

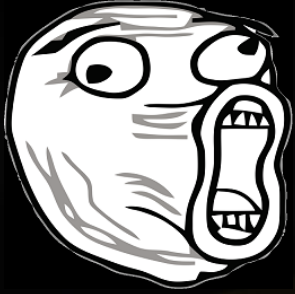
The fast and the continuous

Andrei Solntsev

 **codeborne**

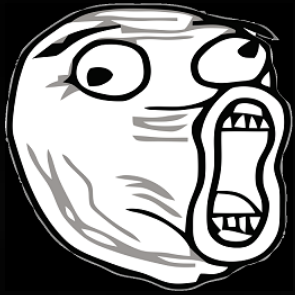
- UI-tests
- are
- evil

- 5 things
that you are doing
wrong
in
test automation



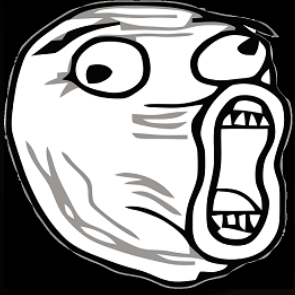
- Dream of
parallelizing
tests?

Oops #1



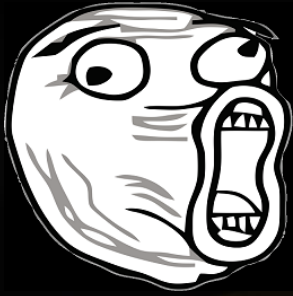
- And run tests in
ONLY
a few hours?

Oops #1



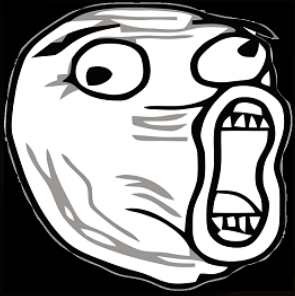
Love *PageObjects* and *Steps?*

Oops #2



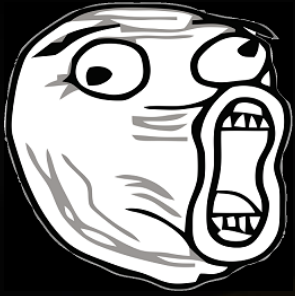
- TestNG
is better than
JUnit?

Oops #3



• “How to tell
webdriver
to ignore
invalid SSL certs?”

Oops #4



How to overcome StaleElementException?

Oops #5

- These questions
are
basically wrong

- And that's why:

- Integration tests are
 - SCAM!

