Test Automation with Selenium 5 and Java

Devoxx Belgium 2024

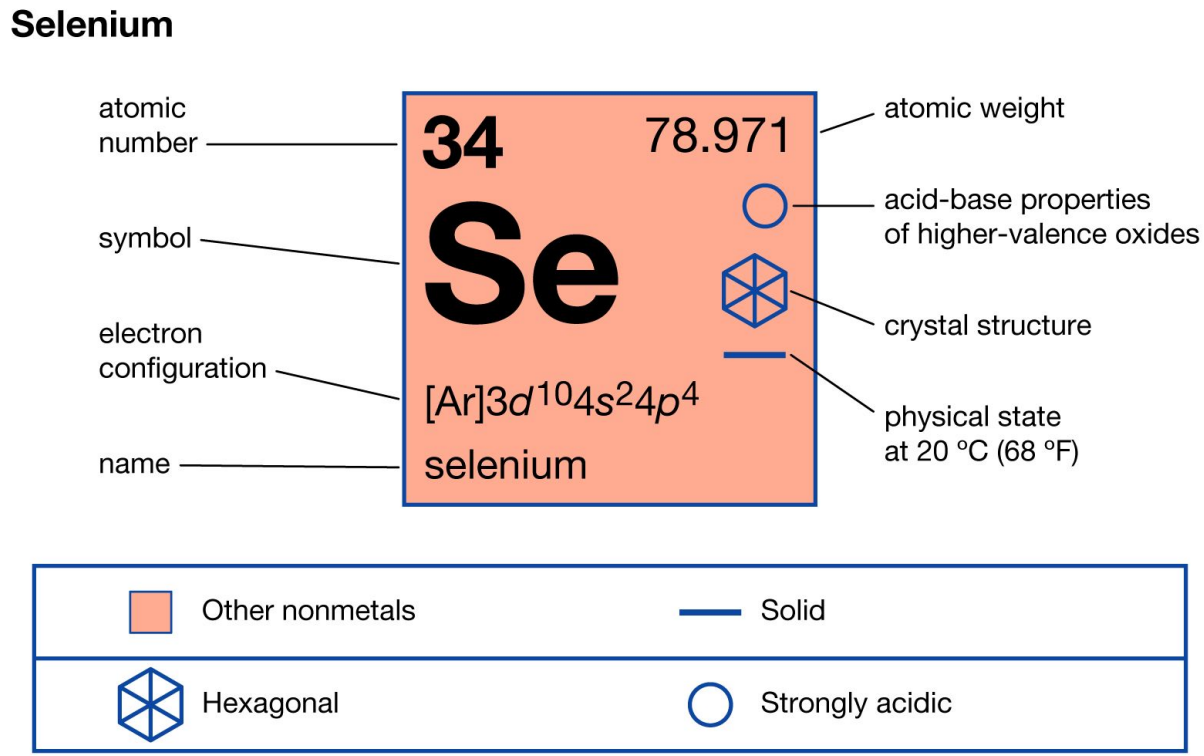
10 October 2024

Boni García

Universidad Carlos III de Madrid, Spain

boni.garcia@uc3m.es

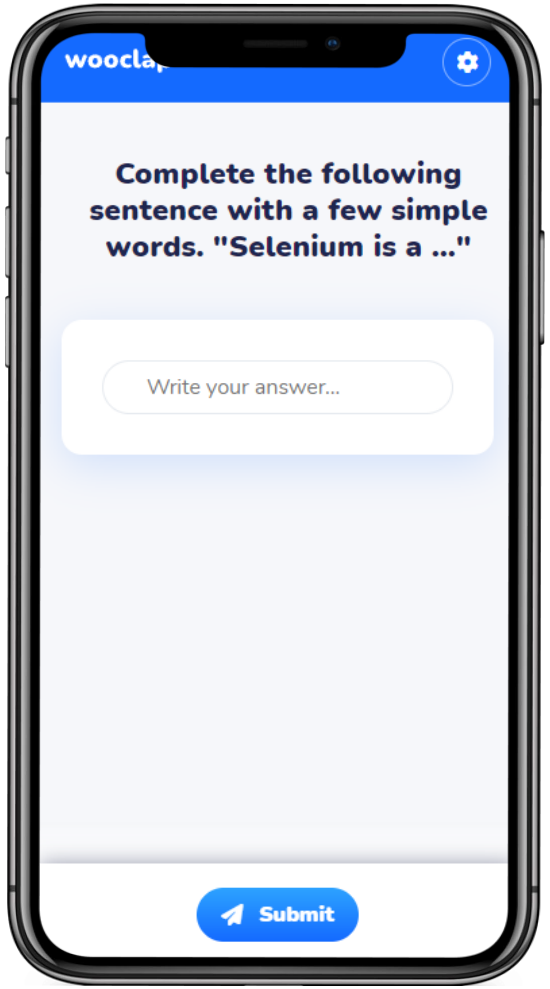
What is Selenium?



What is Selenium?

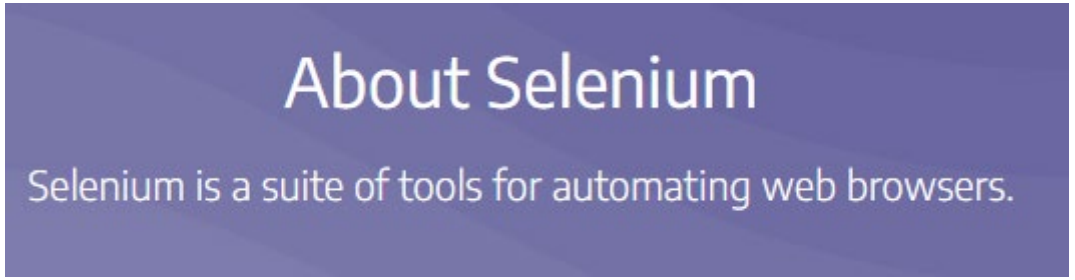
(for software people)

What is Selenium?



https://app.wooclap.com/EUHMZU

What is Selenium?



https://www.selenium.dev/about/

Selenium WebDriver

Library

Selenium IDE

Plugin

Selenium Grid

Server

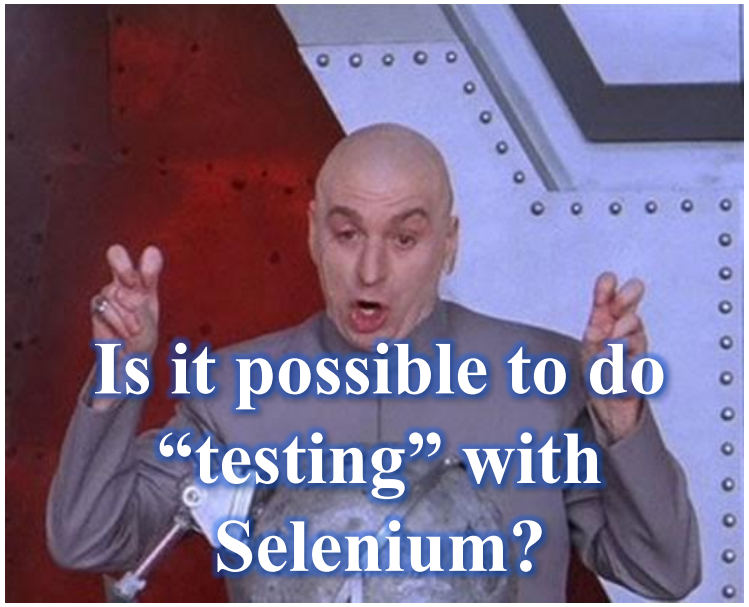
What is Selenium?

