

# Integration Tests with Containers

**Philipp Krenn**

**@xeraa**



**Developer** 🥑



Yours is without a doubt the worst code I've ever run



But it runs

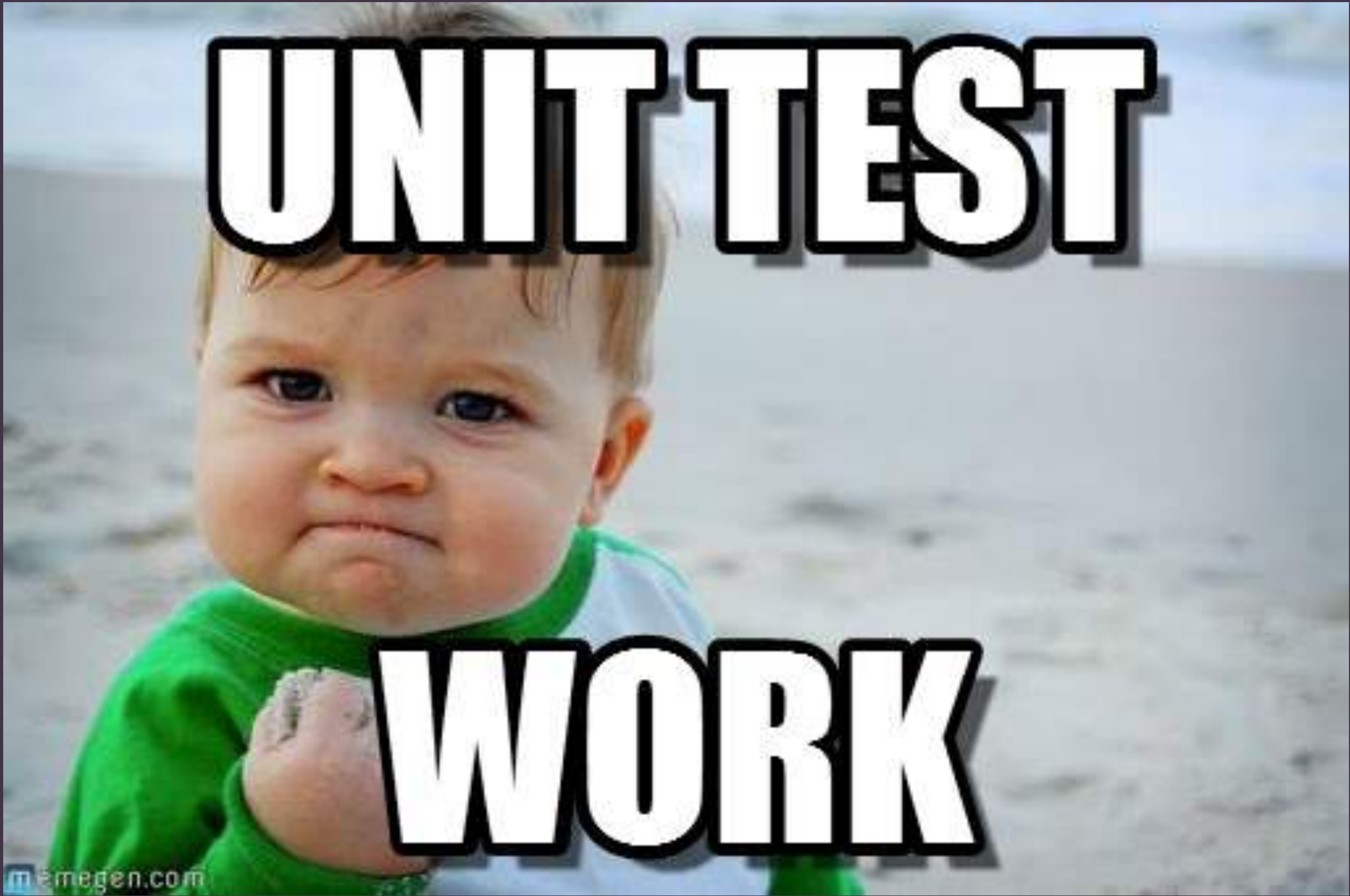
**UNIT TEST?**



**IT COMPILED: PASS**

# Tests



A close-up photograph of a baby with light brown hair and blue eyes, wearing a green long-sleeved shirt. The baby has a grumpy or determined expression and is holding a small clump of sand in their right hand. The background is a blurred outdoor setting, likely a beach with sand and some greenery.

