Efficiency Matters: Mastering Dockerized Scalable Grid for UI Tests

Sargis Sargsyan
Senior Test Architecture Manager at Align Technology

HUSTEF
HUNGARIAN SOFTWARE TESTING FORUM

Why you need **Spee** in Automation testing

d

76% of leaders find improving speed with quality is key to achieving top business goals

b **n**

54%

l
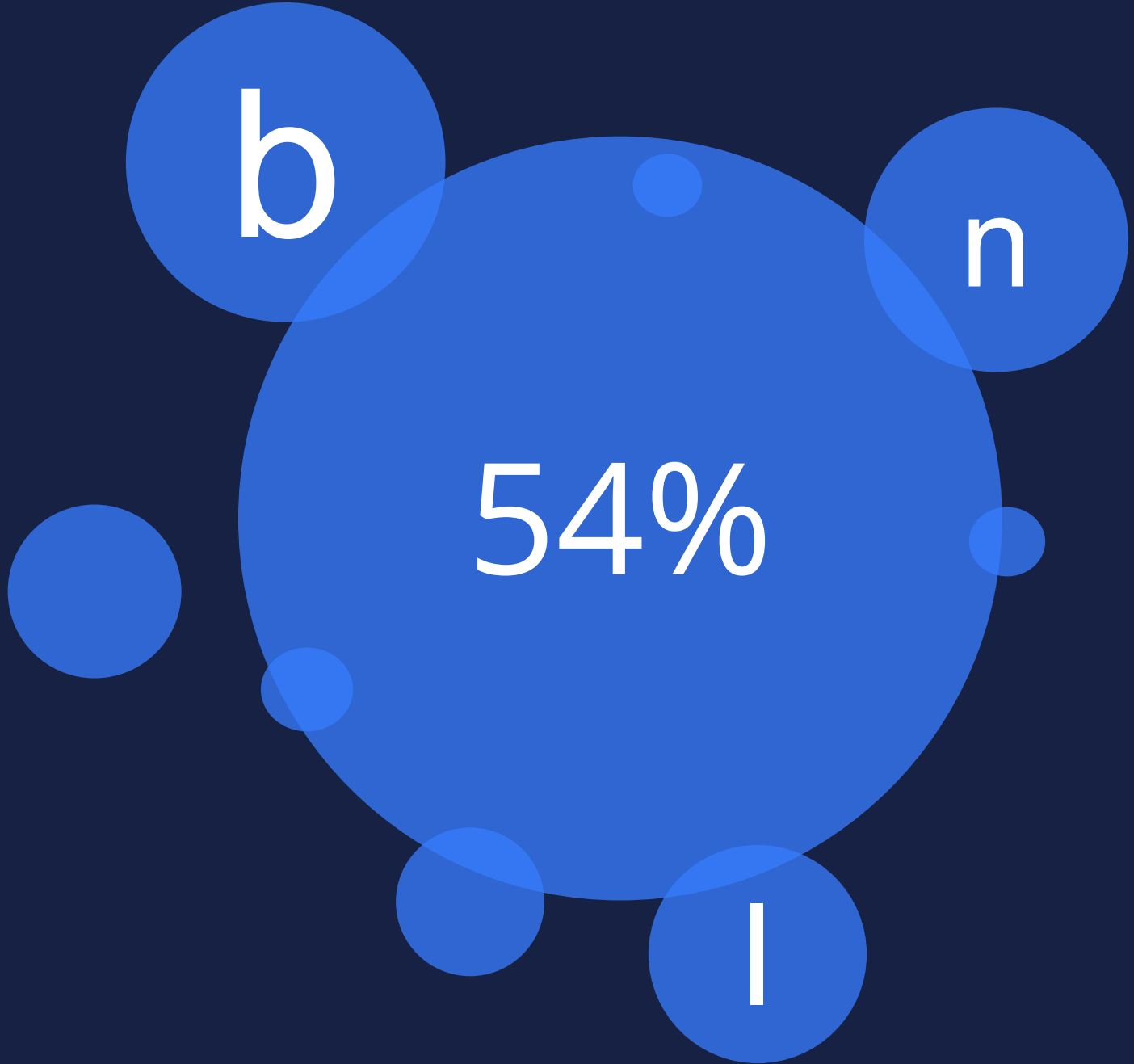
Over half of organizations have trouble keeping up with the changing customer demands

