

Histoire



Comment j'ai commencé BDD

Pourquoi je fais du BDD

2005

Test After Development

Un test est mieux que pas de test...

Tests d'intégration majoritairement

2006 : Test d'intégration ?

Lent : troooooop lent

Mauvais feedback

Fragilité : Plein de rouge d'un coup

Obscur : Qu'est-ce qui n'est pas bon ?

Ca veut dire quoi, lent ?

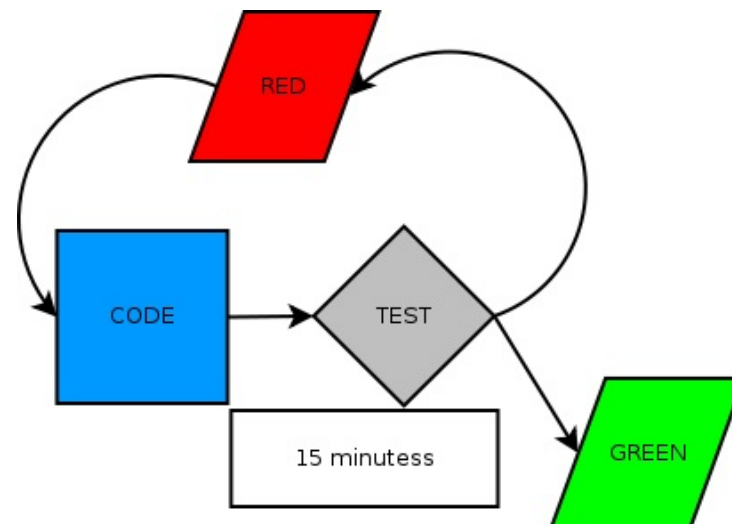
On ne s'améliore pas si on ne mesure pas !



Marie Pia Ignace, Présidente Lean Institute France

Lent = 9..18 minutes

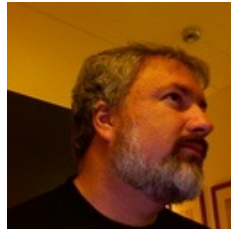
Mauvais workflow !



**Comment avoir plus de
robustesse et de précision ?**

C'est quoi un test unitaire

[A Set of Unit Test Rule:](#)



Michael Feathers, @mfeathers

Un test n'est pas unitaire si

...

il touche la base

il touche le réseau

il touche un fichier

il a des effets de bords sur les autres tests lorsqu'on les lance tous

on a besoin de modifier une configuration pour le lancer

2007 : Tests plus rapides

Les tests ne partent plus sur le réseau
(jmock, easymock, ...)

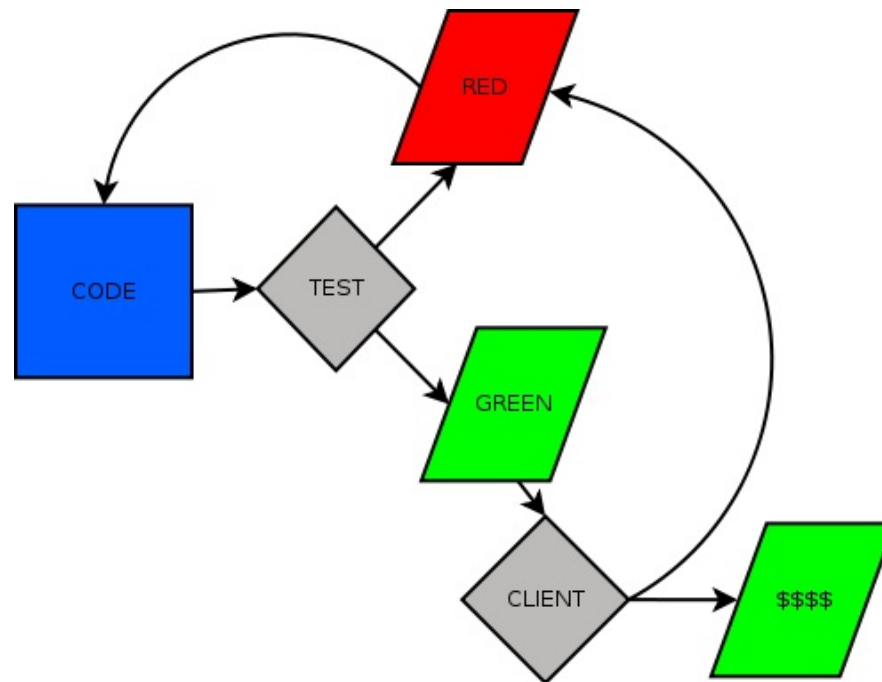
Temps de test réduit (2..3 minutes)

