Génie Logiciel : Conception
Socle Technique & Méthodologique

Sébastien Mosser
INF-5153, Hiver 2019, Cours #1.1

UQÀM

Atelier A1

(semaines #3 & #4)

Comprendre un petit exemple

Identifier des problèmes de conception

Proposer des évolutions simples

Pourquoi concevoir ?

1

«Maybe the problem is not that it's so **hard** to write **good software**, ...

5

... but that it's so **easy to write crap**.»

@calvinb

6

Une balançoire ?

7

8

9

10

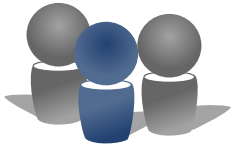
Vision Chef de Projet

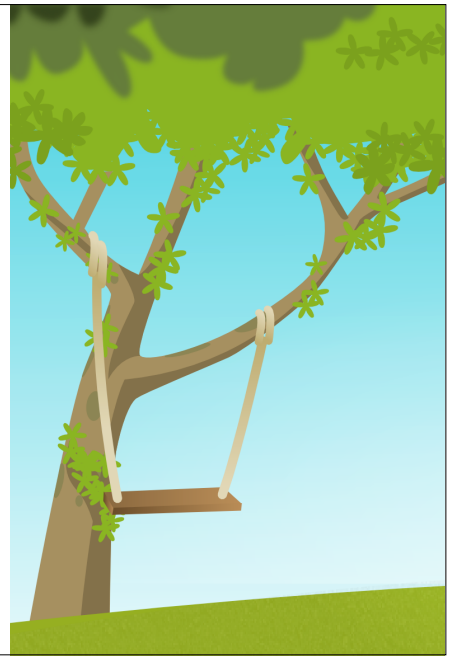
"Ce que j'ai compris du client"

11

Vision Développeurs

"Ce que l'on a livré"





12

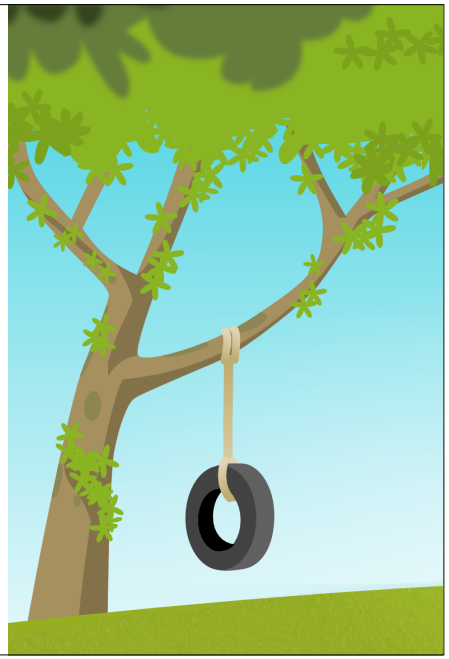
Version Utilisateur

"Ce que je dois utiliser au quotidien"

13

Vision Client

"Ce dont j'avais vraiment besoin"



14

La conception permet d'aligner tous les intervenants sur un projet

Modèles d'exigences, modèles de structure, modèles d'intégration, ...

15

La conception permet d'aligner tous les intervenants sur un projet

Modèles d'exigences, modèles de structure, modèles d'intégration, ...

16

Cycle(s) de développement

2

Activités de création de logiciel

Exigences

Production

Conception

Développement

Spécification

Validation

Maintenance

Tests

Processus de Développement ?

Organisation

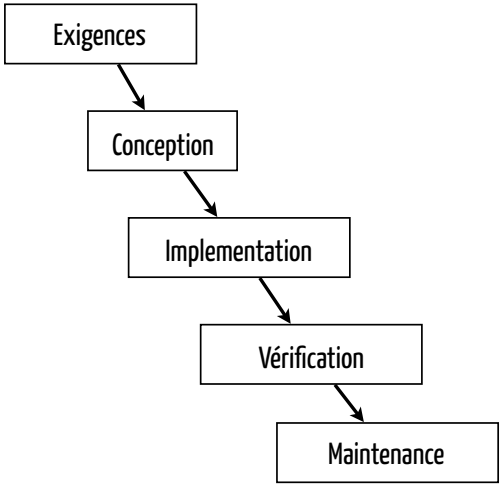
de ces **activités**

Linéaire

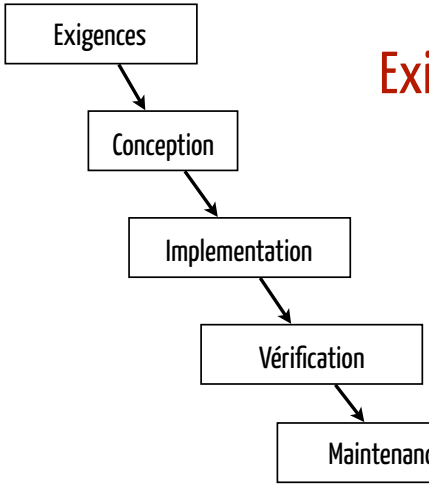
ou

Non-linéaire

Modèle en Cascade (Linéaire, ca. 1970)