J. B. Rainsberger

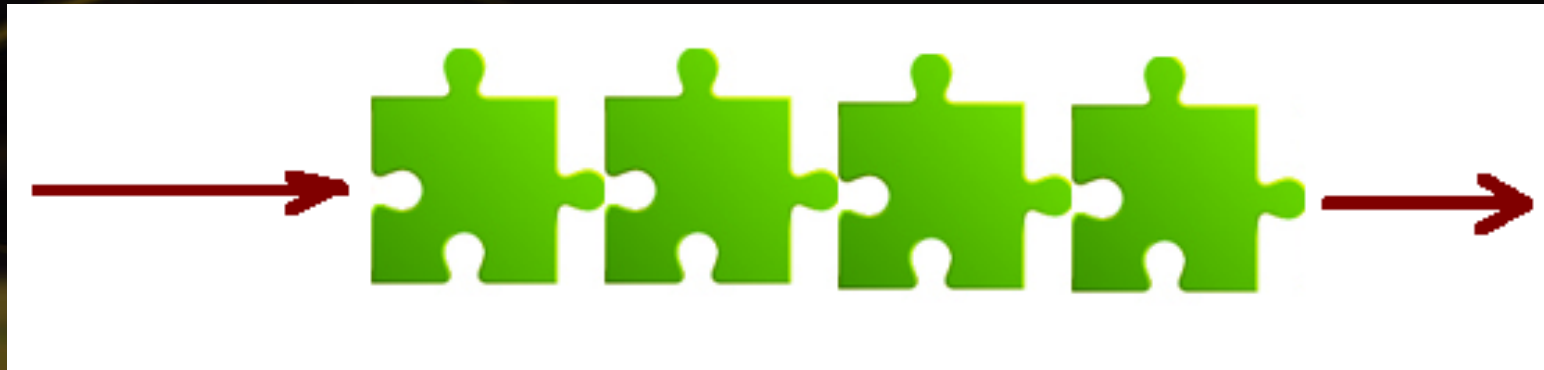
Integration tests are SCAM



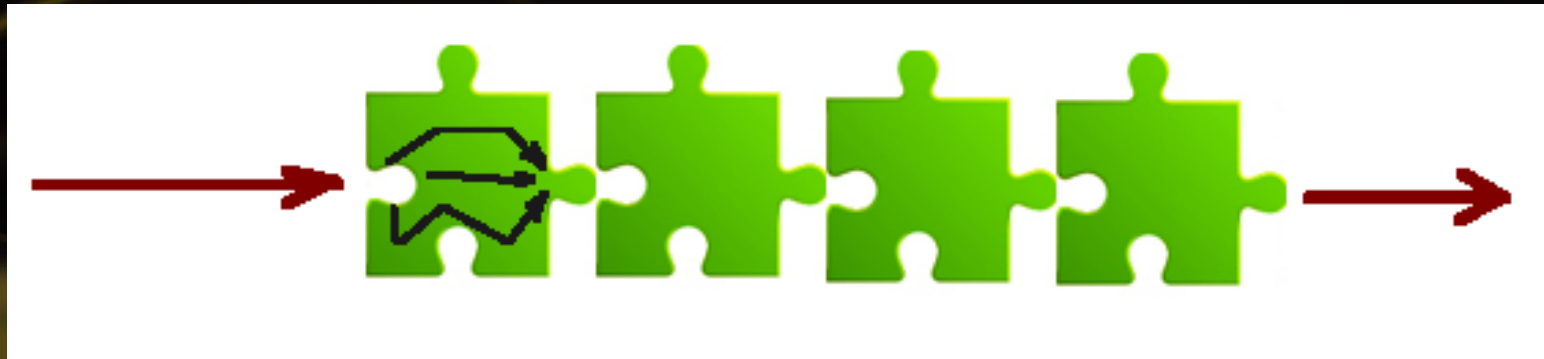
J. B. Rainsberger

<http://devconfu.eu/video/239>

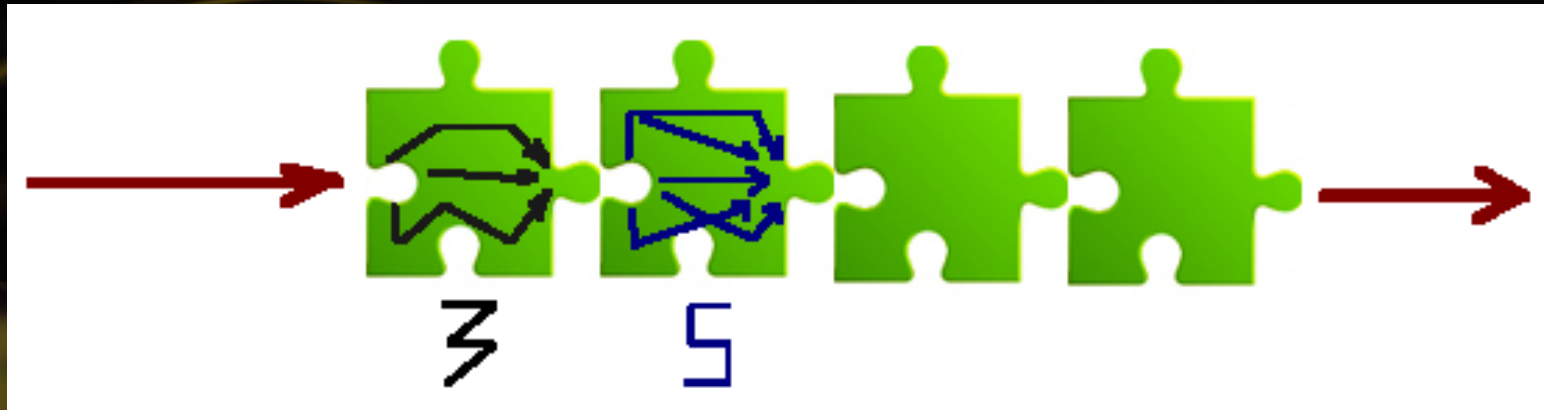
Integrations tests are SCAM



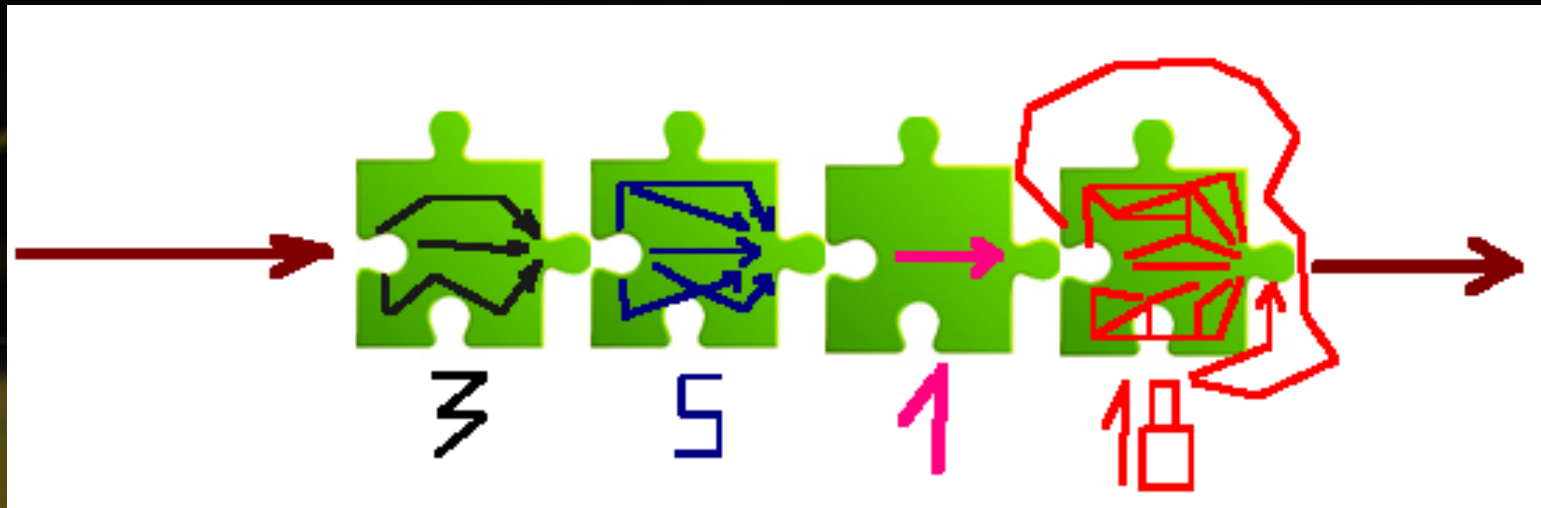
Integrations tests are SCAM