Cool !

2007 : Et le client, il est content ?

Pas toujours !!

Mauvais workflow !



Pourquoi est-ce que le client
n'est pas content ?

2007 : Card, Conversation, Confirmation



Ron Jeffries, @ronjeffries, 2001

Card

prétexte

Conversation



Confirmation

le test d'acceptance

2007 : Test Driven Development

J'ai du rater quelque chose

Bof ...

#fail

TDD est dur

par QUOI je commence ?

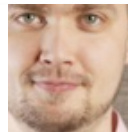
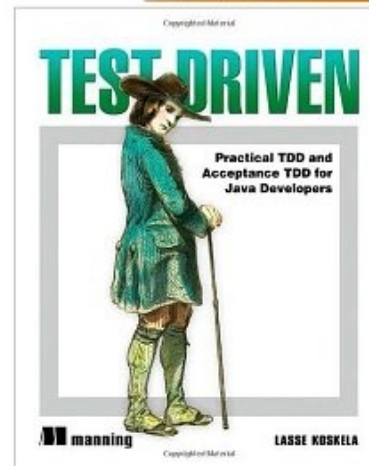
et par OU ?

QUAND est-ce que j'arrête ?

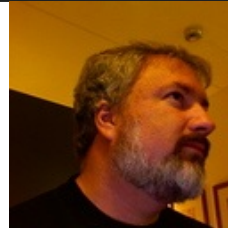
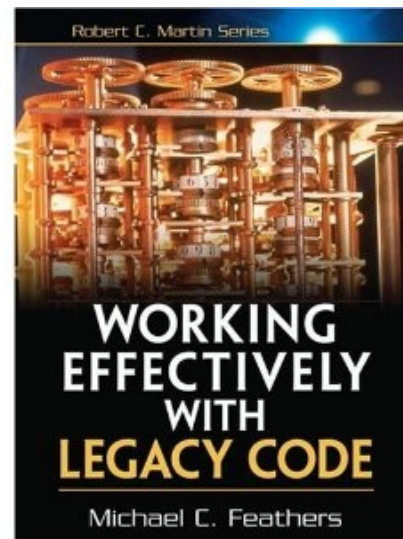
Aide

Livres

Cliquez pour **Feuilleter!**



Lasse Koskela, @lassekoskela



Michael Feathers, @mfeathers

2008 : Test Driven Development

(le retour)

Aha

Introducing BDD

As an example, when I was first getting to grips with TDD, I was pairing with an experienced agile coach, writing little test methods, then writing the code, and generally feeling good about life.

Then I went ahead and wrote some code without a test.

The coach, JR, asked me why I'd written the code. I answered: "we'll need it in a minute", to which JR replied "**yes, we might**".

By using the word **might**, he introduced the possibility that we **might not**. As it turned out, we didn't. - Dan North

<http://behaviour-driven.org/GettingTheWordsRight>

Introducing BDD

I decided it **must** be possible to present TDD in a way that gets straight to the good stuff and avoids all the pitfalls

