

# Экономически эффективный процесс тестирования

Андрей Солнцев

//codeborne



# Интернет-банк Санкт-Петербург

- [idemo.bspb.ru](http://idemo.bspb.ru)

## Сроки

Релиз “для своих”

1 мес.

Публичный релиз

5 мес.

# Ресурсы

тест-аналитиков

0

автоматизаторов

0

тестировщиков

0

разработчиков

4

# Автотесты

ЮНИТ-ТЕСТЫ

1200 шт

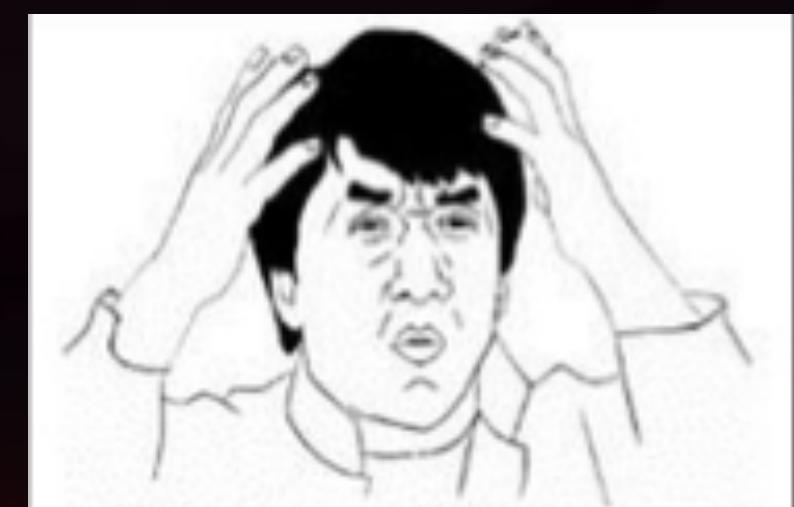
10 сек

UI ТЕСТЫ

120 шт

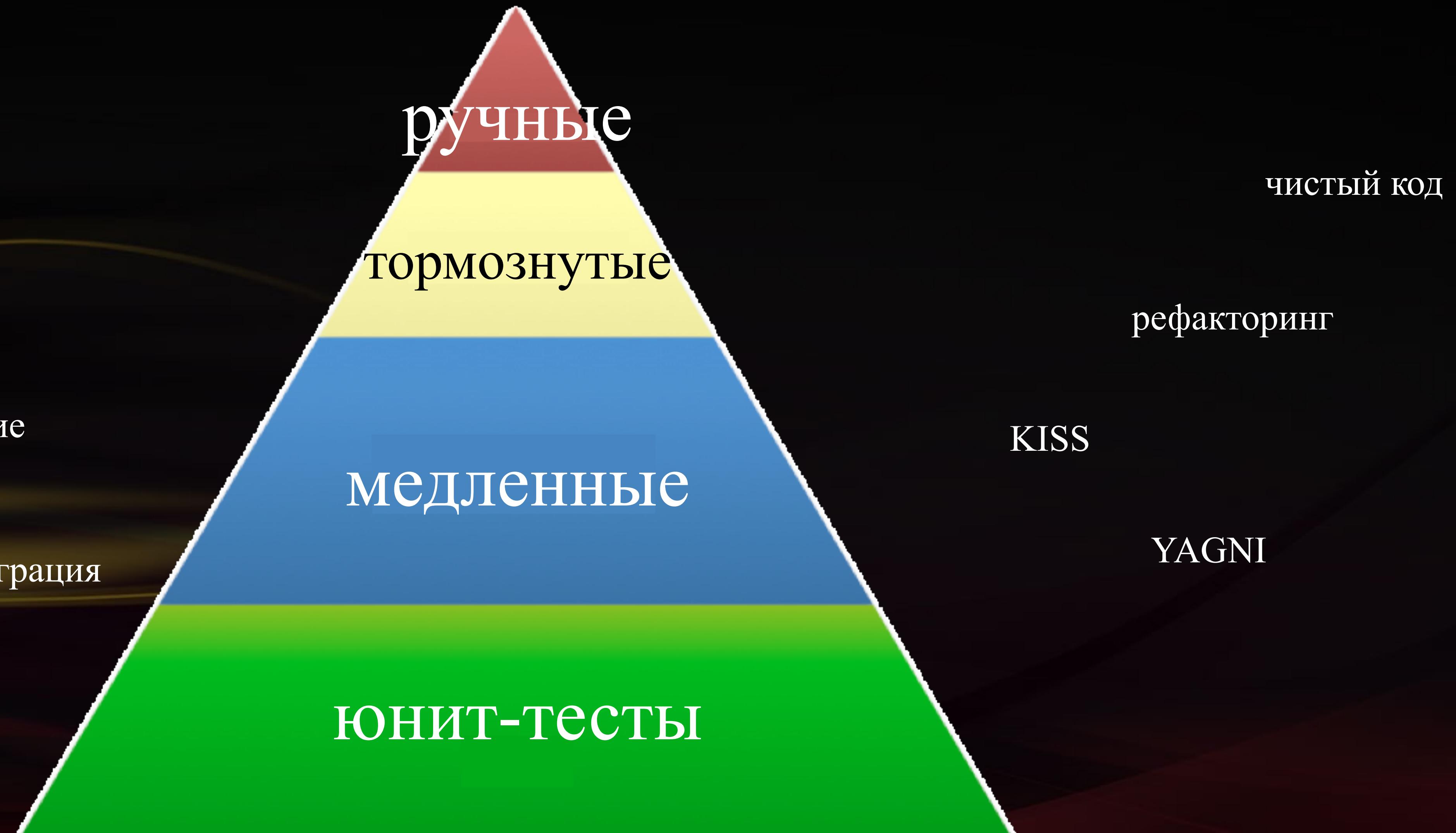
4 мин

Как это возможно?

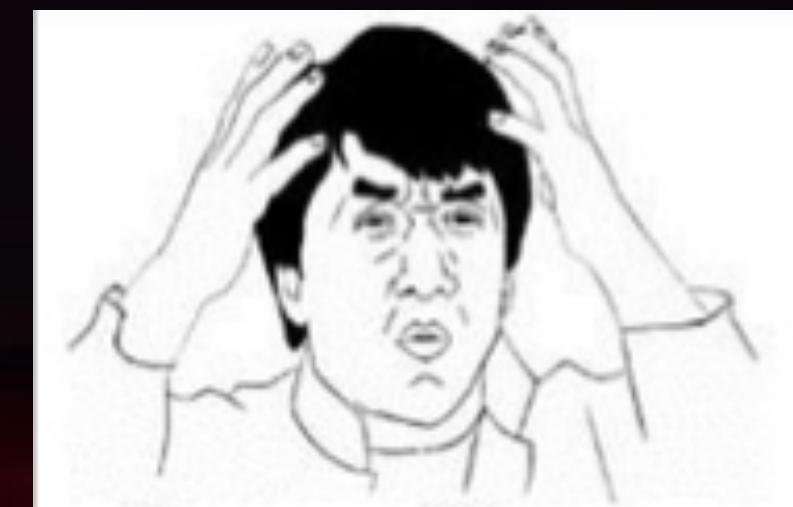


В чём секрет?

# Вот секрет:



Почему  
никто  
его  
не использует?