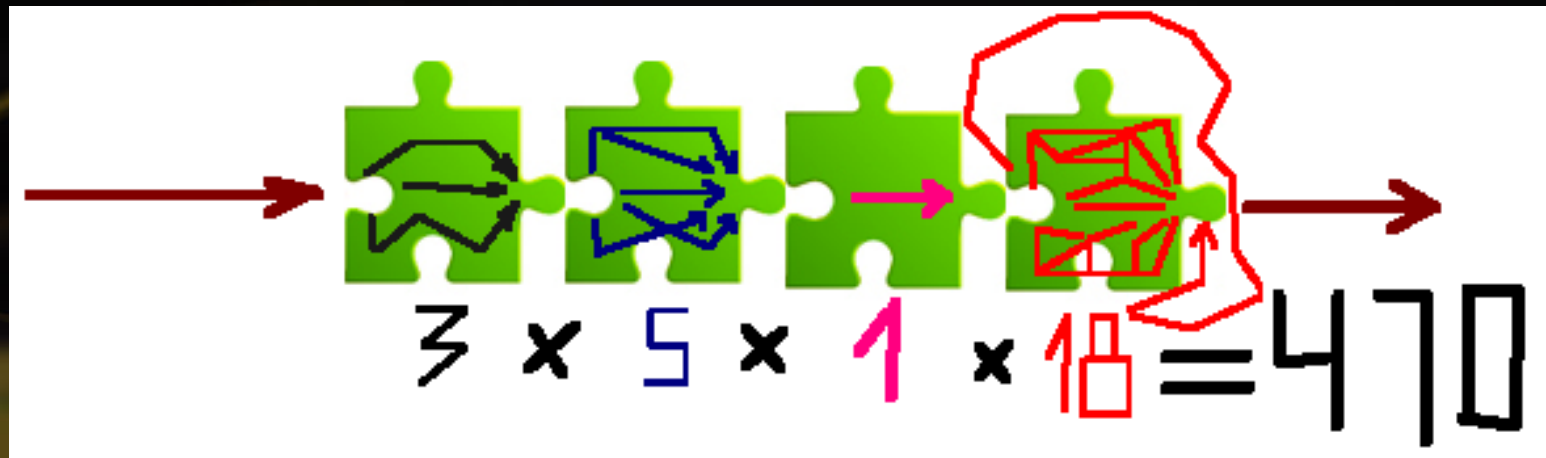
Integrations tests are SCAM



Integrations tests are SCAM



Integrations tests are SCAM

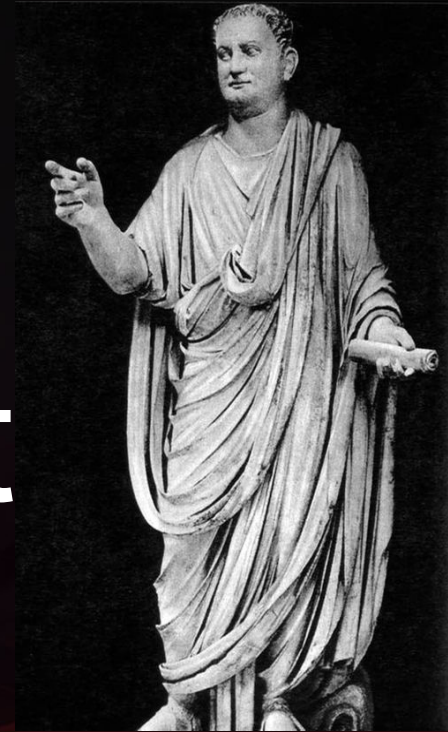


Exponential growth!

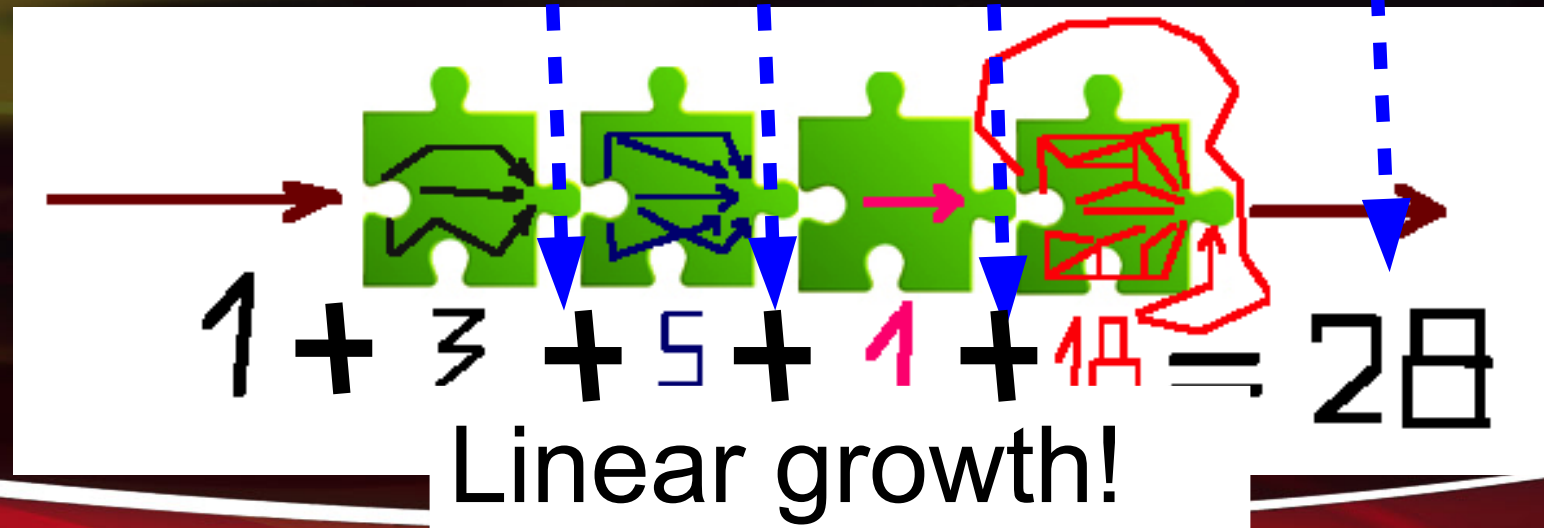
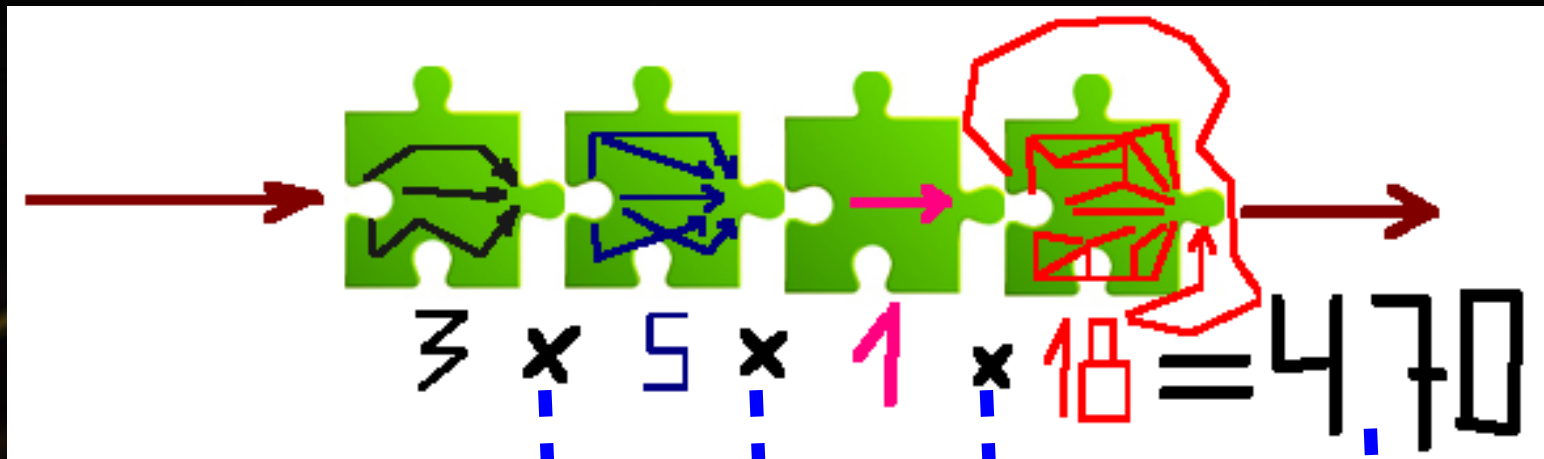


What is the cure?