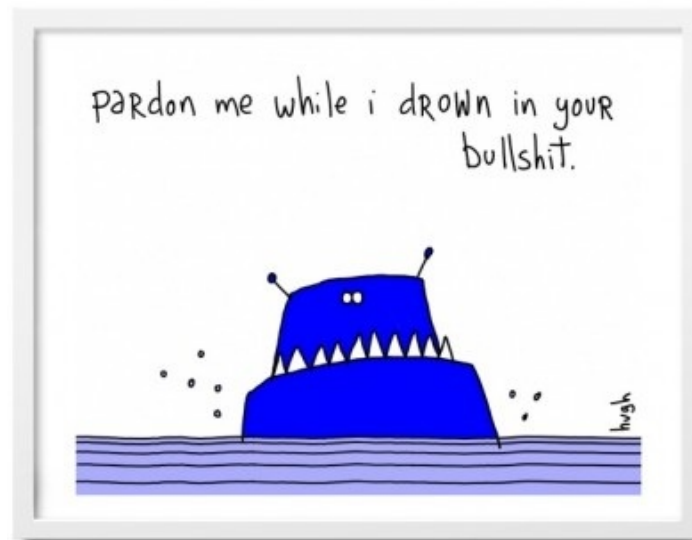
[Introducing BDD, 2006](#)



Dan North, @tastapod

Et Vous ?

Pas trop perdu ?



Jeu

4 questions

trouvez avec votre voisin la bonne réponse

timebox : 1 minute

aide : réponse la plus simple et votre voisin
vous demande plusieurs fois pourquoi

1 - Pourquoi fait-on des tests ?

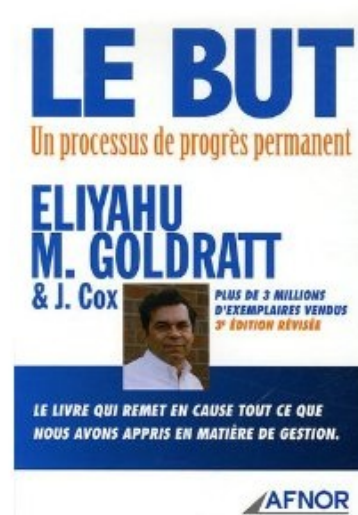
2 - Le client a-t-il plus confiance si le test est fait avant ou après ?

3 - Quel est le but de TDD ?

4 - Quel est le but de BDD ?

1 - Pourquoi fait-on des tests ?

Pour gagner de l'argent



Eli Goldratt (1947 - 11/06/2011) <http://www.eligoldratt.com/messages/> <http://kevinrutherford.posterous.com/rip-eli-goldratt>

**2 - Le client a-t-il plus
confiance si le test est fait
avant ou après ?**

Pareil

Ce qui compte est que le produit soit testé
:)

3 - Quel est le but de TDD ?

