

Fake it until you `make it`

API Integration testing
with Testcontainers and WireMock



Oleg Nenashev,
WireMock Inc
@oleg_nenashev

`make it` ?

> whoami



[@oleg_nenashev](#)



[oleg-nenashev](#)



CD.FOUNDATION



CLOUD NATIVE
COMPUTING FOUNDATION

> whoami --java



[@oleg_nenashev](#)



[oleg-nenashev](#)

2006 - J# and Java ME

2010+ - patches to Hudson/Jenkins

2013+ - Jenkins & Java non-stop

2021+ - Testcontainers, Quarkus, etc.

2023+ - WireMock

Outline

- Intro to API integration testing
- Classification of the approaches
- Testcontainers and WireMock
- Using them together
- And demos!



These slides:

bit.ly/jugh-wiremock-testcontainers

Disclaimer

- This is a talk about “WireMock AND Testcontainers”, not “VERSUS” ir “OR”
- Your mileage may vary



Integration testing?

- Multiple technology stacks and toolchains
- Many teams involved
- Sacred knowledge
- [Almost] always too late



The new ecosystem is about APIs

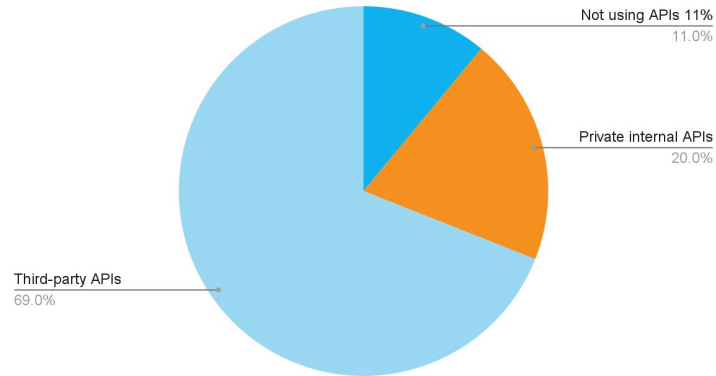


Over 90% of developers use APIs.

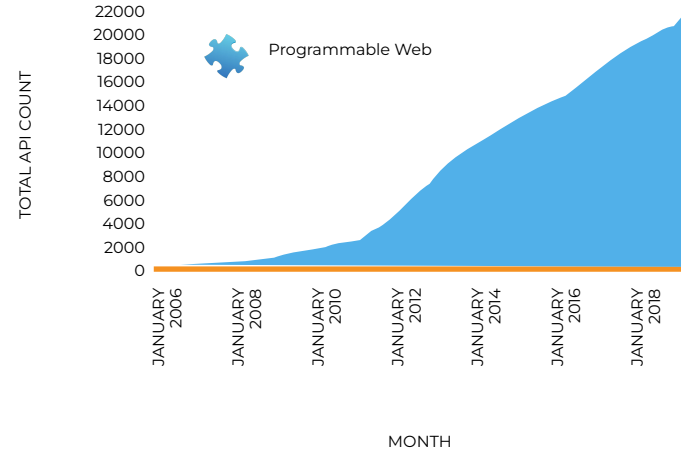


Skyrocket growth of APIs

Points scored



GROWTH IN WEB APIS SINCE 2005

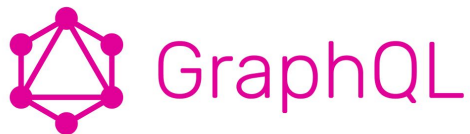


The growth over time of the Programmable Web API directory to more than 22,000 entries

APIs are essential for Integration and Integration Testing

APIs? Not just REST!

- Any interface!
- Async interfaces too
- Binary interfaces too



Open Standards

- Standard specifications for APIs
- Developer tools for the specification
- Ways to share specifications between teams at early stages
- Tools for serving APIs based on specifications



Open API
Specification

www.openapis.org



Swagger

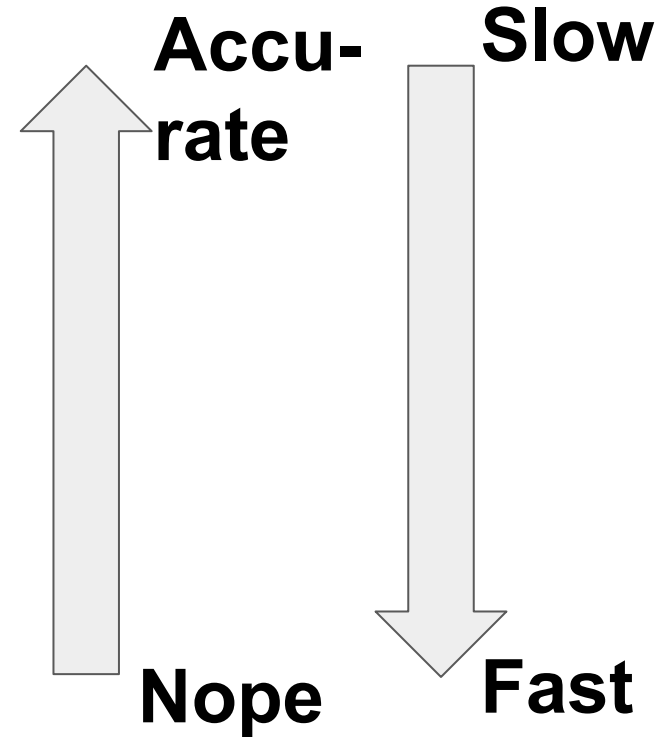


AsyncAPI

www.asyncapi.com

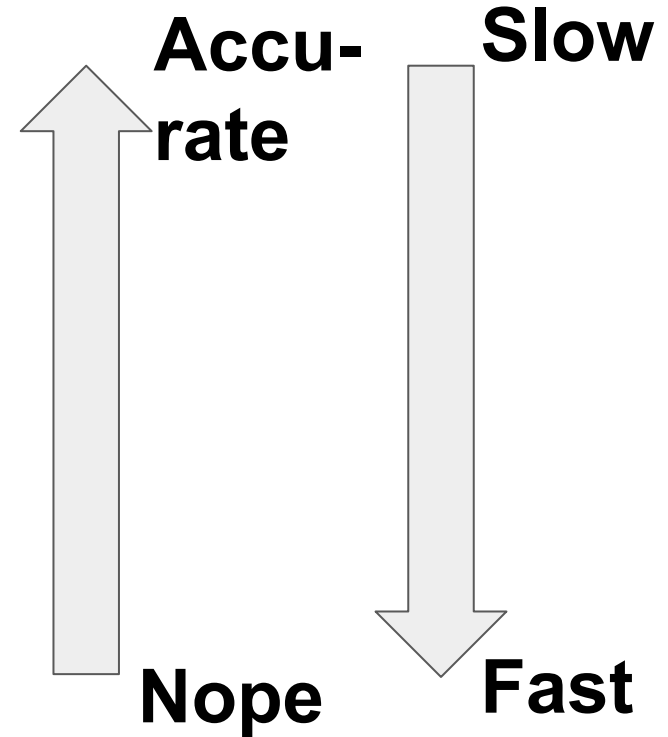
Ways to do API integration testing

1. Testing against Production/Staging servers
2. Testing against a simplified/containerized instance (e.g. Testcontainers)
3. Mocking at the API provider level (e.g. WireMock)
4. Mocking at the code level

