



wemanity
The Agile Driving Force

Renovating your code base

@nelisboucke
@matteo_pierro



wemanity
The Agile Driving Force

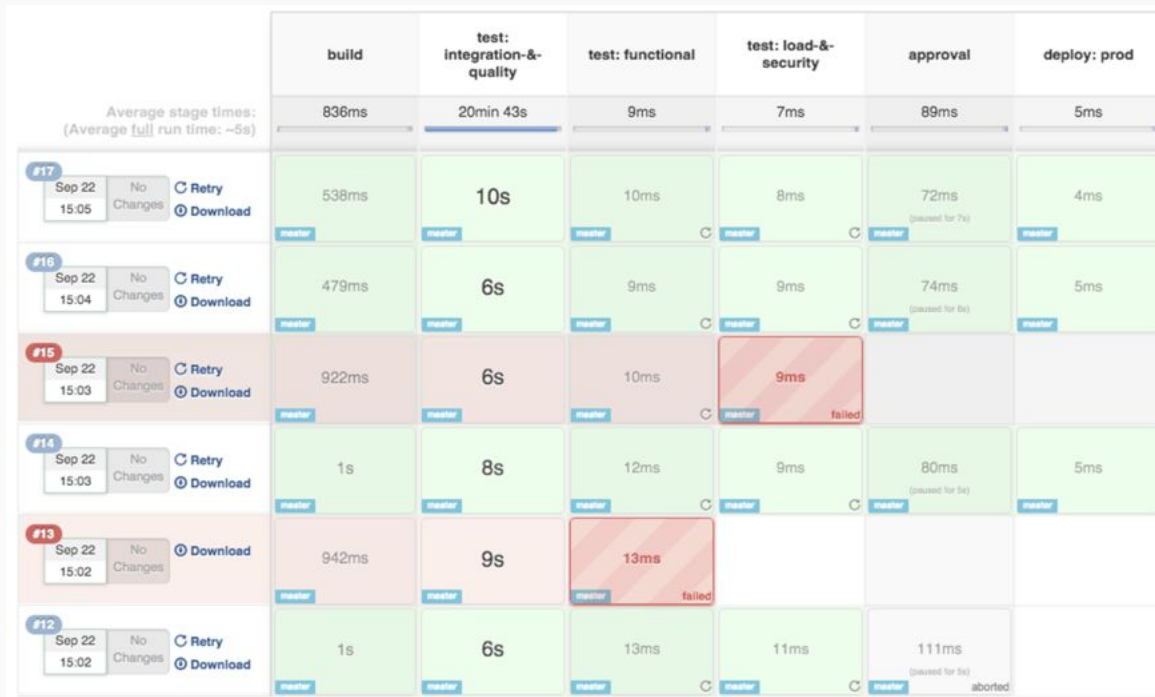
Agenda

- Why?
- Effective Golden Master
 - Introduction
 - Pair/MOB
 - Reflection
- Refactor
 - Introduction
 - Pair/MOB
 - Reflection

Why?



wemanity
The Agile Driving Force





wemanity
The Agile Driving Force

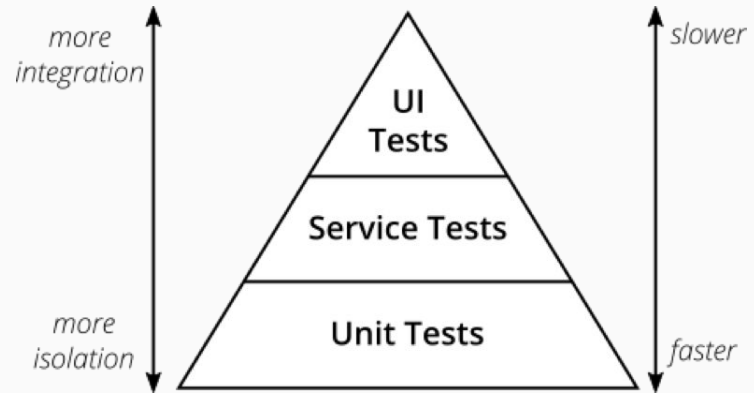
CI/CD on existing code bases

Reliable and Fast test suite

- tests pass → we can release
- failures → pinpoint to real problems

Unit Tests

- Testable design (well defined boundaries)