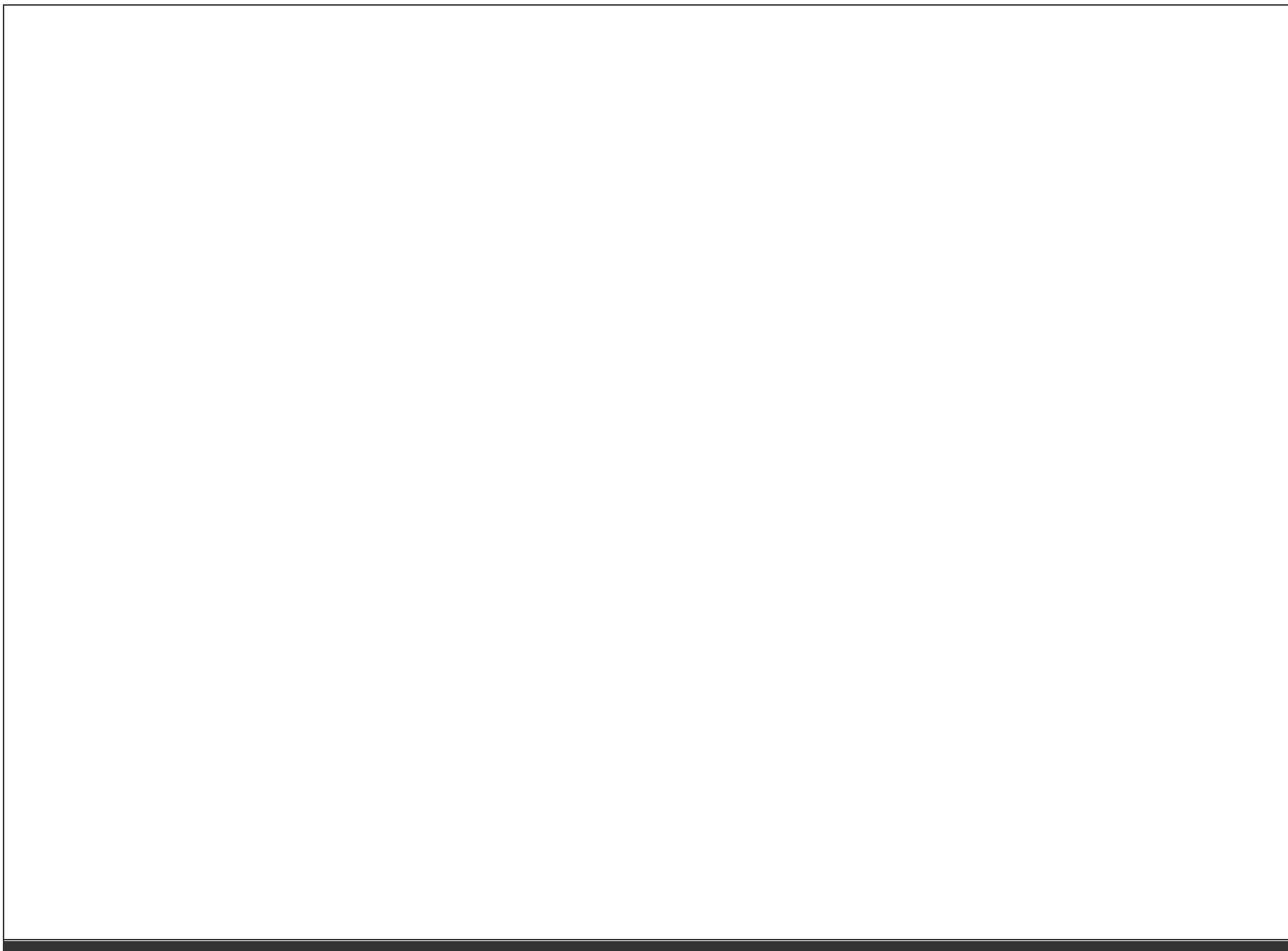
le design

Le test est un effet de bord de TDD

4 - Quel est le but de BDD ?

On va voir

:)



Introduction BDD

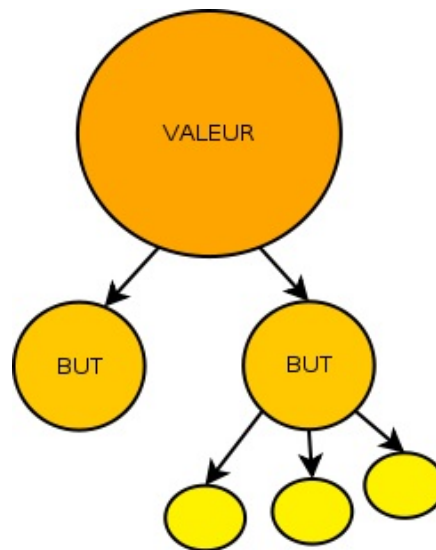
**BDD part d'une d'une
expérience linguistique**

should **plutôt que** test

conditionnel plutôt qu'impératif

Découverte de la valeur

Outside In



Story (classique)

As a [role]
I want [behavior]
In order to [outcome]

Story (à l'envers)

In order to [outcome]
I want [behavior]

Should ?

Pourquoi cette fonctionnalité ?

Demander pourquoi, recommencer tant que :

Protect / Increase revenue

Increase brand value

Make the product remarkable

Provide more value to your customers

<https://github.com/cucumber/cucumber/wiki/>

Ou avec moins d'items

<http://jennidoyle.wordpress.com/2011/01/12/why-ask-why/>

MAKE MONEY

SAVE MONEY

REDUCE RISK

(or reduce inventory (Goldratt))

Outils

Forgot the tools. Just write a
test. Any test. Now.