# Потому что

Кто-то  
когда-то  
решил,  
что  
разработчики дорогие,  
а тестировщики на дороге валяются.

# И поэтому

Чтобы сэкономить,  
Пусть разработчики фигачат код,  
а всякое г....  
спихнём на тестировщиков!



# Локальная оптимизация!

аналитик

разраб

тестировщик

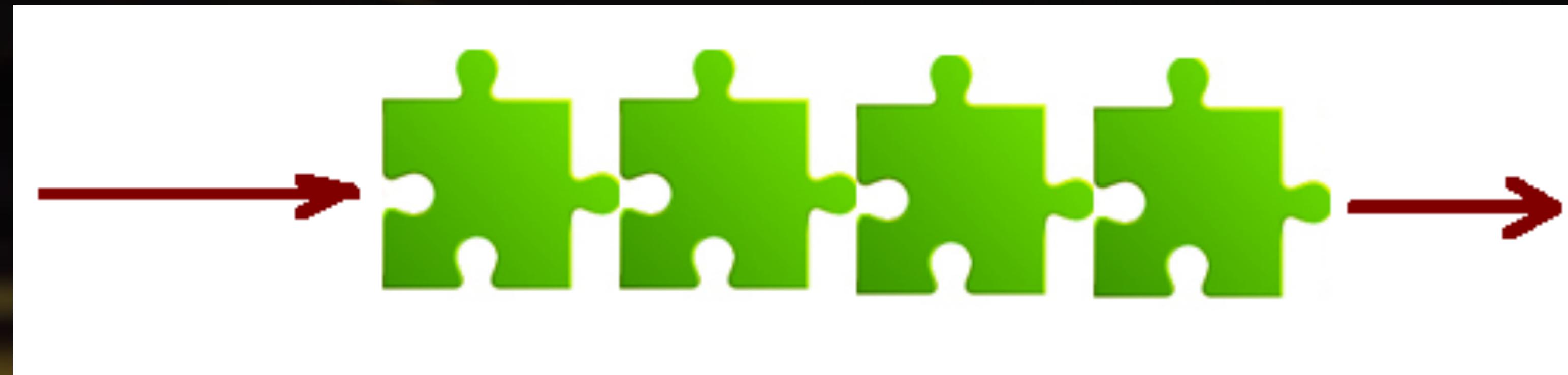


больше  
спеков!

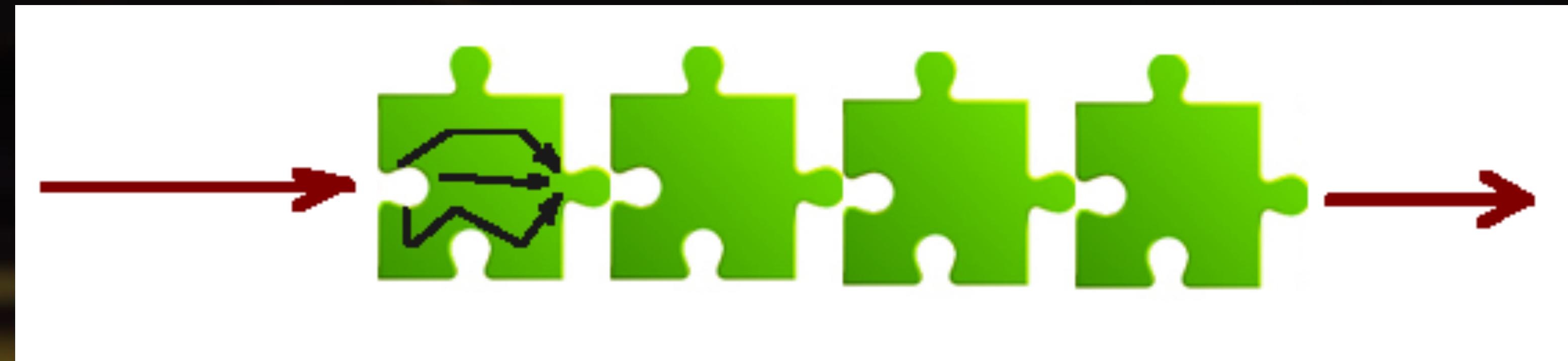
больше  
кода!

больше  
тестов!