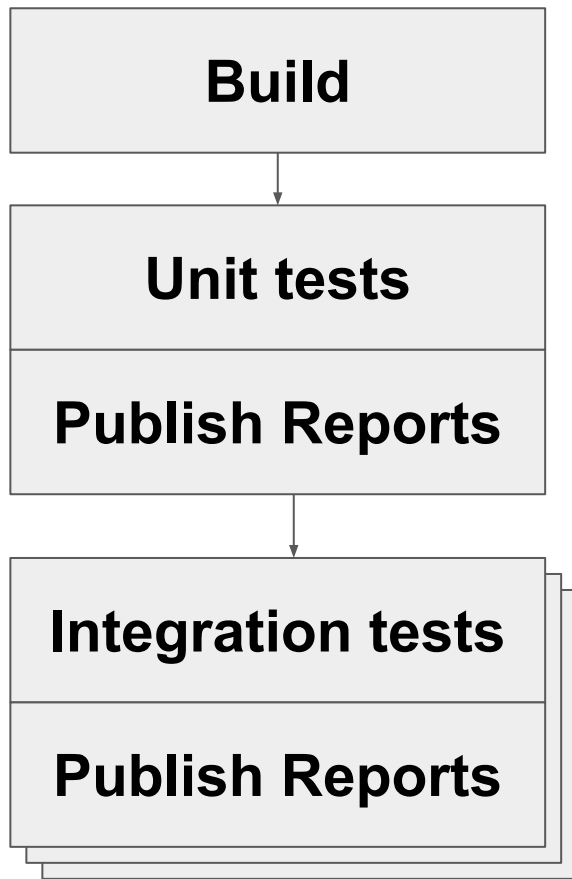


Ways to do API integration testing

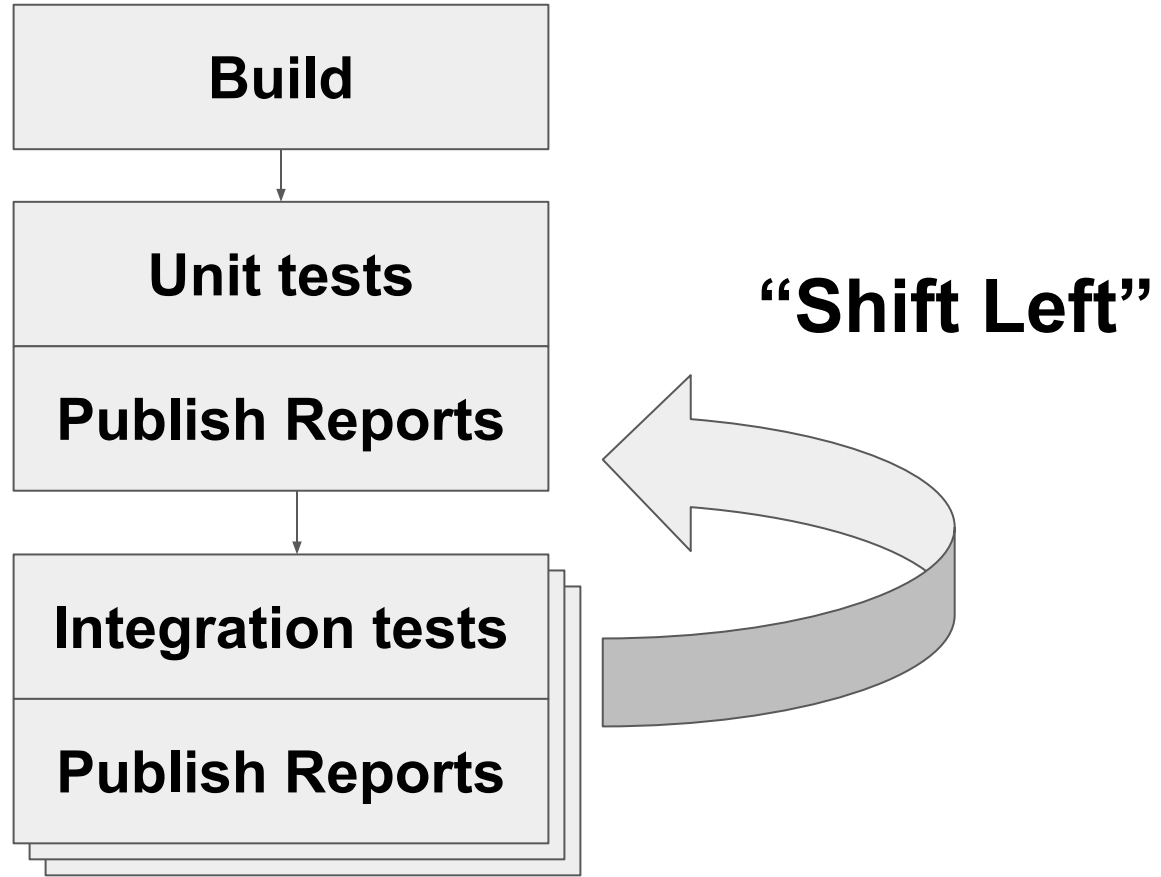
1. Testing against Production/Staging servers
- 2. Testing against a simplified/containerized instance (e.g. Testcontainers)**
- 3. Mocking at the API provider level (e.g. WireMock)**
4. Mocking at the code level



WAS

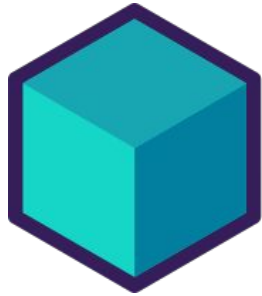


NOW





Shifting ITs left
with Testcontainers and WireMock

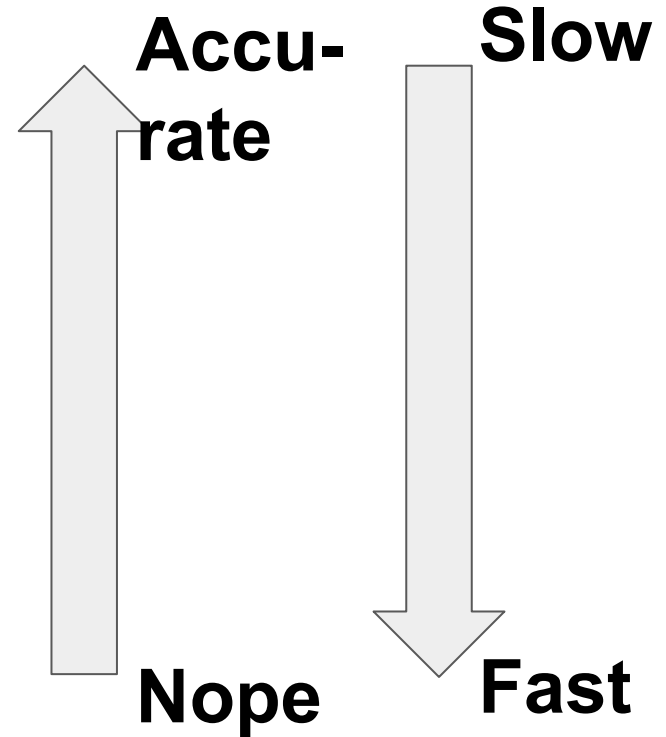


Testcontainers

testcontainers.org

Ways to do API integration testing

1. Testing against Production/Staging servers
- 2. Testing against a simplified/containerized instance (e.g. Testcontainers)**
3. Mocking at the API provider level (e.g. WireMock)
4. Mocking at the code level