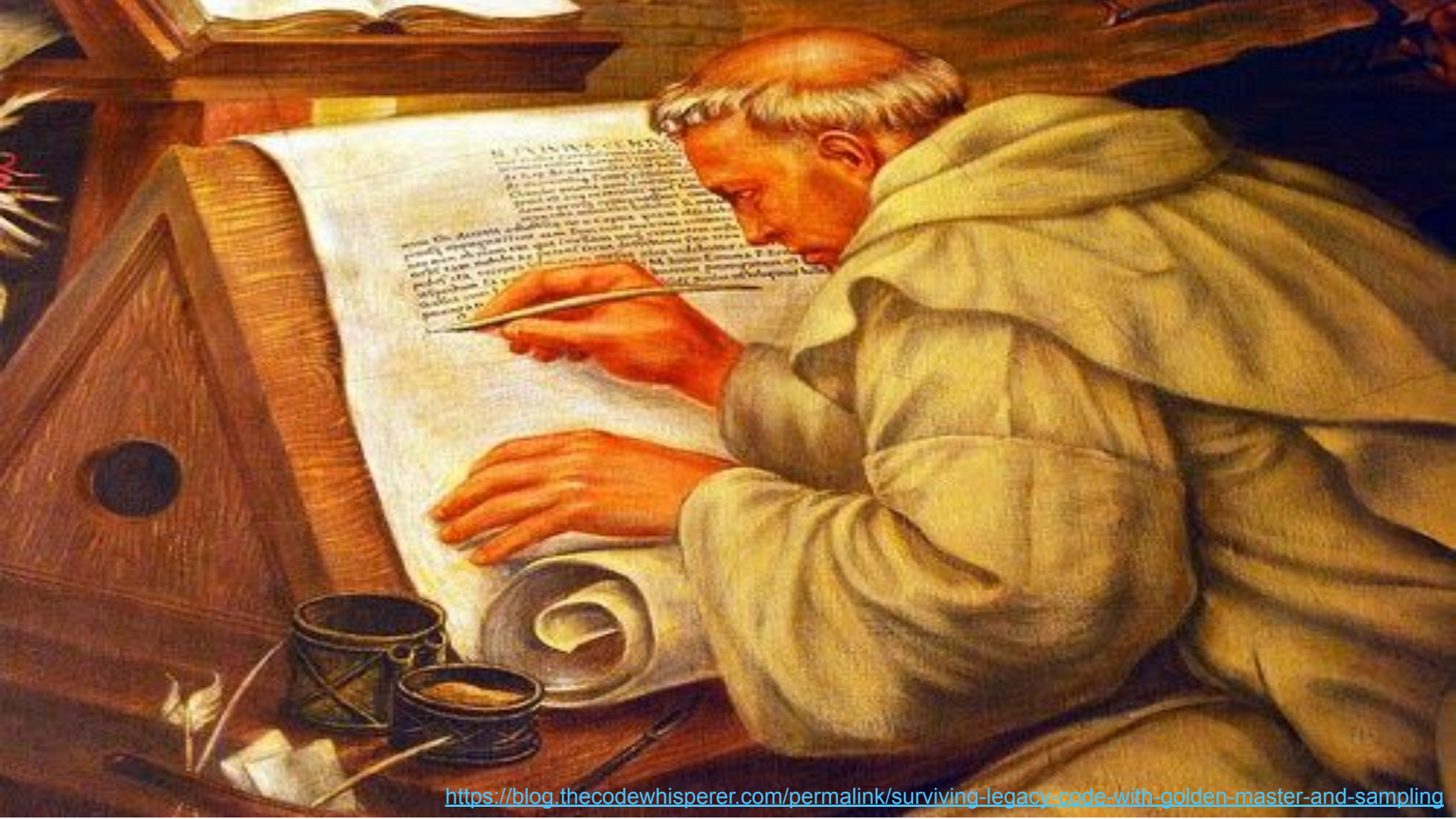




wemanity
The Agile Driving Force

The catch-22 of legacy code

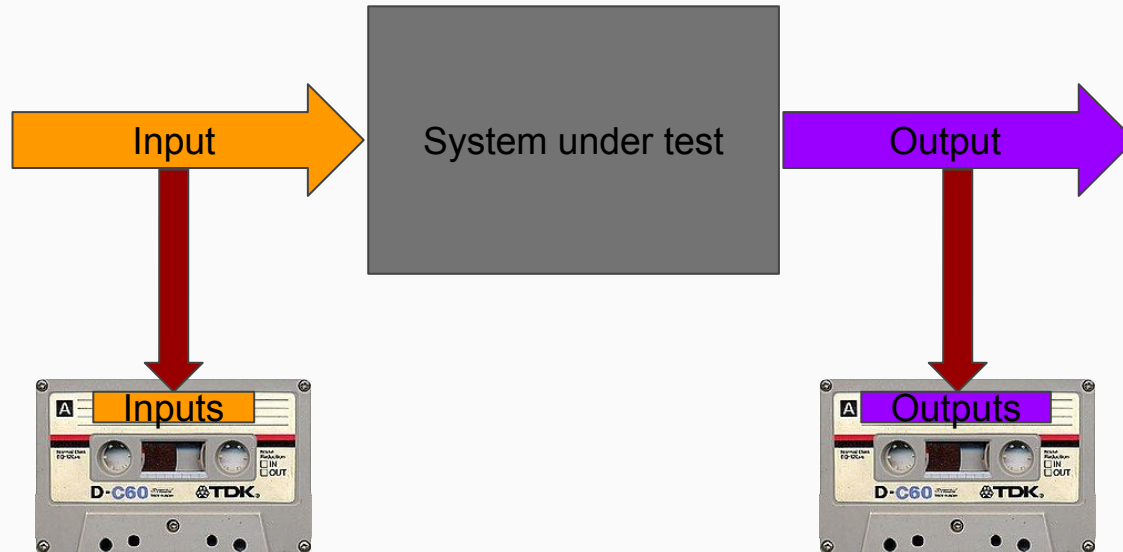
- I can't safely improve my design without decent tests