V **d**

56%

c
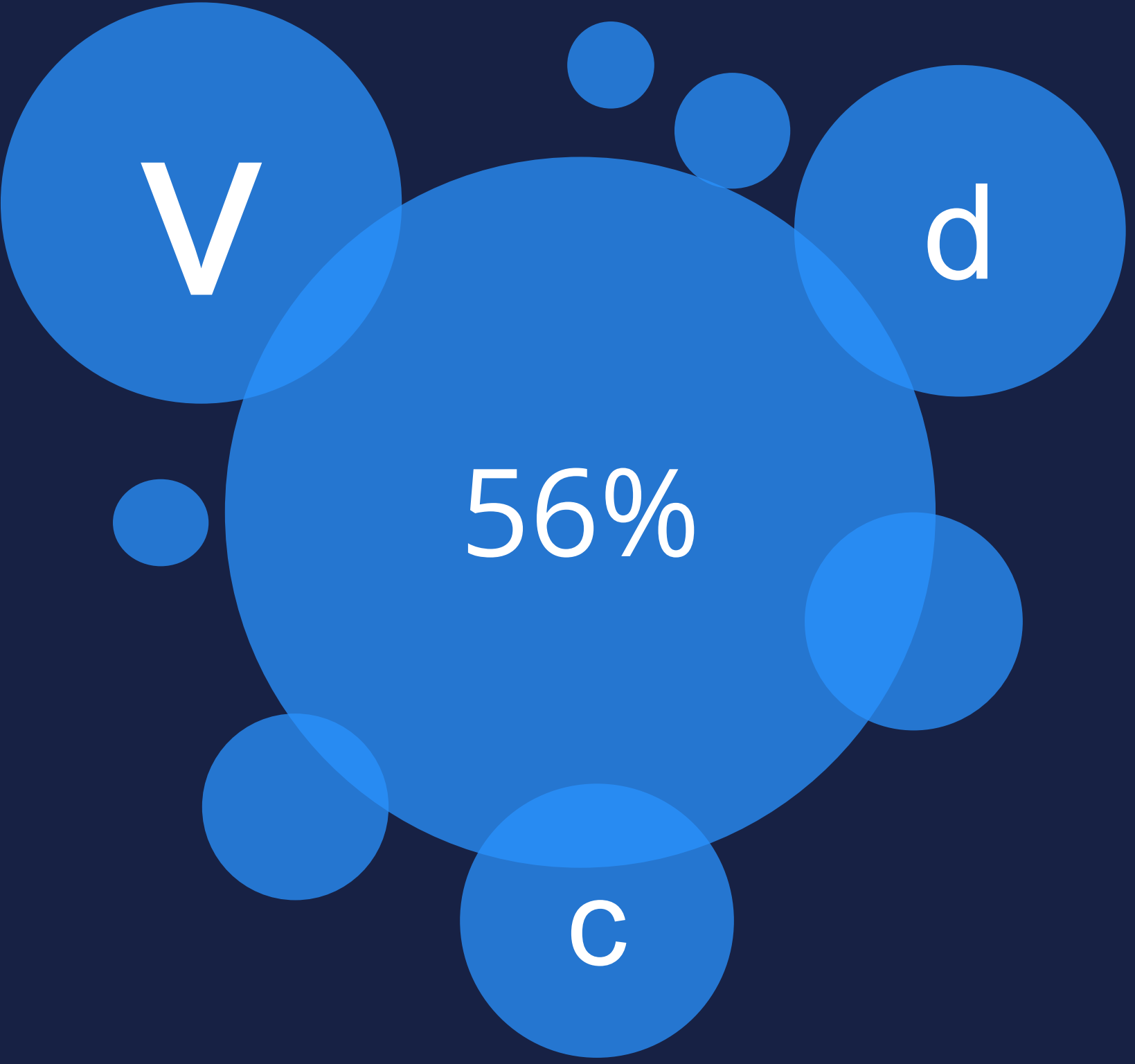
Over half of organizations prioritize speed over quality