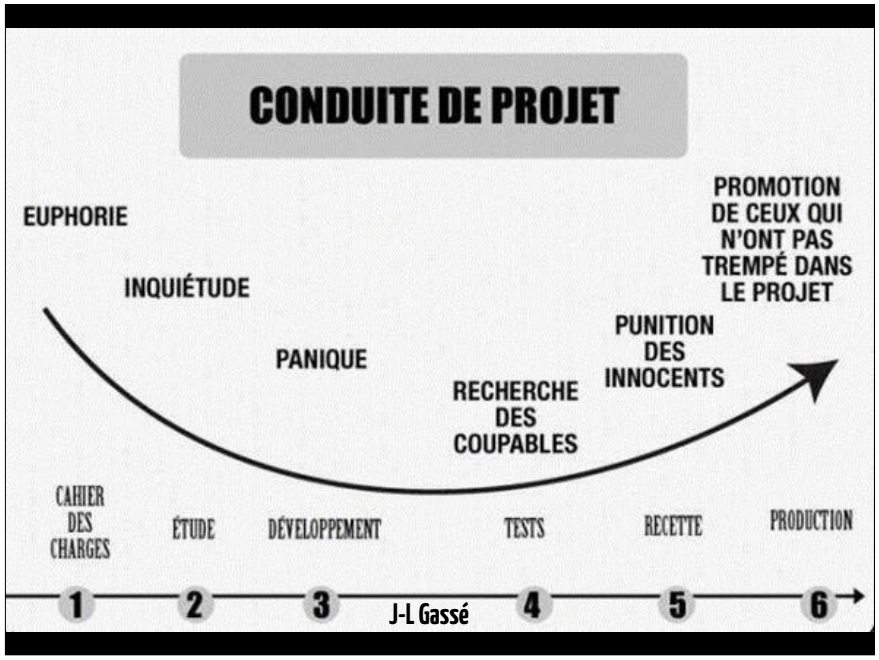
Modèle en Cascade (Linéaire, ca. 1970)



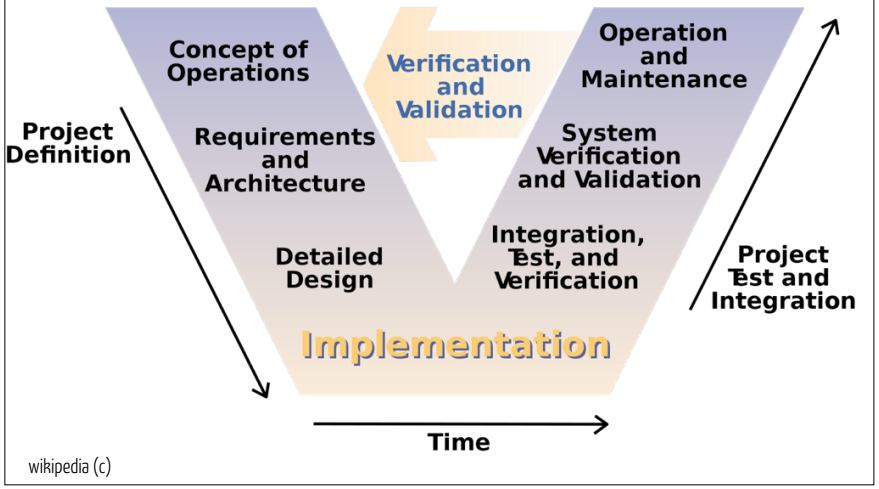
Exigences “au plus tôt”

Effet Tunnel

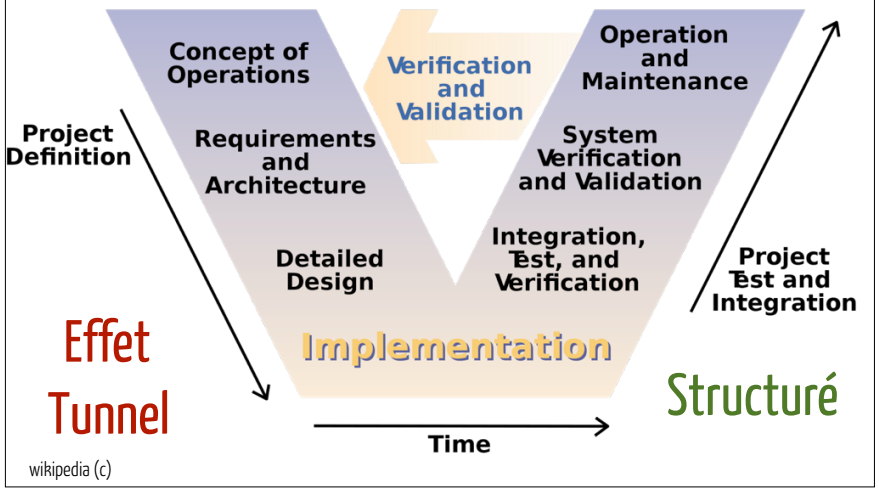
Tests “au plus tard”