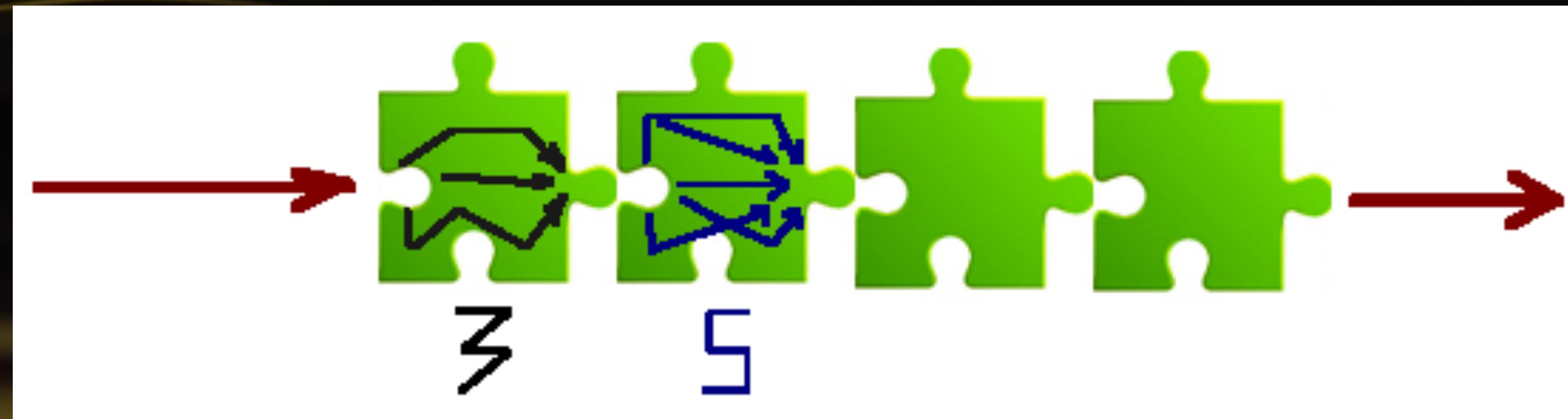
# Интеграционные тесты - ОБМАН



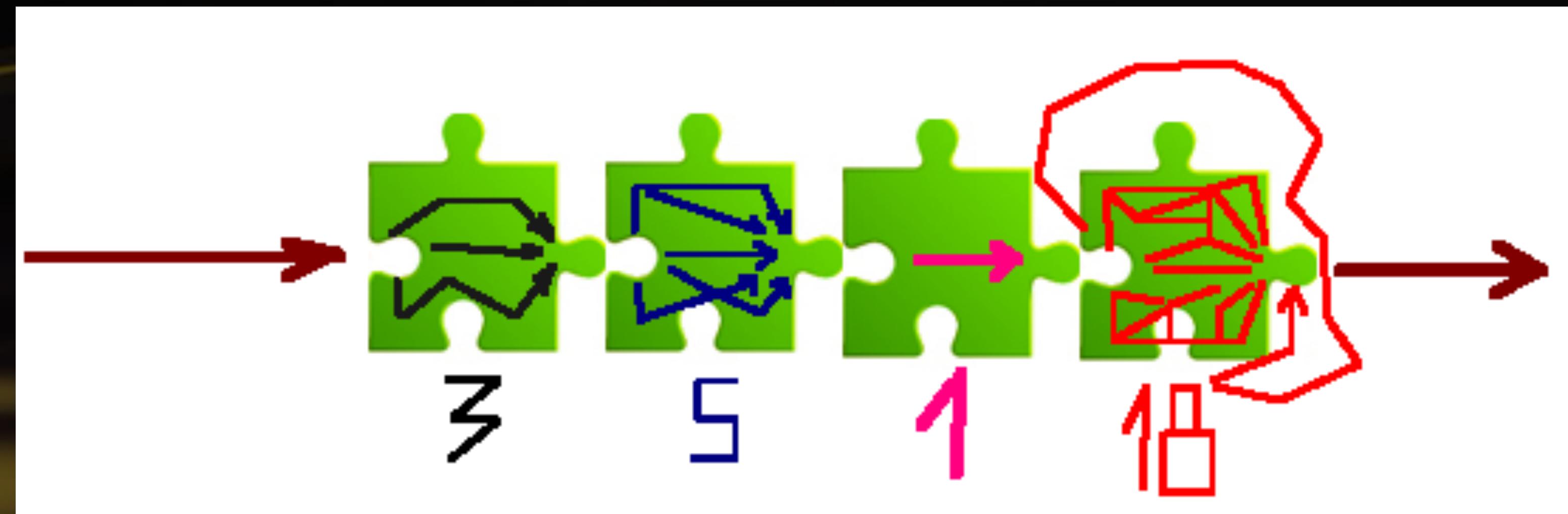
# Интеграционные тесты - ОБМАН



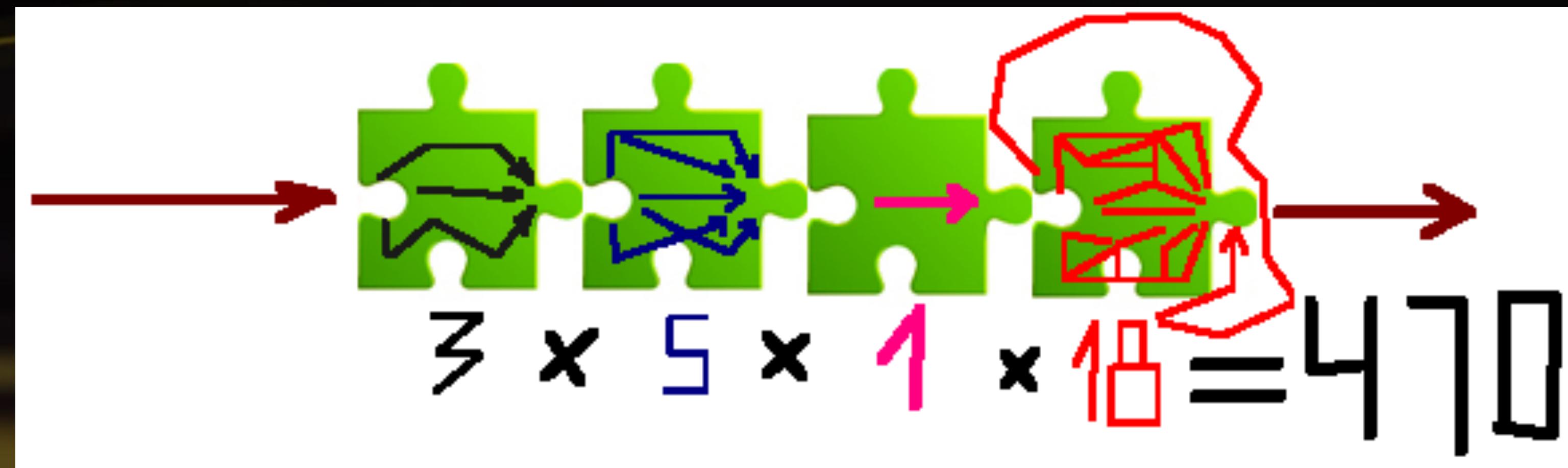
# Интеграционные тесты - ОБМАН



# Интеграционные тесты - ОБМАН



# Интеграционные тесты - ОБМАН



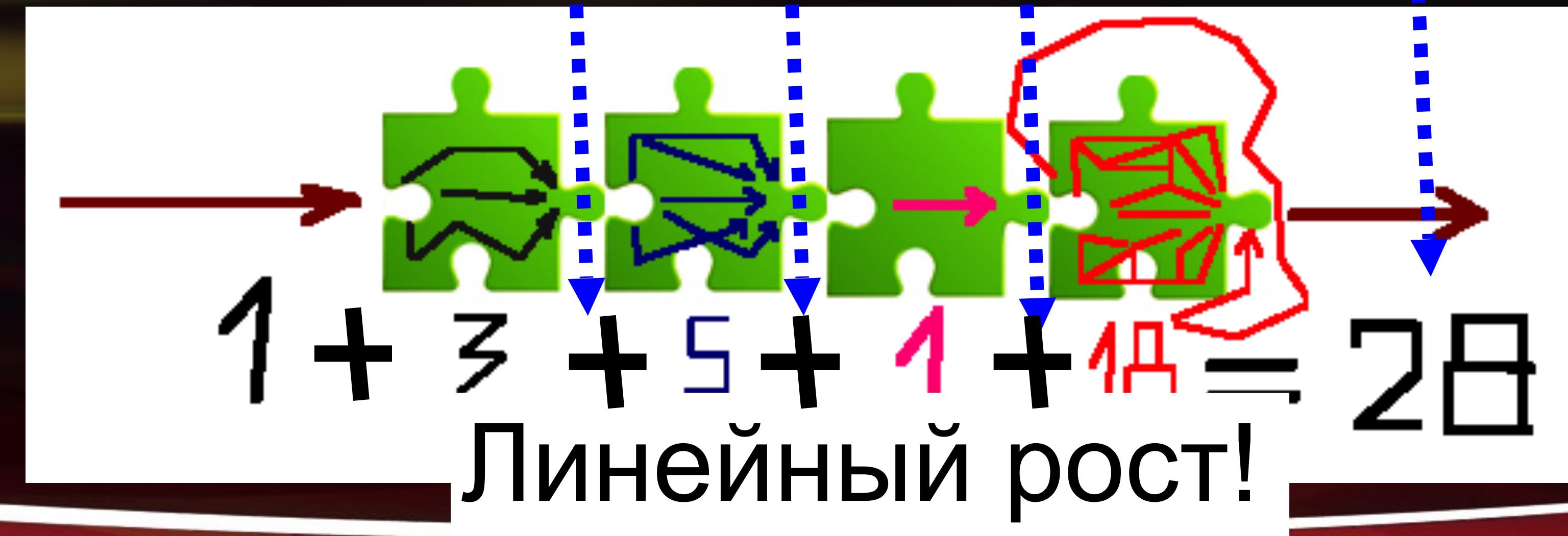
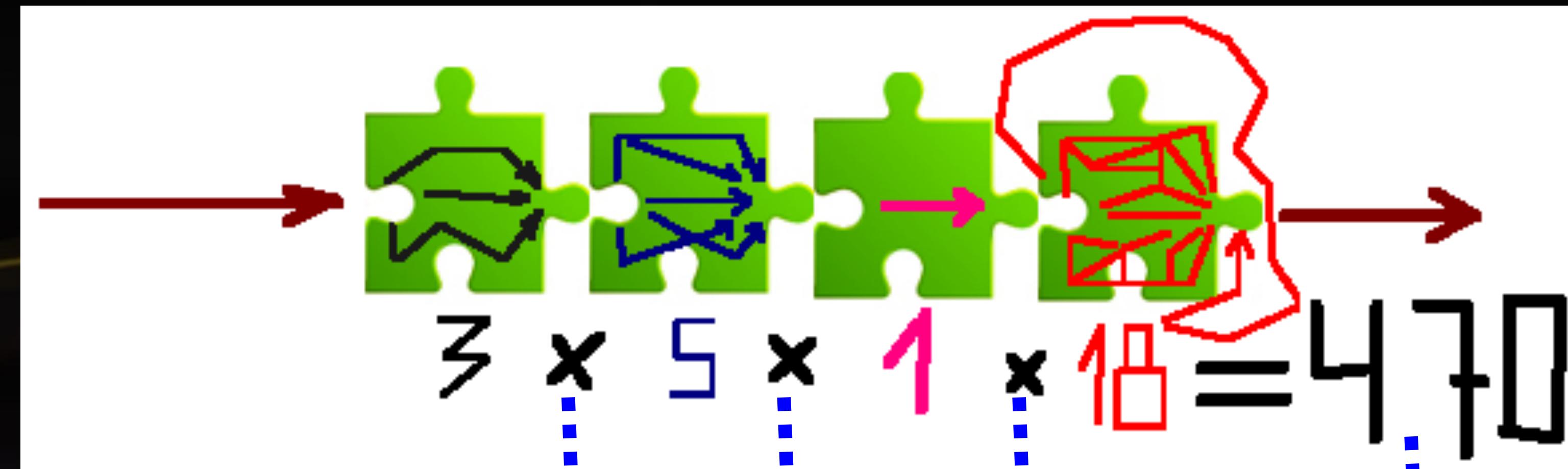
Экспоненциальный рост!

