

Автоматизированное тестирование веб-приложений

Вадим Кадученко

Проблемы

От ошибок никто не
застрахован

Ошибки появляются
ВНЕЗАПНО

Проблемы

Срочное исправление
ошибок нервирует и
отнимает время

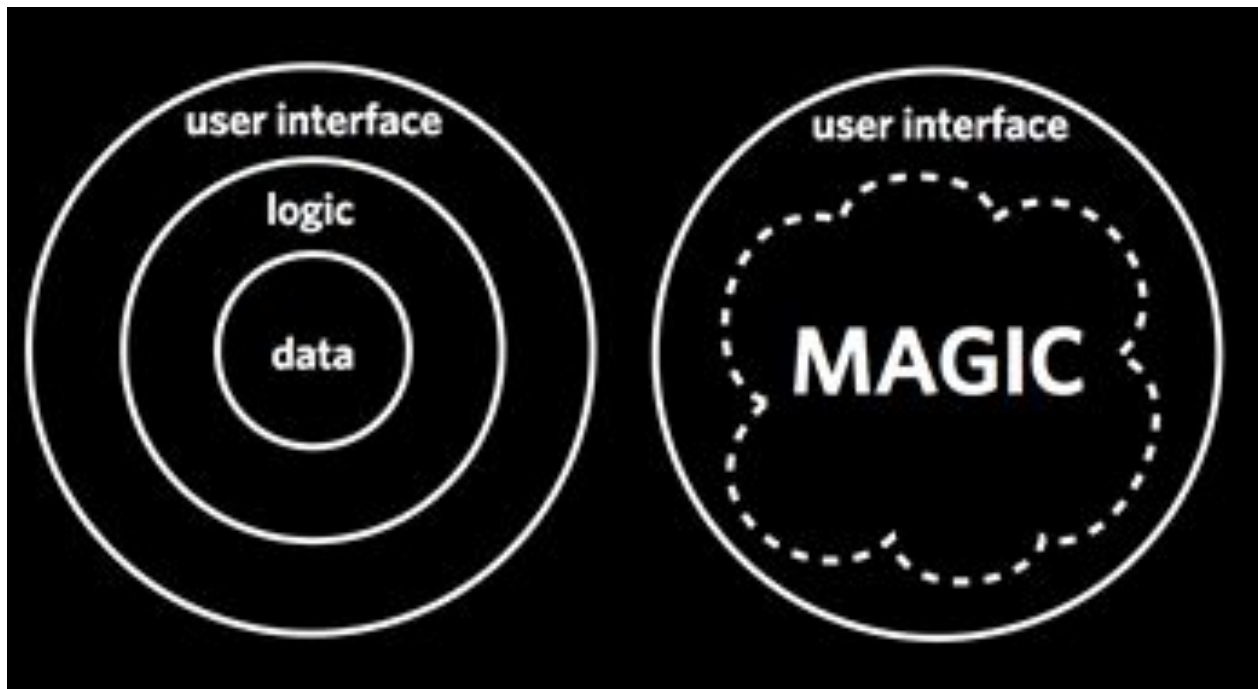
Решение

Автоматизированные и интеграционные тесты

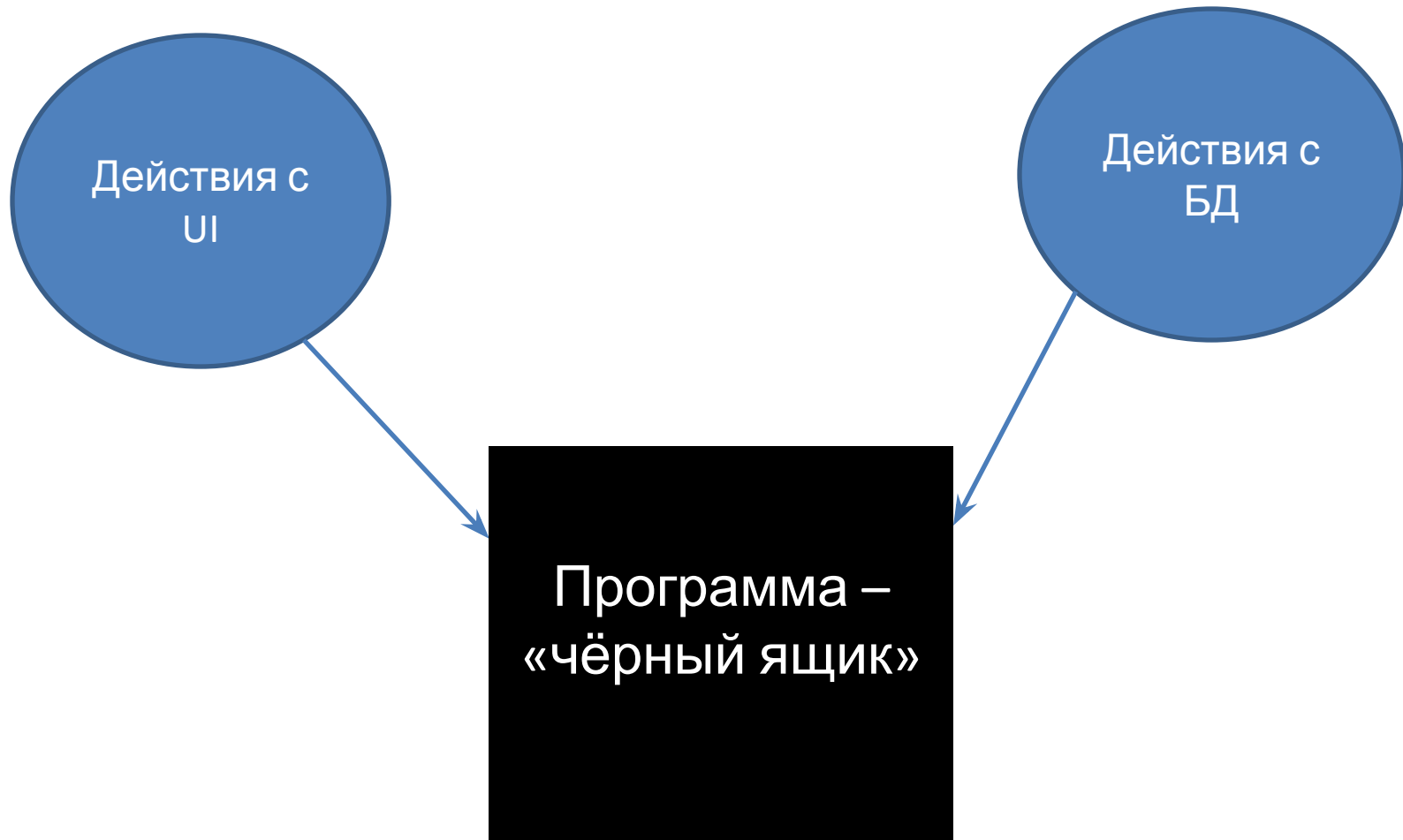
Интеграционные тесты

Имитация действий пользователя

Доступ к программе как к «чёрному ящику»