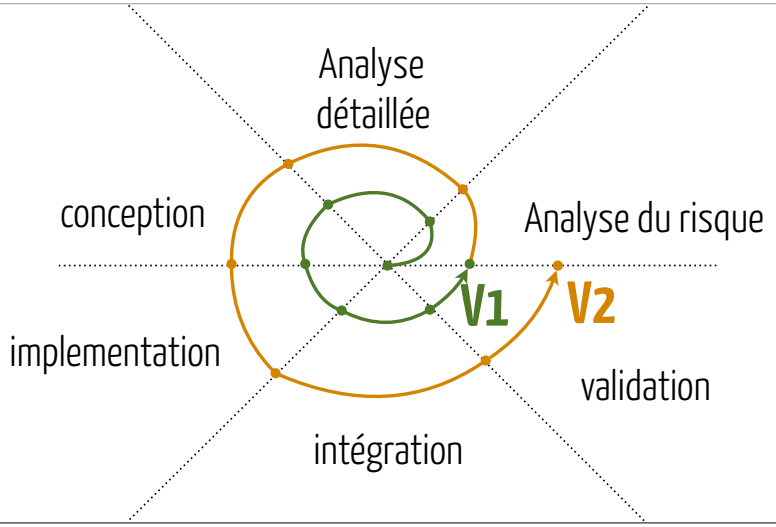
Modèle en V



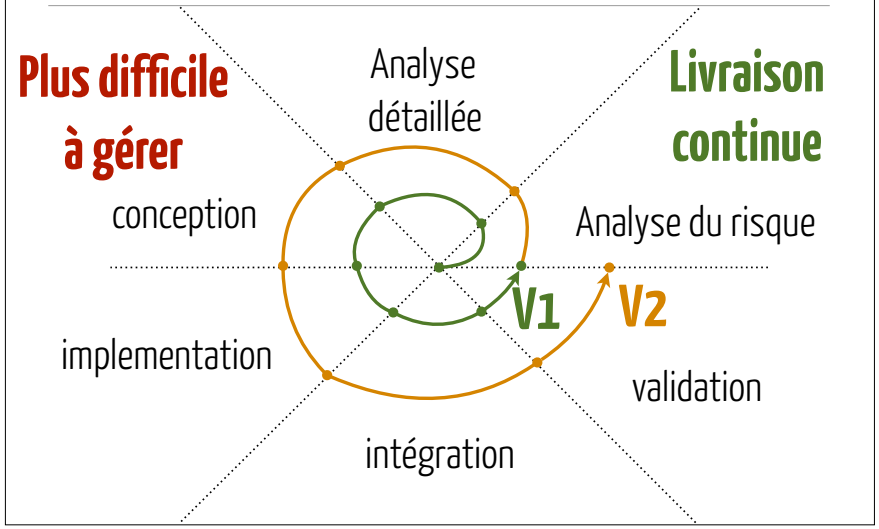
Modèle en V



Modèle en spirale [Boehm86]



Modèle en spirale [Boehm86]