Key features

- Integration into your test frameworks
- Container management layer
- On-demand single-shot containers for your unit and integration tests
- On-demand image build



testcontainers.org

Example - JUnit 5

```
@Testcontainers
class WireMockContainerJUnit5Test {

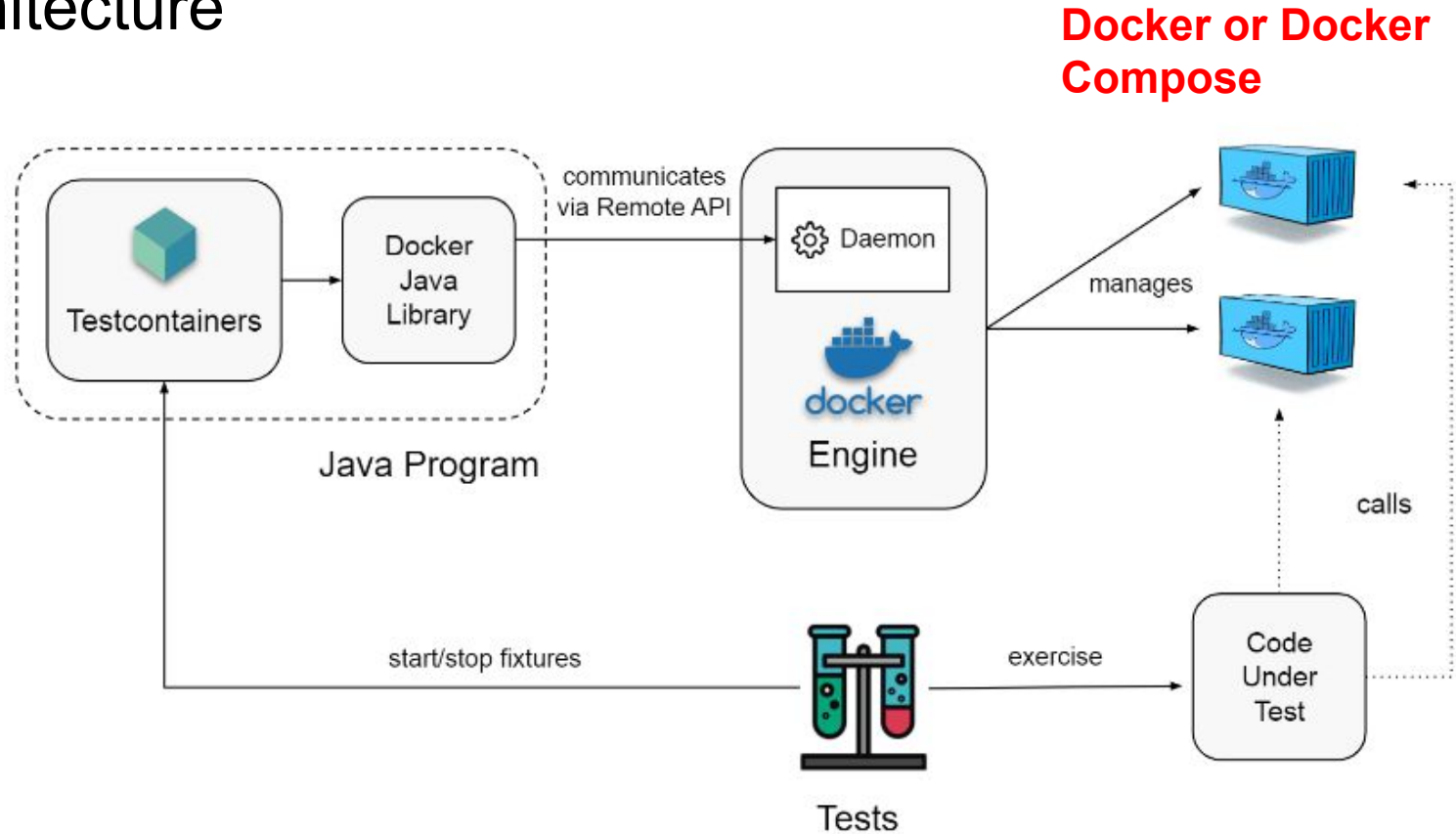
    @Container
    WireMockContainer wiremockServer = new WireMockContainer("2.35.0")
        .withMapping("hello", WireMockContainerJUnit5Test.class, "hello-world.json");

    @Test
    void helloWorld() throws Exception {
        // given
        String url = wiremockServer.getUrl("/hello");

        // when
        HttpResponse response = new TestHttpClient().get(url);

        // then
        assertThat(response.getBody())
            .as("Wrong response body")
            .contains("Hello, world!");
    }
}
```

Architecture



Containers are not always slow!

- Caching Docker image builds
- Suspending containers between tests
- Graceful termination
 - github.com/testcontainers/moby-ryuk



testcontainers.org

Modules!

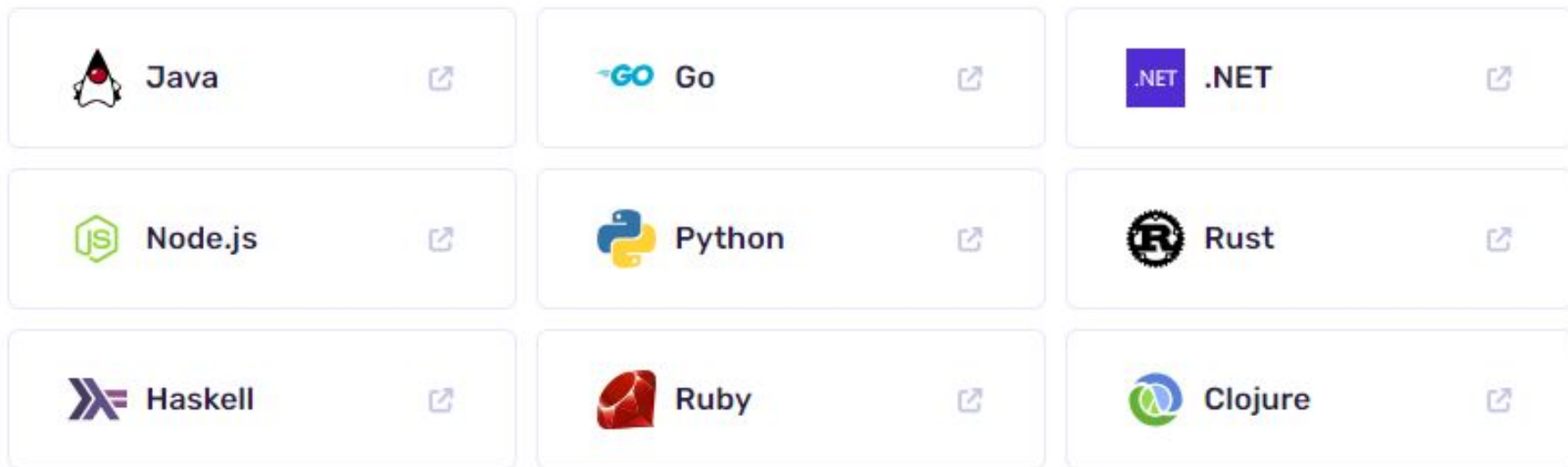
Test Anything You Can Containerize: Database, Message Broker, And More