- I can't write decent tests without change my design





wemanship
The Agile Driving Force

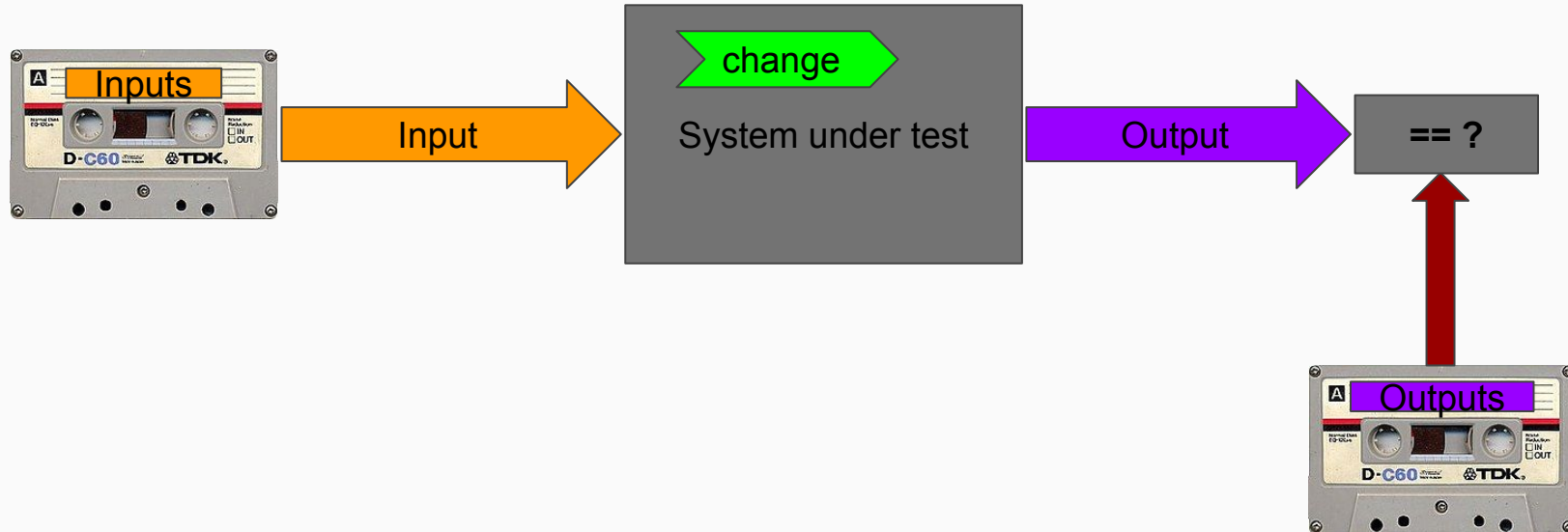
How to implement it





wemanship
The Agile Driving Force

Refactoring





wemanity
The Agile Driving Force

DEMO!