@RonJeffries

Talk is cheap. Show me the code.



Linus Torvalds, torvalds@ppc970.osdl.org

rails : Triple A

pas bdd'ish


```
# actionmailer/test/asset_host_test.rb
class AssetHostTest < Test::Unit::TestCase
  def setup
    set_delivery_method :test
    ActionMailer::Base.perform_deliveries = true
    ActionMailer::Base.deliveries.clear
    AssetHostMailer.configure do |c|
      c.asset_host = "http://www.example.com"
      c.assets_dir = ''
    end
  end

  def test_asset_host_as_string
    mail = AssetHostMailer.email_with_asset
    assert_equal(
      %Q{<img alt="Somelogo" src="http://www.example.com/images/sc
      mail.body.to_s.strip
    )
  end
end
```

```
# actionmailer/test/asset_host_test.rb
class AssetHostTest < Test::Unit::TestCase
  ### ARRANGE
  ###
  def setup
    set_delivery_method :test
    ActionMailer::Base.perform_deliveries = true
    ActionMailer::Base.deliveries.clear
    AssetHostMailer.configure do |c|
      c.asset_host = "http://www.example.com"
      c.assets_dir = ''
    end
  end

  def test_asset_host_as_string
    mail = AssetHostMailer.email_with_asset
    assert_equal(
      %Q{<img alt="Somelogo" src="http://www.example.com/images/sc
      mail.body.to_s.strip
    )
  end
end
```

```
# actionmailer/test/asset_host_test.rb
class AssetHostTest < Test::Unit::TestCase
  def setup
    set_delivery_method :test
    ActionMailer::Base.perform_deliveries = true
    ActionMailer::Base.deliveries.clear
    AssetHostMailer.configure do |c|
      c.asset_host = "http://www.example.com"
      c.assets_dir = ''
    end
  end

  def test_asset_host_as_string
    ### ACT
    ###
    mail = AssetHostMailer.email_with_asset
    assert_equal(
      %Q{<img alt="Somelogo" src="http://www.example.com/images/sc
      mail.body.to_s.strip
    )
  end
end
```

```
# actionmailer/test/asset_host_test.rb
class AssetHostTest < Test::Unit::TestCase
  def setup
    set_delivery_method :test
    ActionMailer::Base.perform_deliveries = true
    ActionMailer::Base.deliveries.clear
    AssetHostMailer.configure do |c|
      c.asset_host = "http://www.example.com"
      c.assets_dir = ''
    end
  end

  def test_asset_host_as_string
    mail = AssetHostMailer.email_with_asset
    ## ASSERT
    ##
    assert_equal(
      %Q{<img alt="Somelogo" src="http://www.example.com/images/sc
      mail.body.to_s.strip
    )
  end
end
```

```
# actionmailer/test/asset_host_test.rb
class AssetHostTest < Test::Unit::TestCase
  def setup
    set_delivery_method :test
    ActionMailer::Base.perform_deliveries = true
    ActionMailer::Base.deliveries.clear
    AssetHostMailer.configure do |c|
      c.asset_host = "http://www.example.com"
      c.assets_dir = ''
    end
  end

  ## est-ce qu'on sait ce que fait le test
  ## en lisant le nom de la méthode ?
  ##
  def test_asset_host_as_string
    mail = AssetHostMailer.email_with_asset
    assert_equal(
      %Q{<img alt="Somelogo" src="http://www.example.com/images/sc
      mail.body.to_s.strip
    )
  end
end
```