p **q**

49%

g
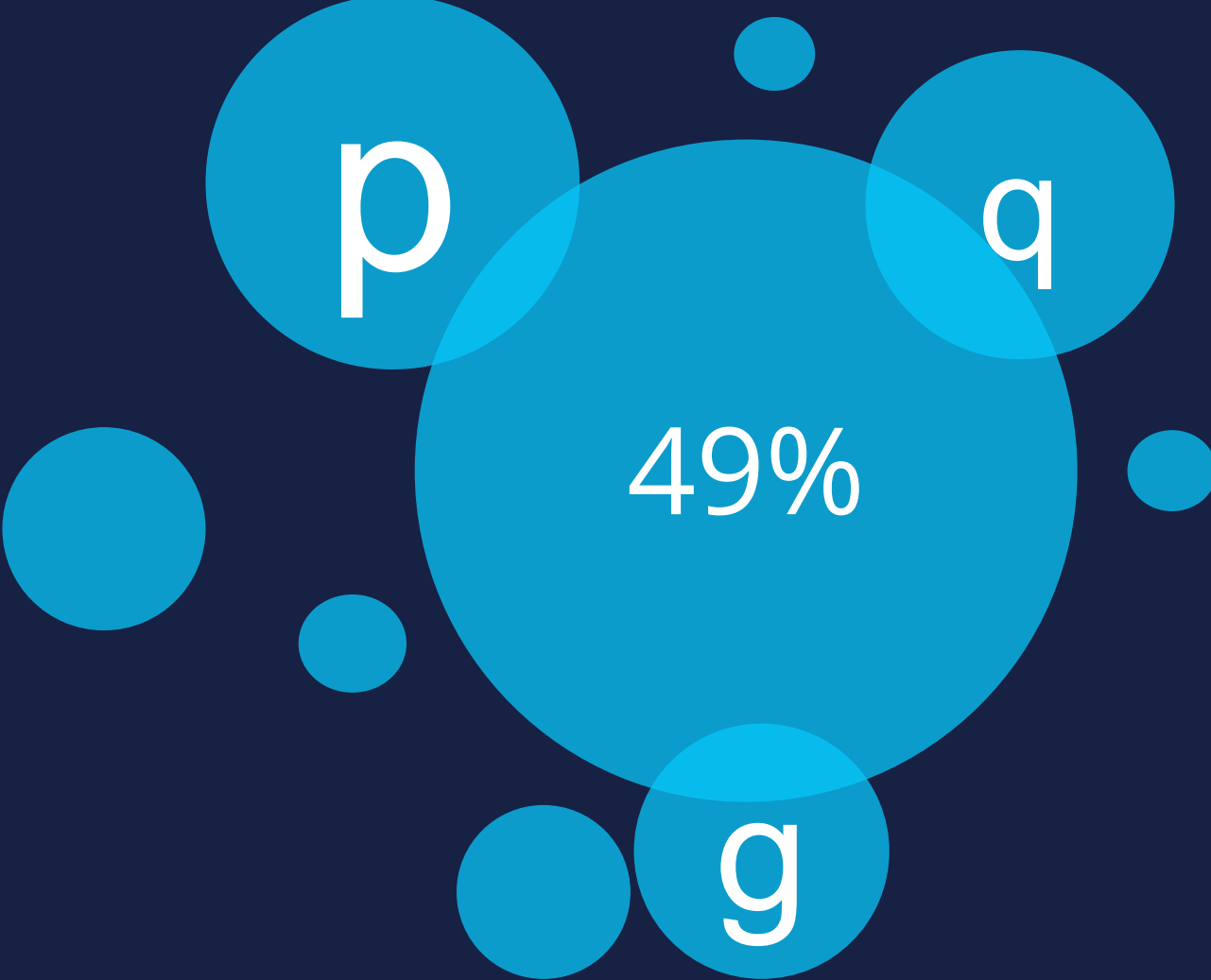
Organization has lost revenue due to poor Customer experience

