

UNIT 5

Automation Testing Tools / Performance Testing Tools



Contents

Automation Testing:

- What is automation testing
- Automated Testing Process
- Automation Frameworks
- Benefits of automation testing
- how to choose automation testing tools.

Selenium Automation Tools:

- Selenium's Tool Suite- Selenium IDE
- Selenium RC
- Selenium Web driver
- Selenium Grid

Automation Tools:

- SoapUI
- Robotic Process Automation (RPA)
- Tosca
- Appium

Automation testing

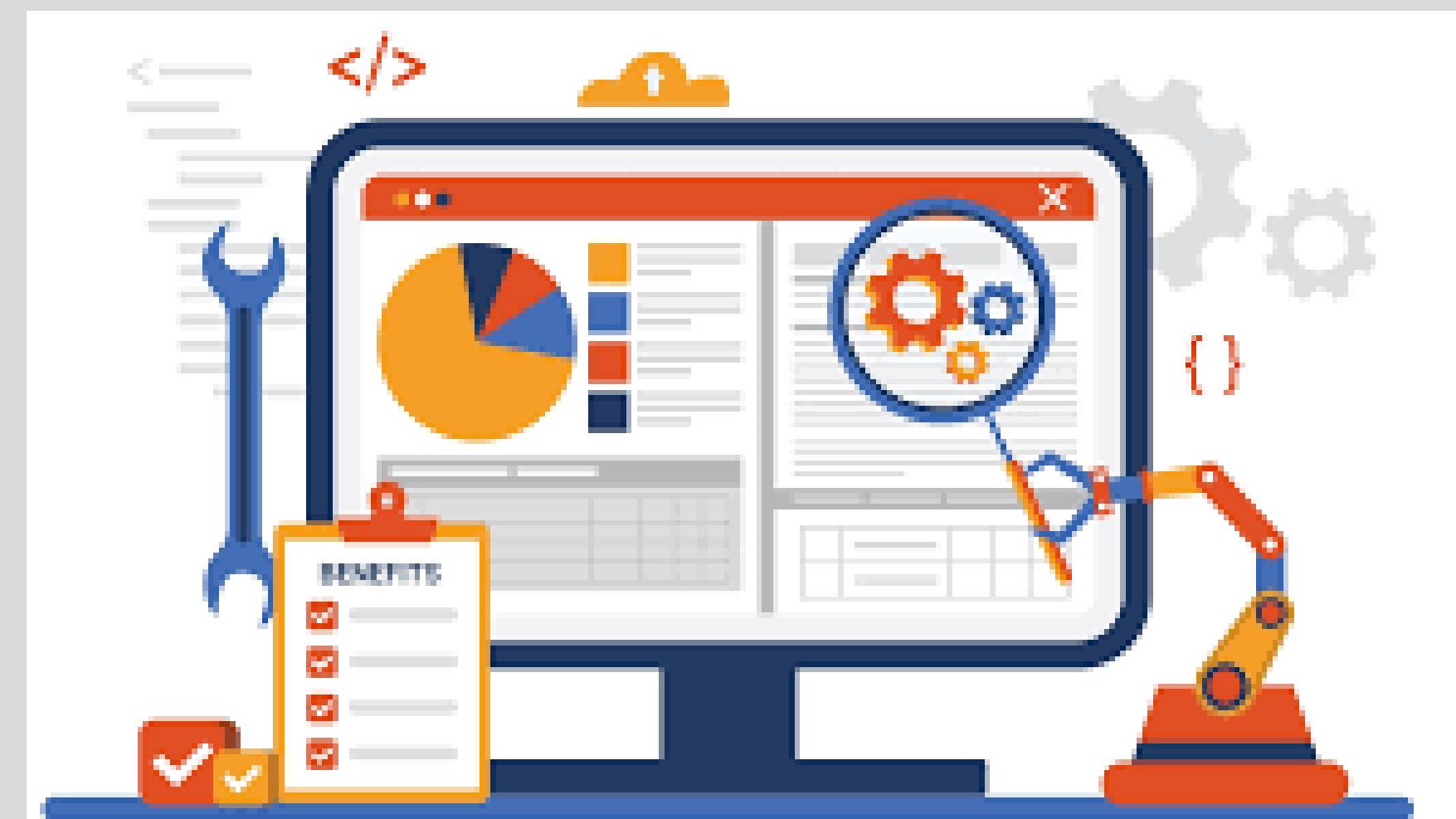
- Automation testing is a software testing technique where test cases are executed automatically using tools instead of manual effort.
- It uses scripts and software tools to check whether an application works correctly.
- It helps in saving time, reducing human errors, and increasing test coverage.
- Mainly used for repetitive tasks, regression testing, and large applications

Example

When testing a login page, an automation tool can:

- Enter username and password
- Click the login button
- Verify successful login

This test can be run multiple times automatically without human intervention.