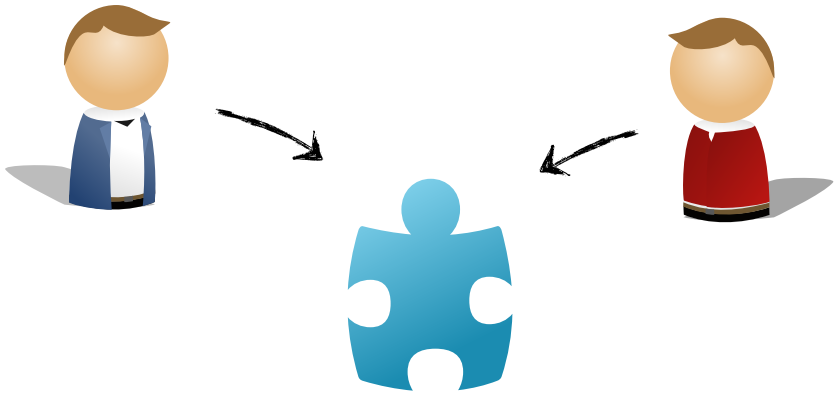


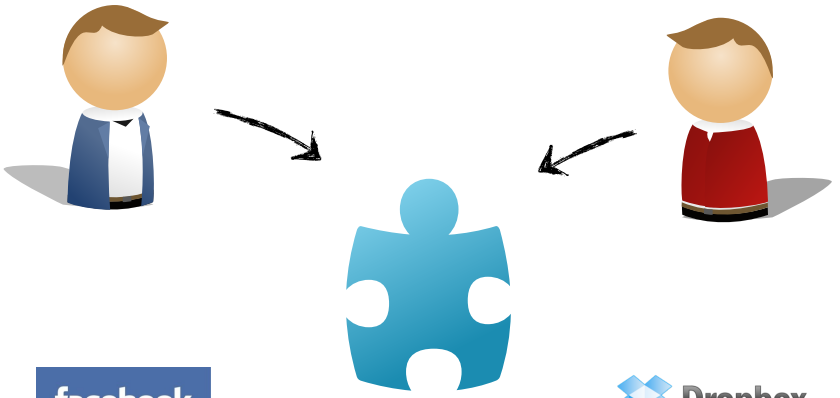
Gestion de versions

3

développeur Logiciel

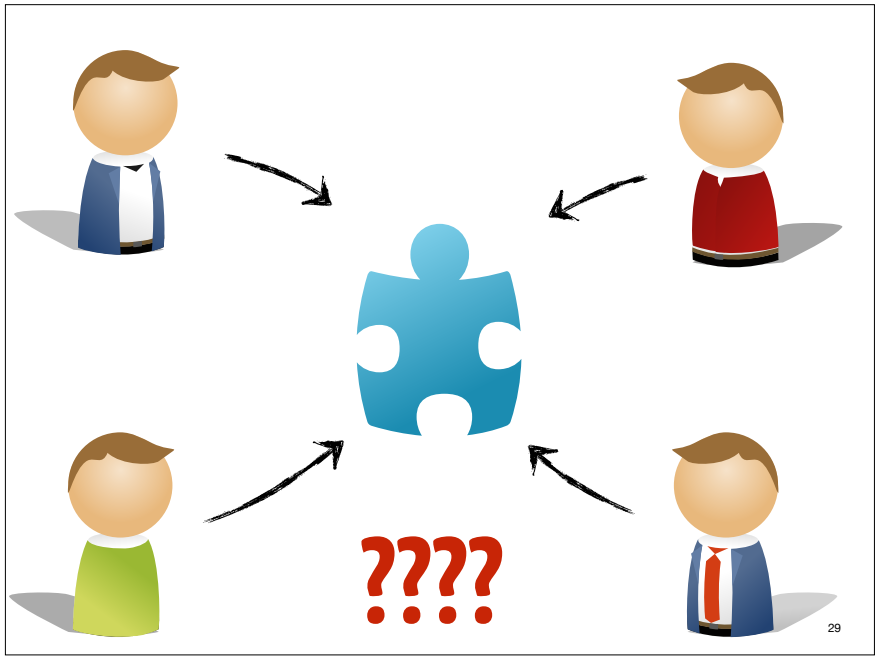
Développement Collaboratif





clé USB ?

Courriel ? Répertoire partagé ?



29

Pourquoi Versionner ?

30

Pourquoi utiliser la gestion de version

Pour tracer les changements !

Pour oublier les changements !

Pour partager les changements !

(among others)