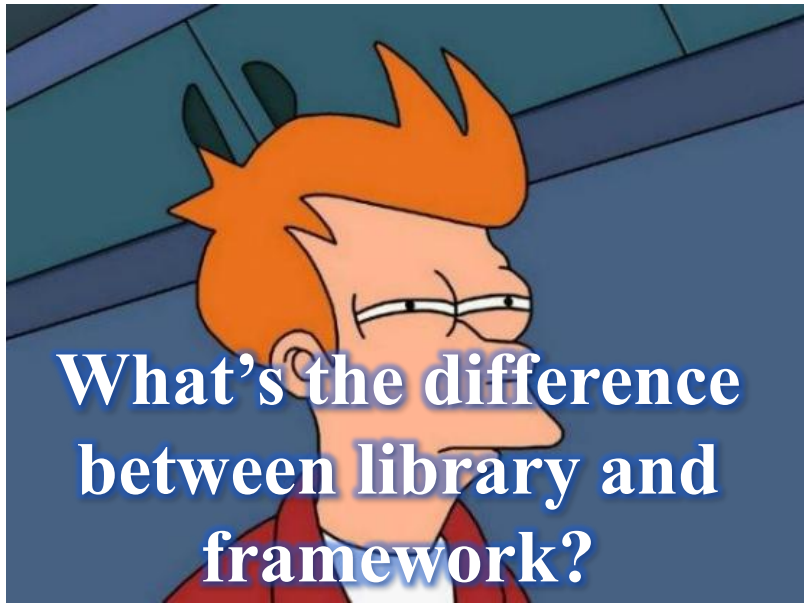
- **Selenium WebDriver** is the heart of the Selenium project and it is often known as simply **Selenium**

> *Selenium is a browser automation library*

What is NOT Selenium?

- Selenium is NOT a testing framework 
- Selenium is NOT a testing library 





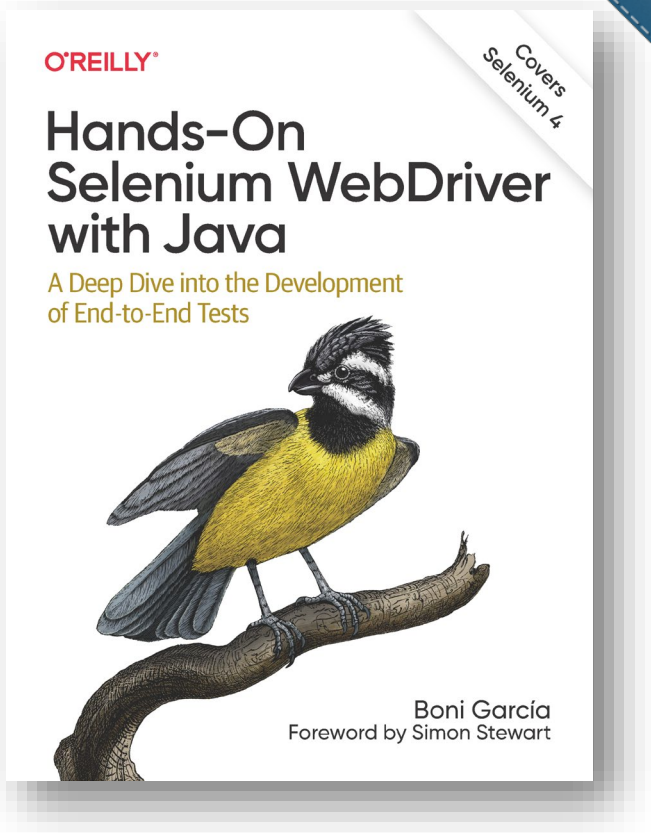
Library vs. Framework

- A **library** is a collection of code that developers can call using an API to solve a given problem

- A **framework** is collection of libraries, tools, and best practices that provides a structure for developing software

What can I do with Selenium?

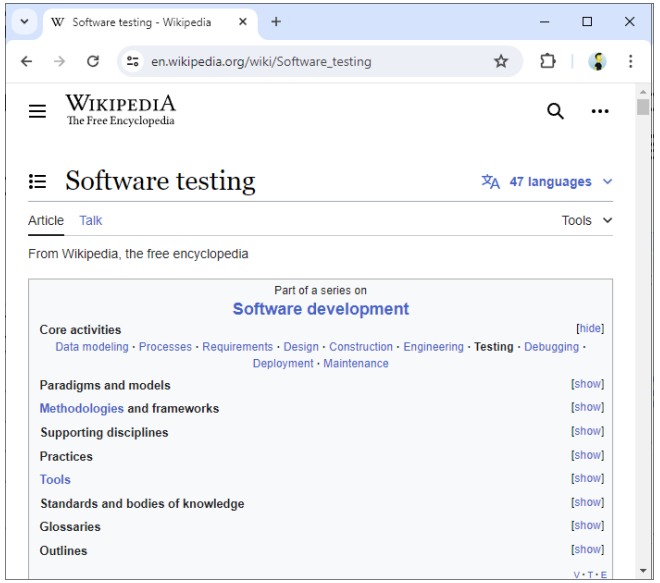
- Navigate to URLs
- Locating web elements (DOM)
- Impersonate user actions (keyboard, mouse)
- Execute JavaScript
- Make screenshots
- Manage browser size, position, history, …
- Manage browser APIs like web storage, user media, …
- …