Longer string



wemanship
The Agile Driving Force

Is my Golden Master effective?



wemanity
The Agile Driving Force

Is my Golden Master effective?

You can use two metrics to know if you have a good golden master:

- code coverage
- mutation testing: number of mutants killed



wemanity
The Agile Driving Force

DEMO!

Longer string



wemanity
The Agile Driving Force

It's your turn now!



wemanity
The Agile Driving Force

Trivia Game

clone the repository <https://github.com/MatteoPierro/trivia.git>

checkout the branch **workshop-starting-point**

guide

<https://github.com/supernelis/workshop-renovating-legacy-codebase/blob/master/StepsTowardsGoldenMaster.md>



wemanity
The Agile Driving Force

Reflections regarding the Golden Master



wemanity
The Agile Driving Force

Questions?

- Are you comfortable to touch this code now?
- Can I also apply this for my (REST, message api, SQL)?
- Capture real data?
- Perfect, so this is the only types of tests I need?
- Do I have to do the whole system at once?



wemanity
The Agile Driving Force

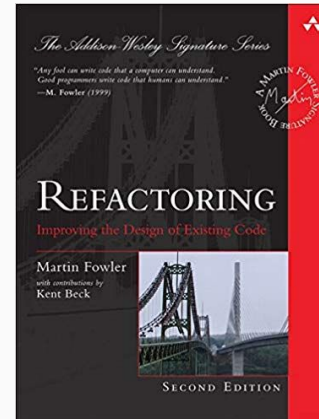
Let's start the refactoring



wemanity
The Agile Driving Force

What is refactoring?

“Refactoring is the process of changing a software system in such a way that it does not alter the external behavior of the code yet improves its internal structure.” cit. Martin Fowler