31

git add

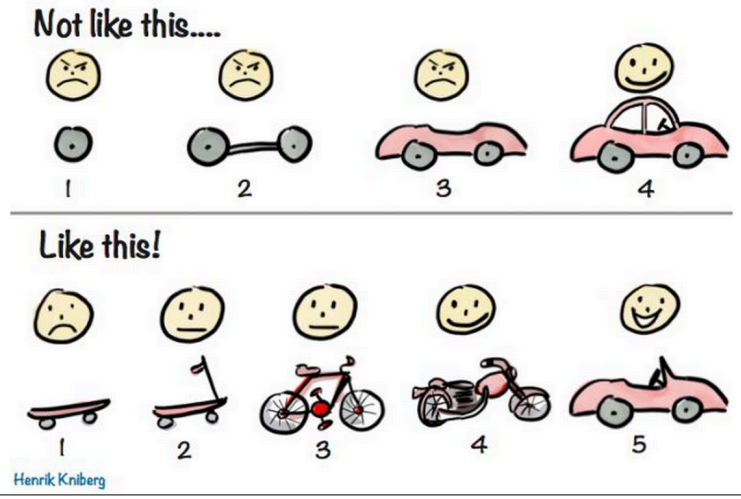
git commit

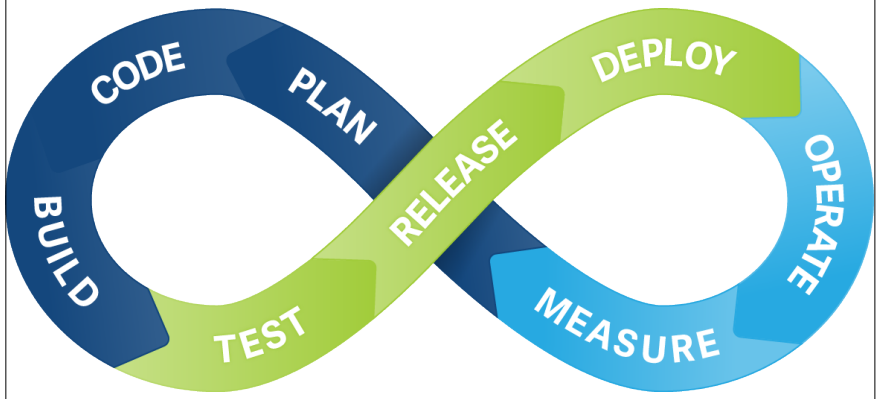
git push

git pull

32

Versions & Conception ?





http://meshfields.de/continuous-integration-testing-delivery-ionic2-hybrid-mobile-apps-buddybuild/

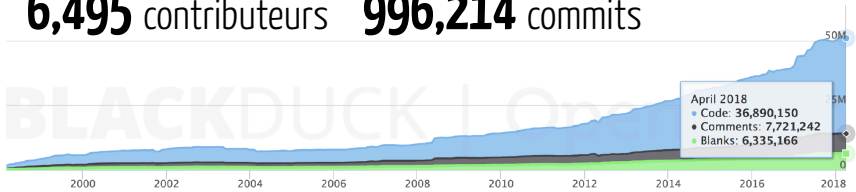
Construction automatique 4

Quelle est la taille de Firefox ?

https://www.openhub.net/p/firefox

Bienvenue dans la “vraie vie”

6,495 contributeurs **996,214** commits



Language Breakdown

Language	Code Lines	Comment Lines	Comment Ratio	Blank Lines	Total Lines	Total Percentage
C++	11,790,199	2,425,138	17.1%	2,222,045	16,437,382	32.3%
JavaScript	7,910,535	2,517,612	24.1%	1,746,715	12,174,862	23.9%
HTML	5,056,599	250,983	4.7%	523,000	5,830,582	11.4%
C	4,946,470	1,294,655	20.7%	823,635	7,064,760	13.9%
Rust	2,058,338	329,541	13.8%	200,316	2,588,195	5.1%
XML	1,414,688	33,442	2.3%	79,942	1,528,072	3.0%
Python	1,232,582	360,603	22.6%	325,064	1,918,249	3.8%
Java	663,704	245,681	27.0%	136,001	1,045,386	2.1%

Et alors ?

1 commit → Construire les binaires

Jouer les tests → Mettre à disposition

Et alors ?

1 commit → Construire les binaires

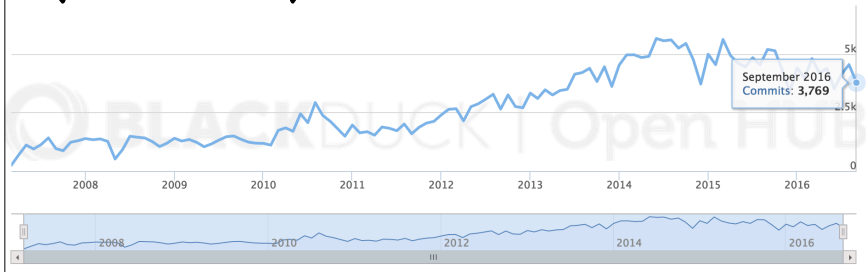
Jouer les tests → Mettre à disposition

137 hours! (08.2012)