Divide and ~~conquer~~ test



Divide and conquer



Divide and conquer

$$\prod_{i=1}^N P_i > \sum_{i=1}^N P_i + K$$

UI-tests are evil

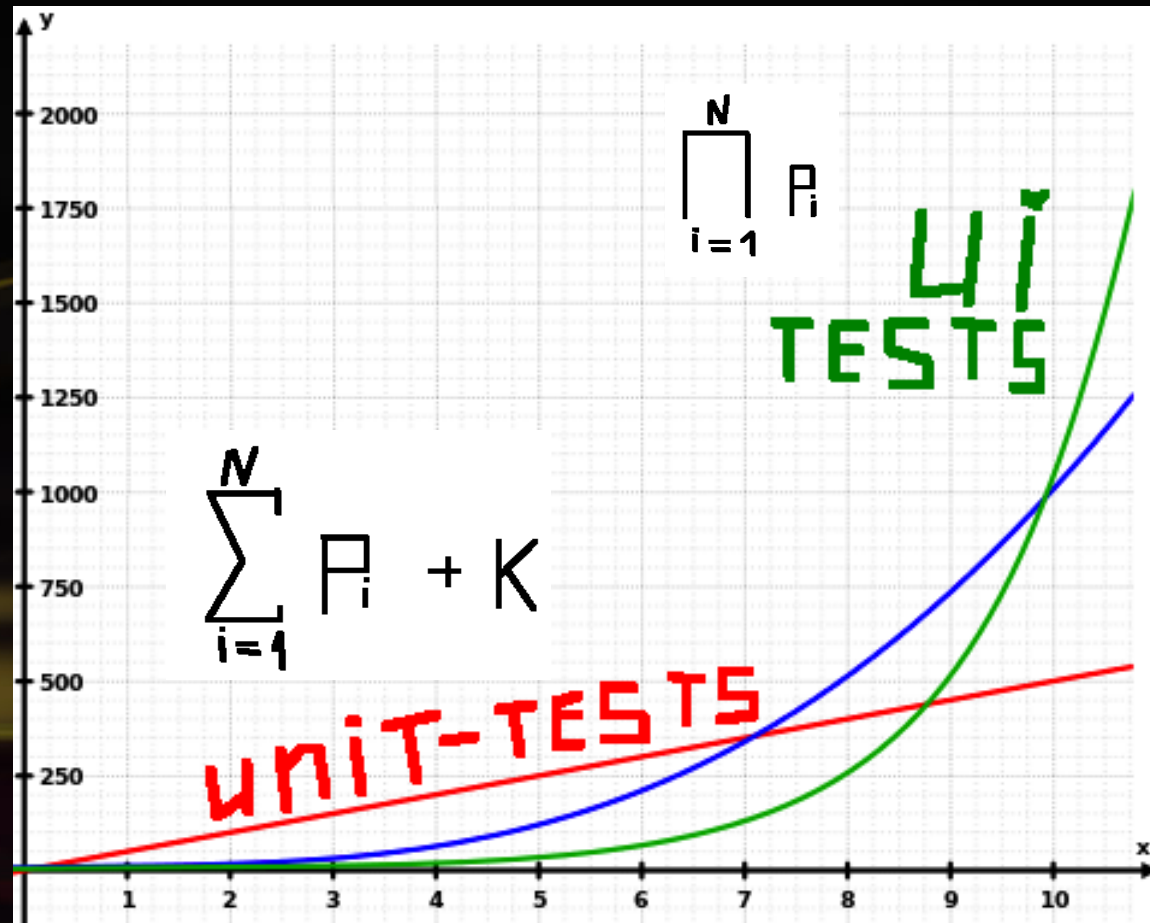
If you have only UI-tests

Divide and conquer

- This is general principle
 - Unit-tests
 - UI-tests

It's profitable
to write
unit-tests

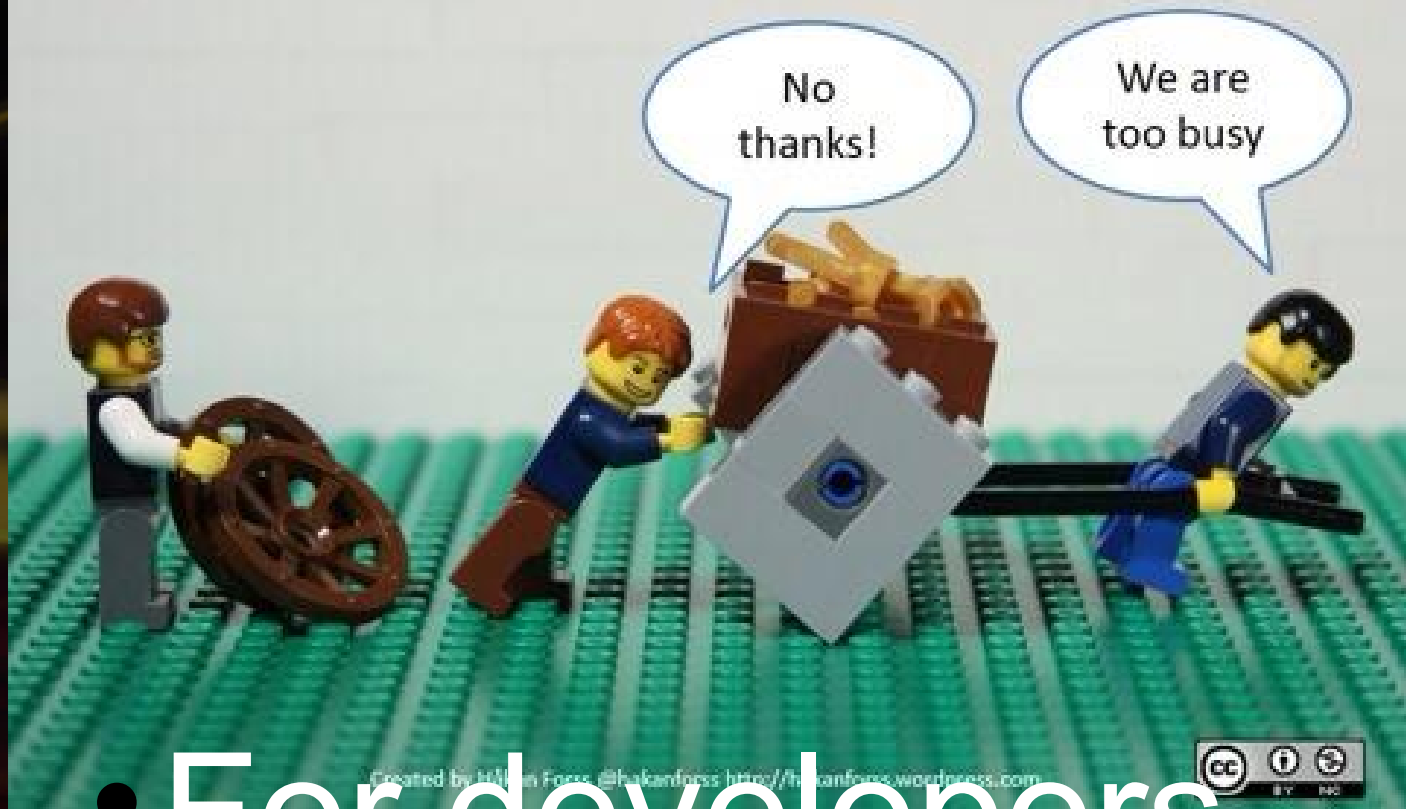
Unit-tests are profitable



- For managers

Unit-tests are profitable

Are you too busy to improve?



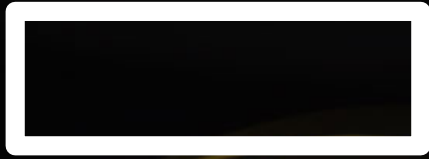