rspec'ish

https://github.com/mongoid/mongoid-spec/unit/mongoid/criterion/optional_spec.rb

```
describe Mongoid::Criterion::Optional do

  let(:base) do
    Mongoid::Criteria.new(Person)
  end

  describe "#ascending" do

    context "when providing a field" do

      let(:criteria) do
        base.ascending(:title)
      end

      it "adds the ascending sort criteria" do
        criteria.options[:sort].should == [[:title, :asc]]
      end
    end
  end
end
```

```
## LE MODULE
##
describe Mongoid::Criterion::Optional do
  let(:base) do
    Mongoid::Criteria.new(Person)
  end

  ## LE COMPORTEMENT DE LA METHODE
  ##
  describe "#ascending" do

    context "when providing a field" do
      let(:criteria) do
        base.ascending(:title)
      end

      it "adds the ascending sort criteria" do
        criteria.options[:sort].should == [[:title, :asc]]
      end
    end
  end
end
```



```
describe Mongoid::Criterion::Optional do
  ## GIVEN
  ##
  let(:base) do
    Mongoid::Criteria.new(Person)
  end

  describe "#ascending" do

    context "when providing a field" do
      let(:criteria) do
        base.ascending(:title)
      end

      it "adds the ascending sort criteria" do
        criteria.options[:sort].should == [[:title, :asc]]
      end
    end
  end
end
```

```
describe Mongoid::Criterion::Optional do

  let(:base) do
    Mongoid::Criteria.new(Person)
  end

  describe "#ascending" do

    context "when providing a field" do
      ## WHEN
      ##
      let(:criteria) do
        base.ascending(:title)
      end

      it "adds the ascending sort criteria" do
        criteria.options[:sort].should == [[:title, :asc]]
      end
    end
  end
end
```

```
describe Mongoid::Criterion::Optional do

  let(:base) do
    Mongoid::Criteria.new(Person)
  end

  describe "#ascending" do

    context "when providing a field" do

      let(:criteria) do
        base.ascending(:title)
      end

      ## THEN
      ##
      it "adds the ascending sort criteria" do
        criteria.options[:sort].should == [[:title, :asc]]
      end
    end
  end
end
```

cucumber

Scenario: compare using eq (==)

Given a file named "compare_using_eq.rb" with:

```
"""
require 'spec_helper'

describe "a string" do
  it "is equal to another string of the same value" do
    "this string".should eq("this string")
  end

  it "is not equal to another string of a different value" do
    "this string".should_not eq("a different string")
  end
end

describe "an integer" do
  it "is equal to a float of the same value" do
    5.should eq(5.0)
  end
end
"""
```

When I run "rspec compare_using_eq.rb"

Then the output should contain "3 examples, 0 failures"

Et ça produit une documentation
compréhensible...

Scenario: compare using eq (==)

Given a file named "compare_using_eq.rb" with:

```
require 'spec_helper'

describe "a string" do
  it "is equal to another string of the same value" do
    "this string".should eq("this string")
  end

  it "is not equal to another string of a different value" do
    "this string".should_not eq("a different string")
  end
end

describe "an integer" do
  it "is equal to a float of the same value" do
    5.should eq(5.0)
  end
end
```

