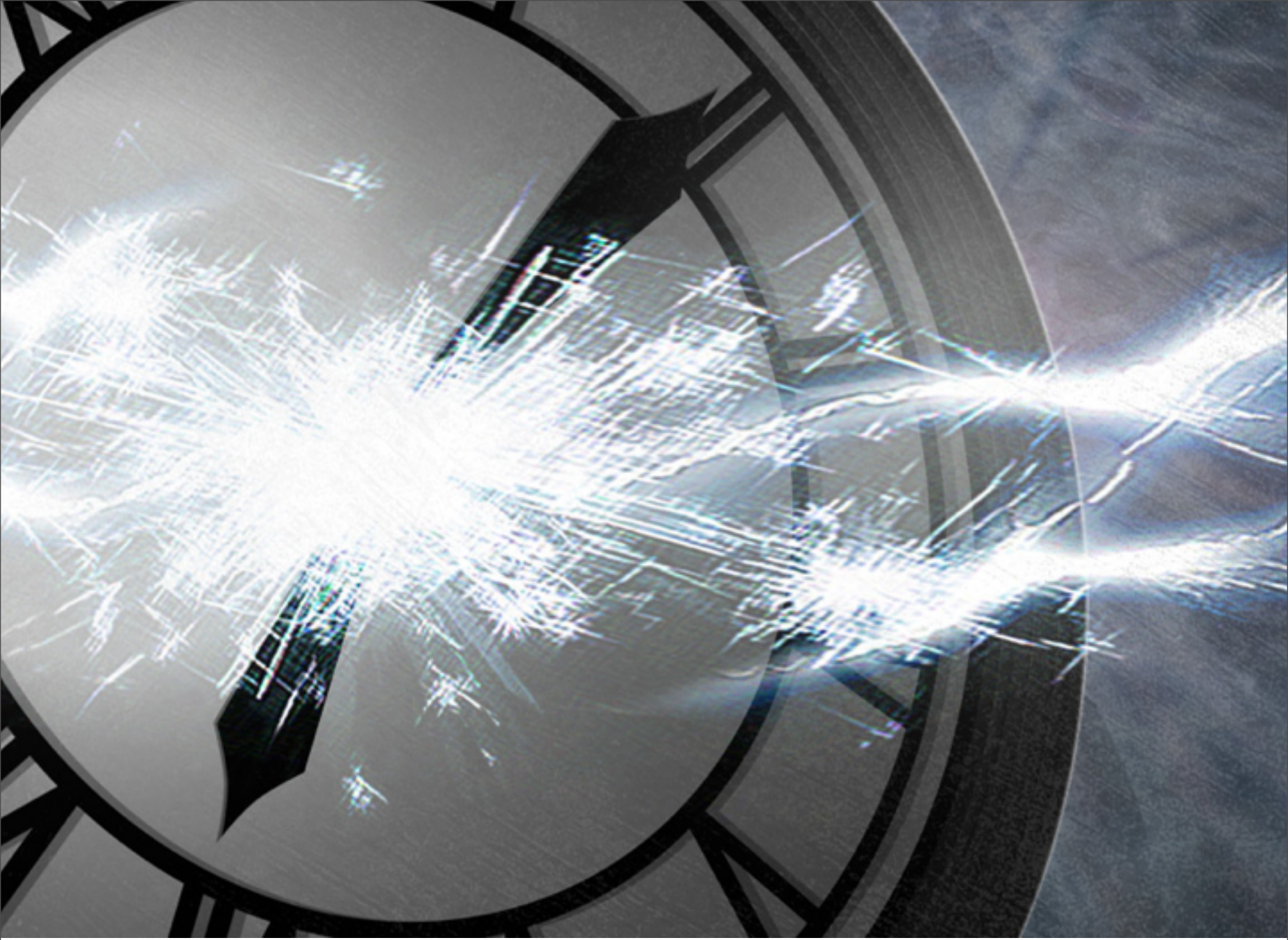
Documentação



*Testes servem como documentação
(são a melhor documentação que
você vai ter)*

*Documentação viva, especificação
executável.*

*Testes devem informar a intenção de
um desenvolvedor*



quinta-feira, 11 de julho de 2013

Opção 1

*Máquina gigante na Amazon, rodando
Windows ou Linux com Xvfb +
Mycontainer*

Servidor dedicado no dev

Pros: *Setup parecido*

Cons: *\$, difícil de debugar (aws)*





Opção II

myContainer Grid

Pros: Desenvolvido para paralelismo

Cons: Difícil de adaptar para paralelismo, difícil de adaptar meu projeto, projeto em construção

<https://github.com/feroult/mycontainer-grid>



Opção III

Selenium Grid

Pros: Fácil e estável

Cons: Difícil paralelizar na mesma máquina (ex. BD)



Opção IV

CloudBees + Sauce Labs

Pros: *Cloud, ;)*

Cons: *Código de produção em servidor externo*

@leguimas <- pode te ajudar



quinta-feira, 11 de julho de 2013



Foque a cobertura nas funcionalidades que mudam com maior frequência

Faça testes para os bugs em produção - você automaticamente estará testando o mais crítico



Não deixe o teste para o final

Não rode a suíte somente antes de gerar a versão

Faça integração contínua

Não se preocupe com a performance de execução de um teste, mas sim, com o paralelismo e concorrência.



Mas o que é mais importante?

Possuir uma suíte de testes



Saber se sua suíte está indo para o buraco





TDC2013

the developer's conference



@ederign



@flsusp