*Forrester Consulting, on behalf of Tricentis, research Sept 2021

The speed of iteration accelerates the pace of innovation

DEV

h

OPS

a

Code

Plan

02

01

Test

Release

03

04

Monitor

Deploy

06

05

Accelerating Iteration through **Tes**X**t** Automation

Test Automation
Turbocharger for Iteration

Shifting Left — a

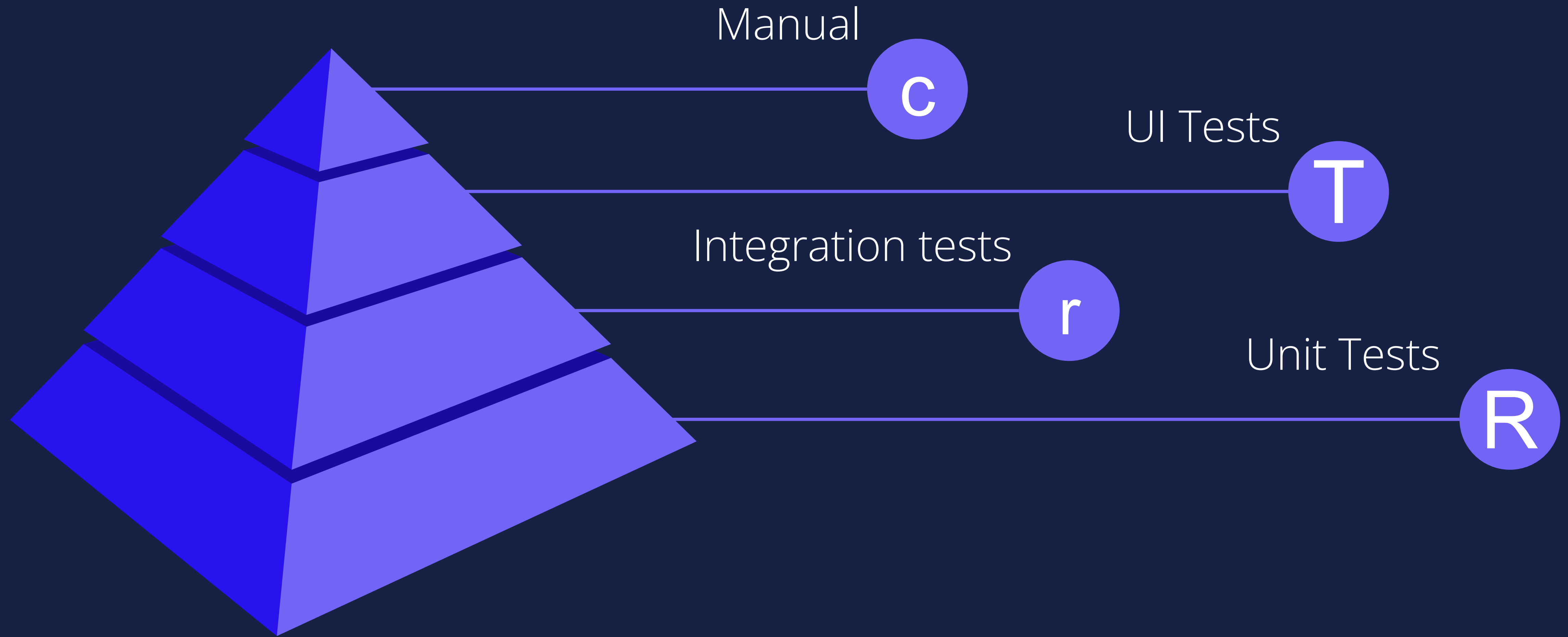Continuous Testing — o

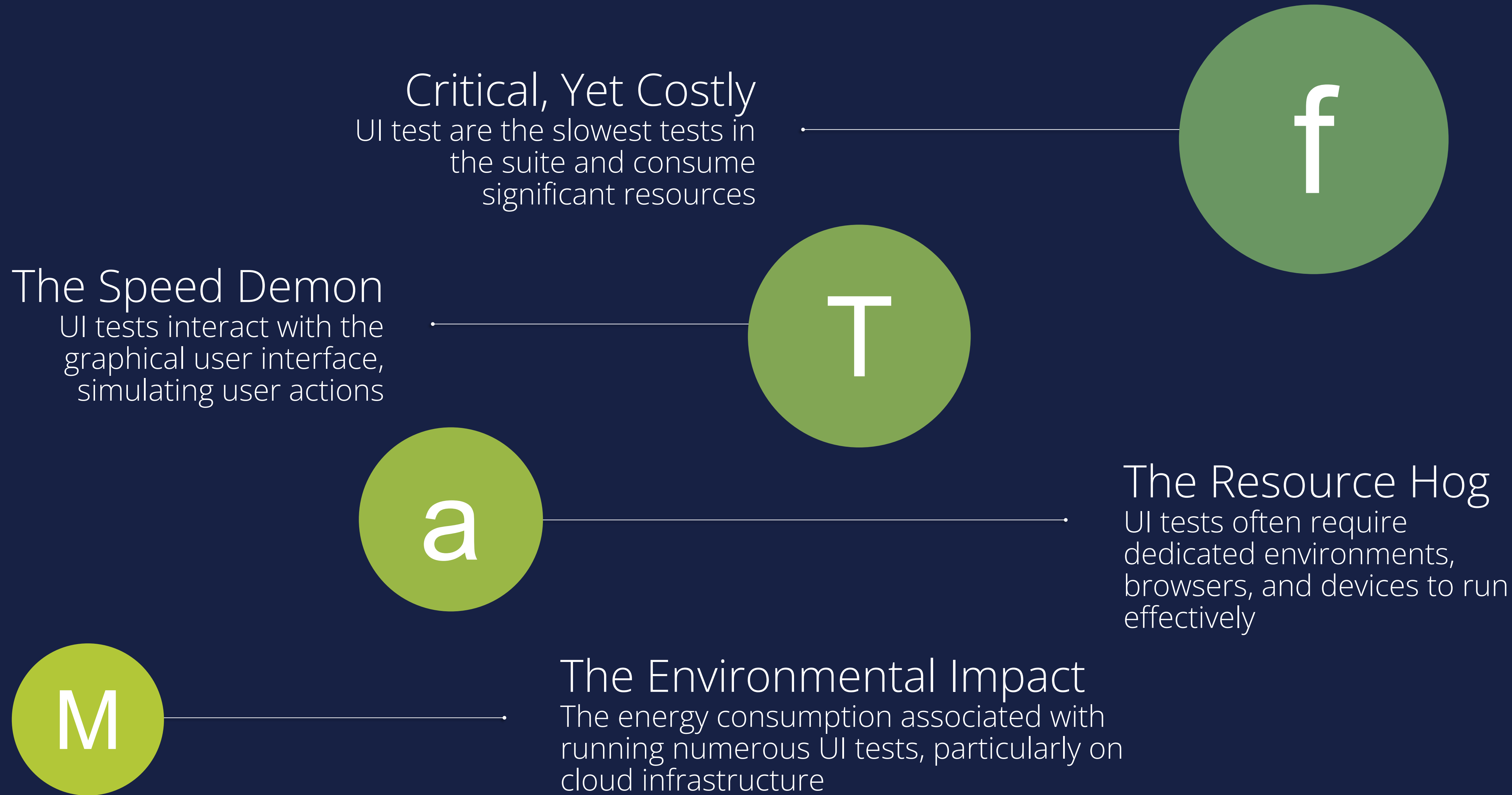Faster Feedback Loops — I

Reliable Results — R

Test automation supercharges the development process by integrating tests early and running them continuously throughout the software development lifecycle.