Что же делать?

• Разбивай  
• И  
• Тестируй!



# Разбивай и тестируй!

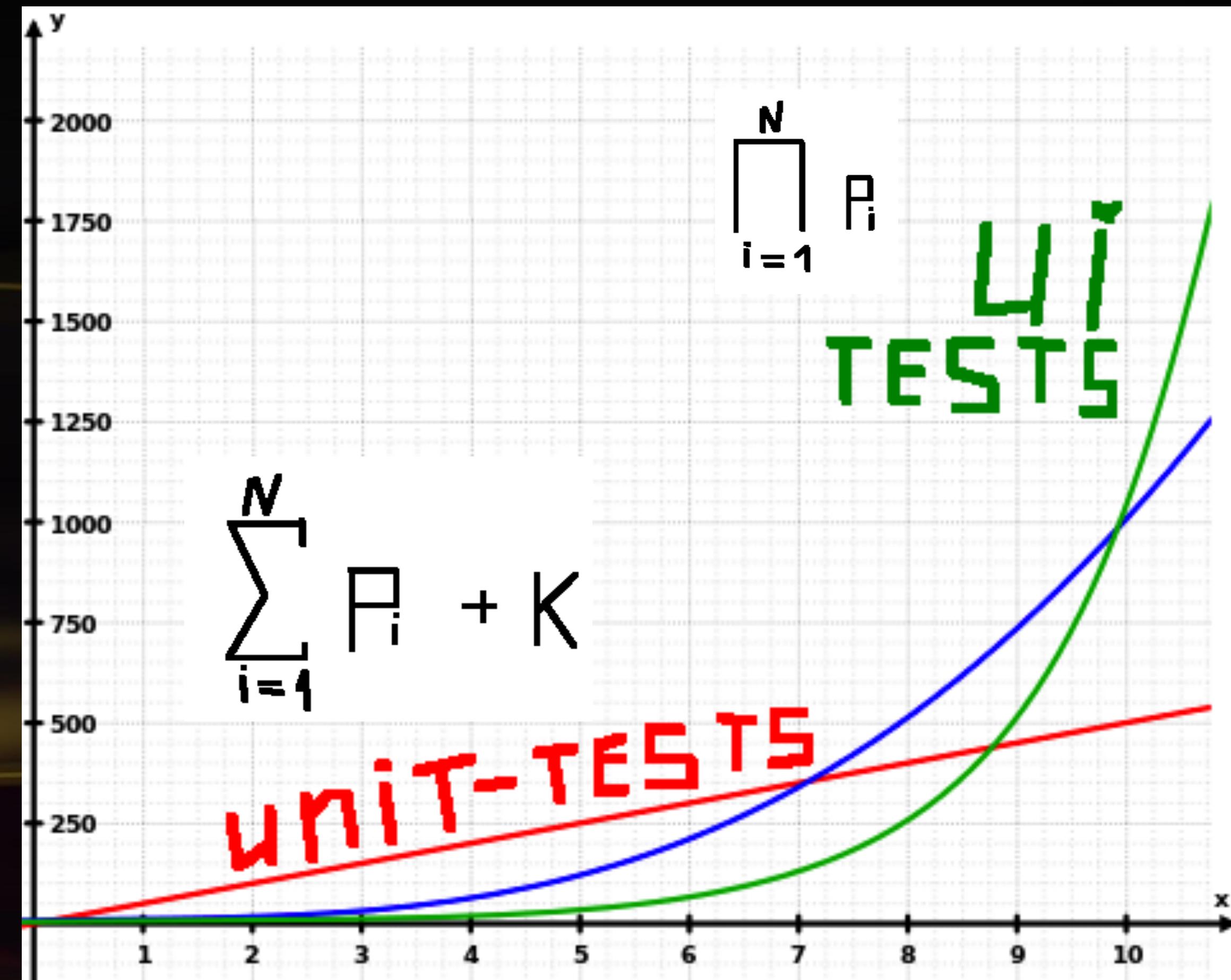


# Разбивай и тестируй!

- Это общий принцип
  - - Юнит-тесты
  - - UI-тесты

- Писать юнит-тесты выгодно

# ЮНИТ-ТЕСТЫ - ВЫГОДНО



- Менеджерам

- Чак Норрис не ходит на охоту

- охота = возможность неудачи



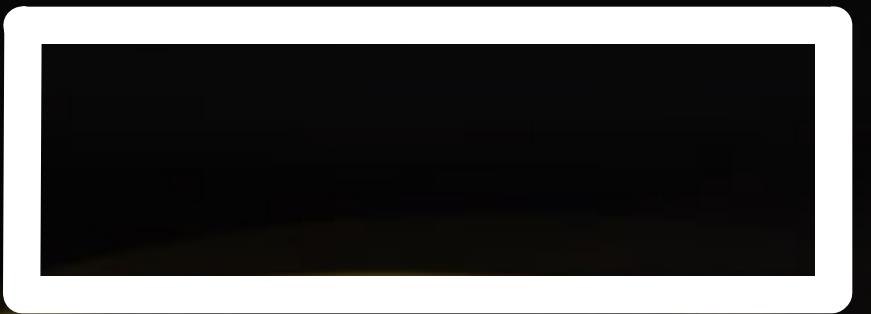
- Чак Норрис ходит убивать!