Automated Testing Process



Automation Framework

- An automation framework is a structured set of guidelines, tools, and rules used to create and run automated tests.
- It helps in code reuse, easy maintenance, and better test management.
- Frameworks make automation faster and more reliable.

Types of Automation Frameworks

- Linear Framework – Simple scripts, no reuse.
- Modular Framework – Application divided into modules.
- Data-Driven Framework – Test data stored externally (Excel, CSV).
- Keyword-Driven Framework – Actions written as keywords.
- Hybrid Framework – Combination of two or more frameworks.

Benefits of automation testing

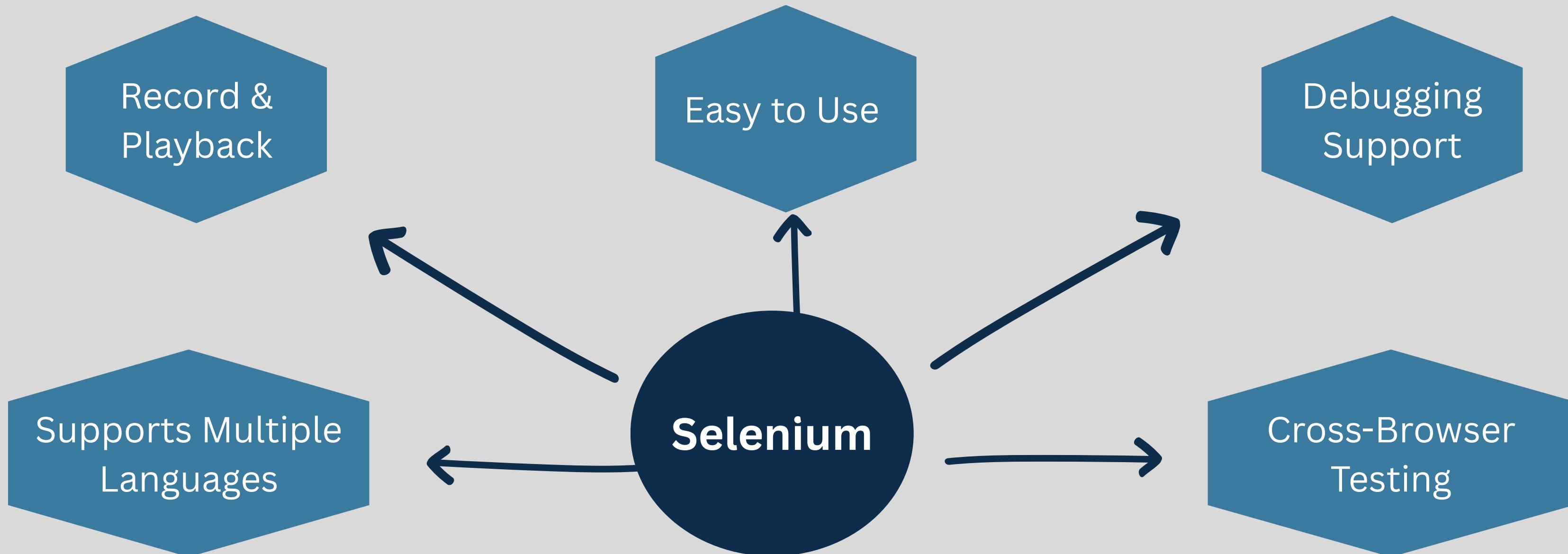
- **Saves Time** – Tests run faster than manual testing.
- **Reusable Test Scripts** – Same scripts can be used again and again.
- **Reduces Human Errors** – Automated tests are accurate and consistent.
- **Faster Feedback** – Bugs are detected quickly after changes.
- **Improves Test Coverage** – More test cases can be executed.
- **Cost Effective (Long Term)** – Saves effort and money over time.
- **Supports Regression Testing** – Old features can be tested after updates.
- **Runs 24/7** – Tests can run anytime without human involvement.