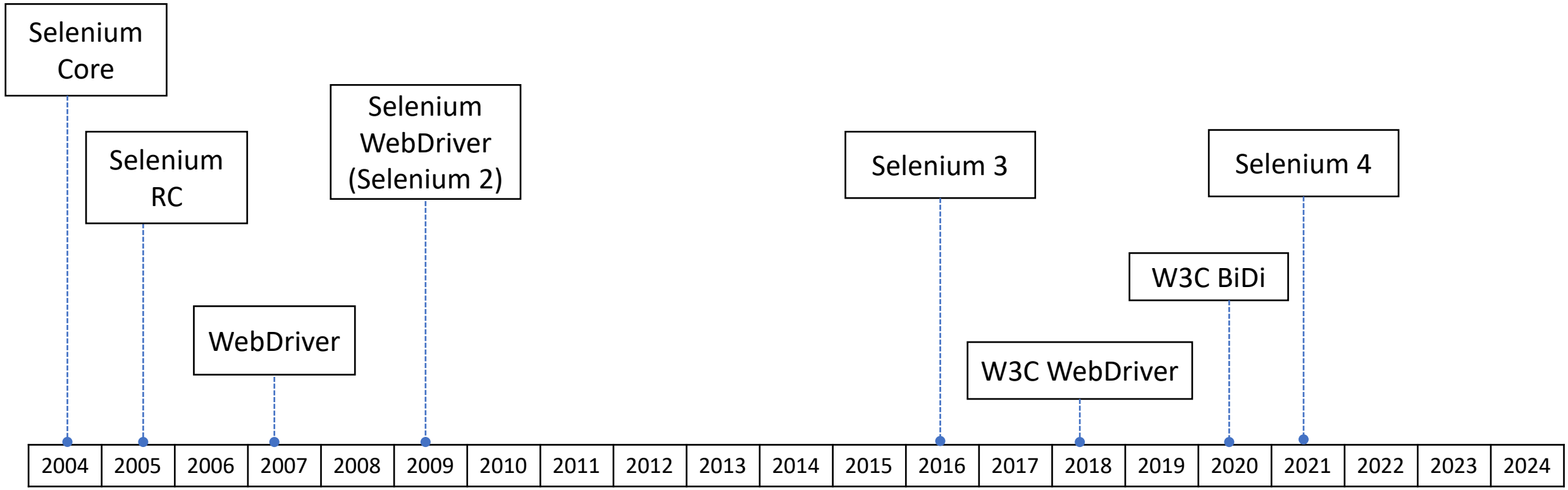
https://github.com/bonigarcia/selenium-webdriver-java

What can I do with Selenium?

```java
// Create object to drive Chrome
WebDriver driver = new ChromeDriver();

// Navigate to a website (e.g., Wikipedia)
driver.get("https://en.wikipedia.org/");

// Type "software testing" in the search box
WebElement searchBox = driver.findElement(By.name("search"));
searchBox.sendKeys("software testing");

// Click on search button
WebElement searchButton = driver
        .findElement(By.className("cdx-search-input__end-button"));
searchButton.click();

// Close browser
driver.quit();
```



Selenium History



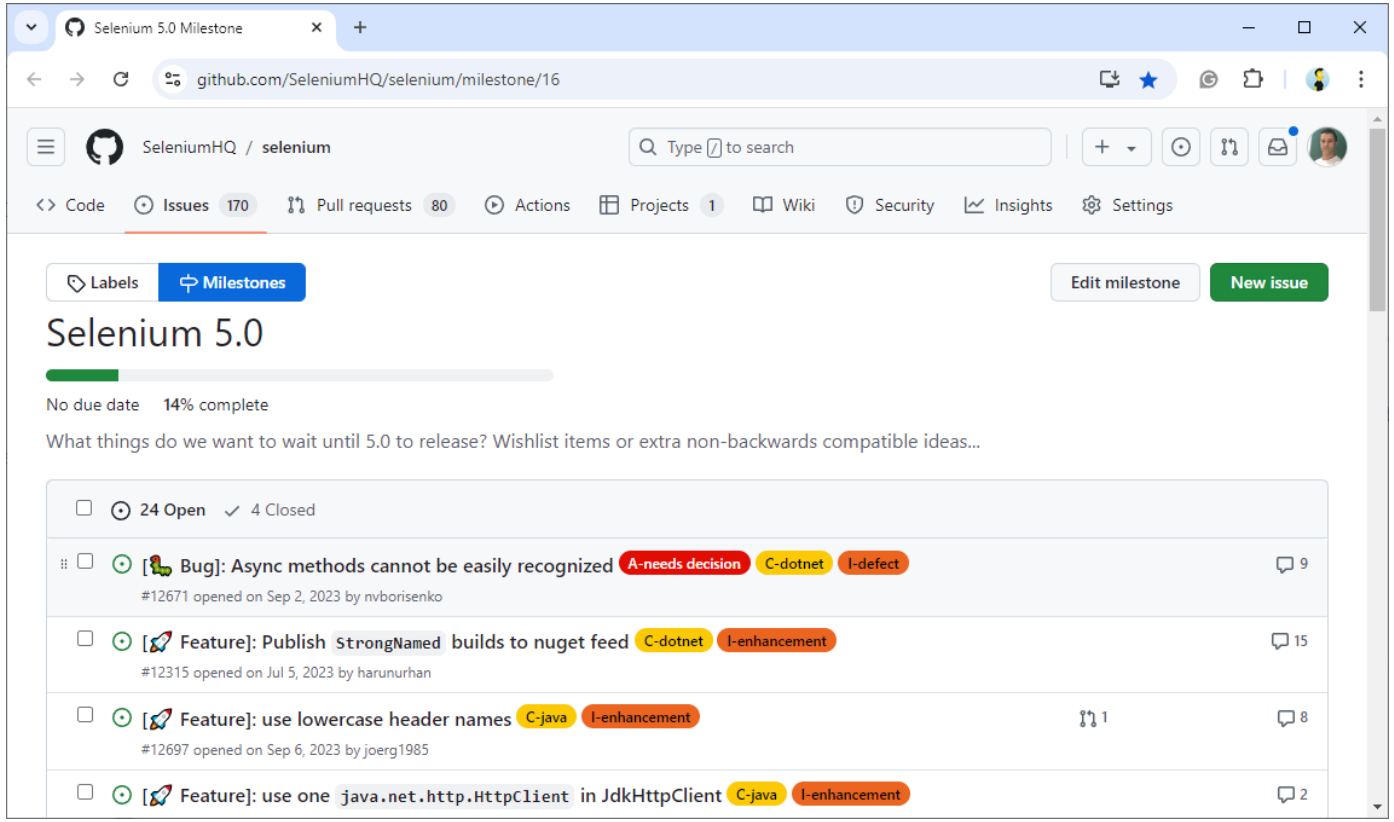
Selenium History

- Current Selenium stable version: **4.25** (released on September 23)