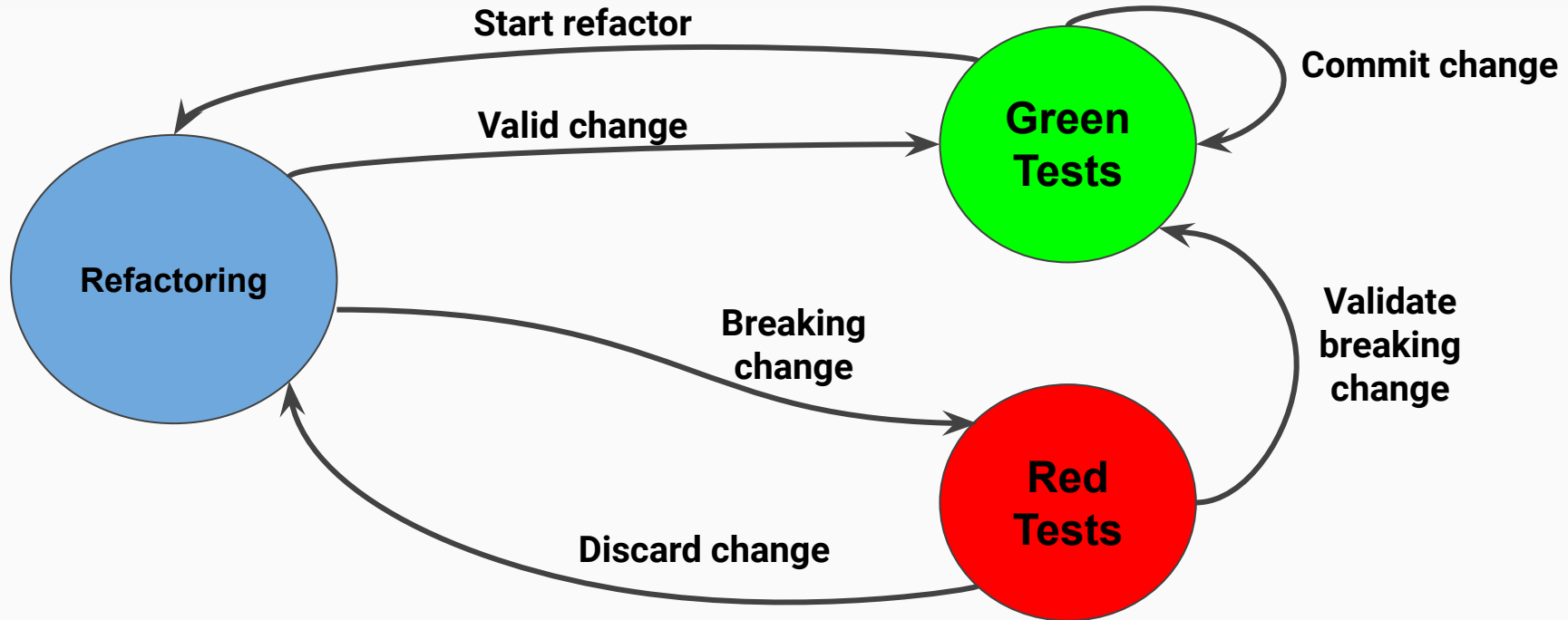
Typical Refactorings

- Extract function (method)
- Extract constant
- Extract variable
- Extract parameter
- Extract module (class)
- Invert if condition
- Replace Nested Conditional with Guard Clauses
-



wemanity
The Agile Driving Force

Refactoring flow





wemanity
The Agile Driving Force

Refactoring

- Clean starting point (if you need it)

java ***java-golden-master***

javascript ***js-golden-master***

- Guide

<https://github.com/supernelis/workshop-renovating-legacy-code-base/blob/master/StepsRefactoring.md>



wemanity
The Agile Driving Force

Reflections regarding Refactoring



wemanship
The Agile Driving Force

Questions?

- What is the biggest takeaway for you?
- What to work on to align it better with the trivia game domain?
- Can you now add unit tests for this codebase?
- Is it easier to estimate a change to this system?
- Perfect solution?



wemanship
The Agile Driving Force

ROTI

(Return On Time Invested)

Who are we?



wemanship
The Agile Driving Force

Nelis Boucké



[@nelisboucke](#)

Matteo Pierro



[@matteo_pierro](#)



wemanity
The Agile Driving Force

We Are Hiring!

<https://wemanity.com/join-us/>



wemanimity
The Agile Driving Force

Add a new behaviour

1. When testing a change, run the same inputs through the new version of the system and flag any output variation.
2. For each variation, have a human determine whether or not the change is expected and desirable. If it is, update the persisted gold master records



wemanship
The Agile Driving Force

First Step

Write a test that runs the **GameRunner**

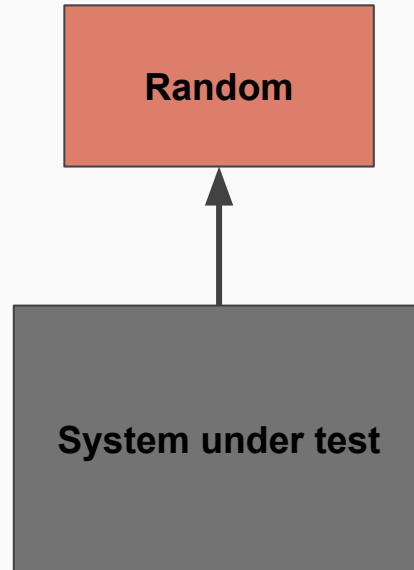
run it several times and manual check the result

Something strange????