http://oduinn.com/blog/2012/08/21/137-hours-compute-hours-every-6-minutes/

3769 Commits en 30 jours ...

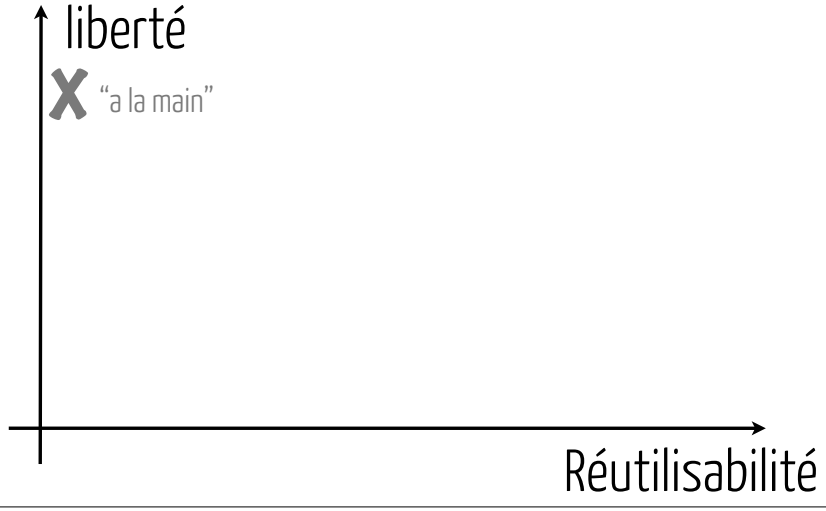
(30×24×60) / 3769



~1commit toutes les 11 minutes

http://www.ohloh.net/p/firefox

Systèmes de constructions



40-1

Systèmes de constructions

40-2

Systèmes de constructions

40-3

Convention over Configuration

41

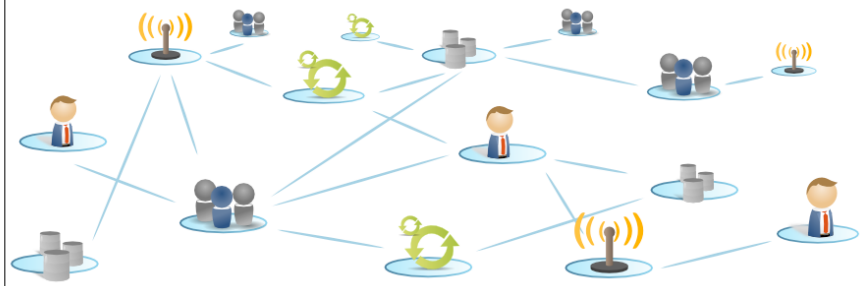
41

Exploiter

de conventions

Exemple: **Internet des Objets**

SensApp



Ecosystème Technologique

SensApp

Scala

akka

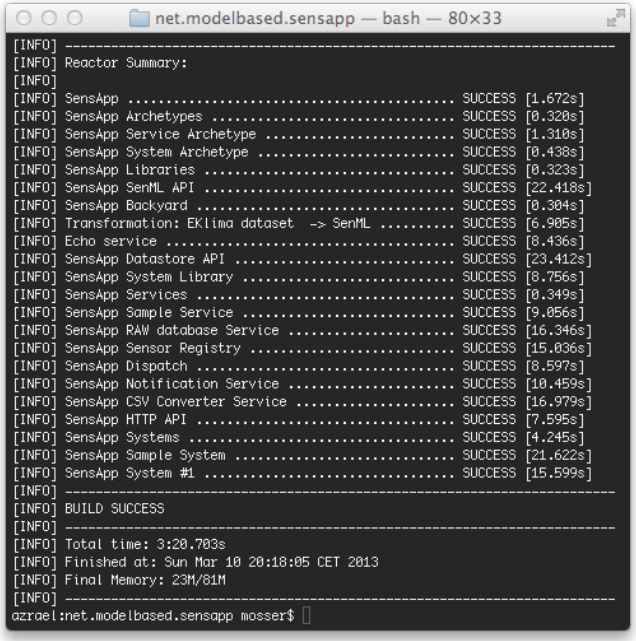
Comment l'installer ?

mongoDB

spray

Conventions à la rescousse

```
$ git clone git://github.com/...
$ cd sensapp/
$ mvn install
```



Convention ?

choix

non-complètement-stupide

fait par

des **gens expérimentés***

(*) En tout cas plus que vous

Convention **Structurelle**

Code source

dans «**src**»

code compilé pour la cible

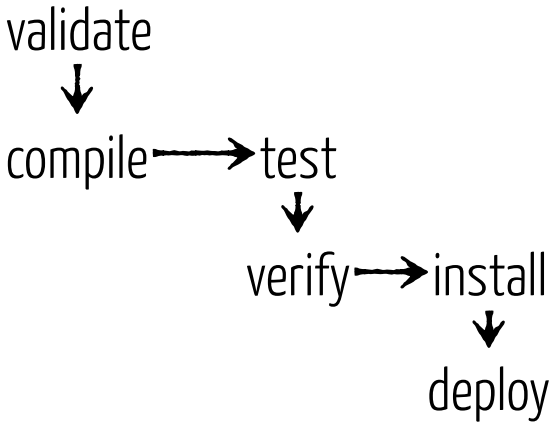
dans «**target**»

Fichiers ressources

dans «**resource**»

...