https://www.selenium.dev/blog/2024/selenium-4-25-released/

Selenium History

- Next major release (**Selenium 5**) is currently in development



There is no official date for Selenium 5

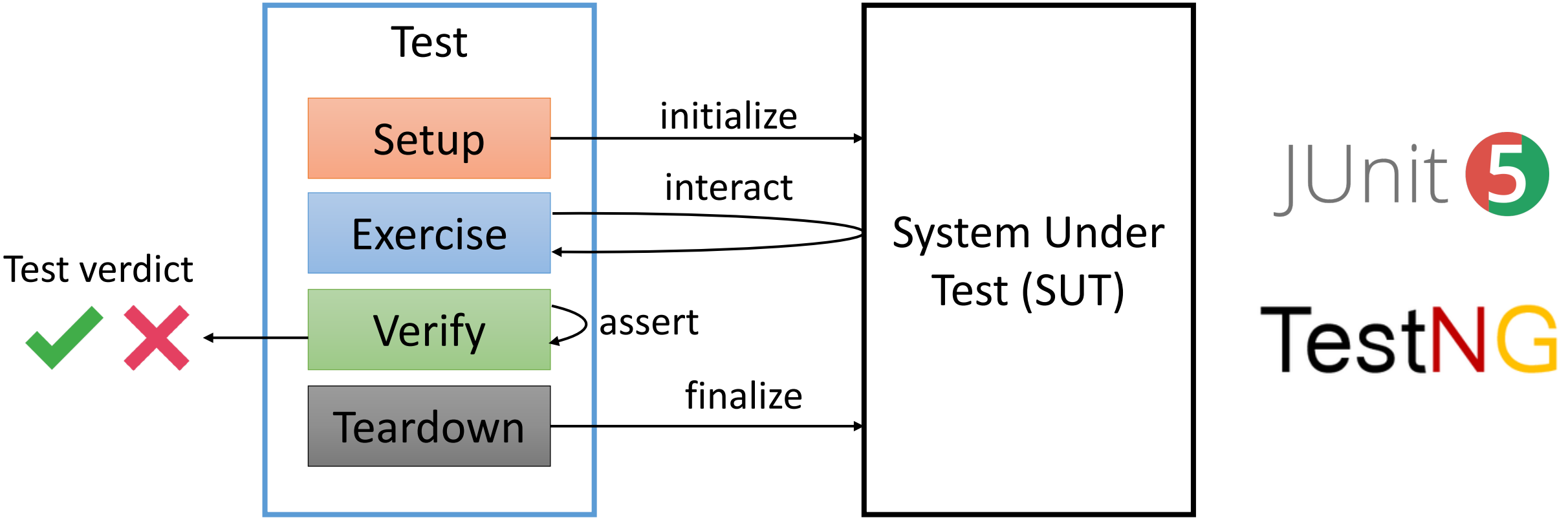
Selenium Manager

WebDriver BiDi

https://github.com/SeleniumHQ/selenium/milestone/16

Testing with Selenium

- In addition to the Selenium library, to create Selenium tests, we typically use a **unit testing framework**



Testing with Selenium – Build Tool

```xml
<dependencies>
    <dependency>
        <groupId>org.seleniumhq.selenium</groupId>
        <artifactId>selenium-java</artifactId>
        <version>4.25.0</version>
        <scope>test</scope>
    </dependency>
    <dependency>
        <groupId>org.junit.jupiter</groupId>
        <artifactId>junit-jupiter</artifactId>
        <version>5.11.1</version>
        <scope>test</scope>
     </dependency>
    <dependency>
        <groupId>org.assertj</groupId>
        <artifactId>assertj-core</artifactId>
        <version>3.26.3</version>
        <scope>test</scope>
    </dependency>
<dependencies>
```

Maven™

```
dependencies {
    testImplementation("org.seleniumhq.selenium:selenium-java:4.25.0")
    testImplementation("org.junit.jupiter:junit-jupiter:5.11.1")
    testImplementation("org.assertj:assertj-core:3.26.3")
}
```

Gradle

Testing with Selenium – Browsers

- We need one or more browsers to be driven with Selenium

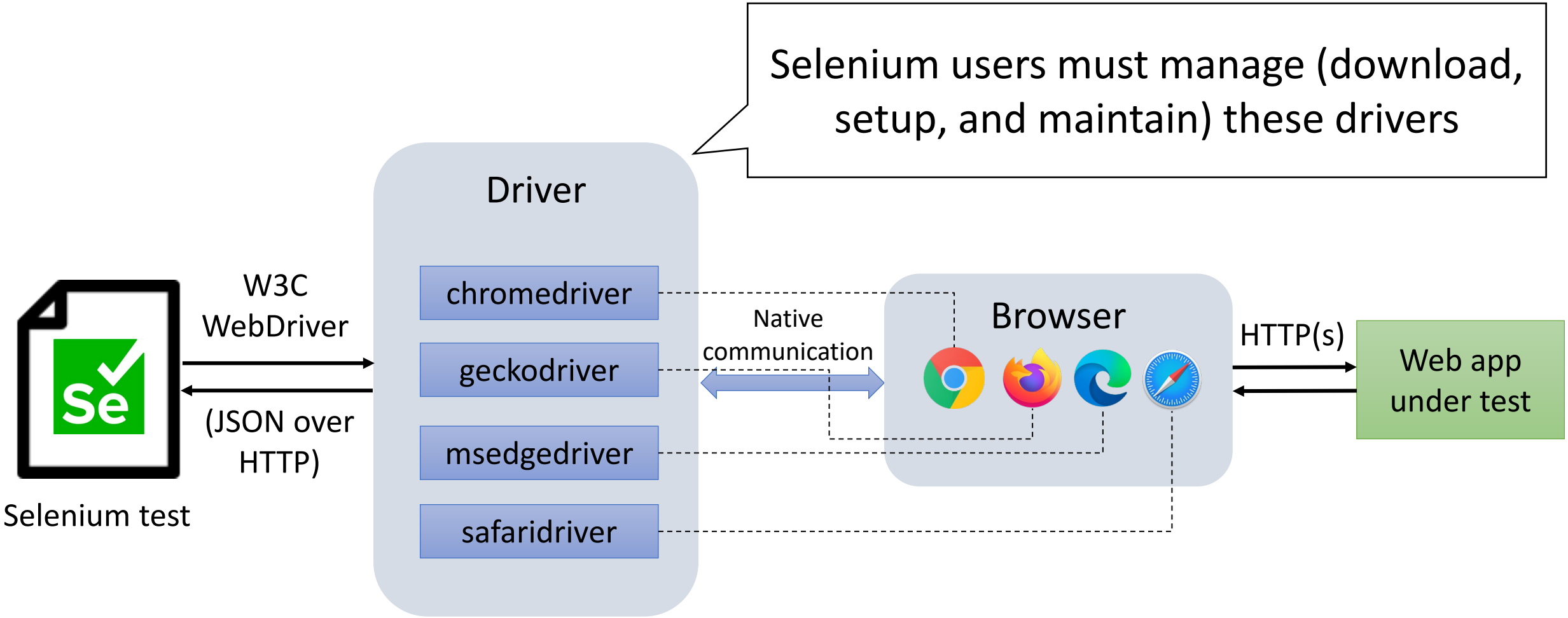
1. Local browser

```
WebDriver chrome = new ChromeDriver();
WebDriver firefox = new FirefoxDriver();
WebDriver edge = new EdgeDriver();
WebDriver safari = new SafariDriver();
```