**UNIT TEST**

**WORK**

Please do not turn off  
the light switch.

It also operates the  
elevator.

Thank You

**But the unit tests  
passed, so...**

**[https://twitter.com/Aaronius/status/  
933497253347463168](https://twitter.com/Aaronius/status/933497253347463168)**





















# Integration Tests



# Mocks

**Mockito, EasyMock,  
JMock,...**

**[https://twitter.com/Nick\\_Craver/status/1159795478554402816](https://twitter.com/Nick_Craver/status/1159795478554402816)**

**"But Nick, you can mock those things"**

**Sure. You can mock most things. But what are you testing then? The mock.**



**We need to see if our Redis client library works *against Redis*. That sockets work. That the data types work. When server breaks happen, we need to see and fix them.**

**[https://twitter.com/Nick\\_Craver/status/1159796131976679424](https://twitter.com/Nick_Craver/status/1159796131976679424)**

**Let's say we are testing against Server of X, version 1. They break an API in version 2.**

**A mock isn't going to show you that. It'll just keep testing and working.**

**So we need to spin up *actual* servers/  
services to test against, so we know the  
whole thing works. It's critical.**



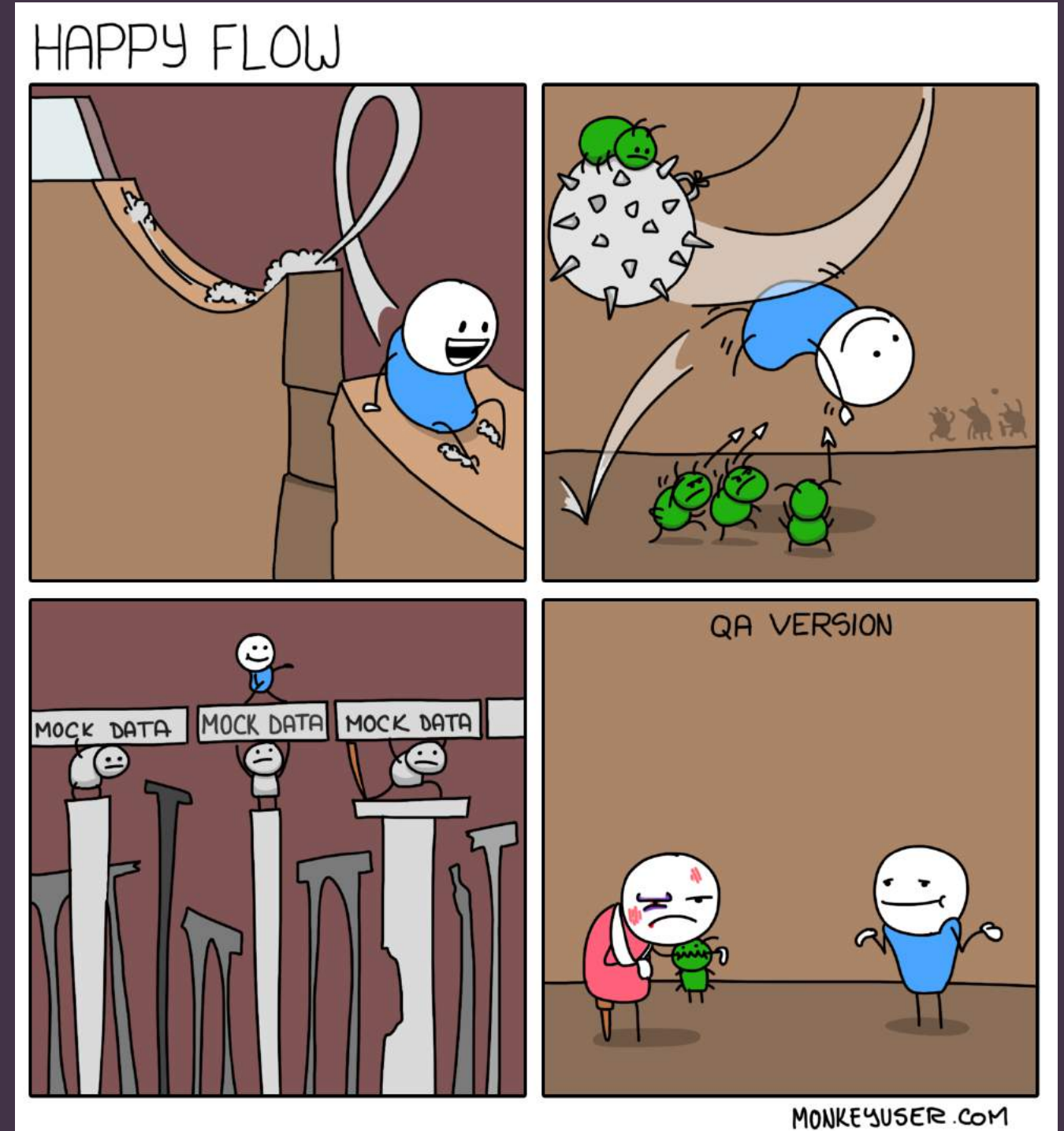




A close-up photograph of a woman with vibrant red hair styled in two pigtails. She is holding a small, light-colored dog with a yellow collar. The image has a soft, slightly blurred background. Overlaid on the image is a large, bold, white text quote.