Нам некогда писать юнит-тесты,  
мы и так не успеваем.



Вы потому и не успеваете,  
что у вас нет юнит-тестов!

# Без юнит-тестов



# С юнит-тестами



# Без юнит-тестов

разработка

тест

багфикс

тест

# С юнит-тестами

разработка

тест

багфикс

в бой!



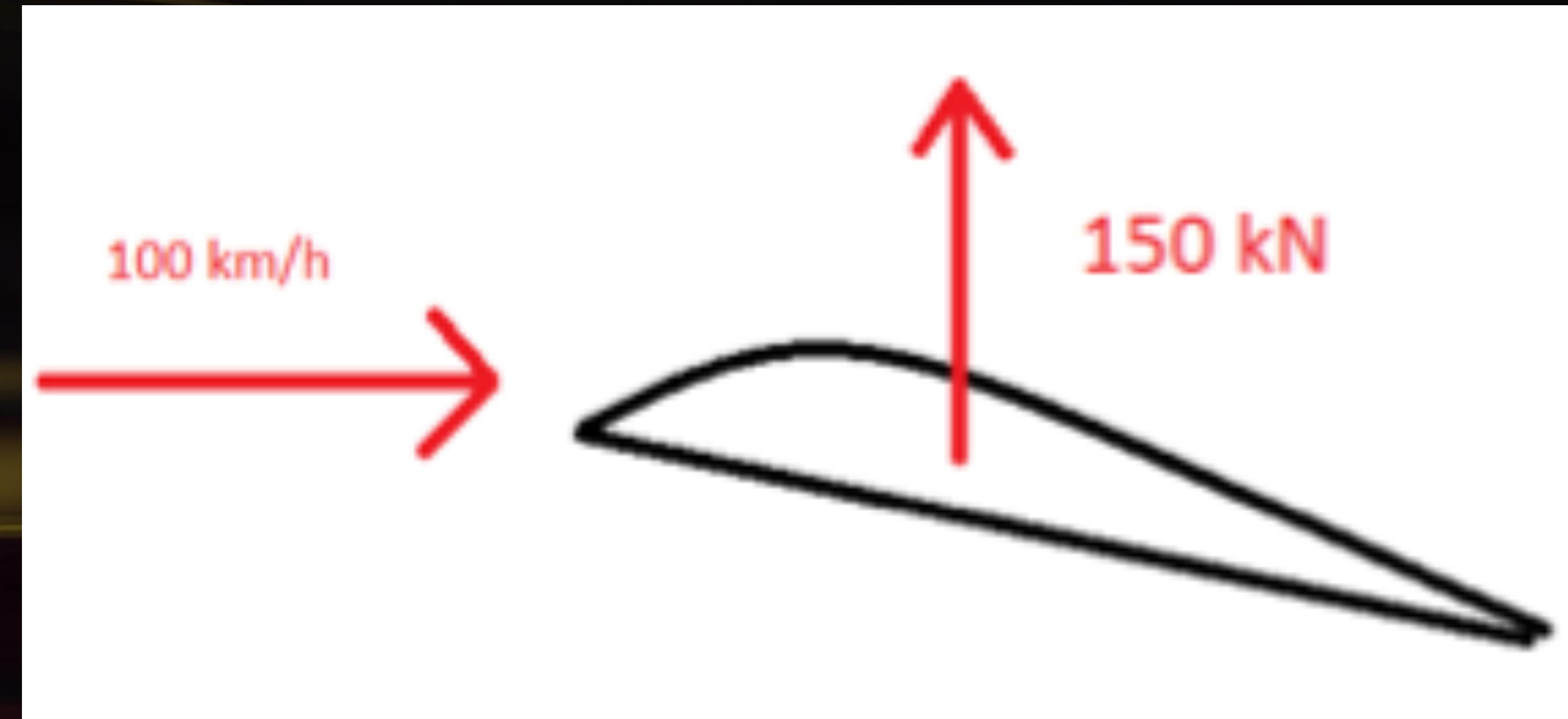
Юнит-тест  
делает из мальчика мужчину

- А что же с UI-тестами?
- То же самое:
- Разбивай и тести!

# Rocket science

Требования:

- При скорости ветра 100 км/ч
- Подъёмная сила крыла должна быть 150 кН



Как бы вы стали это тестировать?