- For developers

We have no time for unit-tests,
we are in a hurry.



You are in a hurry
because you don't have unit-tests

Without unit-tests



With unit-tests



Without unit-tests

dev

test

bugfix

test

With unit-tests

dev

test

bugfix

live!

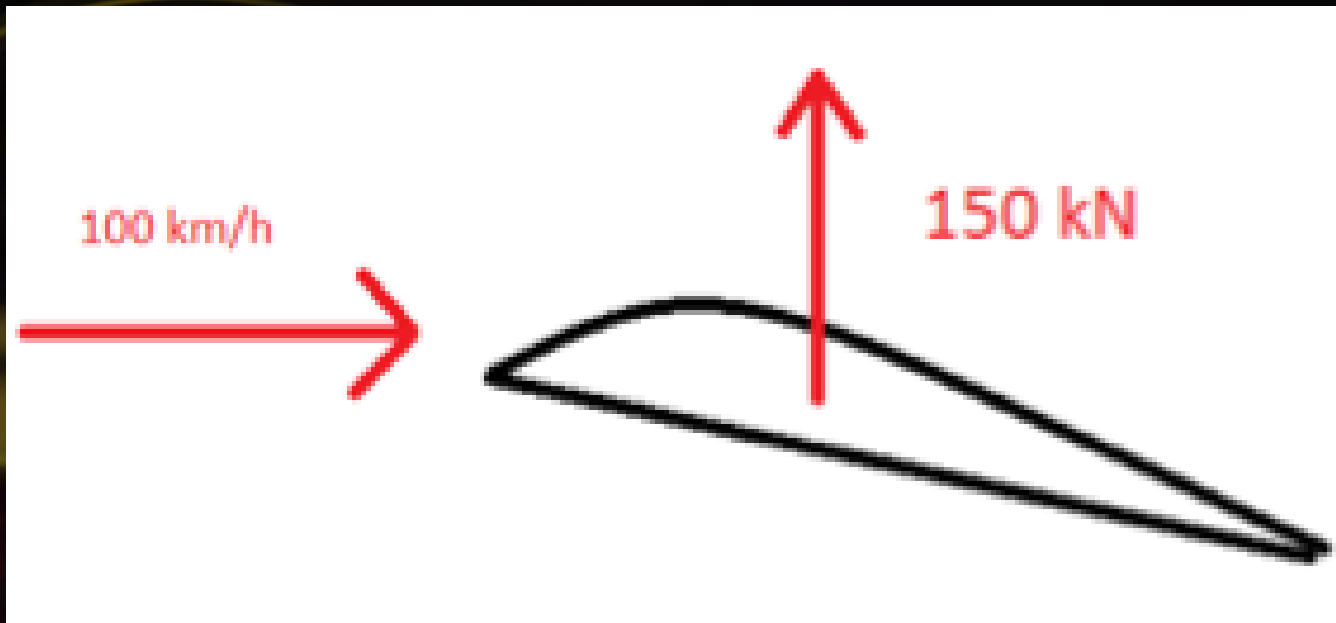
And what about UI-tests?

The same!

Rocket science

Requirements:

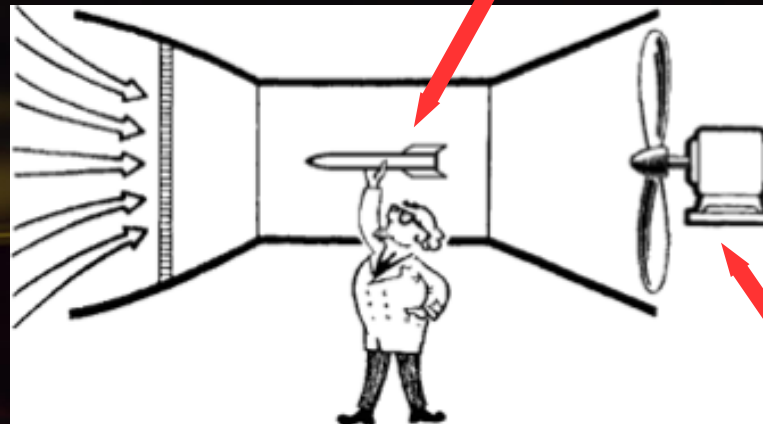
- Wind speed 100 km/h
- Lift wing should be 150 kN



How would you test that?