testcontainers.com/modules

Not just Java

<https://www.testcontainers.org/>



C / C++

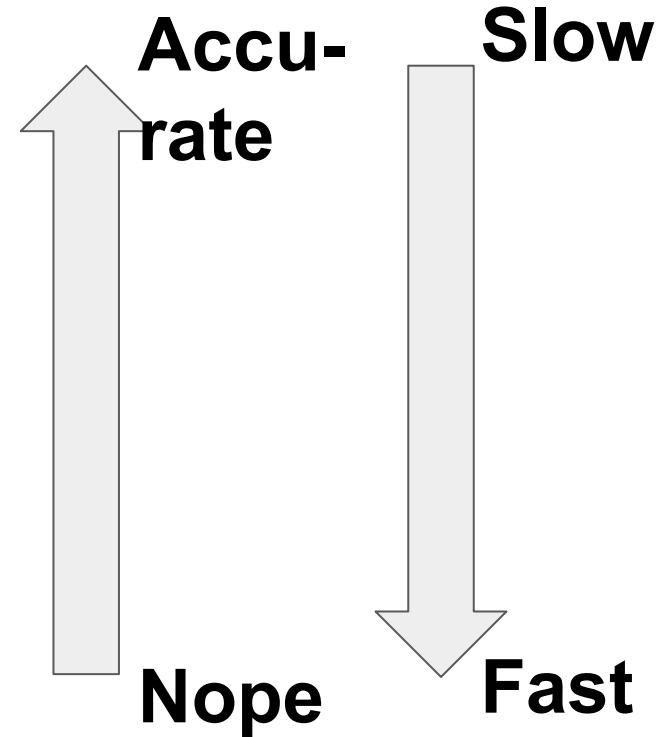
WiP: [oleg-nenashev/testcontainers-c](https://github.com/oleg-nenashev/testcontainers-c)

WIREFMOCK

<https://wiremock.org>

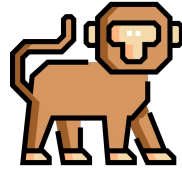
Ways to do API integration testing

1. Testing against Production/Staging servers
2. Testing against a simplified/containerized instance (e.g. Testcontainers)
- 3. Mocking at the API provider level (e.g. WireMock)**
4. Mocking at the code level



API Mocking for Java

Open Source



MockServer



Proprietary



POSTMAN



Traffic Parrot®



WireMock 101

WireMock is a tool for building mock APIs

Created in 2011 as a Java test library and CLI

You can can:

- Create stable development environments
- Isolate yourself from flakey 3rd parties
- Simulate APIs that don't exist yet



WireMock 101

IF (request_url)

*THEN (response) **

* it gets **MUCH** more complex

WireMock Config JSON:

```
{
  "request": {
    "method": "GET",
    "url": "/some/thing"
  },
  "response": {
    "status": 200,
    "body": "Hello, world!",
    "headers": {
      "Content-Type": "text/plain"
    }
  }
}
```

wiremock.org/docs/stubbing

Breaking down JSON files

Using WireMock Cloud

app.wiremock.cloud

Hacker News Mock API

Share

API template...

https://w02r3.wiremockapi.cloud/

+ New

Refresh

Filter stub list

Get Hacker News Item - ask

GET /item/{id}.json

200

```
["by": "tel", "descendants": 16, "id": 121003, "kids": [121016, 121
```

Get Hacker News Item - comment

GET /item/{id}.json

200

```
["by": "norvig", "id": 2921983, "kids": [2922097, 2922429, 292
```

Get Hacker News Item - poll

GET /item/{id}.json

200

```
["by": "pg", "descendants": 54, "id": 12
```

Get Hacker News Item - job

GET /item/{id}.json

Request

Name

Get Hacker News Item - ask

+ Add description

GET

/item/{id}.json

> Advanced ?

> Scenario ?

Response

Direct

Fault

Proxy

Status

200

Enable dynamic response templating ?

Save

Clone

Cancel

Example - JUnit 5

```
@WireMockTest
public class DeclarativeWireMockTest {

    @Test
    void test_something_with_wiremock(WireMockRuntimeInfo wmRuntimeInfo) {
        // The static DSL will be automatically configured for you
        stubFor(get("/static-dsl").willReturn(ok()));

        // Instance DSL can be obtained from the runtime info parameter
        WireMock wireMock = wmRuntimeInfo.getWireMock();
        wireMock.register(get("/instance-dsl").willReturn(ok()));

        // Info such as port numbers is also available
        int port = wmRuntimeInfo.getHttpPort();

        // Do some testing...
    }
}
```


WireMock 201 and beyond - Key features



Advanced request
matching



Dynamic response
templating



Run in your unit tests, on your
laptop or in your test environment.



Fault and latency
injection



Record / Playback



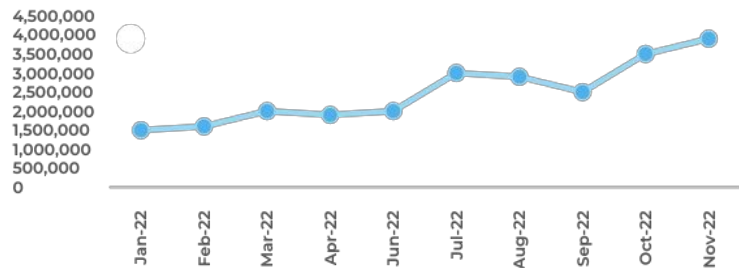
Java, Python, HTTP and
JSON file APIs

<https://github.com/wiremock>

WireMock

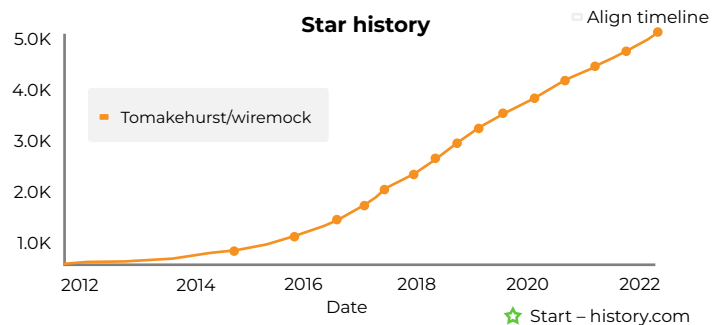
~4M MONTHLY DOWNLOADS OF THE JAVA LIBRARY

Downloads Over the Last 12 Months For com..github.tomakehurst



5K STARS IN GITHUB AND GROWING

Star history



>5M

Downloads a month

3,036

Used by 3,036 Open
Source Projects

>46M

Pulls of WireMock
Docker Image

181

Contributors

1.3k

Forks



wiremock.org



ONE TOOL TO MOCK
THEM ALL

Not just a Java tool

- WireMock Docker - standalone server as a container
 - github.com/wiremock/wiremock-docker
 - Helm charts for Kubernetes too
 - Testcontainers modules
- Adapters and Implementations for technology stacks
 - All major stacks: Python, JavaScript/TS, .NET, Golang, Rust ...
 - Native support: Golang, Rust
 - Protocol adapters, e.g. gRPC
 - Android support
- Soon: gRPC and GraphQL adapters

* Many components are maintained by contributors outside main orgs. Working on consolidation

WireMock Ecosystem



WireMock & JVM Ecosystem - projects for...

