



AUTOMATION

*with*



QUARKUS

YOANN RODIÈRE

HIBERNATE

QUARKUS

[yoann@hibernate.org](mailto:yoann@hibernate.org)

GUILLAUME SMET

QUARKUS

[guillaume@hibernate.org](mailto:guillaume@hibernate.org)

***You already figured it out: we are French!***

*"Death by a thousand paper cuts"*

## *Triaging*


- Constant flow of issues/pull requests
- Many components with different maintainers
- Need to ping appropriate people

## *Affecting milestones*

- Minor release every month (micro every week)
- ~ 200 issues / pull requests without milestone when releasing
- Heterogeneous team makes it harder

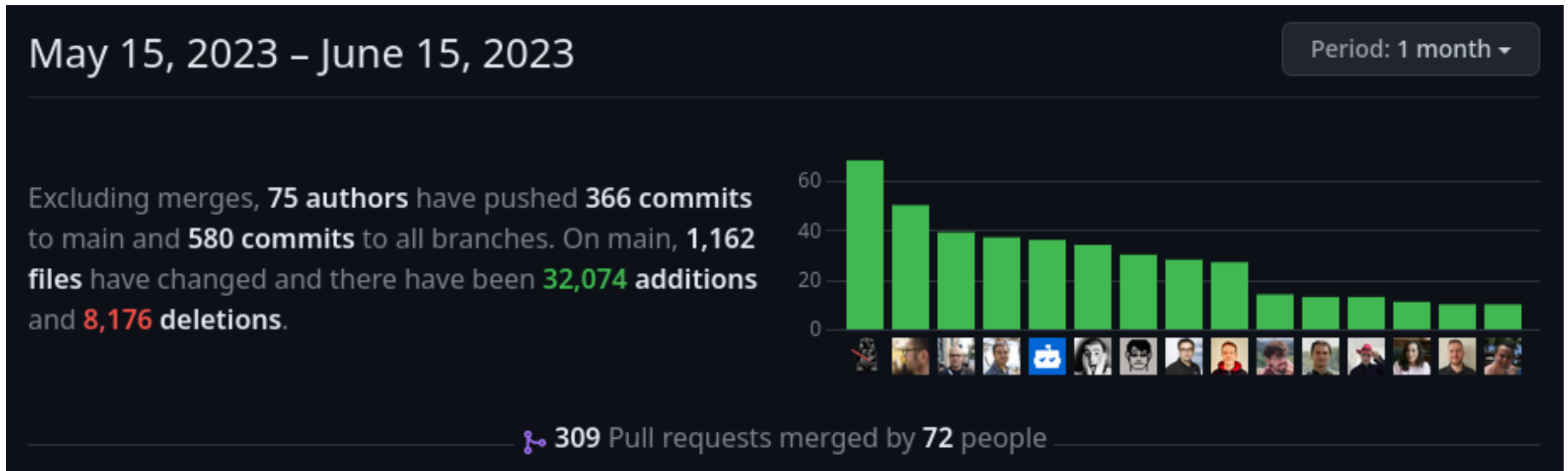
# Hard to find cause when CI fails

```
27717     at io.netty.util.concurrent.SingleThreadEventExecutor$4.run(SingleThreadEventExecutor.java:986)
27718     at io.netty.util.internal.ThreadExecutorMap$2.run(ThreadExecutorMap.java:74)
27719     at io.netty.util.concurrent.FastThreadLocalRunnable.run(FastThreadLocalRunnable.java:30)
27720     at java.base/java.lang.Thread.run(Thread.java:833)
27721
27722 2022-05-18 09:48:15,165 ERROR [io.qua.ver.htt.run.QuarkusErrorHandler] (vert.x-ev
```

This step has been truncated due to its large size. Download the full logs from the  menu once the workflow run has completed.