11 12 1 2 3 4 5 6 7

Manual

c

UI Tests

T

Integration tests

r

Unit Tests

R

The UI **Automation** Dilemma

Critical, Yet Costly
UI test are the slowest tests in the suite and consume significant resources

f

The Speed Demon
UI tests interact with the graphical user interface, simulating user actions

T

a

The Resource Hog
UI tests often require dedicated environments, browsers, and devices to run effectively

M

The Environmental Impact
The energy consumption associated with running numerous UI tests, particularly on cloud infrastructure

Parallelism and Optimization Strategies

Optimized Test Data Preparation: The Fuel for Efficiency

Data-Driven Testing: Use external data sources to create a variety of test scenarios without manually inputting data each time.

i

Test Data Management: Implement strategies for creating, maintaining, and reusing test data to minimize setup time.

d

Mock Data: Use mock data for testing components that rely on external services to avoid dependencies and improve test stability.

J

Optimal Test Suite Setup: Efficiency by Design

Atomic Tests: Focus on small, independent tests that target specific functionalities, leading to faster and more reliable tests.

X

Prioritize Critical Tests: Identify and prioritize tests that cover high-risk or high-impact areas of your application.

m

Modular Design: Break down tests into reusable components to reduce duplication and maintenance effort.

w

Parallel Execution: Unleashing the Speed Boost

The Power of Many: Run multiple UI tests simultaneously across different environments, browsers, or devices.

u

Shorter Feedback Loops: Dramatically reduce the time it takes to complete your entire UI test suite, enabling faster cycles

p

Increased Test Coverage: Parallel execution allows for more comprehensive testing without sacrificing speed