Choose automation testing tools

- **Application Type** - Choose tool based on application: Web, Mobile, or Desktop
Example: Selenium for web apps
- **Programming Language Support** - Tool should support languages you know (Java, Python, C#)
- **Ease of Use** - Tool should be easy to learn and use
GUI-based tools are better for beginners
- **Cost & Licensing** - Check if tool is open-source or paid
Example: Selenium is free
- **Test Environment Support** - Should work on different browsers, OS, and devices
- **Integration Capability** - Tool should integrate with CI/CD tools (Jenkins, Git)

Selenium IDE

- Selenium IDE is a record and playback tool in the Selenium tool suite.
- It is mainly used for creating simple automation test cases.
- Works as a browser extension (Chrome & Firefox).
- Suitable for beginners with little or no programming knowledge.



Advantages

- Easy installation and learning.
- Quick test creation.
- Good for prototyping and small projects.

Limitations

- Not suitable for complex test scenarios.
- Limited control compared to Selenium WebDriver.
- Not ideal for large-scale testing.

Selenium RC (Remote Control)

- Selenium RC is an older Selenium tool used for automating web applications.
- It allows testers to write test scripts in multiple programming languages.
- It works by sending commands to the browser through a Selenium Server.
- Selenium RC is now deprecated and replaced by Selenium WebDriver.