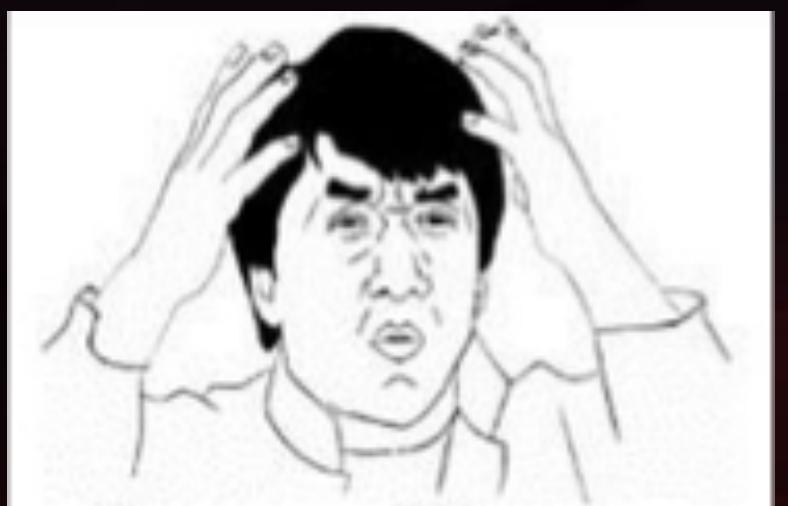
# Rocket science

- Запустить ВЕСЬ самолёт
- Протестировать крыло ОТДЕЛЬНО

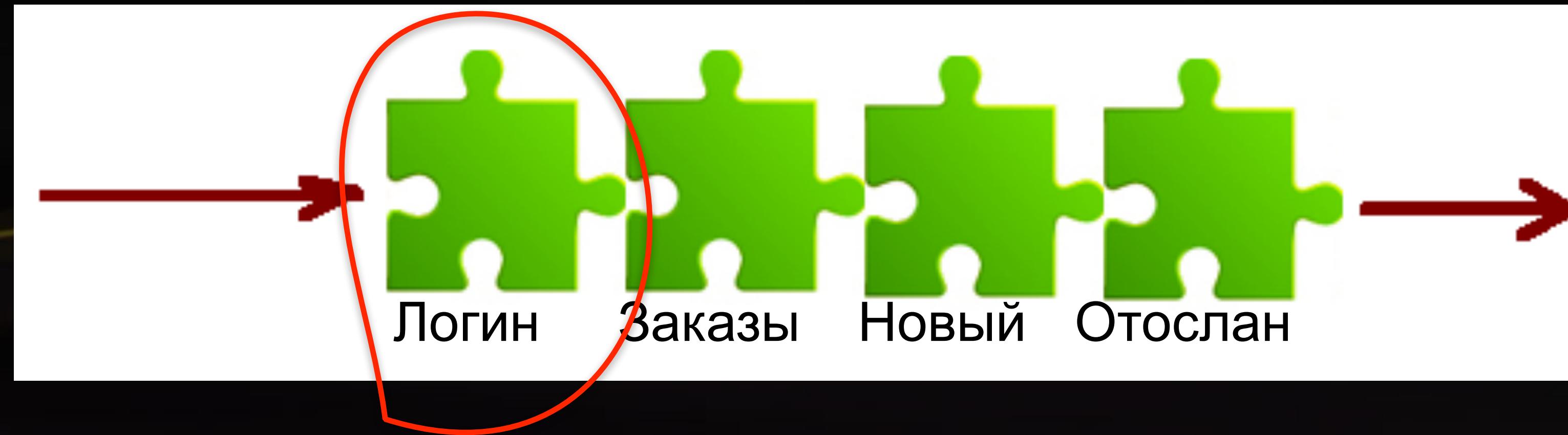


# Проблема нашего поколения

WebDriver позволяет  
слишком легко  
запустить  
ВЕСЬ  
самолёт



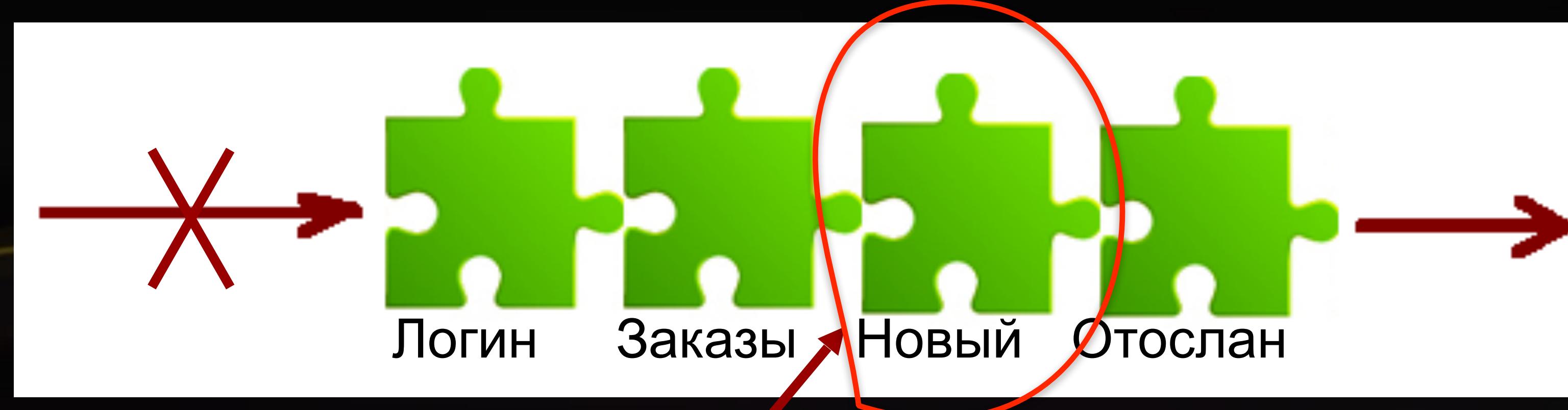
# Разбивай и тестируй!