B

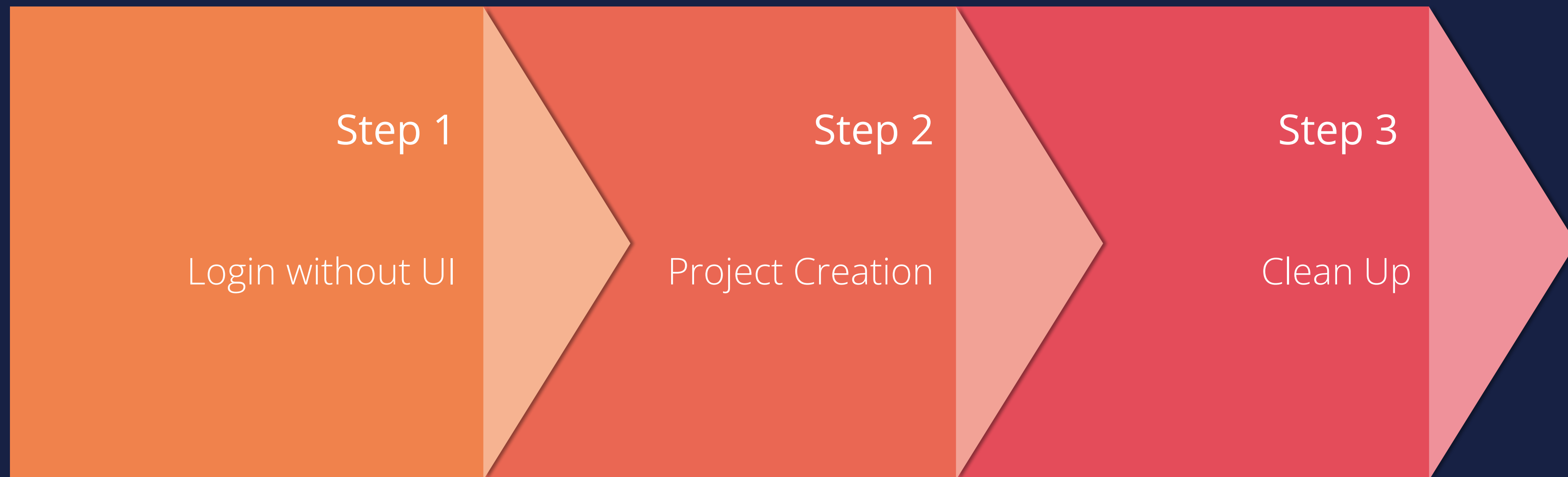Design **Test Data** Preparation

Test Data
Preparation

Test Run

Clean Up

Test Data Preparation with API

Step 1

Login without UI

Step 2

Project Creation

Step 3

Clean Up

Atomic **Tes**t Design

Importance of Atomic Tests

▷ Isolates specific functionalities

▷ Enables PARALLEL EXECUTION

▷ Easier maintenance

▷ Reduces debugging time

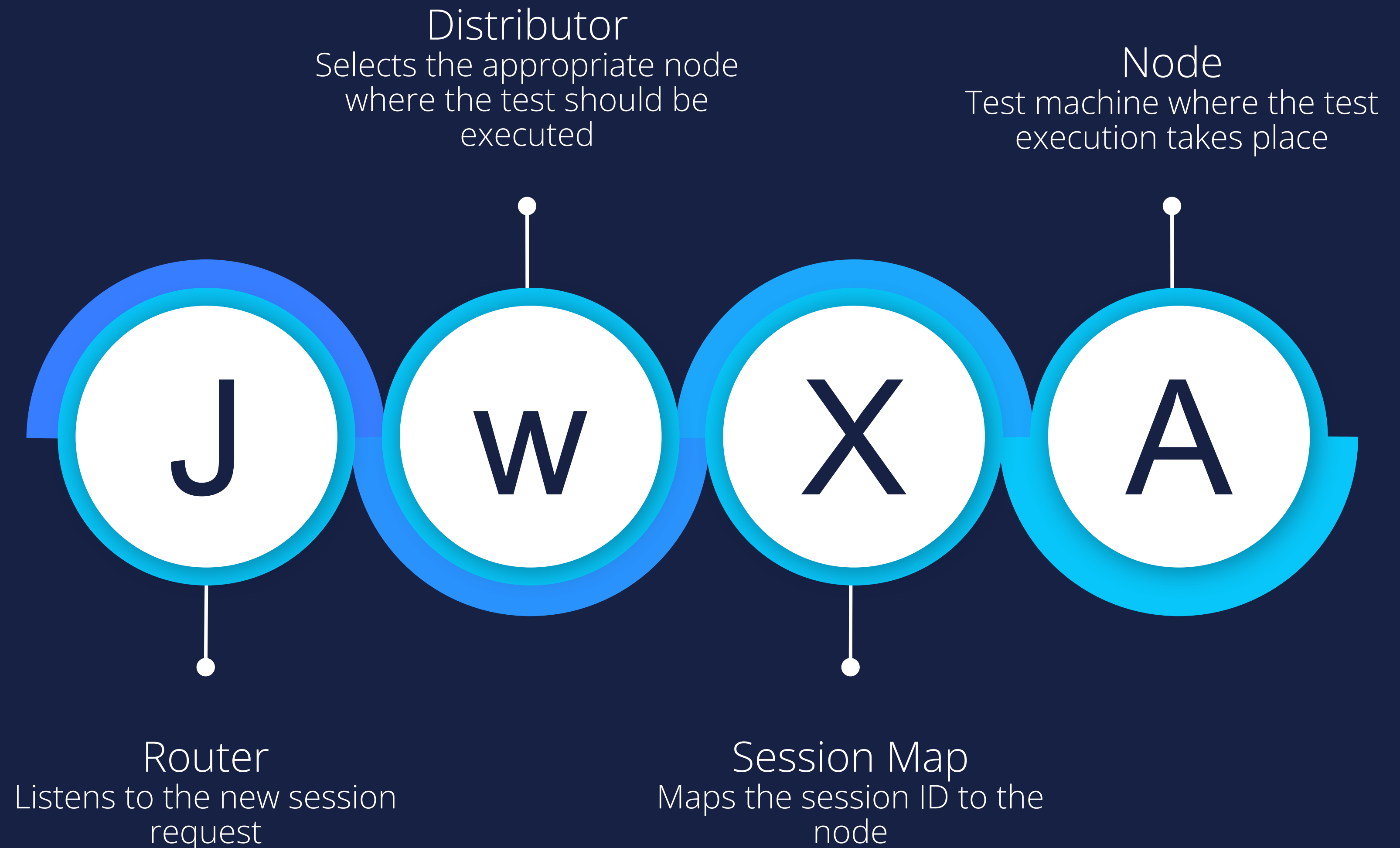▷ Provides quicker feedback in CI pipelines