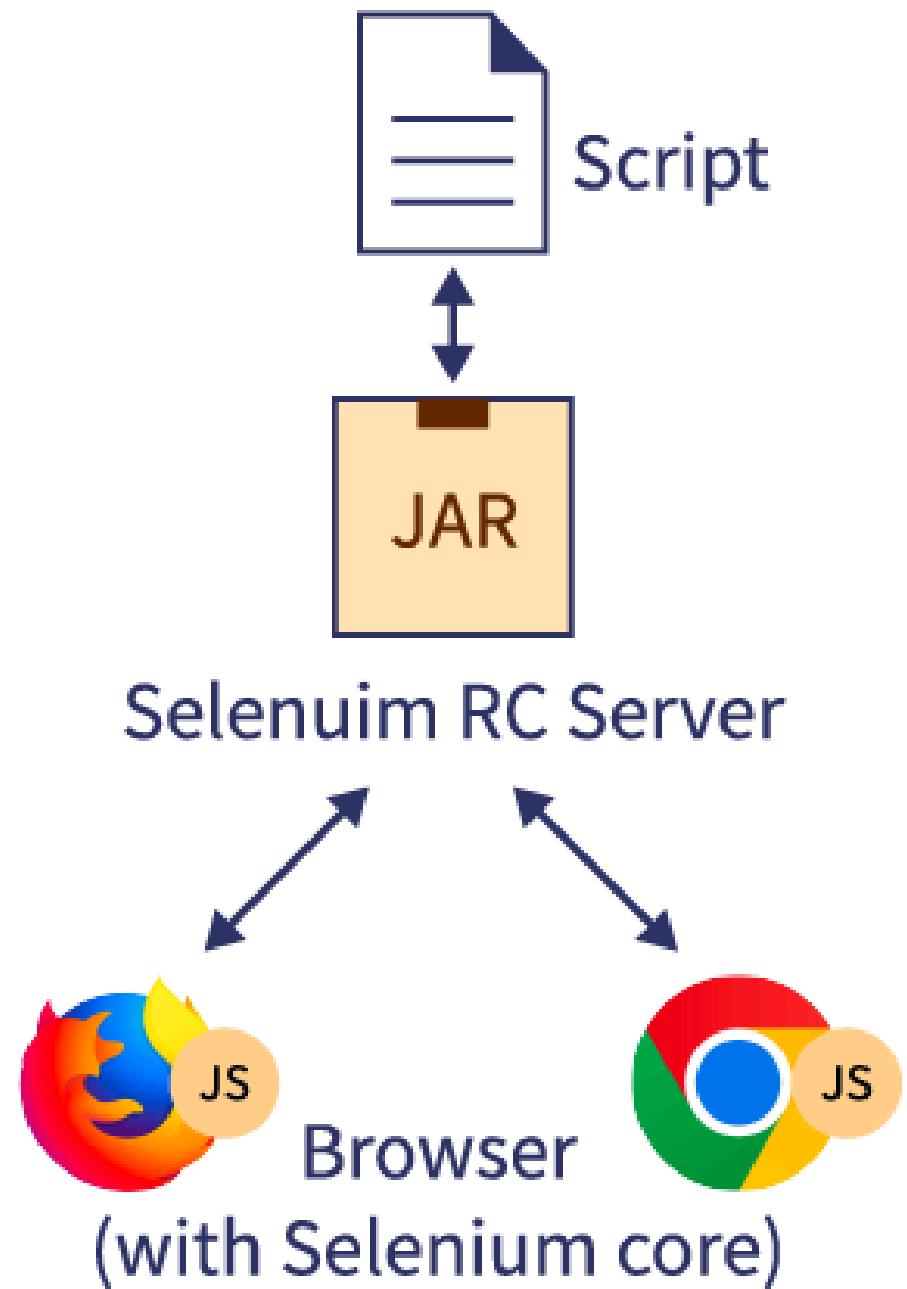
Features

- Supports multiple browsers.
- Supports multiple programming languages.
- Allows data-driven testing.

Limitations

- Slower execution due to server dependency.
- Complex architecture.
- Less stable compared to WebDriver.
- No longer supported in latest Selenium versions.

Selenium RC



Selenium WebDriver

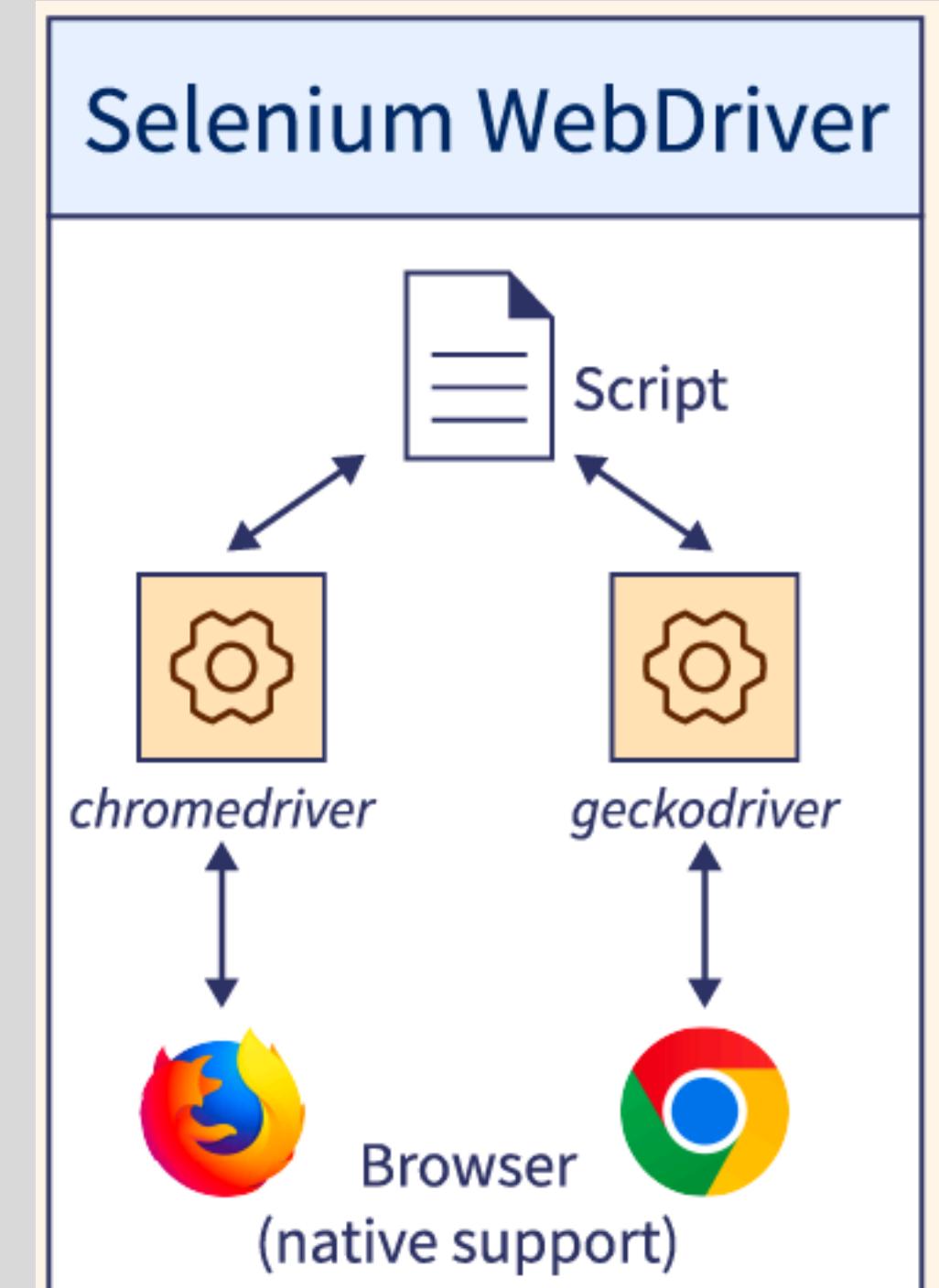
- Selenium WebDriver is a modern automation tool used for testing web applications.
- It directly communicates with the web browser without using any server.
- It is part of the Selenium tool suite.
- WebDriver is faster and more reliable than Selenium RC.

