

02

OPEN ORIENTED

凹凸实验室

# 前端UI测试

小立

# 目录

02

- 背景
- 工具
- 方法

- 开发过程自测难以全部覆盖
- 除了开发的功能点之外，几乎空白
- 机械 繁琐

# 目录

02

- 背景
- **工具**
- 方法



**Selenium**

# Selenium

- 英[səˈliːniəm] 美[sɪˈliːniəm]
- A browser automation framework and ecosystem. (一个浏览器的自动化框架/生态系统)

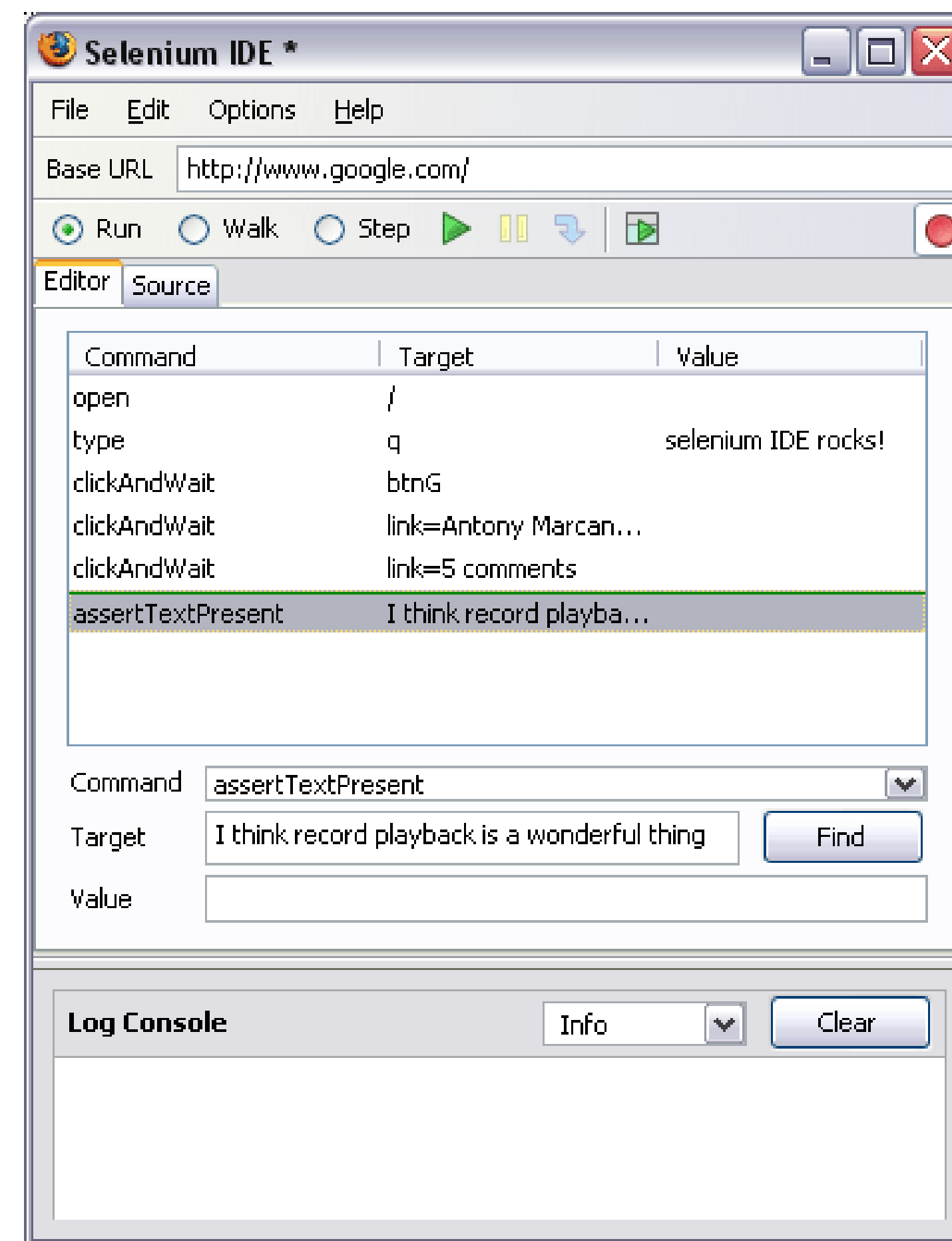
# Selenium

- **Selenium IDE**
- Selenium Remote Control
- Selenium Webdriver

# Selenium IDE

02

- 拥有诸如录制，编辑，调试代码  
自动化脚本的功能
- 一个完整的开发自动化脚本的  
( IDE )
- 其他功能：
  - 注入用户脚本
  - 代码补全