## *In a word: scale*



Quarkus 3.0 = 1000+ pull requests and issues

# *Quarkus GitHub Bot*



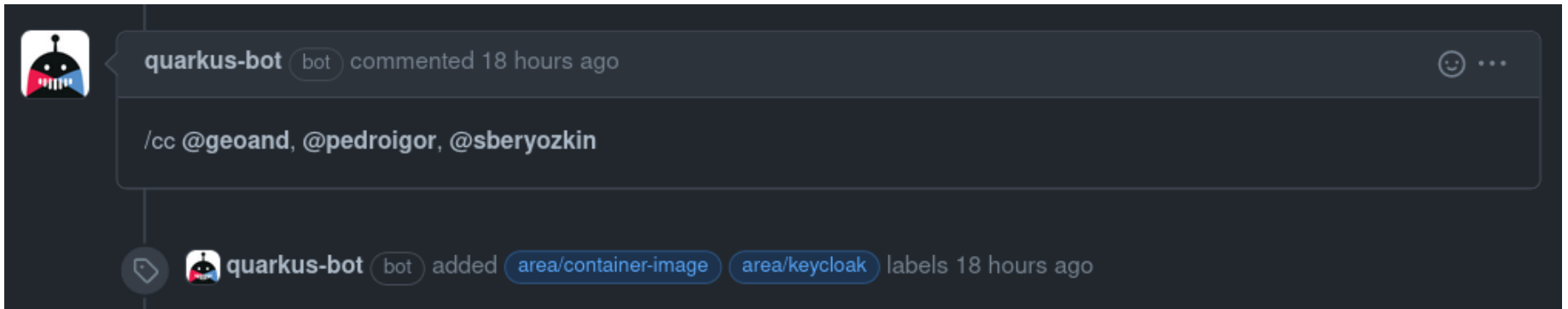
Probot




QUARKUS





# *Automated triaging*



The screenshot shows a GitHub interface with a dark theme. At the top, a comment from the bot 'quarkus-bot' is displayed, mentioning three users: @geoand, @pedroigor, and @sberyozkin. Below the comment, the bot has added two labels to the issue: 'area/container-image' and 'area/keycloak'. The bot's profile picture is a stylized robot head.

 **quarkus-bot** bot commented 18 hours ago 😊 ⋮

/cc @geoand, @pedroigor, @sberyozkin

  **quarkus-bot** bot added [area/container-image](#) [area/keycloak](#) labels 18 hours ago

# *Automated workflow*



quarkus-bot

bot

added this to the **2.10 - main** milestone 15 hours ago

# Automated build report



quarkus-bot (bot) commented 2 days ago



## Failing Jobs - Building [66cf0e2](#)

Status	Name	Step	Failures	Logs	Raw logs
✗	Devtools Tests - JDK 11	<a href="#">Build</a>	<a href="#">Failures</a>	<a href="#">Logs</a>	<a href="#">Raw logs</a>
✓	JVM Tests - JDK 11				
✗	JVM Tests - JDK 11 Windows	<a href="#">Build</a>	<a href="#">Failures</a>	<a href="#">Logs</a>	<a href="#">Raw logs</a>
✓	JVM Tests - JDK 17				

Full information is available in the [Build summary](#) check run.

## Failures

### ⚙️ Devtools Tests - JDK 11 #

- Failing: [integration-tests/devtools](#)

### 📦 integration-tests/devtools

✗ `io.quarkus.platform.catalog.RegistrySnapshotCatalogCompatibilityTest.testRegistrySnapshotPlatformCatalog` line 30 - [More details](#) - [Source on GitHub](#)

▶ Details

# Annotations

extensions/vertx/deployment/src/test/java/io/quarkus/vertx/mdc/VertxMDCTest.java

```
118     MDC.put("requestId", mdcValue);
119     LOGGER.info("Test 1");
120
121     assertThat(inMemoryLogHandler.logRecords(),
```

✗ Check failure on line 121 in extensions/vertx/deployment/src/test/java/io/quarkus/vertx/mdc/VertxMDCTest.java



**quarkus-bot** / Build summary for d11d7305a36ab6e122f5998219be950b78bb3dc2

## JVM Tests - JDK 11 Windows

java.lang.AssertionError:

Expected: a collection containing "Test MDC value ### Test 1"

[Raw output](#)

java.lang.AssertionError:

Expected: a collection containing "Test MDC value ### Test 1"

but: was empty

```
at org.hamcrest.MatcherAssert.assertThat(MatcherAssert.java:18)
at org.hamcrest.MatcherAssert.assertThat(MatcherAssert.java:6)
at io.quarkus.vertx.mdc.VertxMDCTest.mdcNonVertxThreadTest(VertxMDCTest.java:121)
at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at java.base/jdk.internal.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at java.base/jdk.internal.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.base/java.lang.reflect.Method.invoke(Method.java:566)
```

*This bot is Open Source*

*Want your own? Easy-peasy!*

## *Quarkus GitHub App*

- Quarkus extension
  - Develop GitHub Apps in Java
  - Zero boilerplate
- Stable - in production for 3 years
- Used by Quarkus, Hibernate, Keycloak, WildFly...
- Supports GitHub Enterprise too
- Be creative, create your own bot!



**Richard Startin**

@richardstartin



Would it be disingenuous to automatically close bugs related to OSGi with “could not reproduce?”

1:15 PM · May 31, 2022





*Demo time!*

***Fully documented!***

<https://docs.quarkiverse.io/quarkus-github-app/dev/>

## *In short*

- Quarkus!
  - Developer joy: live reload...
  - CDI, persistence, Kafka...
  - Native compilation & Containers
- GitHub!
  - Extensive Hub4j GitHub API
  - GraphQL client
- More!
  - Event annotations
  - Configuration in the repository
  - Replay UI
  - Mockito-based test framework

# Comment-based commands



The screenshot shows a GitHub issue with three comments. The first comment, from user 'gsmet', is highlighted with a blue border and contains the command '@all-contributors add @yrodier' for code. The second comment, from the bot 'allcontributors', mentions the issue and shows the command 'docs: add yrodier as a contributor #50' with a 'Merged' status. The third comment, from the bot 'allcontributors', responds to 'gsmet' by stating 'I've put up a pull request to add @yrodier!'.


 **gsmet** commented on Dec 1, 2020 Member Author ...

@all-contributors add **@yrodier** for code



 **allcontributors** bot mentioned this issue on Dec 1, 2020

**docs: add yrodier as a contributor #50** Merged

 **allcontributors** bot commented on Dec 1, 2020 Contributor ...

@gsmet

I've put up a [pull request](#) to add **@yrodier!** 🎉



Demo time\*!

*\* Yes! Another one!*

## *In short*

- Airline library: no manual parsing!
- Permissions with `@Permission(ADMIN)` or `@Team("your-team")`
- Scope: issues, pull requests or both
- Help generation
- And many more options

## *Quarkus GitHub Action*

- Our newest addition
- Same but for GitHub Actions
- Tons of features to make the development of GitHub Actions in Java easier
- Fully documented (again!): <https://docs.quarkiverse.io/quarkus-github-action/dev/> !

## *Create an action*

```
quarkus create app \  
  -x io.quarkiverse.githubaction:quarkus-github-action \  
  org.acme:my-github-action:999-SNAPSHOT \  
  --name="My Action" \  
  --description="Description of my action" \  
  --data=github-action-codestart.github-repository=org/my-github-action
```





## *Publish your action*

- Push the repository to GitHub
- Automagically published
- That's all, folks!

## *Use your action*

```
- name: Run action
  uses: org/my-github-action@main
  with:
    who: JCon Slovenia
    github-token: ${{ secrets.GITHUB_TOKEN }}
```

## *Automation vs paper cuts*

- Small paper cuts add up
- Automation streamlines your work
- Quarkus GitHub App and Quarkus GitHub Action can help, quick and easy
- Start small, put in production, and grow from there