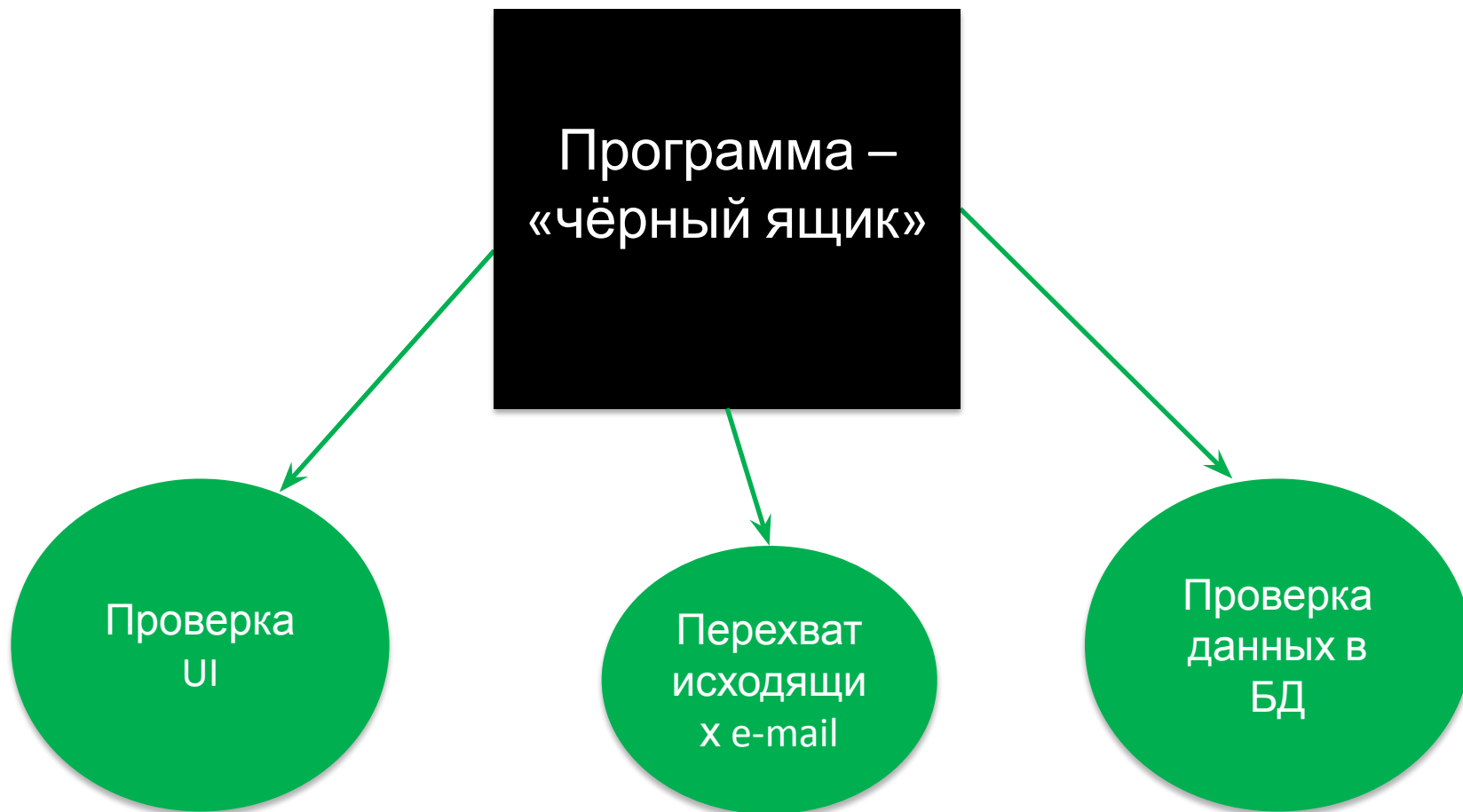
Взаимодействие тестов с приложением



Интеграционные тесты

**Код программы никак
не изменяется**

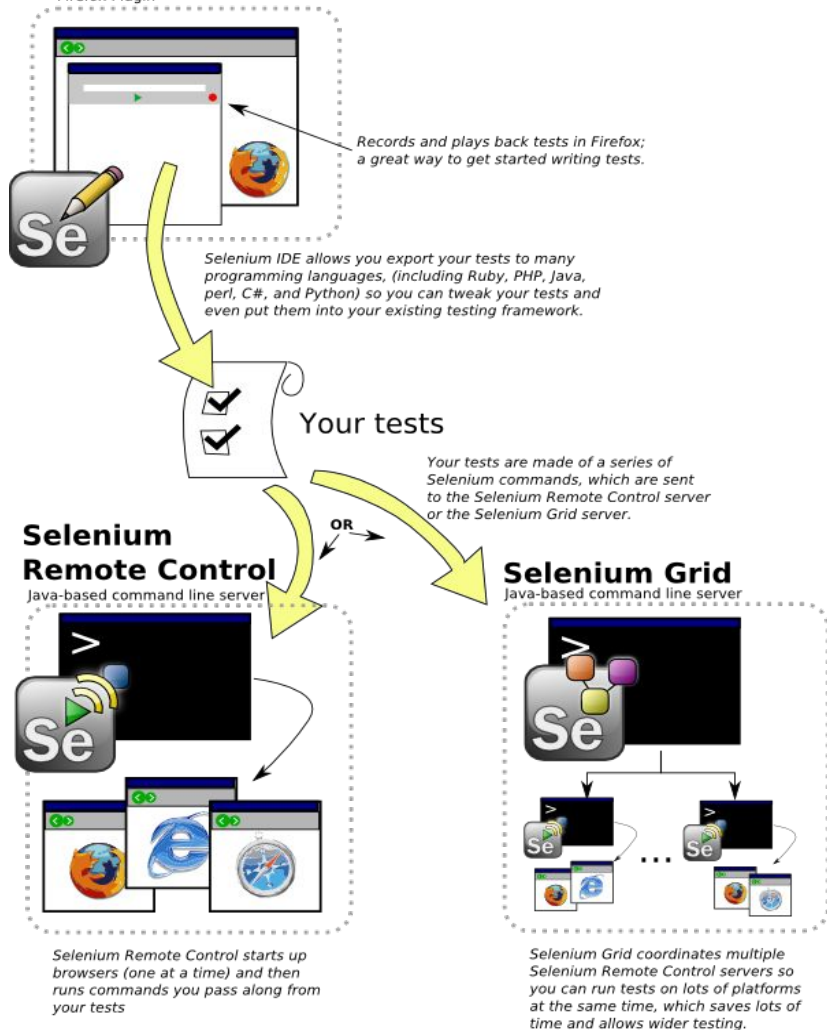
Проверка корректности теста



Взаимодействие с UI

Selenium IDE

Firefox Plugin



Selenium для
ИМИТАЦИИ
ДЕЙСТВИЙ
ПОЛЬЗОВАТЕЛЯ



http://www.google.com/



screenflow

Web [Images](#) [Maps](#) [News](#) [Shopping](#) [Gmail](#) [more](#)

[iGoogle](#) | [Sign in](#)



Google Search

I'm Feeling Lucky

[Advanced Search](#)
[Preferences](#)
[Language Tools](#)

[Advertising Programs](#) - [Business Solutions](#) - [About Google](#)

©2008 - [Privacy](#)

In Memoriam: [Randy Pausch](#) (1960-2008)



Взаимодействие с UI

```
[Fact]
public void SeleniumExample1()
{
    ISelenium selenium = new DefaultSelenium("localhost", 5561, "*firefox", "localhost:8099");

    selenium.Start();

    selenium.Open("/Account/LogOn");
    selenium.WaitForPageToLoad("10000");

    selenium.Type("Name", "username");
    selenium.Type("Password", "12345");

    selenium.Click("//input[@value='Войти']");
    selenium.WaitForPageToLoad("10000");

    Assert.True(selenium.IsTextPresent("Вход успешен!"));

    selenium.Stop();
}
```

Взаимодействие с базой данных

- Fluent NHibernate для работы с БД
- Тестировщик работает с эталонной базой данных

Роль интеграционных тестов в проекте

**Интеграционные тесты –
НЕ инструмент