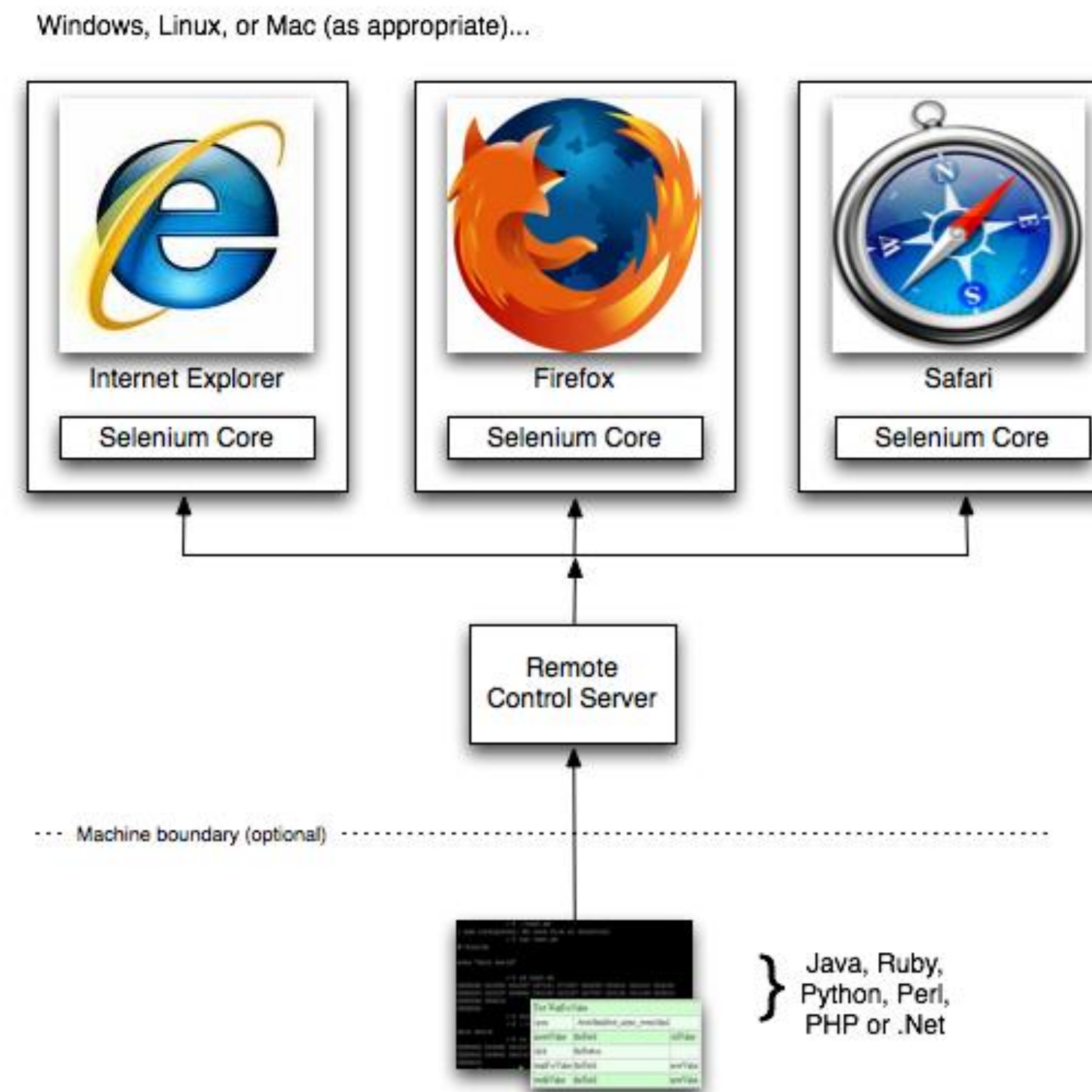
# Selenium

- Selenium IDE
- **Selenium Remote Control**
- Selenium Webdriver

# Selenium Remote Control

02

- 通过http的方式远程进行UI测试的工具
- 包含两部分：
  - 服务端：接收用户的请求，代理用户的操作
  - 客户端：发送指令给服务端



# Selenium

- Selenium IDE
- Selenium Remote Control
- **Selenium Webdriver**

# Selenium Webdriver

- 在本地进行UI测试的工具
- 可以驱动多种浏览器：
  - 带界面：Chrome Firefox Safari IE Edge
  - 无界面：PhantomJS HtmlUnit Remote

# Selenium Webdriver

02

- 安装
- 用法
- APIS

# Selenium Webdriver – 安装

02

- 安装依赖包
  - `npm install selenium-webdriver`
- 下载对应的浏览器驱动
  - Chrome Driver
  - Safari Driver
  - 不够用看这里

# Selenium Webdriver

02

- 安装
- **用法**
- APIS

# Selenium Webdriver – 用法

02

```
var webdriver = require('selenium-webdriver'),  
    By = webdriver.By,  
    until = webdriver.until;
```

获取driver

```
var driver = new webdriver.Builder()  
    .forBrowser('firefox')  
    .build();
```

做动作

```
driver.get('http://www.google.com/ncr');  
driver.findElement(By.name('q')).sendKeys('webdriver');  
driver.findElement(By.name('btnG')).click();  
driver.wait(until.titleIs('webdriver - Google Search'), 1000);  
driver.quit();
```

结束



# Selenium Webdriver

02

- 安装
- 用法
- **APIS**

# Selenium Webdriver – APIS

02

- **分类：**
  - **元素选择**