Rocket science

- Run THE WHOLE plane
- Test the wing SEPARATELY

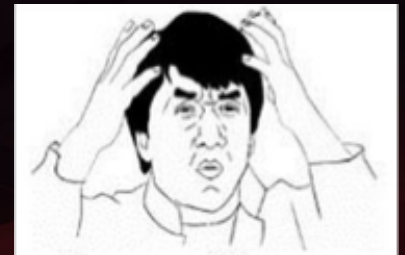


Testable

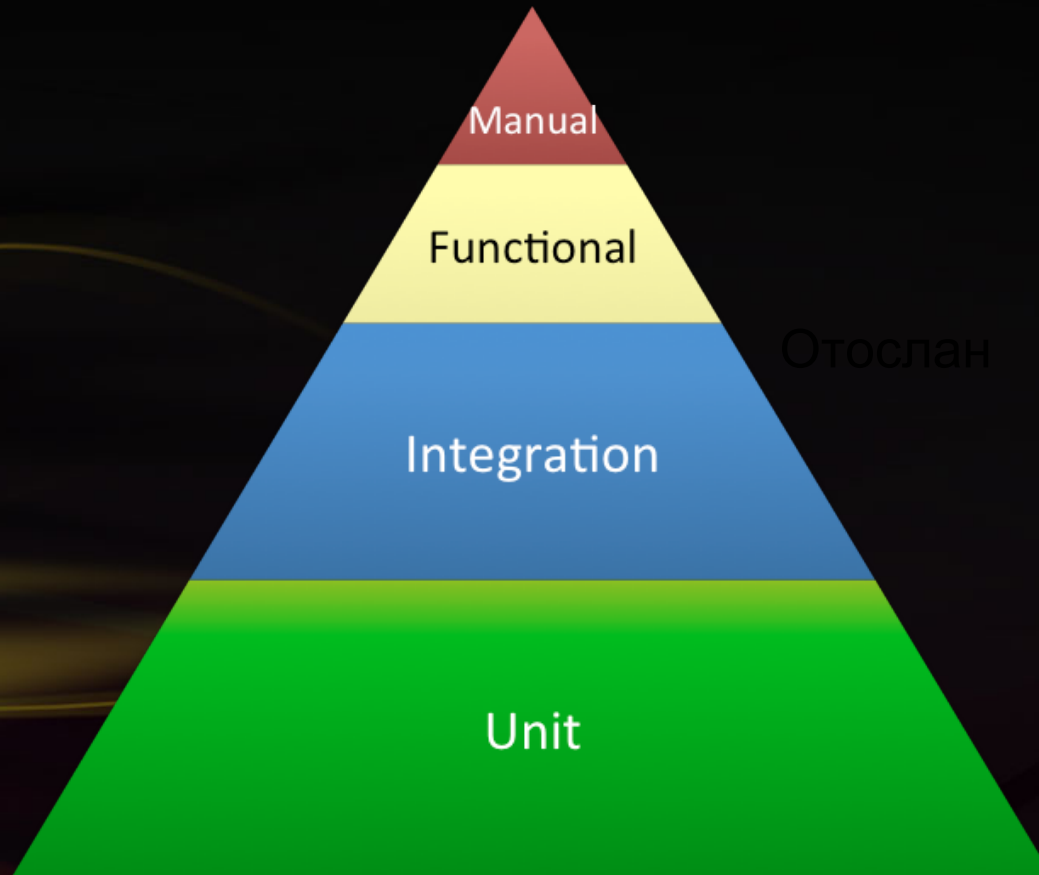
Emulator

The problem of our generation

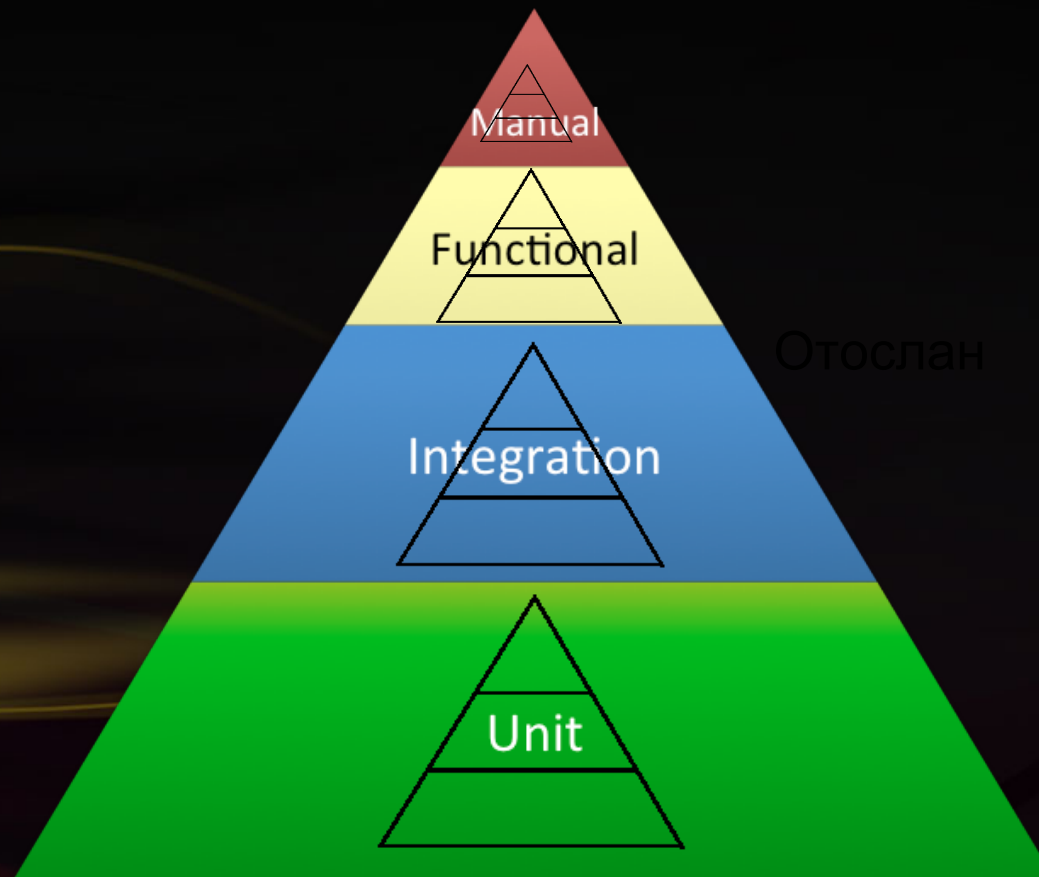
WebDriver allows
too easily
running
THE WHOLE
plane



The testing pyramid



The testing matryoshka



Split and conquer!