нахождения багов!**

Роль интеграционных тестов в проекте

**Функциональность
реализована только тогда,
когда на неё написан
интеграционный тест**

Схема разработки продукта

- 1) Unit-тест
- 2) Код реализации
- 3) Интеграционный тест
- 4) Исправления в коде
- 5) ???
- 6) PROFIT!

Схема разработки продукта

**Программист не пишет
интеграционные тесты**

Схема разработки продукта

**Интеграционные тесты
применимы не только к
Web-приложениям**

Преимущества интеграционных тестов

**Лёгкость масштабных
изменений в проекте**

Преимущества интеграционных тестов

Легко обнаружить
breaking changes в
библиотеках и
фреймворках

Преимущества интеграционных тестов

**Если ошибка
исправлена, то она
гарантированно не
появится снова**

Преимущества интеграционных тестов

Раннее обнаружение
ошибок обходится
дёшево

Преимущества интеграционных тестов

**Автоматизированные
тесты дешевле ручных**

Возможные проблемы

Окружение для тестов
сложное и долго
настраивается

Возможные проблемы

- Тесты сложно рефакторить, так как они проходят слишком долго.
- Из-за этого при написании тестов нарушается ритм работы программиста

Возможные проблемы

Опасность превращения
тестов в нечитаемый и
трудно поддерживаемый
КОД

Контакты и полезные ссылки

- Selenium: <http://www.seleniumhq.org>
- Fluent NHibernate:
<http://fluentnhibernate.org/>
- Мой e-mail: vadim.kaduchenko@gmail.com