2. Remote browser

```
ChromeOptions options = new ChromeOptions();
WebDriver driver = new RemoteWebDriver(
          new URL("http://server:4444/"), options);
```

Testing with Selenium – Browsers



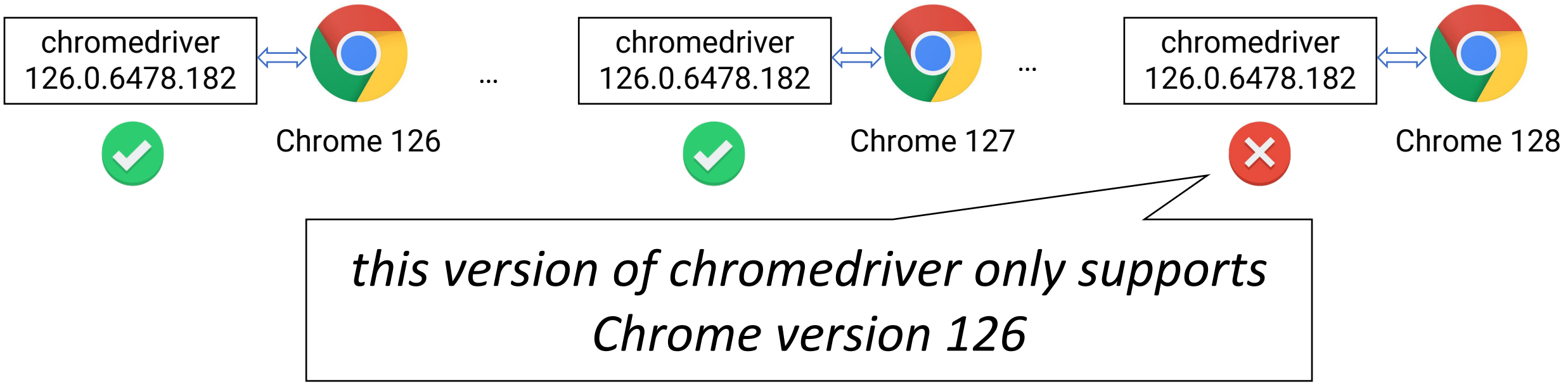
Testing with Selenium – Hello World (Selenium 3)

```java
class ChromeManualTest {

    WebDriver driver;

    @BeforeAll
    static void setupClass() {
        System.setProperty("webdriver.chrome.driver", "/path/to/chromedriver");
    }

    @BeforeEach
    void setup() {
        driver = new ChromeDriver();
    }

    @Test
    void test() {
        driver.get("https://bonigarcia.dev/selenium-webdriver-java/");
        String title = driver.getTitle();
        assertThat(title).contains("Selenium WebDriver");
    }

    @AfterEach
    void teardown() {
        driver.quit();
    }

}
```

https://github.com/bonigarcia/selenium-examples

Driver Management

- Modern web browsers are ***evergreen***



Automated Driver Management

https://bonigarcia.dev/webdrivermanager/

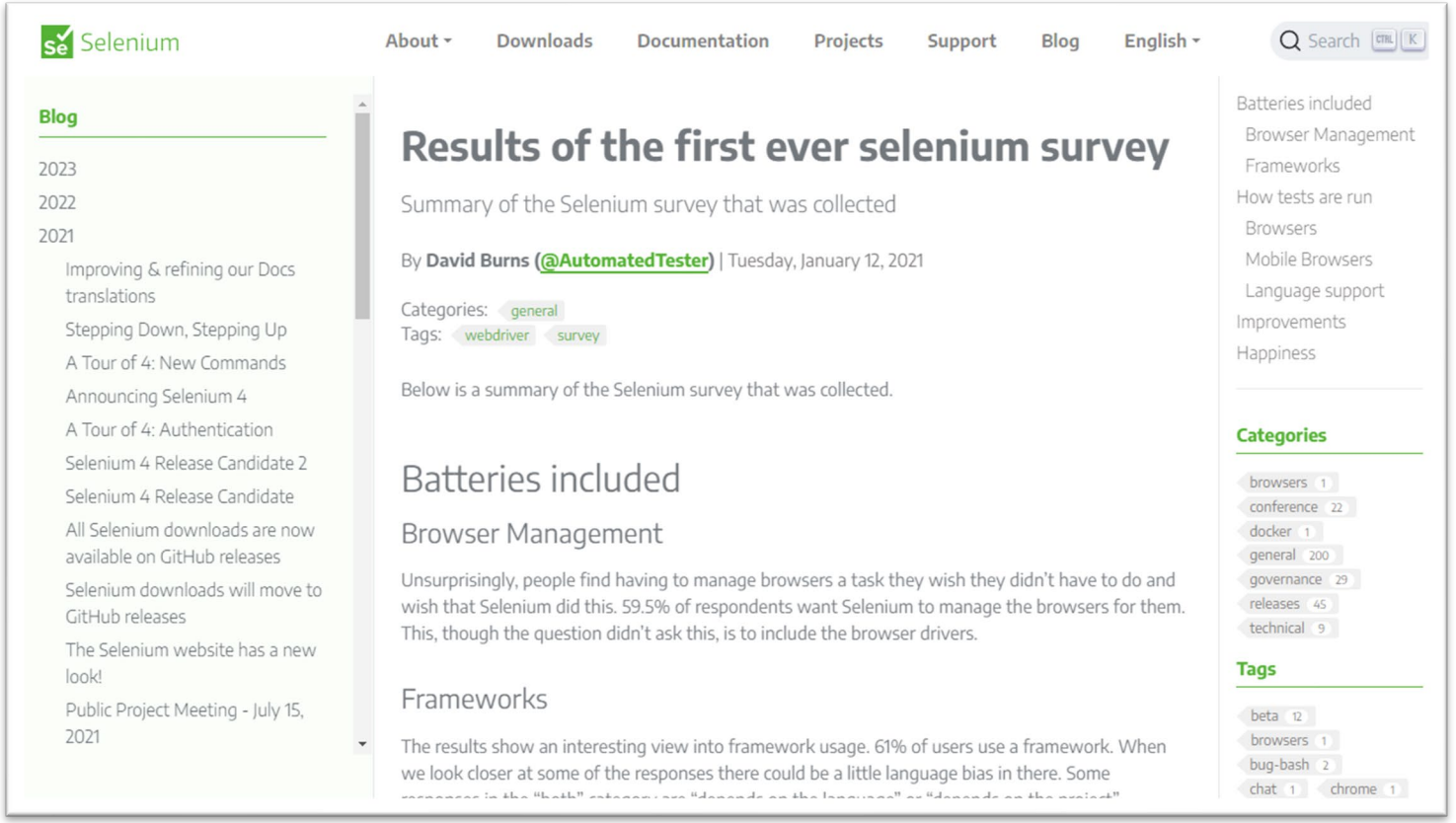
> *Automated driver management and other helper features for Selenium WebDriver in Java*