- Java
- Kotlin
- Scala
- Groovy
- Clojure
- (?)Android
- JUnit 4/5
- Testcontainers
- Spring Boot
- Spring Cloud Contract
- Quarkus
- Maven/Gradle
- Spock
- Pact

Extensibility

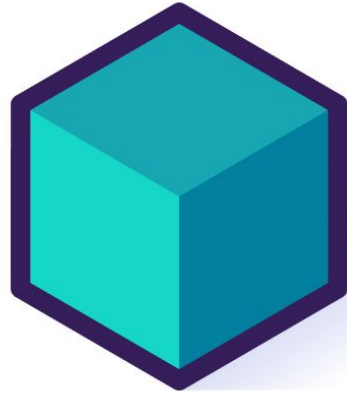
- Flexible definition formats
- Open Source, Apache v2 License
- Open APIs
- Extensions engine



<https://wiremock.org/docs/>



Disclaimer



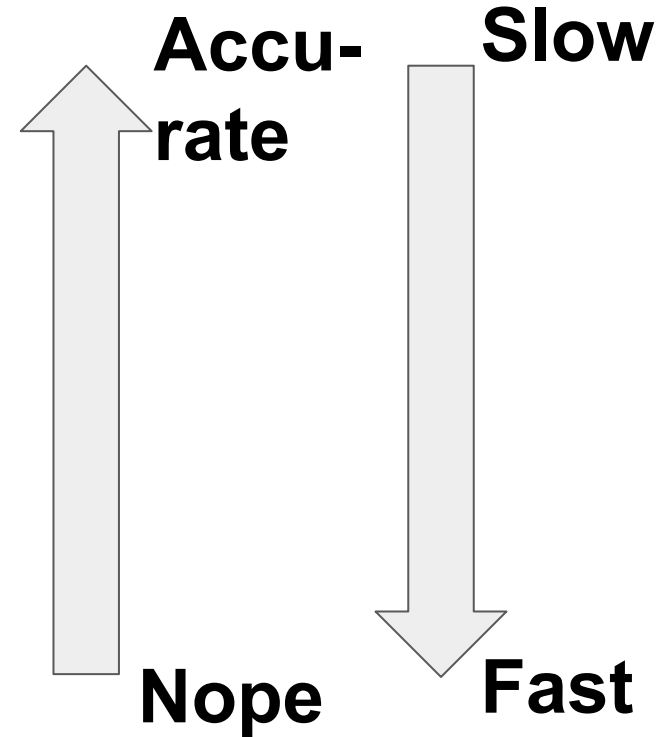
OR



Choosing a Tool for your job

Ways to do API integration testing

1. Testing against Production/Staging servers
2. **Testing against a simplified/containerized instance (e.g. Testcontainers)**
3. **Mocking at the API provider level (e.g. WireMock)**
4. Mocking at the code level



Feature / Project	WireMock	Testcontainers
JUnit Integrations	+	+
Battle-tested	+	+
Server side is NOT needed	+	-
Tests the server side	-	+
Multi-platform	+	+
“Marketplace”	+	+
Extensible	+	+
SaaS / Enterprise Support	+	+

There is no silver bullet!

Choosing a right tool * **

- Need database or complex dynamic environment?
 - YES => Testcontainers
- Do you have a ready implementation?
 - NO => WireMock
- Is it containerized and lightweight?
 - NO => WireMock
- Can you easily trigger the desired scenarios?
 - NO => WireMock
- => Testcontainers

* Your mileage may vary

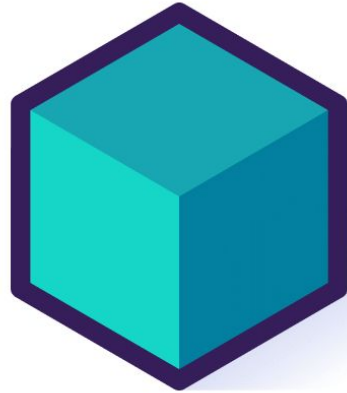
** More later

*** I just like footnotes





Disclaimer




AND



Better Together!

WireMock offers an official Testcontainers module!












WireMock

Cloud, Web

Java Go Python

Documentation

-  Java  
-  Go  
-  Python  

Official Modules

AtomicJar partners with software vendors to maintain and certify official modules.

testcontainers.com/modules/wiremock

Description

This module allows provisioning the WireMock server as a standalone container within your tests, based on [WireMock Docker](#).

WireMock is a popular open-source tool for API mock testing with over 5 million downloads per month. It can help you to create stable test and development environments, isolate yourself from flakey 3rd parties and simulate APIs that don't exist yet. WireMock has a rich matching system, allowing any part of an incoming request to be matched against complex and precise criteria.

Read more:

- [WireMock Documentation](#)
- [WireMock and Testcontainers](#)

Example - JUnit 5

```
@Testcontainers
class WireMockContainerJUnit5Test {

    @Container
    WireMockContainer wiremockServer = new WireMockContainer("2.35.0")
        .withMapping("hello", WireMockContainerJUnit5Test.class, "hello-world.json");

    @Test
    void helloWorld() throws Exception {
        // given
        String url = wiremockServer.getUrl("/hello");

        // when
        HttpResponse response = new TestHttpClient().get(url);

        // then
        assertThat(response.getBody())
            .as("Wrong response body")
            .contains("Hello, world!");
    }
}
```


Get started

- wiremock.org/docs/solutions/testcontainers
- For Java:
 - <https://github.com/wiremock/wiremock-testcontainers-java>

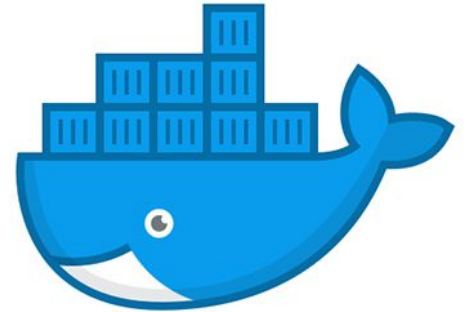
WireMock and Testcontainers - Coming soon

- Maven Central releases (now GitHub Packages / Jitpack)
- 1.0 releases for Java, Golang and Python
- SDKs for WireMock gRPC and GraphQL extensions
- Official Quarkus extension and images based on it

Demo time! Integration Testing of WireMock Docker images

Demo 1

- Sources location:
github.com/wiremock/wiremock-docker/tree/main/test/integration-tests
- Scope:
 - JUnit 5 and JUnit 4 examples
 - Multi-version testing