LoginTest

- - 1
- - 2
- - 3

Split and conquer



NewOrderTest

- - 1
- - 2

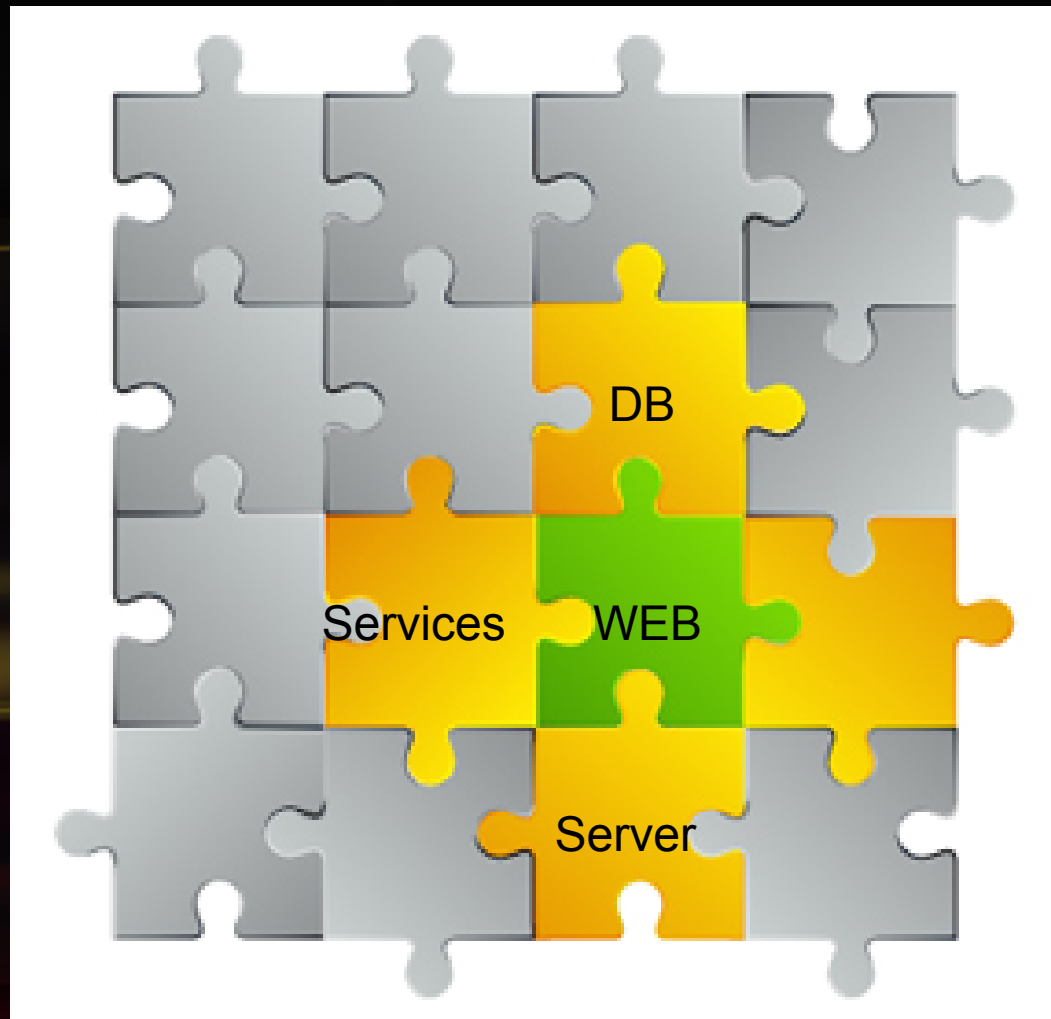
/fastLogin?user=bob&url=/orders/new

• Isn't it dangerous?

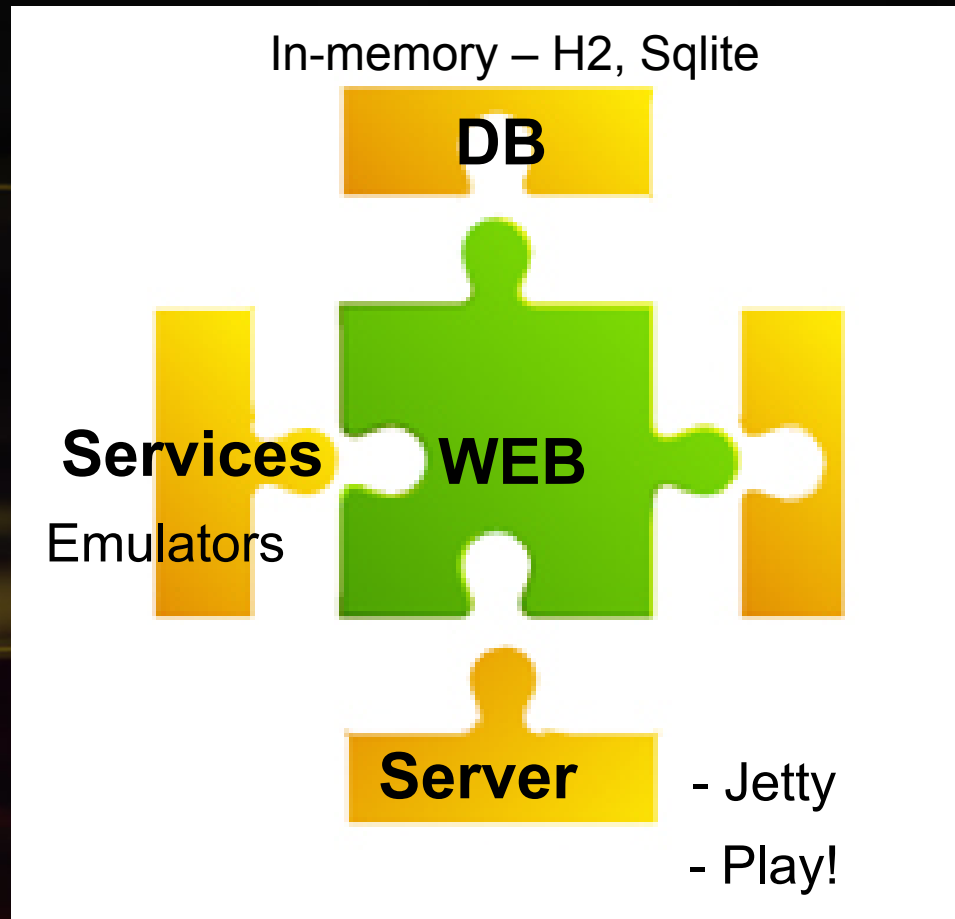
- No!
- Only in tests
- Benefit overweights risk