Testing with Selenium – Hello World (Selenium 4)


```java
class ChromeWdmTest {

    WebDriver driver;

    @BeforeAll
    static void setupClass() {
        WebDriverManager.chromedriver().setup();
    }

    @BeforeEach
    void setup() {
        driver = new ChromeDriver();
    }

    @Test
    void test() {
        driver.get("https://bonigarcia.dev/selenium-webdriver-java/");
        String title = driver.getTitle();
        assertThat(title).contains("Selenium WebDriver");
    }

    @AfterEach
    void teardown() {
        driver.quit();
    }

}
```

Automated Driver Management

- In 2021, the Selenium project surveyed its users:



Selenium users wanted *batteries included*

https://www.selenium.dev/blog/2021/selenium-survey-results/

Selenium Manager

- I joined the Selenium project as a committer in August 2022

Selenium Manager (beta)

- It is a CLI (Command-Line Interface) tool

- It has been developed in Rust




- It is shipped in each Selenium release





Selenium Manager – Automated Driver Management

- Selenium Manager automatically discovers, downloads, and caches the drivers required by Selenium

1. Browser version discovery
2. Driver version discovery
3. Driver download and cache

Chrome 128

chromedriver
128.0.6613.137

`~/.cache/selenium`

Selenium Manager – Hello World (Selenium 5)


```
class ChromeBasicTest {

    WebDriver driver;

    @BeforeEach
    void setup() {
        driver = new ChromeDriver();
    }

    @Test
    void test() {
        driver.get("https://bonigarcia.dev/selenium-webdriver-java/");
        String title = driver.getTitle();
        assertThat(title).contains("Selenium WebDriver");
    }

    @AfterEach
    void teardown() {
        driver.quit();
    }

}
```

Selenium Manager – Automated Browser Management

- Selenium Manager automatically discovers, downloads, and caches the browsers driven with Selenium when these browsers are not installed in the local system

			
	✓	✓	✓ *
	✓	✓	✓
	✓	✓	✓

*Requires admin permissions

Selenium Manager – Automated Browser Management

```
class FirefoxBasicTest {

    WebDriver driver;

    @BeforeEach
    void setup() {
        driver = new FirefoxDriver();
    }

    @Test
    void test() {
        driver.get("https://bonigarcia.dev/selenium-webdriver-java/");
        String title = driver.getTitle();
        assertThat(title).contains("Selenium WebDriver");
    }

    @AfterEach
    void teardown() {
        driver.quit();
    }

}
```

If Firefox is not available, Selenium Manager will manage it

Selenium Manager – Automated Browser Management

```java
class ChromeVersionTest {

    WebDriver driver;

    @BeforeEach
    void setup() {
        ChromeOptions options = new ChromeOptions();
        options.setBrowserVersion("beta");
        driver = new ChromeDriver(options);
    }

    @Test
    void test() {
        driver.get("https://bonigarcia.dev/selenium-webdriver-java/");
        String title = driver.getTitle();
        assertThat(title).contains("Selenium WebDriver");
    }

    @AfterEach
    void teardown() {
        driver.quit();
    }

}
```

Specific browser versions (including "beta", "dev", or "nightly") are supported

Selenium Manager – Other Uses

- Another uses of Selenium Manager include:
  - Advanced configuration (with envs or global configuration file)
  - As a standalone CLI tool

Selenium Manager (beta)

https://www.selenium.dev/documentation/selenium_manager/

Selenium Manager & WebDriverManager

- Is Selenium Manager a replacement for WebDriverManager?
  - For the use case of automated driver management, yes
- WebDriverManager and Selenium Manager have different features
  - Automated browser management is different in each project
  - WebDriverManager provides other additional features
    - Self-managed browsers in Docker containers
    - Monitoring features
    - Video recording

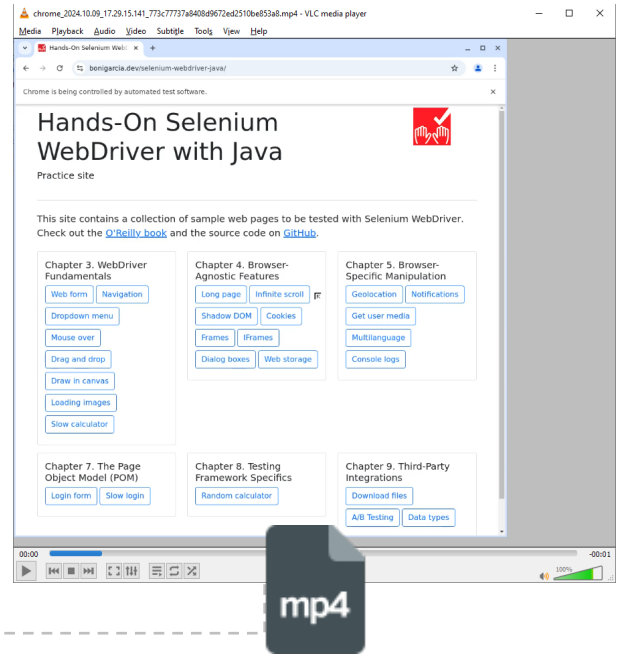
https://bonigarcia.dev/webdrivermanager/#webdrivermanager-and-selenium-manager

WebDriverManager Browser Management


```
class DockerChromeTest {

    WebDriver driver;
    WebDriverManager wdm;

    @BeforeEach
    void setupTest() {
        wdm = WebDriverManager.chromedriver().browserInDocker();
        driver = wdm.create();
    }

    @Test
    void test() {
        driver.get("https://bonigarcia.dev/selenium-webdriver-java/");
        assertThat(driver.getTitle()).contains("Selenium WebDriver");
    }

    @AfterEach
    void teardown() {
        wdm.quit();
    }

}
```