Demo 2 - Quarkus project

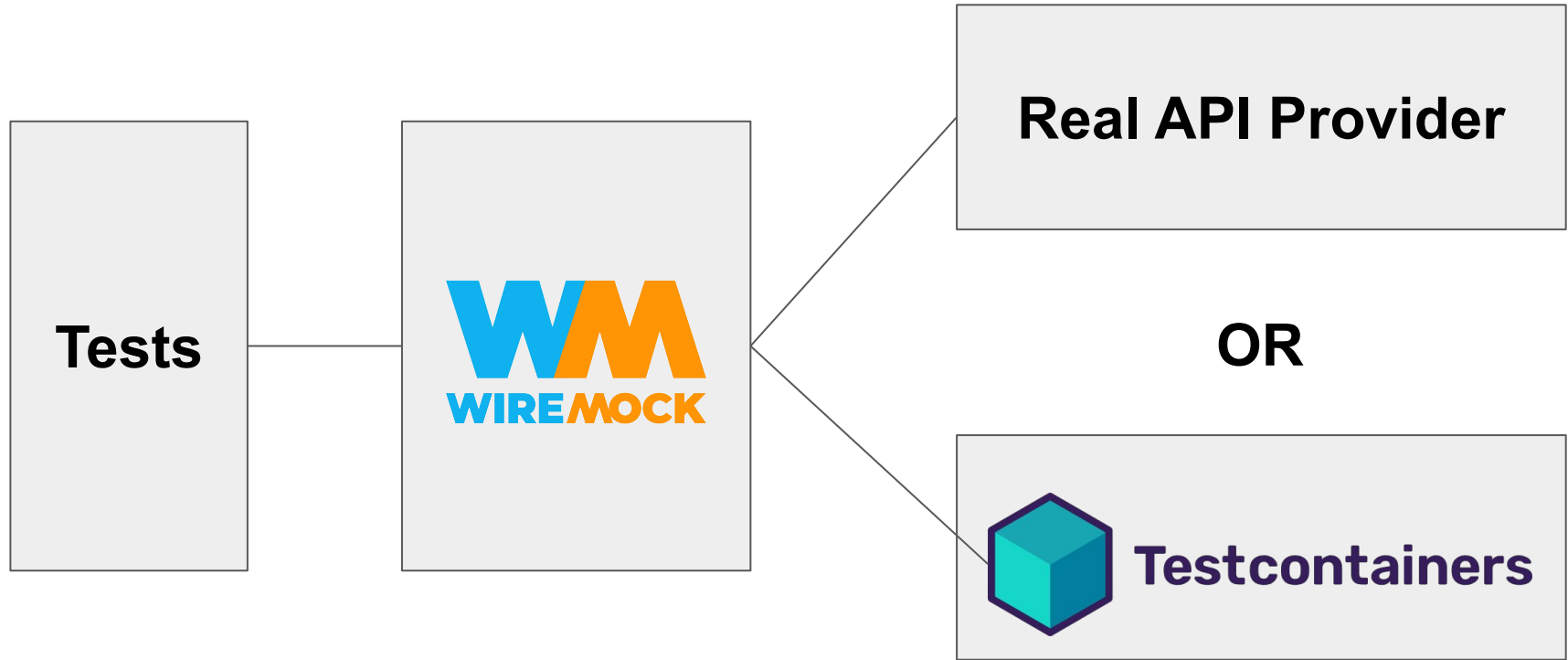
- <https://github.com/quarkusio/quarkus-super-heroes/blob/6d5eb145049d81aa3d533370ef59621230125d8a/rest-fights/src/test/java/io/quarkus/sample/superheroes/fight/client/HeroClientTests.java>



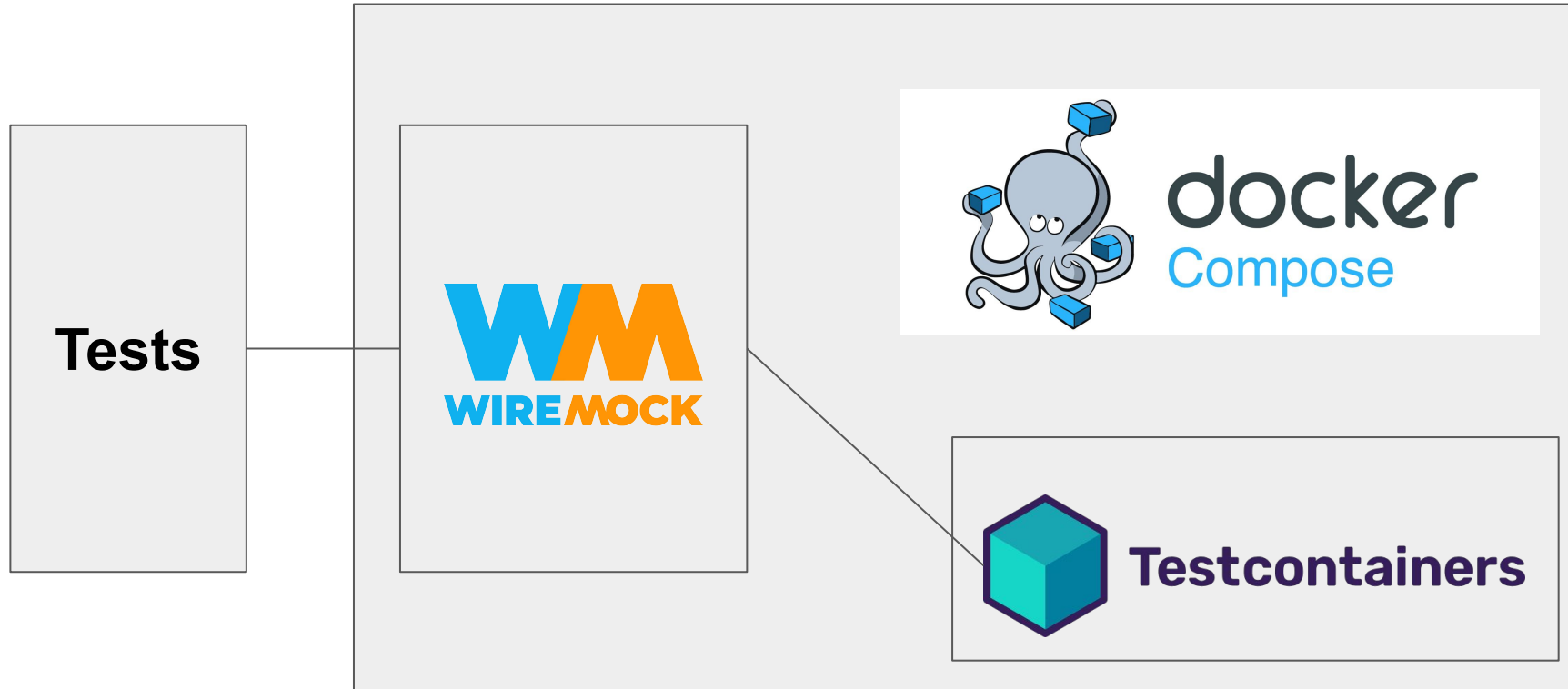
QUARKUS

Chaining WireMock and Testcontainers

WireMock as a Proxy



Proxy can be made transparent in your test!