**Ich mock mir die Welt  
widdewidde wie sie mir  
gefällt**

<https://www.monkeyuser.com/2018/happy-flow/>



 **Unit tests, systems not  
under your control**

 **Test real datastore**



# In-Memory

# H2, HSQLDB, Apache Derby,...



**same**



# Embedded Elasticsearch unsupported in 5.0+

<https://www.elastic.co/blog/elasticsearch-the-server>



**Often good enough**



**Test real datastore,  
supported systems**

# Actual Datastore



# Local installation

# Docker container

# Cloud

# Demo



**"Good old approach"**



**External dependency,  
parallelization**

# Embedded



# embedded-elasticsearch

<https://github.com/allegro/embedded-elasticsearch>

# Demo

# Customization & Mappings

```
.withPlugin("analysis-stemmer")  
.withIndex("cars", IndexSettings.builder()  
    .withType("_doc", getSystemResourceAsStream("car-mapping.json"))
```

 **IDE support,  
customization, custom  
lifecycle**

 **Custom integration**



# More embedded datastores

[https://github.com/flapdoodle-oss/  
de.flapdoodle.embed.process](https://github.com/flapdoodle-oss/de.flapdoodle.embed.process)

# Build Tool

# docker-maven-plugin

<http://dmp.fabric8.io>

# Dockerfile or Docker assembly

```
<build>  
  <from>java:8</from>  
  <assembly>  
    <descriptor>docker-assembly.xml</descriptor>  
  </assembly>  
  <cmd>  
    <shell>java -jar /maven/service.jar</shell>  
  </cmd>  
</build>
```