Dependencies



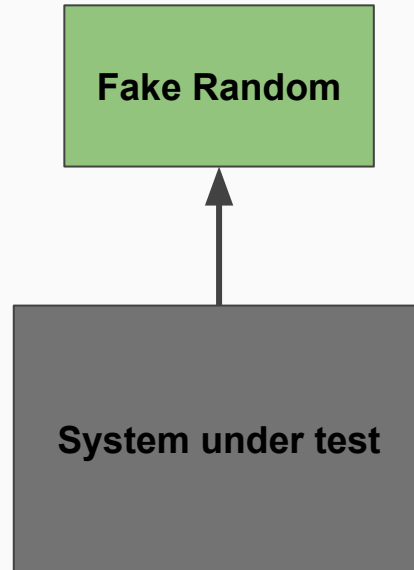
wemanimity
The Agile Driving Force



Dependencies



wemanship
The Agile Driving Force





wemanship
The Agile Driving Force

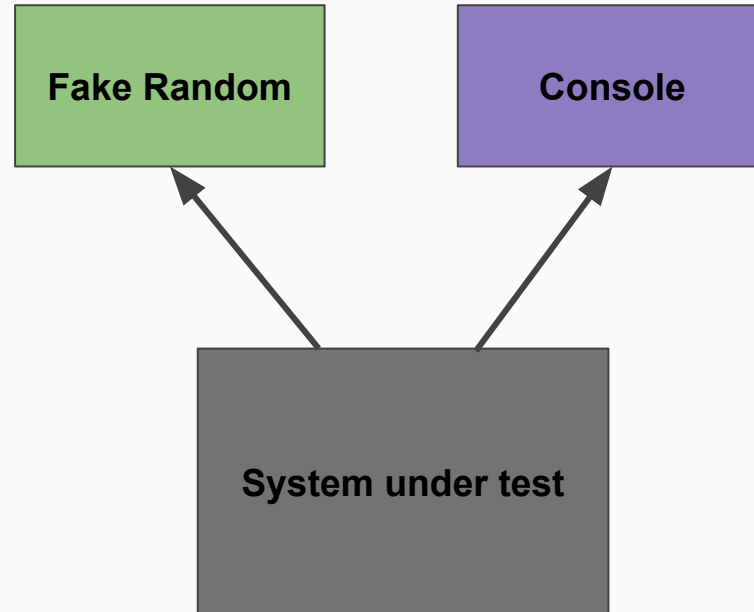
Second Step

Let's write a test to assert the result given a seed.

Which is the Problem?

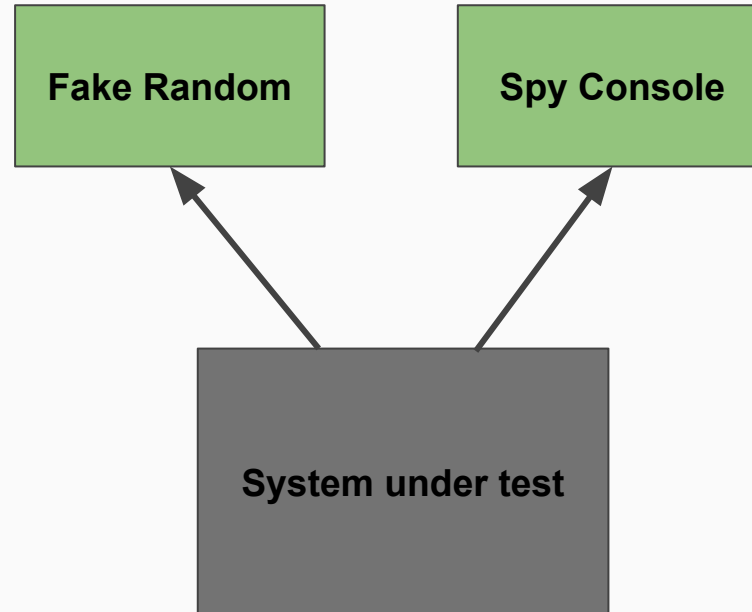


Dependencies





Dependencies





wemanity
The Agile Driving Force

Third Step

Check your code coverage.

How can we improve it?



wemanimity
The Agile Driving Force

Possible Seeds

Javascript : 3,5,7,77

Java: 3, 5



wemanship
The Agile Driving Force

Fourth Step

Check your mutation testing report

How can we improve it?

Tip



wemanship
The Agile Driving Force

Check **GameRunner**

How many users do we have?



Seeds

Javascript: seed 3, 5, 7, 77

```
it("1 player", function () {
    this.verify(runGame(77, ["Matteo"]), { reporters: [] });
});

it("2 player", function () {
    this.verify(runGame(9, ["Matteo", "John"]), { reporters: [] });
});

it("4 player", function () {
    this.verify(runGame(2, ["Matteo", "John", "Pep", "Jin"]), { reporters: [] });
});
```