WireMock Proxy? Why?

- Record & Playback for Request sequences
- Reverse engineering of legacy undocumented APIs
- **NEW**: OpenAPI Standard Compliance Verification (in 3.0 Beta)
- Emulating complex cases
 - Overriding particular responses
 - Emulating response sequences / Stateful behavior
 - Failure Injection

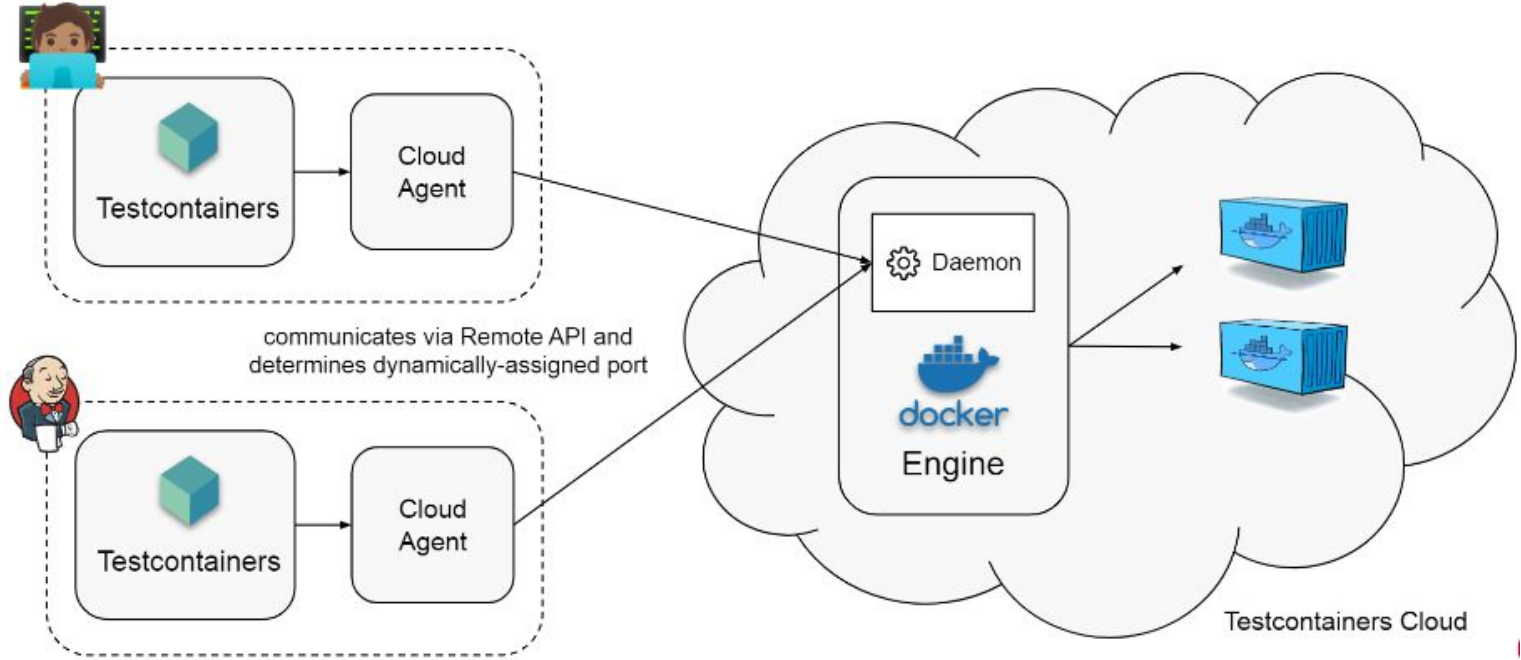


Running API mocks
As a service?

Testcontainers Cloud*

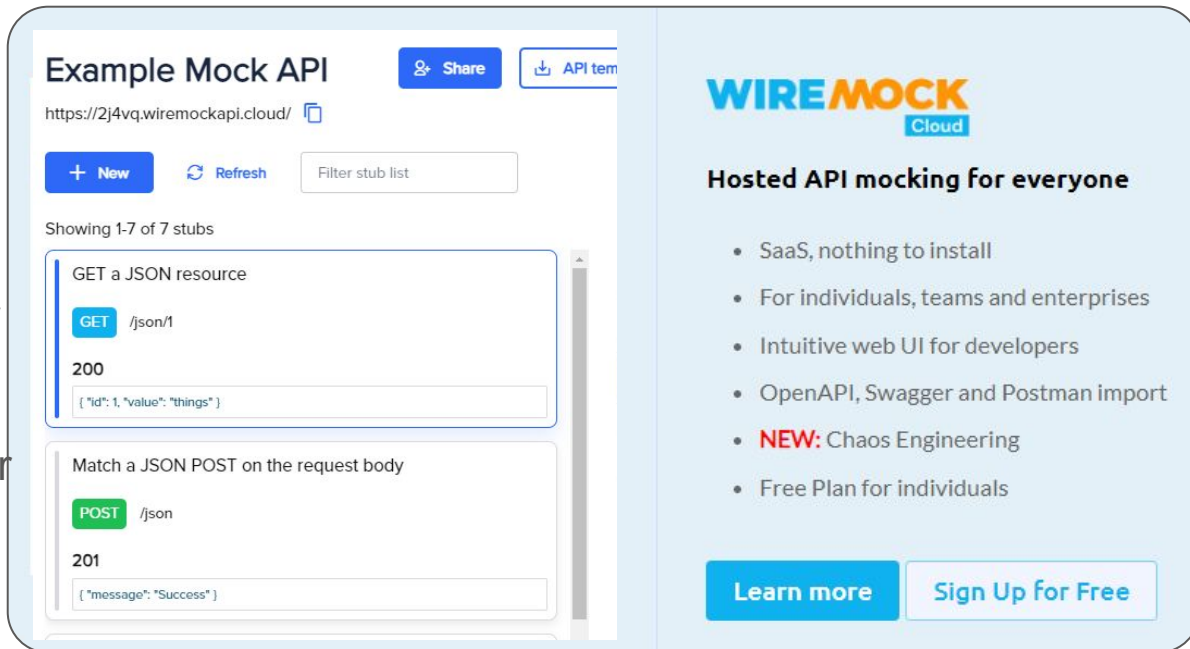
Credits:

<https://github.com/bmuschko/testcontainers-integration-testing>



WireMock Cloud by WireMock Inc.

- Small startup
- WireMock creator is a co-founder of the Inc.
- WireMock Cloud - SaaS for end-to-end API mocking
- Private beta: K8s Edition for managed / on-premises



The screenshot displays the WireMock Cloud interface for an 'Example Mock API'. The URL is `https://2j4vq.wiremockapi.cloud/`. The interface includes a '+ New' button, a 'Refresh' button, and a 'Filter stub list' input field. It shows two stubs:

- GET a JSON resource**: A GET request to `/json/1` returns a 200 status with a JSON response: `{ "id": 1, "value": "things" }`.
- Match a JSON POST on the request body**: A POST request to `/json` returns a 201 status with a JSON response: `{ "message": "Success" }`.