Features

- Supports multiple browsers (Chrome, Firefox, Edge).
- Supports multiple programming languages (Java, Python, C#, etc.).
- Works on multiple operating systems.
- Supports parallel and cross-browser testing.

Advantages

- Fast execution.
- No server dependency.
- Better stability.
- Actively used in the industry.



Selenium Grid

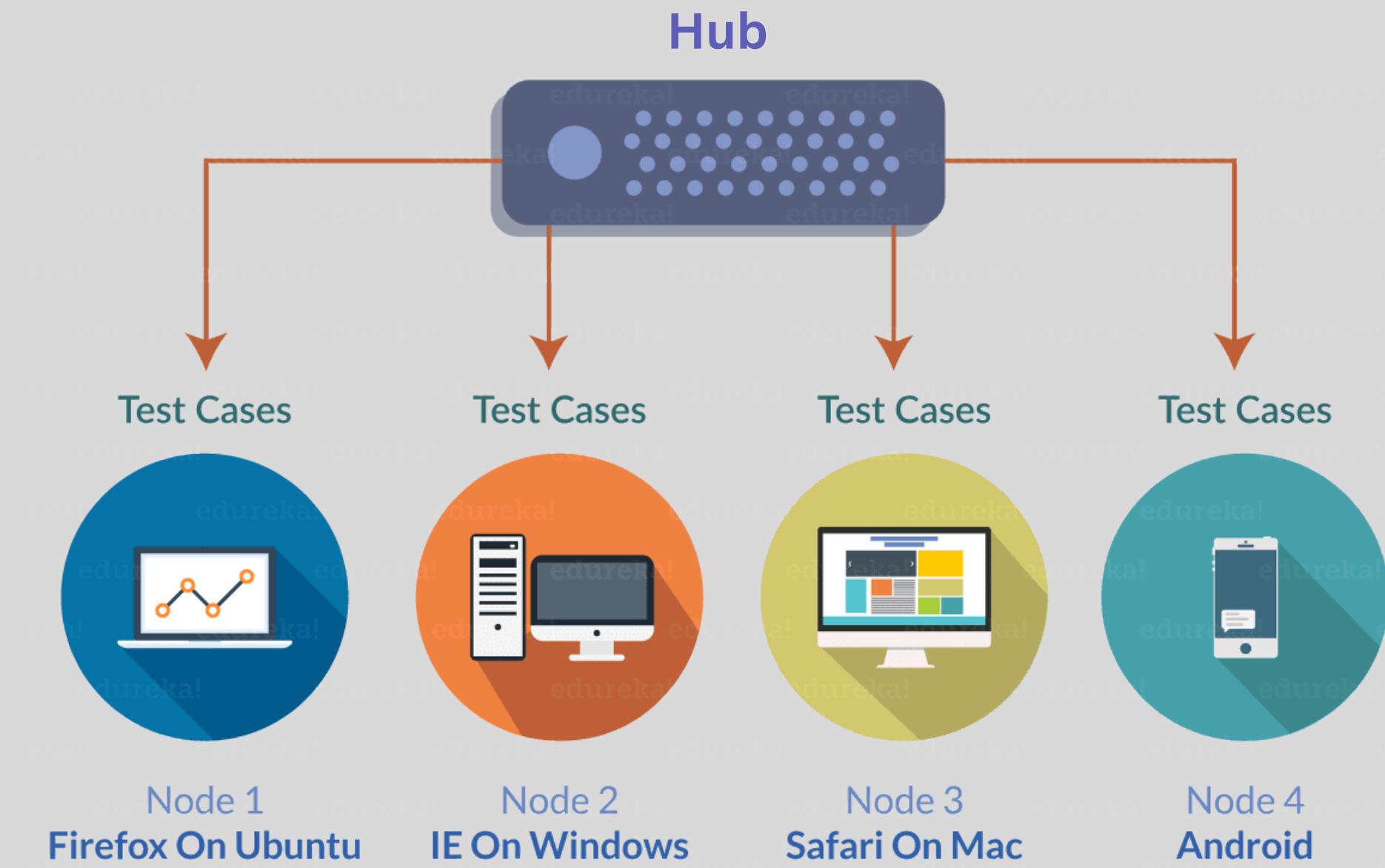
- Selenium Grid is a tool used to run multiple test cases at the same time.
- It helps in parallel testing across different browsers and operating systems.
- Selenium Grid is used with Selenium WebDriver.
- It reduces overall test execution time