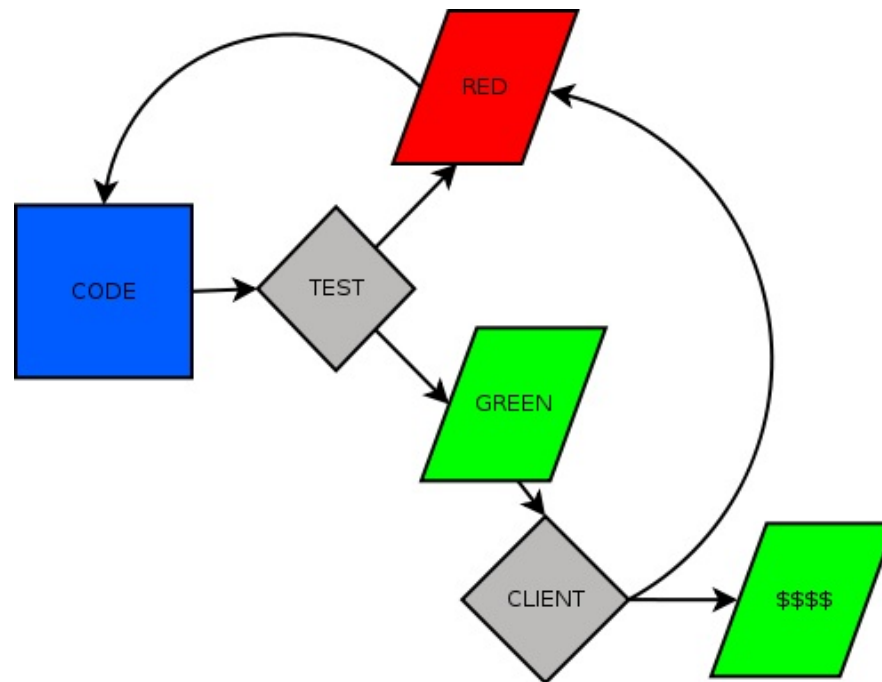
When I run `rspec compare_using_eq.rb`

Then the output should contain "3 examples, 0 failures"

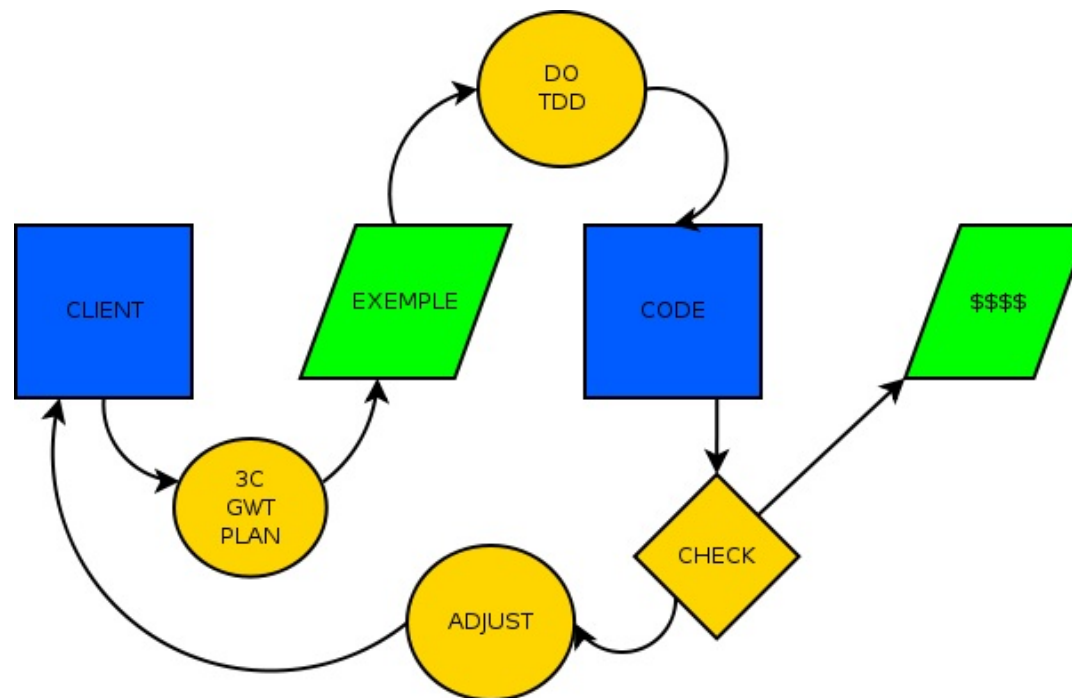
Workflow

**Vous vous souvenez de
mon workflow tout pourri ?**

Mauvais workflow !



Workflow Possible ?



Qu'est-ce que je peux
retenir de ça ?

Valeur pour le client

Chercher la satisfaction du client (Womack)

Ce que je veux

Où je veux

Quand je veux

Ne me faites pas perdre mon temps

Réduisez le nombre de décisions que je dois prendre

Par la livraison de produit de
qualité

Utiliser le même langage,
omniprésent, pour faciliter

la communication

les questions

BDD n'est pas à propos des outils

les outils sont là pour assoir la démarche

Ressources

BDD

Keogh : [bdd-for-life](#)

[référentiel de l'Institut](#)

<http://behaviour-driven.org/>

Lean

The Gold Mine (Ballé)