Parallel Testing with Selenium Grid 4

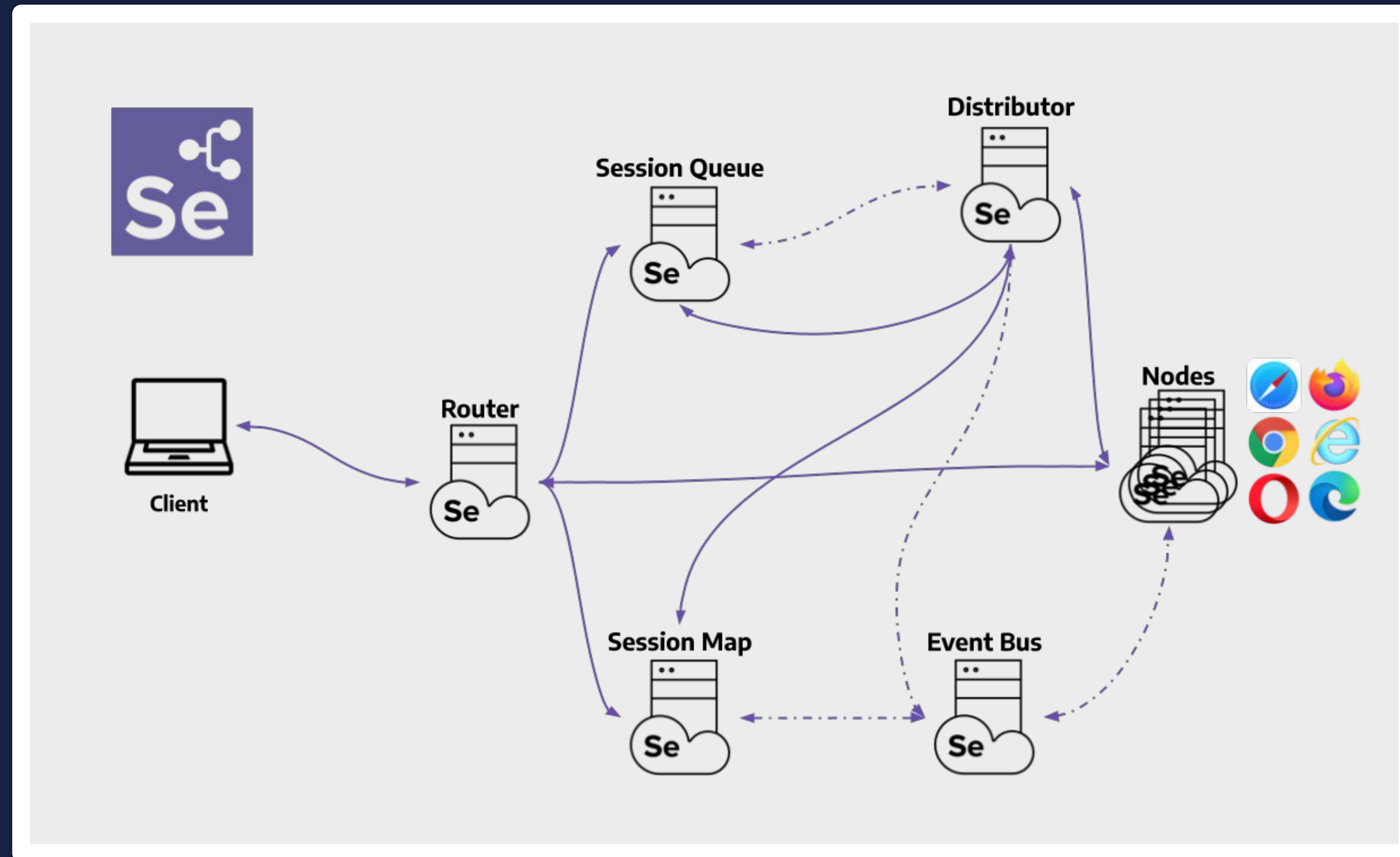# Selenium Grid 4 Architecture

Selenium 4 has a new architecture that supports four separate processes – Router, Session Map, Distributor, and Node.

Distributor
Selects the appropriate node where the test should be executed

Node
Test machine where the test execution takes place

J W X A

Router
Listens to the new session request

Session Map
Maps the session ID to the node

Selenium Grid 4 Architecture

Source: Selenium Grid Components

```yaml
version: "3"
services:
  node-docker:
    image: selenium/node-docker:4.20
    volumes:
      - ./assets:/opt/selenium/assets
      - ./NodeDocker/config.toml:/opt/bin/config.toml
      - /var/run/docker.sock:/var/run/docker.sock
    depends_on:
      - selenium-hub
    environment:
      HUB_HOST: selenium-hub
      SE_EVENT_BUS_HOST: selenium-hub
      SE_EVENT_BUS_PUBLISH_PORT: 4442
      SE_EVENT_BUS_SUBSCRIBE_PORT: 4443
      SE_NODE_PORT: 4444
      SE_NODE_GRID_URL: http://localhost:4444/wd/hub
      SE_NODE_MAX_SESSIONS: 1
      SCREEN_WIDTH: 1024
      SCREEN_HEIGHT: 768
      SCREEN_DEPTH: 24
      DBUS_SESSION_BUS_ADDRESS: "/dev/null"
      SE_OPTS: "--session-timeout 900"
      JAVA_OPTS: "-Xmx2g"

  selenium-hub:
    image: seleniarm/hub:4.20.0
    container_name: selenium-hub
    ports:
      - "4442:4442"
      - "4443:4443"
      - "4444:4444"
    environment:
      SE_SESSION_REQUEST_TIMEOUT: 1800
```

Selenium Grid
4.20.0 (revision 866c76ca80)

Overview

Sessions

Help

URI: http://172.19.0.3:5555

Stereotypes

Sessions: 0   Max. Concurrency: 8

0%

Help - All rights reserved - Software Freedom Conservancy 2024.