Features

- Parallel test execution.
- Cross-browser testing.
- Supports multiple OS and browsers.
- Improves testing speed and efficiency.

Advantages

- Faster execution.
- Better resource utilization.
- Ideal for large projects.



Feature / Aspect	Selenium RC (Remote Control)	Selenium WebDriver	Selenium Grid
Definition	Legacy tool to automate browsers by injecting JavaScript.	Modern browser automation tool controlling browser natively.	Tool to run tests in parallel across multiple machines/browsers.
Architecture	Client-server: Requires RC server to communicate with browser.	Direct communication with browser without server.	Hub-node architecture for distributed testing.
Browser Support	Limited; relies on JavaScript injection.	Supports all major browsers natively (Chrome, Firefox, Edge, Safari).	Supports multiple browsers on multiple machines simultaneously.
Performance	Slower due to client-server communication.	Faster; communicates directly with browser.	Speeds up execution by running tests in parallel.
Language Support	Java, C#, Perl, Python, Ruby, PHP.	Same languages as RC (Java, C#, Python, Ruby, JavaScript).	Works with WebDriver scripts in any language.
Parallel Execution	Not built-in; needs extra setup.	Not built-in; can be combined with Grid or TestNG for parallelism.	Specifically designed for parallel/distributed execution.
Current Relevance	Obsolete; replaced by WebDriver.	Widely used; industry standard.	Still used with WebDriver for parallel execution.

SoapUI (API Automation Tool)

- Open-source and paid tool for API testing
- Mainly used for SOAP and REST web services

Features

- Functional testing of APIs
- Supports SOAP, REST, HTTP, JDBC
- Automated test case execution
- Security testing & load testing

Advantages

- Easy to use GUI
- Good for backend testing
- Supports scripting using Groovy

Limitations

- Not used for UI testing
- Advanced features available only in paid version

Robotic Process Automation (RPA)

- Automation of repetitive business processes
- Uses software bots to mimic human actions

Advantages

- Reduces human effort
- Increases accuracy & speed
- Cost-effective for repetitive tasks

Features

- Automates rule-based tasks
- Works with existing applications
- No coding or low-code tools available

Limitations

- Not suitable for complex decision-making
- Depends on stable UI

Tosca (Test Automation Tool)

- Enterprise-level automation tool by Tricentis
- Used for end-to-end testing

Features

- Scriptless automation
- Model-based testing
- Supports web, desktop, API, SAP, mobile

Advantages

- No coding required
- Faster test creation
- Good for large applications

Limitations

- Expensive
- Requires training

Appium (Mobile Automation Tool)

- Open-source tool for mobile application testing
- Used for Android and iOS apps

Advantages

- Free and open source
- Supports multiple programming languages
- Reusable code for Android & iOS

Features

- Supports native, hybrid & mobile web apps
- Uses WebDriver protocol
- Works with real devices & emulators

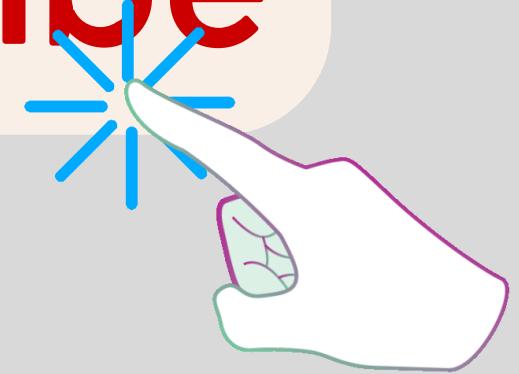
Limitations

- Slower compared to native tools
- Setup can be complex



SHARE

subscribe



Thank You