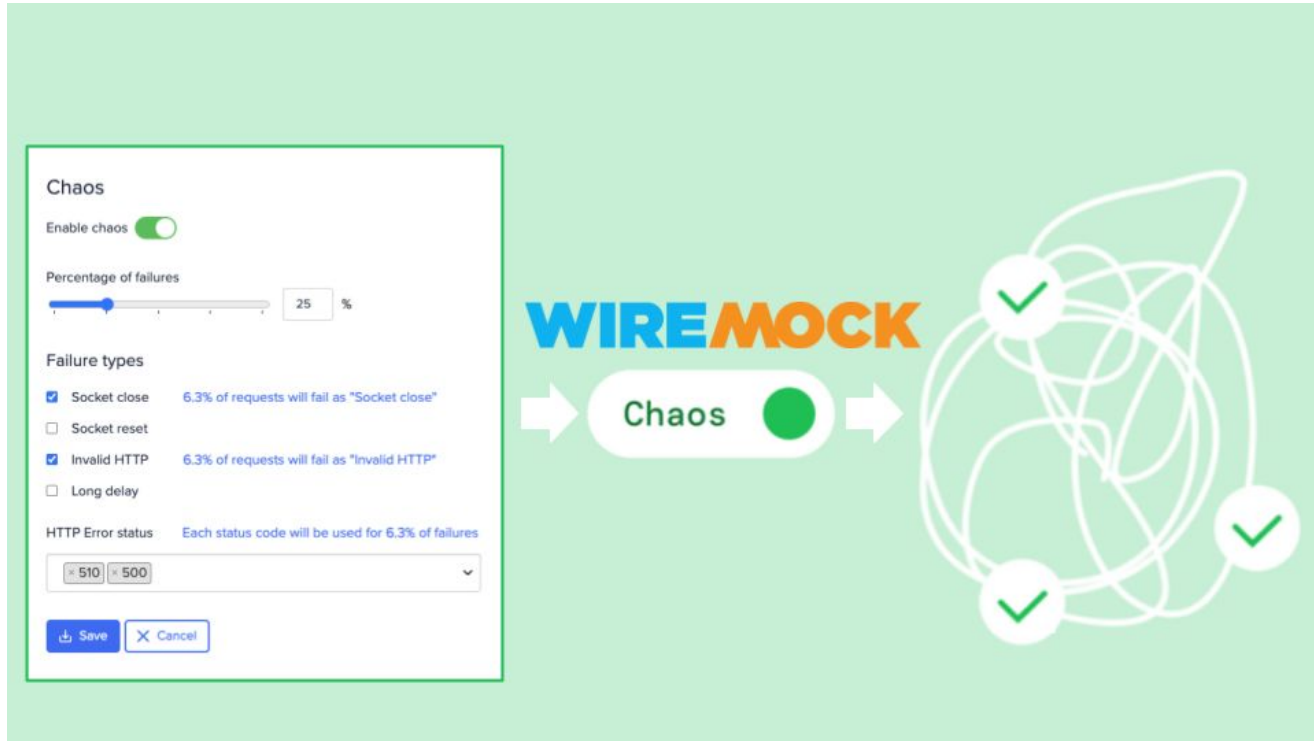
On the right side of the interface, the WireMock Cloud logo is displayed above the text 'Hosted API mocking for everyone'. Below this, a list of features is provided:

- SaaS, nothing to install
- For individuals, teams and enterprises
- Intuitive web UI for developers
- OpenAPI, Swagger and Postman import
- **NEW:** Chaos Engineering
- Free Plan for individuals

At the bottom right, there are two buttons: 'Learn more' and 'Sign Up for Free'.

wiremock.io

Add more Chaos... Engineering!



The image illustrates the WireMock Chaos configuration interface and its application. On the left, a configuration panel titled "Chaos" is shown with the following settings:

- Enable chaos:** A toggle switch is turned on.
- Percentage of failures:** A slider is set to 25%.
- Failure types:**
 - Socket close: 6.3% of requests will fail as "Socket close"
 - Socket reset
 - Invalid HTTP: 6.3% of requests will fail as "Invalid HTTP"
 - Long delay
- HTTP Error status:** A dropdown menu is set to "x 510 x 500", with a note: "Each status code will be used for 6.3% of failures".

At the bottom of the panel are "Save" and "Cancel" buttons. To the right of the panel, the "WIREMOCK" logo is displayed in blue and orange. Below it, a white pill-shaped button labeled "Chaos" with a green circle indicator is shown with arrows pointing to a complex, tangled white network diagram. Three green checkmarks are placed at different points within the network diagram, indicating successful or configured nodes.

The End

Takeaways

- Shift left the integration testing
- API Mocking/Abstraction are great
- There is no silver bullet
- WireMock and Testcontainers address their own use-cases
- They work well together!



It's [always] a great time to contribute!

wiremock.org/participate

testcontainers.org/contributing

Credits to

- All WireMock contributors
- WireMock Inc. Team
- All Testcontainers contributors and AtomicJar folks
- All FOSS contributors



QUESTIONS?



WIREFMCK

Contacts:

✉ E-mail: oleg@wiremock.org

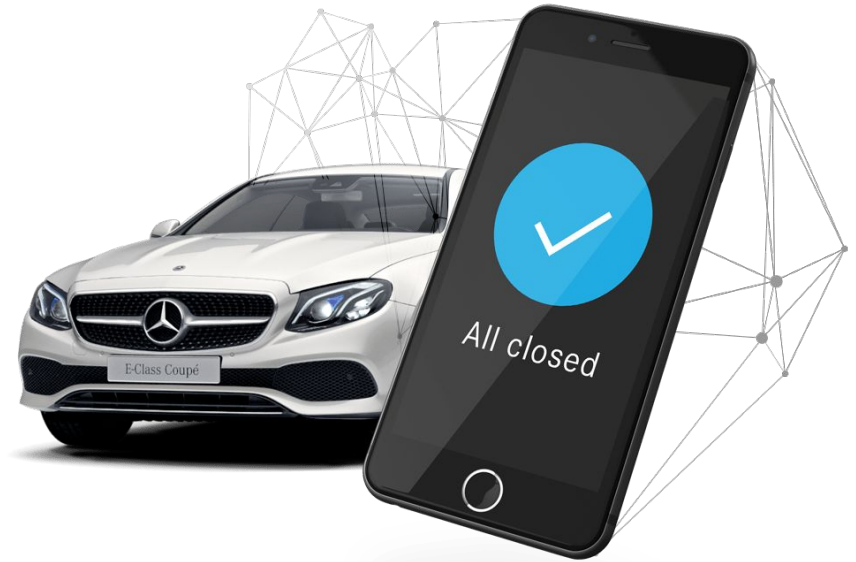
🐙 GitHub [oleg-nenashev](https://github.com/oleg-nenashev)

🐦 Twitter: [@oleg_nenashev](https://twitter.com/oleg_nenashev)

THANK YOU

Software **ATE** the world

- Modern hardware projects are 80% software ones now
- R&D budget for software often exceeds the “core product” parts
- Everything inside: from Android infotainment to K8s clusters



Source: developer.mercedes-benz.com/products/vehicle_status/details