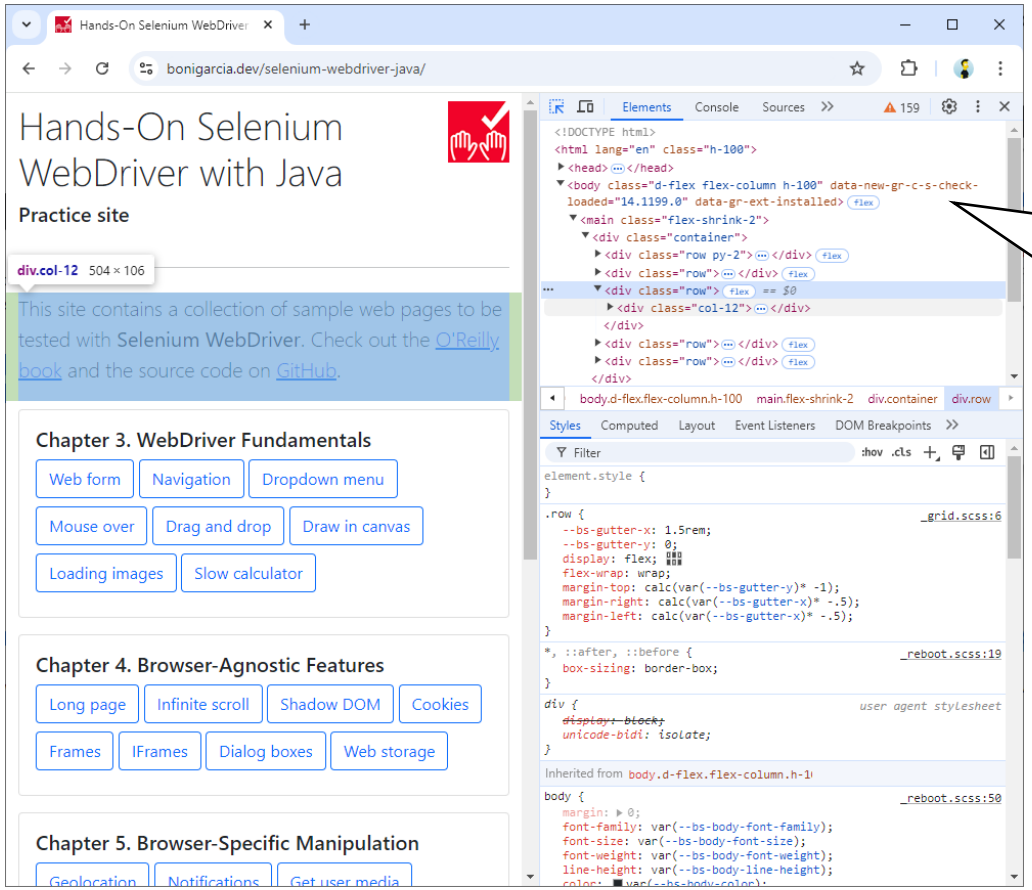
WebDriverManager Browser Management


```java
class DockerChromeRecordingTest {

    WebDriver driver;
    WebDriverManager wdm;

    @BeforeEach
    void setupTest() {
        wdm = WebDriverManager.chromedriver().browserInDocker().enableRecording();
        driver = wdm.create();
    }

    @Test
    void test() {
        driver.get("https://bonigarcia.dev/selenium-webdriver-java/");
        assertThat(driver.getTitle()).contains("Selenium WebDriver");
    }

    @AfterEach
    void teardown() {
        wdm.quit();
    }

}
```



Chrome DevTools

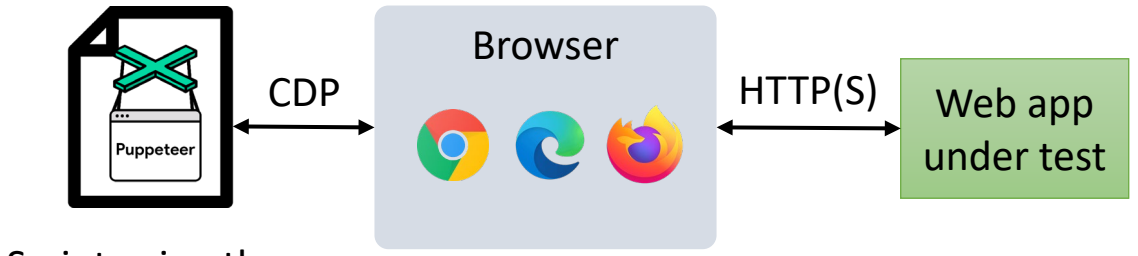
- **Chrome DevTools** is a set of web developer tools built into Chromium-based browser (e.g., Chrome and Edge)



Inspect DOM, console, network, sources, performance, memory, or security

Chrome DevTools Protocol

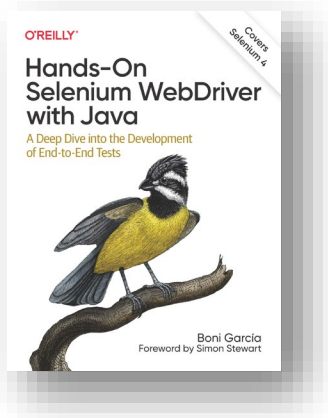
- The **Chrome DevTools Protocol** (CDP) is a set of APIs that allows us to instrument, inspect, debug, and profile Chromium-based browsers
  - CDP is great for browser automation



Puppeteer (a Node.js browser automation library created by Google) is based on CDP

Chrome DevTools Protocol and Selenium

- Selenium supports CDP since version 4 to provide features not available in the W3C WebDriver, such as
  - Network interception and monitoring
  - Device emulation
  - Performance metrics collection
  - Entire page screenshot
  - Security and authentication
  - …