/fastLogin?user=bob&url=/orders/new

Fast tests



Fast tests



Server *jetty://*

@BeforeClass

```
public static void startServer() {  
    Server server = new Server(8080);  
  
    WebAppContext shop =  
        new WebAppContext("webapp",  
                           "/shop");  
    server.setHandlers(shop);  
  
    server.start();  
}
```

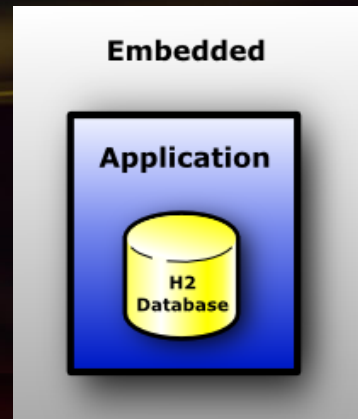
Server has started!
Without installation
In few seconds
Directly from tests

```
@Test  
public void loginTest() {  
    open("http://localhost:8080/shop");  
}
```

In-memory DB

```
db.url=jdbc:oracle:thin:@ip:1521:SID  
db.driver=oracle.jdbc.OracleDriver
```

```
%test.db.url=jdbc:h2:mem:shop  
%test.db.driver=org.h2.Driver
```



For DB structure:

LIQUIBASE

5 things that you are doing wrong in test automation

Parallelize tests

$$\prod_{i=1}^N P_i$$

$$/ K$$

Run tests ONLY in few hours

We have 300 UI Tests that run for 4 minutes!

- In-memory DB
- Test data
- Emulated services
- fastLogin

- Video:
- <https://vimeo.com/73128965>

• PageObjects

An element
should not be used
in 50 tests!

TestNG vs JUnit

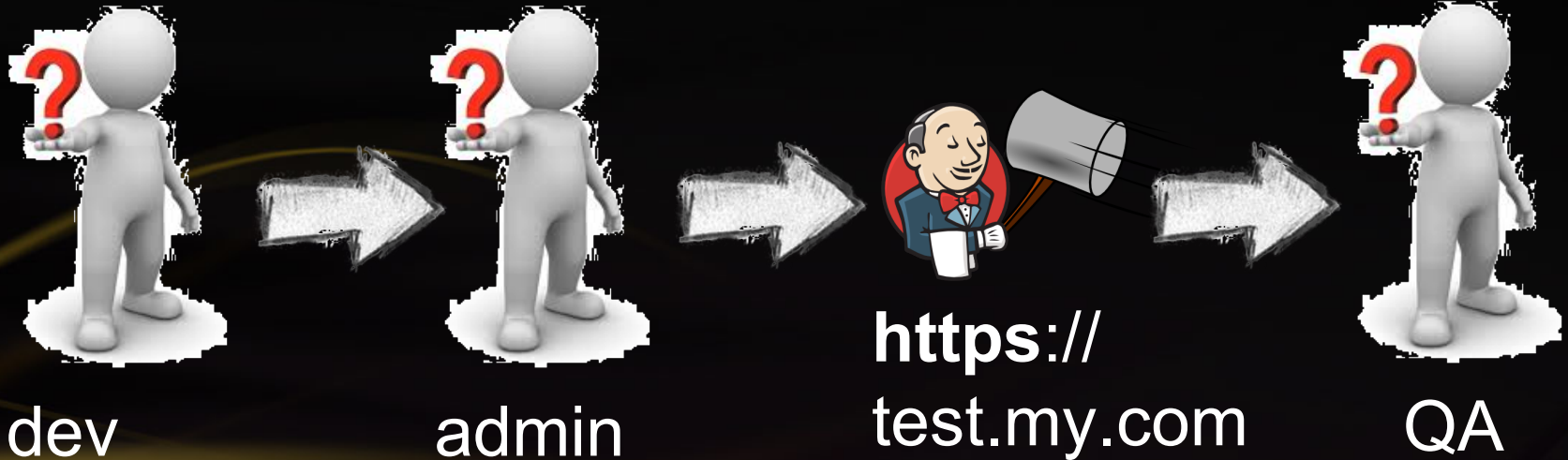
Order of tests

- Bad! - fastLogin

Parametrized tests

- UI-tests – only 1-2 green path
- Other 10010 combinations - unit-tests

Invalid SSL certs



Invalid SSL certs



Invalid SSL certs

- Run tests BEFORE commit!
- Run tests on <http://localhost:8080>
- Commit tests with code in one GIT repo
 - > ant unit-tests
 - > ant ui-tests

How to win StaleElementException?

**Use right
tools!**

Tool can do
many things



Tool solves
my problems



selenide.org

~~StaleElementException~~

~~WebDriverWait.presenceOfElementLocated~~

~~Implicit timeout~~

Use right tools!



Rifle -

Is a newbie weapon

Weapon of professional...

Is knife!

To be master

Video:

<https://dl.dropboxusercontent.com/u/2856647/aikido.mp4>

- Master is not who knows 10010 frameworks
- Master is who can solve problems easily, using minimum energy.

Andrei Solntsev

//codeborne



selenide.org

andrei.solntsev@gmail.com
twitter.com/asolntsev