# **Build a custom image:**

`docker:build`

# **Run container:**

`docker:start`

`docker:stop`

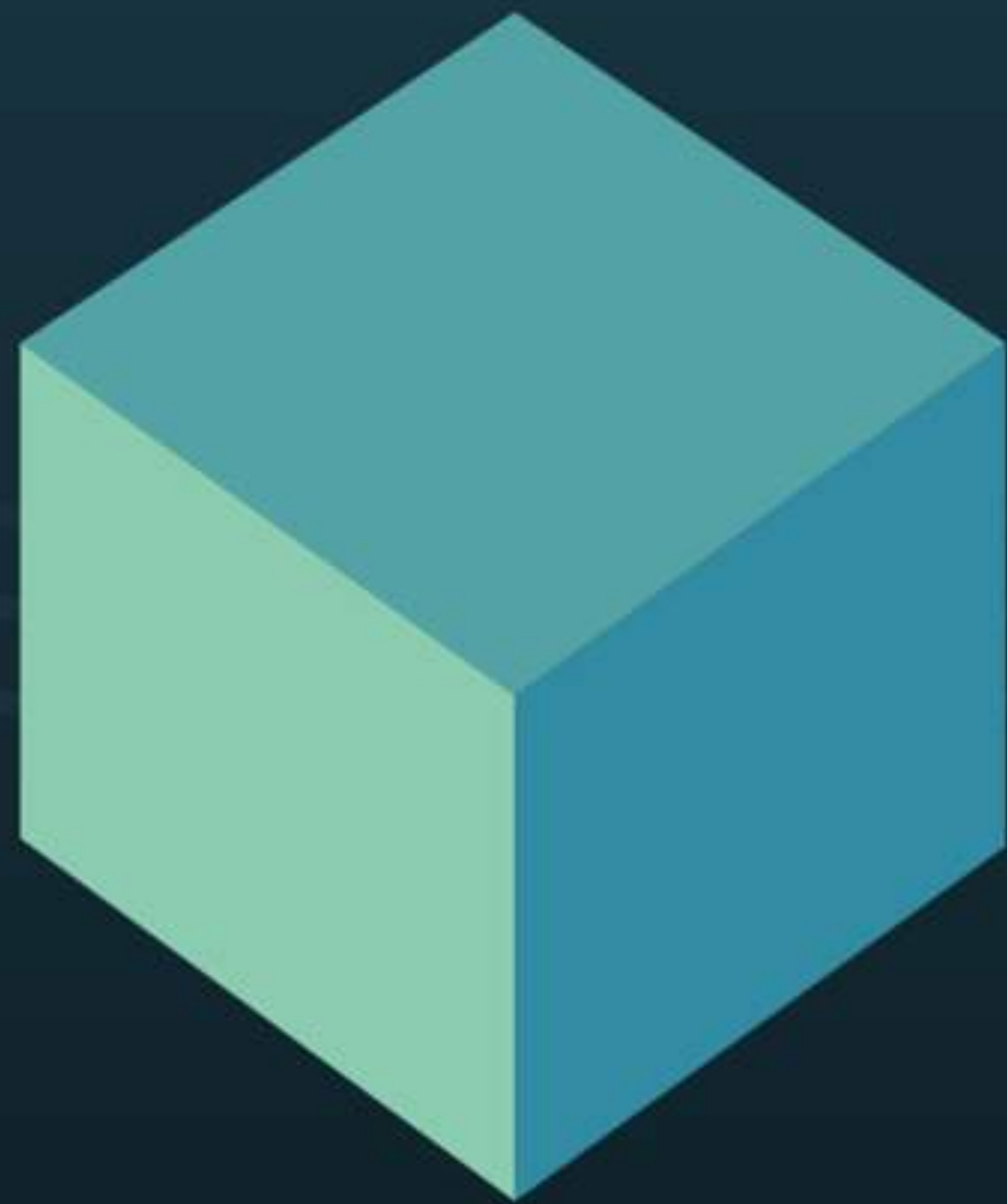
# Demo

# maven-failsafe-plugin

<https://maven.apache.org/surefire/maven-failsafe-plugin/>

 **Standard or custom  
Docker image**

 **One instance for all  
tests, no IDE support**



TESTCONTAINERS



# Testcontainers Generic

# Dependency

```
@ClassRule
public static GenericContainer redis =
    new GenericContainer("redis:5.0.5")
        .withExposedPorts(6379);
```