LoginTest

- 1 тест
- 2 тест
- 3 тест

# Разбивай и тестируй!



NewOrderTest

- 1 тест
- 2 тест

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

# • А это не опасно?

- Нет!
- Только в teste
- Выгода больше риска

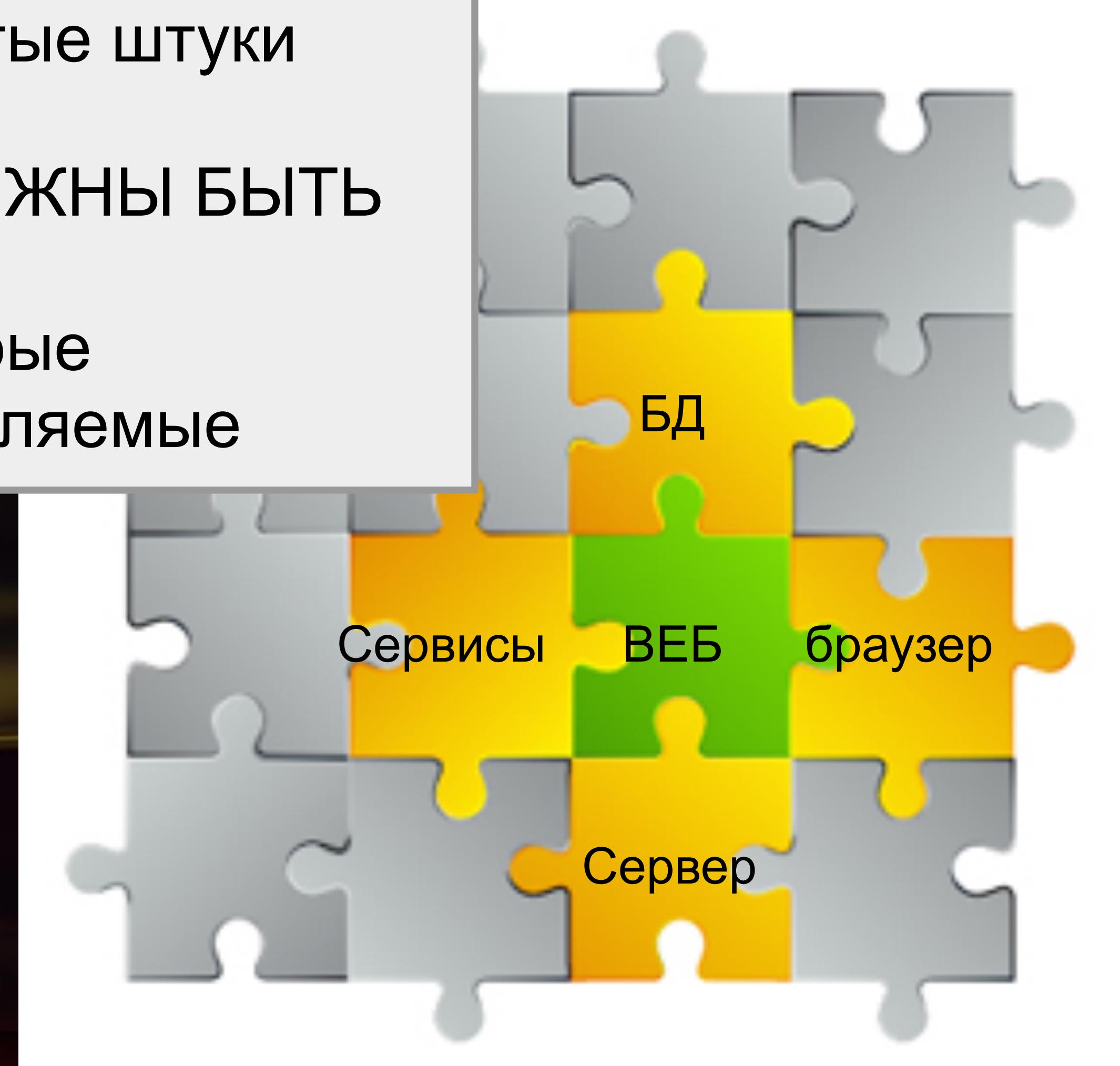
/fastLogin?user=bob&url=/orders/new

# Быстрые тесты

Жёлтые штуки

должны быть

- быстрые
- управляемые



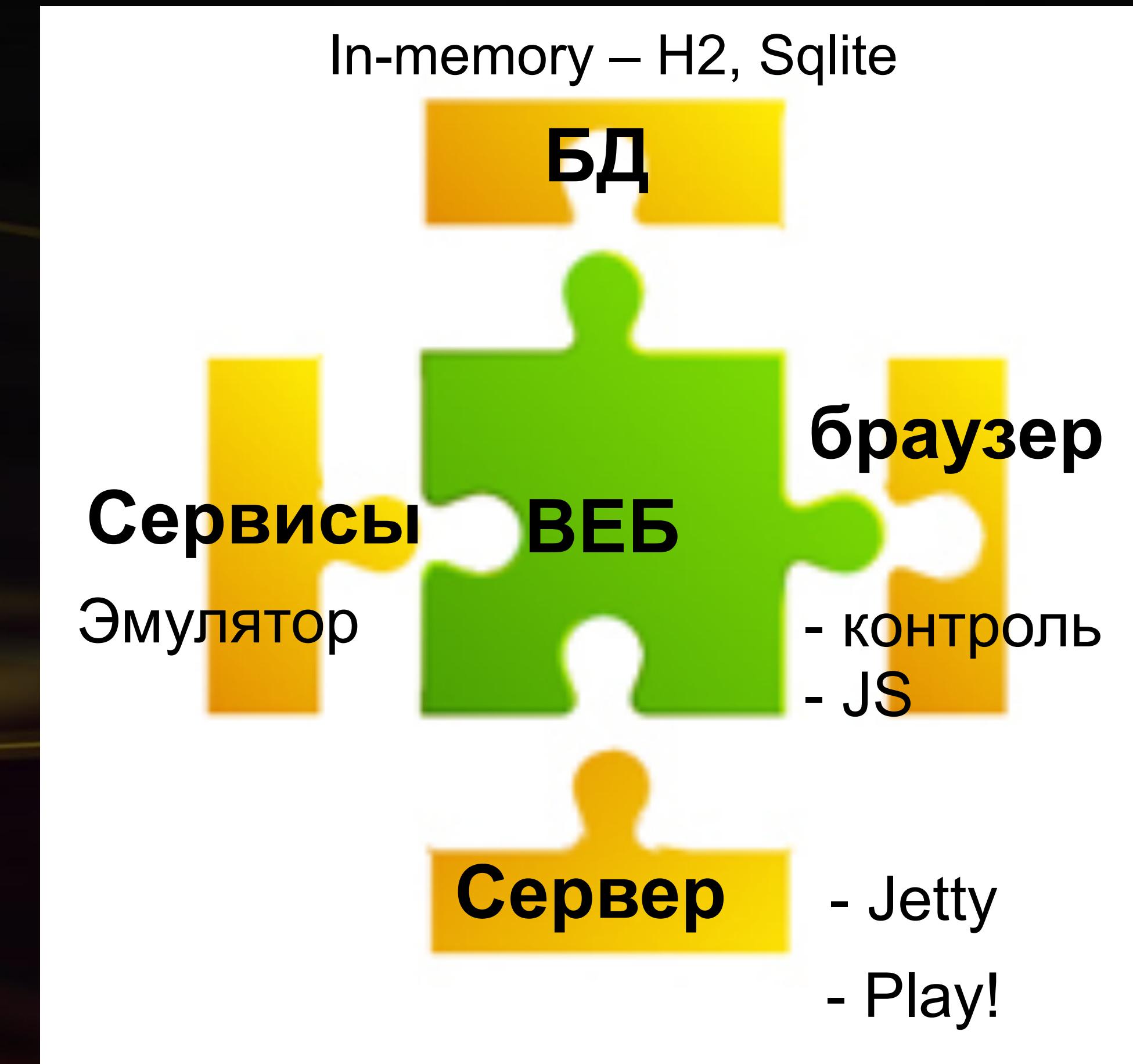
# Суть тестов

Arrange

Act

Assert

# Быстрые тесты



# CepBep **jetty://**

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

**jetty://**

Сервер запустился!

Без всякой инсталляции

В считанные секунды

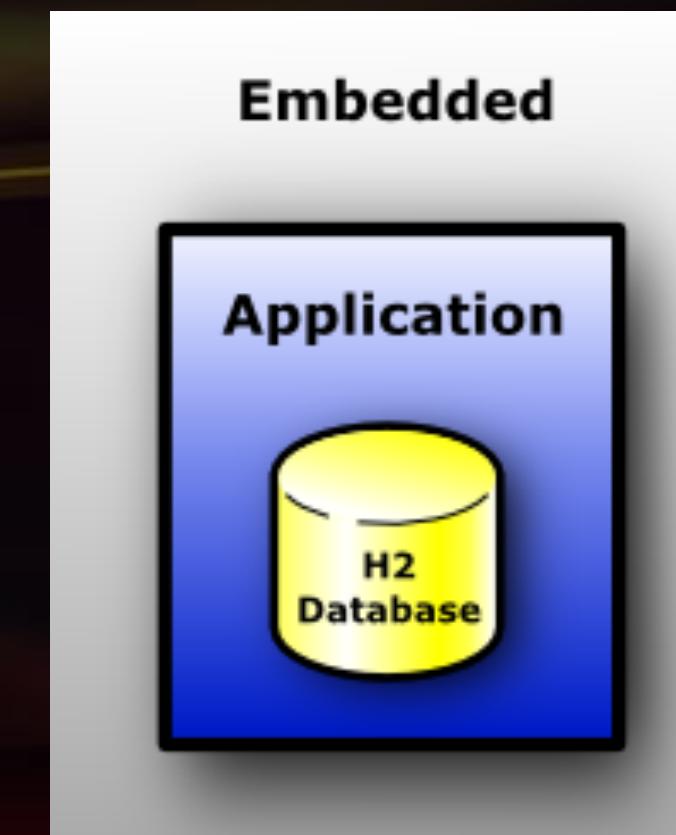
Прямо из тестов

```
@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
```