Convention “Processus”: Cycle de vie



Tests 5

(Unitaire & Acceptation)

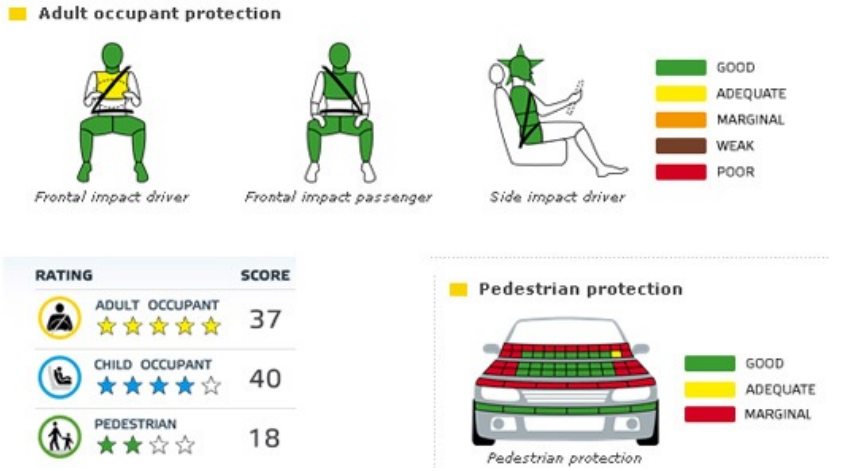
“ New code is **guilty** until proven **innocent**.

Test ≠ Essai

Essai

54

Test

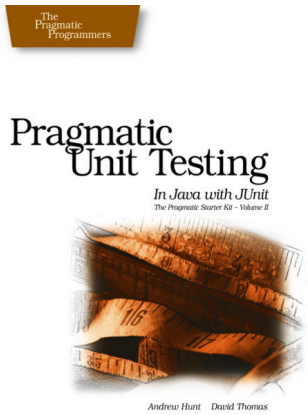


55

Bibliographie supplémentaire

Lisez ce livre.

Vraiment.



Placement produit : le cours de Tests (MGL 7230) devrait ouvrir à l'hiver 2020

56

Les BONS tests sont "A TRIP"

JUNIT

Automatic

Thorough

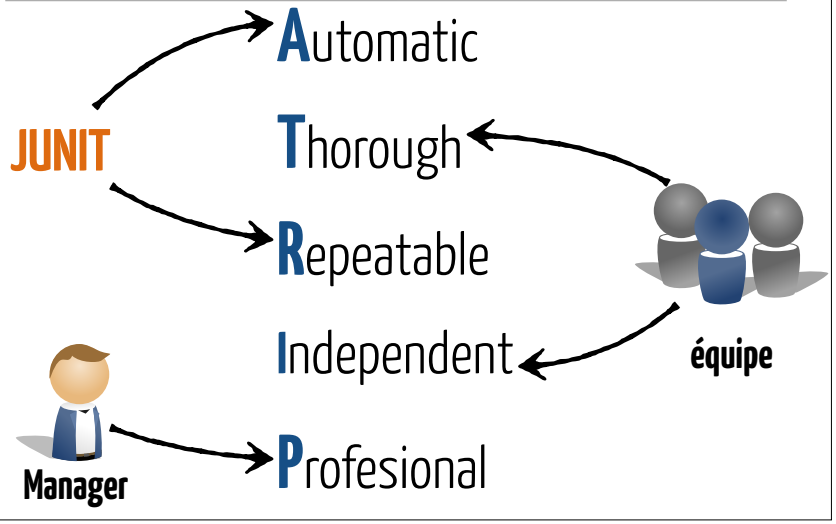
Repeatable

Independent

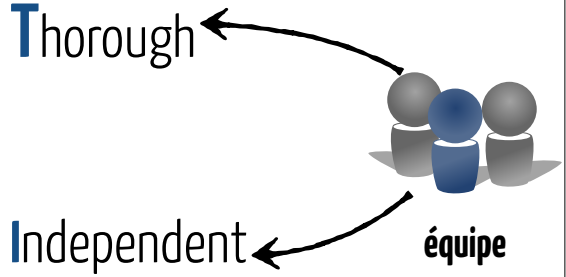
Professional

57

Les BONS tests sont "A TRIP"



Les BONS tests sont "A TRIP"



Test unitaire : Contexte + Assertion

```
@Test
public void addDrinks() {
    assertTrue(theOrder.getContents().isEmpty());
    theOrder.add("a drink");
    theOrder.add("another drink");
    assertFalse(theOrder.getContents().isEmpty());
}
```


Conception ? Tests d'acceptation

```
Feature: Ordering Drinks

  Background:
    Given a brand new micro-brewery named "MaxiBrew"

  Scenario: Creating an Order
    Given the empty order for table 1
    When someone orders a drink
    Then the order is not empty
```

Tests : Unitaire ⇔Acceptation

```
@Given("^the empty order for table (\\d+)$")
public void initializeOrder(int table) {
    theOrder = new Order(table);
}

@When("^someone orders (.*)")
public void addToTheOrder(String drink) {
    theOrder.add(drink);
}

@Then("^the order is not empty$")
public void orderContentsStatus() {
    assertFalse(theOrder.getContents().isEmpty());
}
```

61

Intégration Continue 6

62

Continuous Integration

Continuous Integration is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly. This article is a quick overview of Continuous Integration summarizing the technique and its current usage.