# Docker Compose integration

```
@ClassRule
public static DockerComposeContainer environment =
    new DockerComposeContainer(new File("src/test/resources/compose.yml"))
        .withExposedService("elasticsearch_1", ELASTICSEARCH_PORT,
            Wait.forHttp("/").forStatusCode(200));
```

# Demo

# What's in docker ps -a



# **Docker for Mac API: socat**

## **Linux: Multipurpose relay (SOcket CAT)**

**<https://github.com/alpine-docker/socat>**

# Cleanup: Moby Ryuk

<https://github.com/testcontainers/moby-ryuk>

**[Ryuk] drops a Death Note, a notebook that allows the user to kill anyone simply by knowing their name and face**

[https://en.wikipedia.org/wiki/Ryuk\\_\(Death\\_Note\)](https://en.wikipedia.org/wiki/Ryuk_(Death_Note))



 **IDE support,  
customization, custom  
lifecycle**

 **Hacky configuration**

# Testcontainers Custom

g:org.testcontainers



| Group ID                           | Artifact ID                   | Latest Version |      | Updated     | Download                                                                              |
|------------------------------------|-------------------------------|----------------|------|-------------|---------------------------------------------------------------------------------------|
| <a href="#">org.testcontainers</a> | <a href="#">elasticsearch</a> | 1.10.6         | (7)  | 29-Jan-2019 |    |
| <a href="#">org.testcontainers</a> | <a href="#">influxdb</a>      | 1.10.6         | (14) | 29-Jan-2019 |    |
| <a href="#">org.testcontainers</a> | <a href="#">pulsar</a>        | 1.10.6         | (13) | 29-Jan-2019 |    |
| <a href="#">org.testcontainers</a> | <a href="#">jdbc</a>          | 1.10.6         | (48) | 29-Jan-2019 |    |
| <a href="#">org.testcontainers</a> | <a href="#">docs-examples</a> | 1.10.6         | (1)  | 29-Jan-2019 |    |
| <a href="#">org.testcontainers</a> | <a href="#">jdbc-test</a>     | 1.10.6         | (25) | 29-Jan-2019 |   |
| <a href="#">org.testcontainers</a> | <a href="#">junit-jupiter</a> | 1.10.6         | (7)  | 29-Jan-2019 |  |
| <a href="#">org.testcontainers</a> | <a href="#">kafka</a>         | 1.10.6         | (18) | 29-Jan-2019 |  |
| <a href="#">org.testcontainers</a> | <a href="#">localstack</a>    | 1.10.6         | (20) | 29-Jan-2019 |  |
| <a href="#">org.testcontainers</a> | <a href="#">mariadb</a>       | 1.10.6         | (27) | 29-Jan-2019 |  |
| <a href="#">org.testcontainers</a> | <a href="#">mockserver</a>    | 1.10.6         | (11) | 29-Jan-2019 |  |