Queue size: 0

SD selenium-paralle...   main   /Users/ssargsyan/dev/conferance/...stng.xml

nfigReader.java   config.properties   testng.xml   ProjectTest.java

Project

selenium-parallel-execution-atomic
  .idea
  allure-results
  src
    main
      java
        com.sargissargsyan
          api
          base
          models
          pages
          utils
      resources
        config.properties
        log4j2.properties
    test
  target
  .gitignore
  pom.xml
  README.md
  testng.xml
External Libraries
Scratches and Consoles

```
1   <!DOCTYPE suite SYSTEM "https://testng.org/testng-1.0.dtd" >
2
3   <suite name="Parallel Execution Example" verbose="1" parallel="classes" thr
4       <test name="all-packages">
5           <classes>
6               <class name="com.sargissargsyan.IssueTest"/>
7               <class name="com.sargissargsyan.LoginTest"/>
8               <class name="com.sargissargsyan.ProjectTest"/>
9           </classes>
10      </test>
11  </suite>
12
```

Run   /Users/ssargsyan/dev/conferance/selenium-parallel-ex...

✓ Tests passed: 4 of 4 tests – 3 min 45 sec

Suite1   3 min 45 sec
  all-packages   3 min 45 sec
    IssueTest   1 min 20 sec
      issuePageNewStatus   1 min 18 sec
    ProjectTest   1 min 15 sec
      projectPage   1 min 15 sec
    LoginTest   42 sec 408 ms
      loginViaUI   9 sec 409 ms
      loginViaUIError   6 sec 973 ms

selenium-parallel-execution-atomic-design > testng.xml   3:40   LF UTF-8 4 spaces

Not Secure — 54.82.22.164

playwright-parallel-run   main   project.spec.ts

Selenium Grid
4.20.0 (revision 866c76ca80)

Overview

Sessions

Help

URI: http://172.19.0.3:5555

Stereotypes

Sessions: 0   Max. Concurrency: 8

0%

Help - All rights reserved - Software Freedom Conservancy 2024.

Queue size: 0

Project

- playwright-parallel-run ~/dev/conference
  - .idea
  - api
    - models
      - Issue.ts
      - Project.ts
      - User.ts
    - BaseService.ts
    - IssueService.ts
    - ProjectService.ts
  - node_modules library root
  - pages
    - BasePage.ts
    - HomePage.ts
    - IssuePage.ts
    - LoginPage.ts
    - NewProjectPage.ts
    - ProjectPage.ts
  - playwright
  - playwright-report
  - test-results
  - tests
    - issue.spec.ts
    - login.spec.ts
    - project.spec.ts
  - utils
    - BrowserUtils.ts

.env   project.spec.ts

```
import { User } from '../api/models/User';
import { TestUtils } from '../utils/TestUtils';
import {BaseService, RegisterData} from '../api/BaseService';
import { test, expect } from '@playwright/test';
import {Project} from "../api/models/Project";
import {ProjectService} from "../api/ProjectService";
import {ProjectPage} from "../pages/ProjectPage";
import {BrowserUtils} from "../utils/BrowserUtils";

test.describe('Project details page', () => {    ± Sargis Sargsyan
    let name: string;
    let username: string;
    let email: string;
    const password = 'Armenia2023';
    let user: User;
    let project: Project;

    test.beforeEach(async ({ page }) => {
        name = TestUtils.randomString(5);
        username = TestUtils.randomString(7);
        email = TestUtils.randomEmail(username);
        let registerData: RegisterData = {
            email: email,
            full_name: name,
            password: password,
            type: 'public',
            username: username,
            accepted_terms: true
        };
        project = {
```

Terminal   Local   Local (2)   Local (3)

SELENIUM_REMOTE_URL=http://54.82.22.164:4444/wd/hub npx playwright test

Running 6 tests using 6 workers
  6 passed (28.7s)

To open last HTML report run:

  npx playwright show-report

~/dev/conference/playwright-parallel-run git:[main]

playwright-parallel-run > tests > project.spec.ts   38:10   LF   UTF-8   4 spaces

Thank You!

@sargiset

https://sargissargsyan.com

/sargissargsyan