01 May 2006

Martin Fowler

Translations: Portuguese Chinese Korean French Chinese Czech

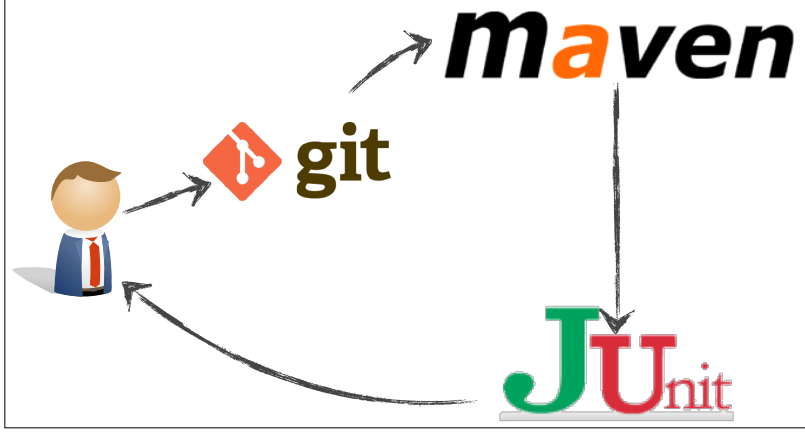
Tags: popular · agile · delivery · extreme programming · continuous integration

01.05.2006!

http://martinfowler.com/articles/continuousIntegration.html

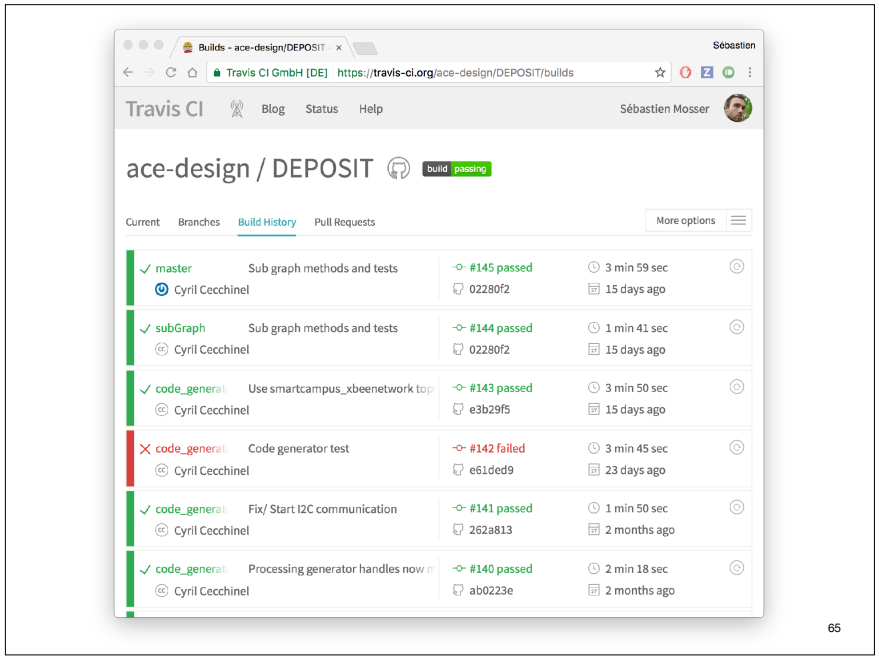
63

Boucle d'intégration



64

64


Builds - ace-design/DEPOSIT
Sébastien
Travis CI GmbH [DE] https://travis-ci.org/ace-design/DEPOSIT/builds
Travis CI
Blog
Status
Help
Sébastien Mosser
ace-design / DEPOSIT
build passing
Current
Branches
Build History
Pull Requests
More options
master
Sub graph methods and tests
Cyril Cecchinel
#145 passed
02280f2
3 min 59 sec
15 days ago
subGraph
Sub graph methods and tests
Cyril Cecchinel
#144 passed
02280f2
1 min 41 sec
15 days ago
code_generat
Use smartcampus_xbeenetwork top
Cyril Cecchinel
#143 passed
e3b29f5
3 min 50 sec
15 days ago
code_generat
Code generator test
Cyril Cecchinel
#142 failed
e61ded9
3 min 45 sec
23 days ago
code_generat
Fix/ Start I2C communication
Cyril Cecchinel
#141 passed
262a813
1 min 50 sec
2 months ago
code_generat
Processing generator handles now
Cyril Cecchinel
#140 passed
ab0223e
2 min 18 sec
2 months ago
65

65

66


15
minutes
Have
a
Break

67