Items per page: 20    1 - 20 of 30





# ElasticsearchContainer

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

# Demo

# More languages

**Scala, Go, .net\*, Python\*, JavaScript\*, Rust\***

\* Early development



# **Selenium 2 / Webdriver**

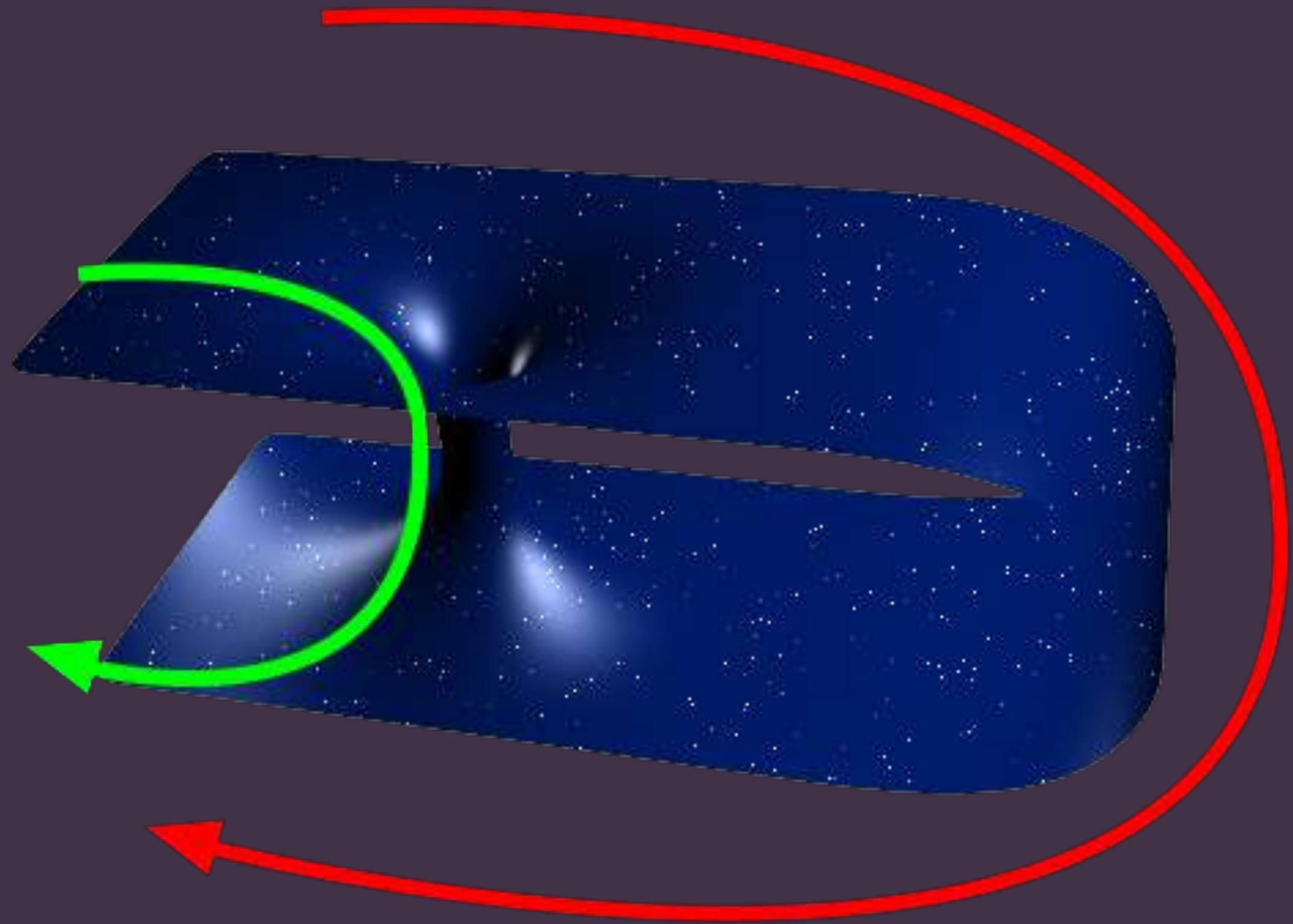
**VNC screen recording**

 **IDE support,  
customization, custom  
lifecycle**

 **Custom integration**

# Inside Containers





# Wormhole Pattern

```
$ docker run -it
--rm
-v $PWD:$PWD
-w $PWD
-v /var/run/docker.sock:/var/run/docker.sock
maven:3 mvn --projects parent,4_testcontainers-custom test
```

```
$ docker ps -a
```

# Conclusion

# Why Integration Tests

**Why not**  
**Mocking**  
**In-Memory**  
**Actual Datastore**



# How Embedded Build Tool Testcontainers

# Code

**<https://github.com/xeraa/integration-test-demo>**

# Questions?

**Philipp Krenn**

**@xeraa**