  - 动作
  - 测试

# Selenium Webdriver – APIS

02

- 元素选择器 ( Object driver.By )
- 分类：
  - 常用：By.css , By.className
  - 不常用：By.js, By.linkText, By.name, By.partialLinkText, By.xpath
- 获取元素：
  - **Thenable** driver.findElement(*locator*)

# Selenium Webdriver – APIS

02

- **分类：**
  - 元素选择
- **动作**
- **测试**

# Selenium Webdriver – APIS

02

- 动作 ( Class driver.actions )
- 分类 :
  - 元素动作 :
    - `$ele.[click|sendKeys|getAttribute|...]`
  - 其他动作 :
    - **Thenable** `driver.actions().[keyDown|click|mouseMove]`
    - **\*\***`.perform()`

```
driver.actions().  
    keyDown(Key.SHIFT).  
    click(element1).  
    click(element2).  
    dragAndDrop(element3, element4).  
    keyUp(Key.SHIFT).  
    perform();
```

# Selenium Webdriver – APIS

02

- **分类：**
  - 元素选择
  - 动作
  - **测试**

# Selenium Webdriver – APIS

- 测试 ( driver.testing )
- 提供API :
  - after, afterEach, before, beforeEach, it, it.only, it.skip, xit
- 仅仅是包装了一下Mocha的接口...

```
function wrapIt() {  
  if (!WRAPPED.it) {  
    let it = getMochaGlobal('it');  
    WRAPPED.it = wrapped(it);  
    WRAPPED.itOnly = wrapped(it.only);  
  }  
}
```

# Selenium Webdriver – 例子



- Q: 懒加载导致元素不存在？
  - A: 手动滚动屏幕
  - A2: 设置屏幕大小
- Q: API不好用？
- Q: 持续集成？
- Q: 生成报表？



THANKS

FOR YOUR WATCHING