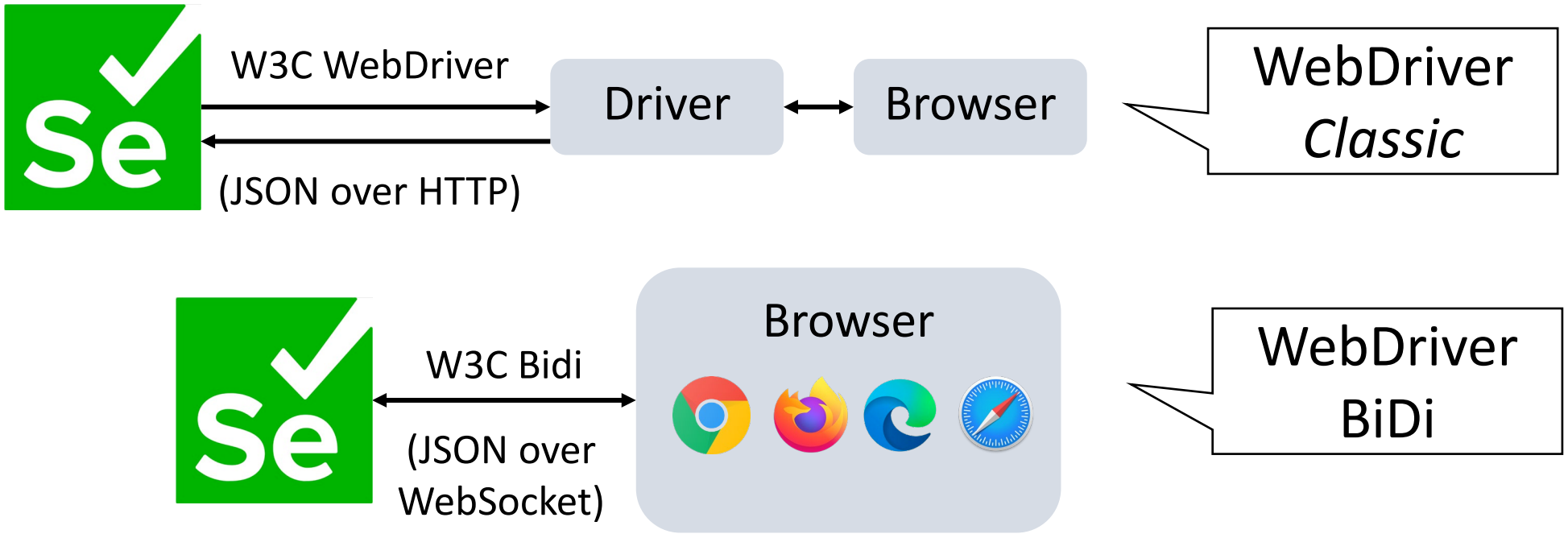


See test examples using CDP and Selenium here

https://github.com/bonigarcia/selenium-webdriver-java

WebDriver BiDi

- W3C WebDriver is evolving to a new specification called **W3C BiDi** (*BiDirectional*)
  - Its goal is to combine the best of WebDriver and CDP



https://w3c.github.io/webdriver-bidi/

WebDriver BiDi

- WebDriver BiDi features:
  - Bidirectional communication using a WebSocket (like CDP)
  - Event-driven architecture (e.g., for log gathering)
  - Support for modern web features (e.g., network interception)
- WebDriver BiDi modules:
  - Browsing context
  - Actions
  - Scripting
  - Logging
  - Network

https://www.selenium.dev/documentation/webdriver/bidi/w3c/

WebDriver BiDi

- Some of the WebDriver BiDi features are already available in the latest versions of Selenium 4

```
@BeforeEach
void setup() {
    ChromeOptions options = new ChromeOptions();
    options.enableBiDi();
    driver = new ChromeDriver(options);

}
```

```
@BeforeEach
void setup() {
    FirefoxOptions options = new FirefoxOptions();
    options.enableBiDi();
    driver = new FirefoxDriver(options);

}
```

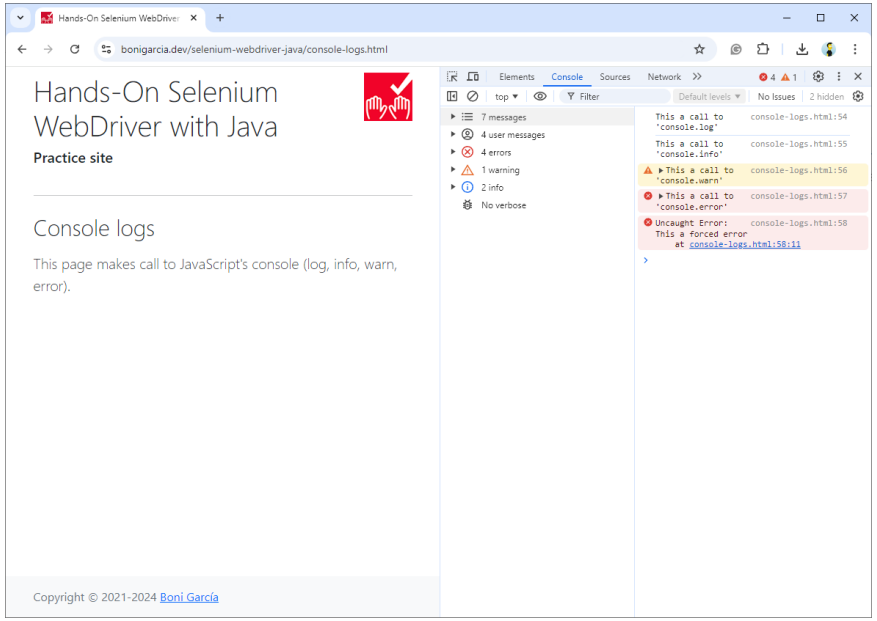
```
@BeforeEach
void setup() {
    EdgeOptions options = new EdgeOptions();
    options.enableBiDi();
    driver = new EdgeDriver(options);

}
```

To use WebDriver BiDi in Selenium, first we need to enable it using browser options

WebDriver BiDi

```
@Test
void test() {
    List<GenericLogEntry> logs = new ArrayList<>();
    try (LogInspector logInspector = new LogInspector(driver)) {
        logInspector.onGenericLog(logs::add);
        logInspector.onConsoleEntry(logs::add);
        logInspector.onJavaScriptException(logs::add);
    }

    driver.get(
            "https://bonigarcia.dev/selenium-webdriver-java/console-logs.html");

    new WebDriverWait(driver, Duration.ofSeconds(5))
            .until(_d -> logs.size() > 3);

    for (GenericLogEntry log : logs) {
        System.out.println(log.getText());
    }
}
```

https://github.com/bonigarcia/selenium-examples

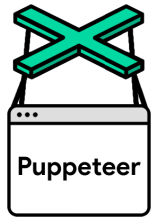


The Selenium Ecosystem

- Tools (e.g., testing frameworks) based on Selenium

- WebDriver ecosystem

- WebDriver BiDi ecosystem



https://www.selenium.dev/ecosystem/

Quiz

- Top-3 respondents will win a Selenium 20th anniversary t-shirt!

https://app.wooclap.com/JNZDZN

Test Automation with Selenium 5 and Java

slides

Thank you!

Boni García
Universidad Carlos III de Madrid, Spain
boni.garcia@uc3m.es