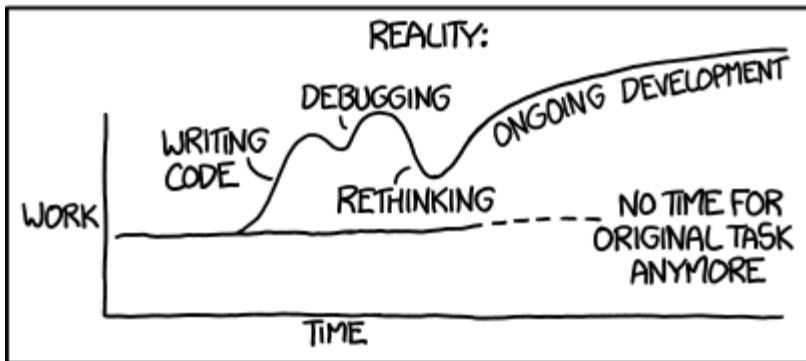
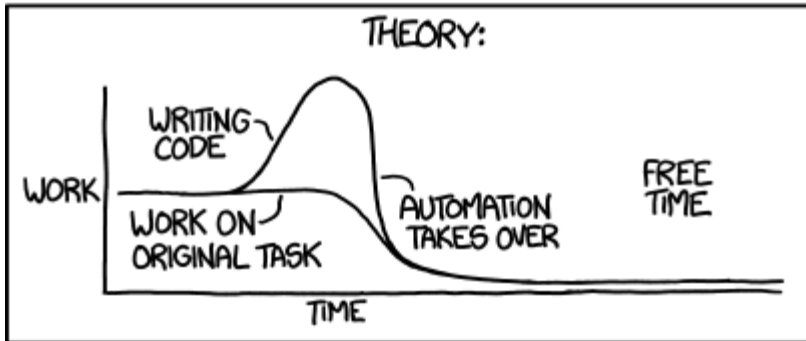
## *Extensions?*

- These extensions are small
- Everything is wired during the build with Gizmo
- Custom extensions are powerful and bring developer joy

***It's easy, just try it!***



*Thanks for listening!*

"I SPEND A LOT OF TIME ON THIS TASK.  
I SHOULD WRITE A PROGRAM AUTOMATING IT!"



<https://xkcd.com/1319/>

## *Resources*

- Quarkus GitHub App - <https://github.com/quarkiverse/quarkus-github-app>
  -  <https://quarkiverse.github.io/quarkiverse-docs/quarkus-github-app/dev/>
- Quarkus GitHub Action - <https://github.com/quarkiverse/quarkus-github-action>
  -  <https://quarkiverse.github.io/quarkiverse-docs/quarkus-github-action/dev/>
- Quarkus GitHub Bot - <https://github.com/quarkusio/quarkus-github-bot>



# *Architecture*



Webhook request with payload

**Reactive Route**

- Check signature of the payload

GitHubEvent via CDI event

**Dispatcher**

- Handle all the heavy lifting on connecting to GitHub REST API
- Parse the payload into a GHEventPayload subclass

GHEventPayload subclass via CDI event

**Multiplexer (subclass of MyGitHubApp)**

@Issue.Opened method      @Issue.Edited method

- One method per event annotation (they are CDI qualifiers)
- Methods annotated with @Observes to listen to CDI events
- Resolves config files

Call the superclass method directly

**Event annotations**  
@Issue.Opened  
@Issue.Edited  
@PullRequest.Opened  
...

**Build**

```
class MyGitHubApp {  
    void onOpen(@Issue.Opened @Issue.Edited GHEventPayload.Issue issuePayload) throws IOException {  
        issuePayload.getIssue().comment("Hello from MyGitHubApp");  
    }  
}
```

# *License*

Logos in this presentation are trademarks or registered trademarks of their respective owners in the United States and other countries.

The copyright of the "Automation" comic on the "Thanks for listening!" slide is owned by Randall Munroe (author of [xkcd](#)), who licenses the comic under the [Creative Commons Attribution-NonCommercial 2.5 License](#).

The rest of this presentation is distributed under CC-BY-4.0. See <https://creativecommons.org/licenses/by/4.0/> for details.

The source code for this presentation is available at <https://github.com/gsmet/talks/>.