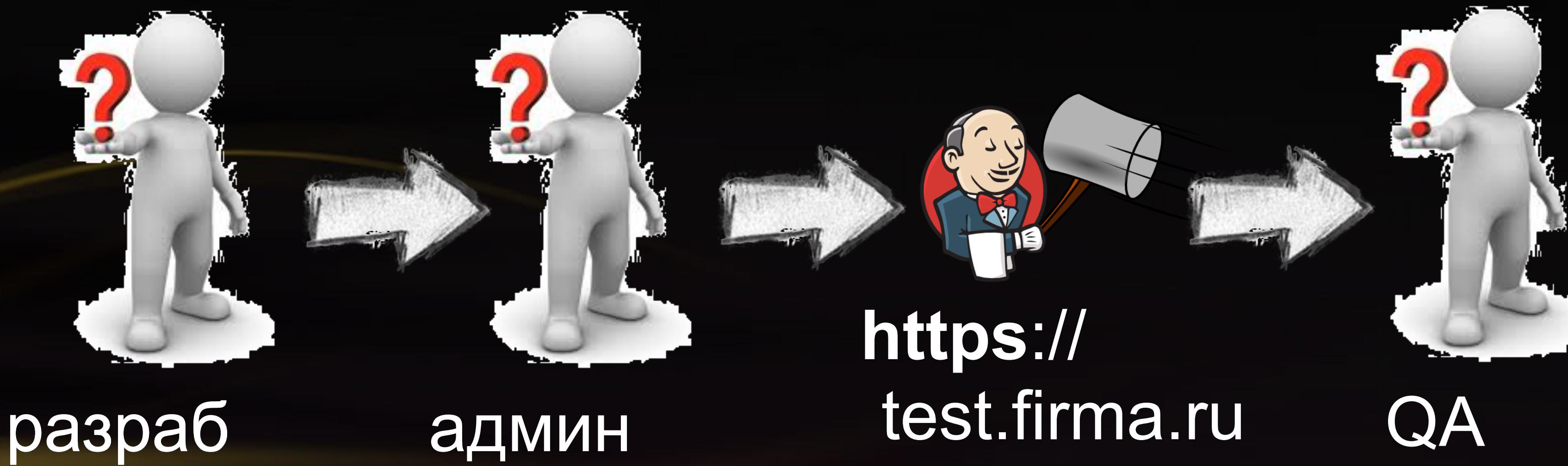
```
%test.db.url=jdbc:h2:mem:shop  
%test.db.driver org.h2.Driver
```



Структура ДБ:

LIQUI<sup>DB</sup>BASE

# Неэффективный процесс



# Эффективный процесс



# Эффективный процесс

- Возможность запускать
  - Тесты ДО КОММИТА!
  - Тесты на <http://localhost:8080>
  - Тесты вместе с кодом в GIT
    - > ant unit-tests
    - > ant ui-tests

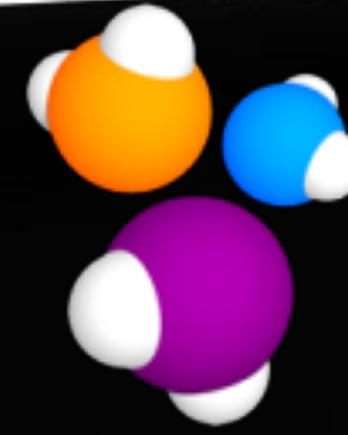
# Как победить StaleElementException?

Используй правильные  
инструменты!

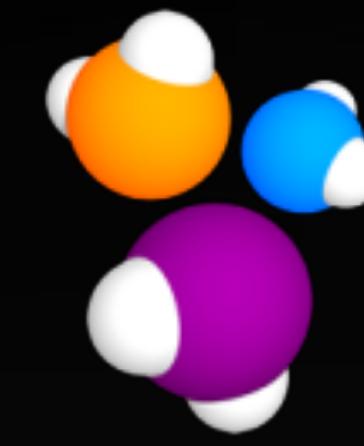
Инструмент  
многое может



Инструмент  
решает мои  
проблемы



```
@Test  
public void userCanLogin() {  
    open("/login");  
    $("#submit").click();  
    $(".menu").shouldHave(text("Есть чо?"));  
}
```



ru.selenide.org

~~StaleElementException~~

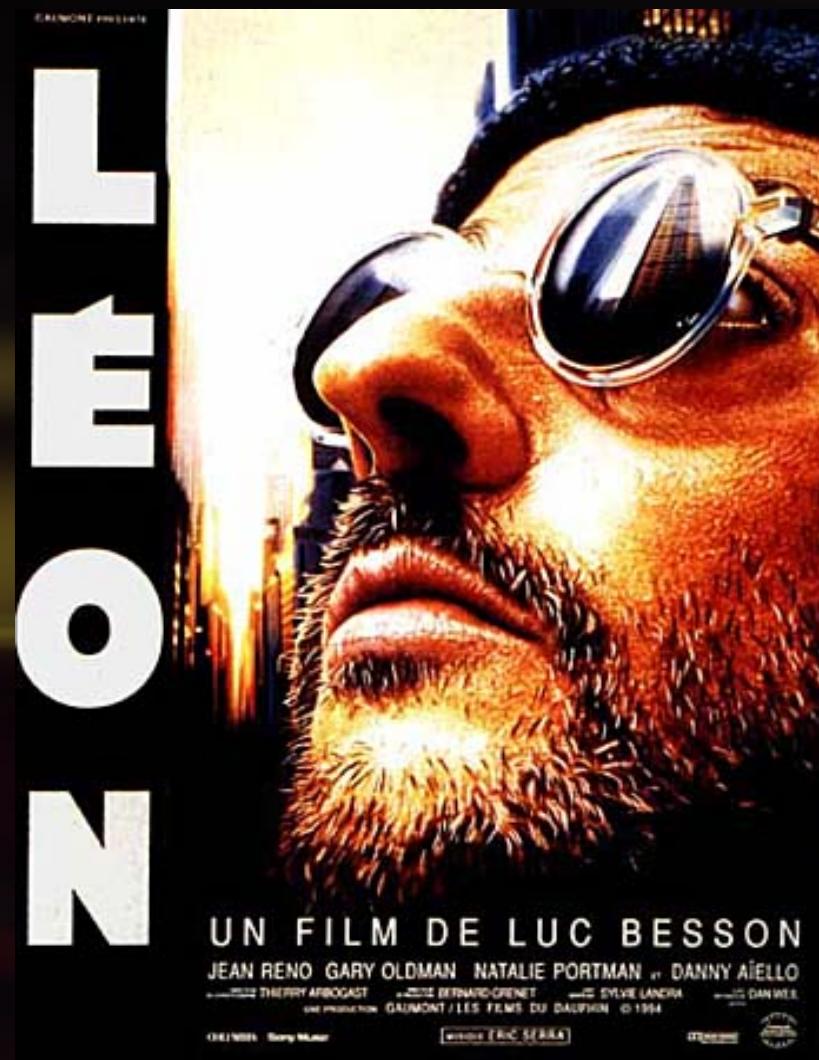
~~WebDriverWait.until(ExpectedConditions.elementLocated~~

~~Implicit timeout~~

Видео:

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

# Используй правильные инструменты!



Винтовка -  
оружие новичка

Выбор профессионала -  
нож!

# Подытожим

- Быстрых - много, медленных - мало.
  - Убеди этих ленивых ушлёпков писать юнит-тесты!
- Arrange. Act. Assert.
  - Эмулируй ненадёжные зависимости.
- Выбирай простые и надёжные средства
  - а не всемогущие фреймворки, “решающие ваши проблемы”.

# Что значит мастер

Видео:

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

- Мастер не тот, кто знает 100500 технологий
- А тот, кто решает задачу легко, потратив минимум энергии

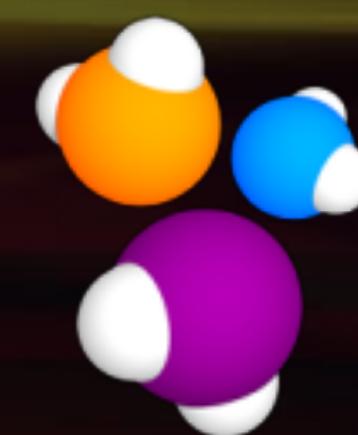
# Андрей Солнцев

//codeborne



@asolntsev

[andrei.solntsev@gmail.com](mailto:andrei.solntsev@gmail.com)



[selenide.org](http://selenide.org)

