

Agile Nacht, 08.07.2024

Raus aus der Wartungshölle ... zumindest ein bißchen

Sandra Parsick

@SandraParsick

@sparsick@mastodon.social

mail@sandra-parsick.de

Wer bin ich?

- Sandra Parsick
- Freiberuflicher Softwareentwickler und Consultant im Java-Umfeld
- Schwerpunkte:
 - Java Enterprise Anwendungen
 - Agile Methoden
 - Software Craftmanship
 - Automatisierung von Entwicklungsprozessen
- Trainings
- Workshops

✉️ mail@sandra-parsick.de

🐦 @SandraParsick

Ⓜ️ @sparsick@mastodon.social

RSS https://www.sandra-parsick.de

🎧 https://ready-for-review.dev



♣ ACE Pro



Testcontainers



Wieso haben wir
das so gemacht?

Welche
Abhängigkeiten
haben wir im
Einsatz?

Wie können wir
am besten up to
date bleiben?

Entspricht die
Implementierung
noch den
Architekturbildern?

Welche
Softwarelizenzen
haben wir im
Einsatz?

Sind wir von
Sicherheitslücken
betroffen?

Onboarding neuer
Devs

Wie bekommen
wir die
Anwendung lokal
zum Laufen?

Willkommen in der Wartungshölle



Nicht-Techies so













Laut „The Cost of poor Software Quality in the US: A 2022 Report“

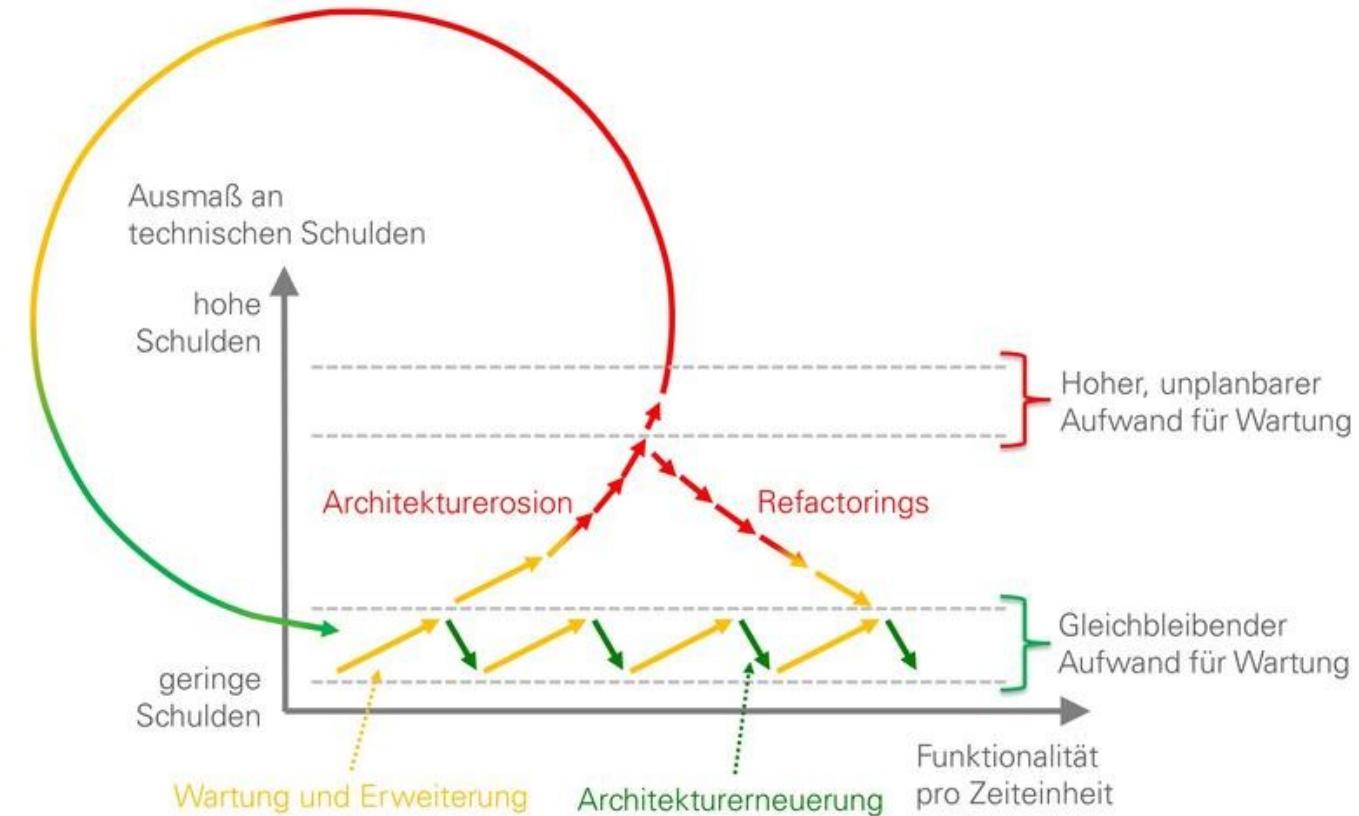
- 23% ihrer Zeit verschwenden Entwickler, aufgrund von technischen Schulden!
- 33% ihrer Zeit nutzen Entwickler, um technische Schulden zu verringern!
- 40% des IT-Budgets fallen bis 2025 auf technische Schulden ab!







Effekte von technischen Schulden



Von Carola Lilienthal



Automation is Queen :)

Wieso haben wir
das so gemacht?

Welche
Abhängigkeiten
haben wir im
Einsatz?

Wie können wir
am besten up to
date bleiben?

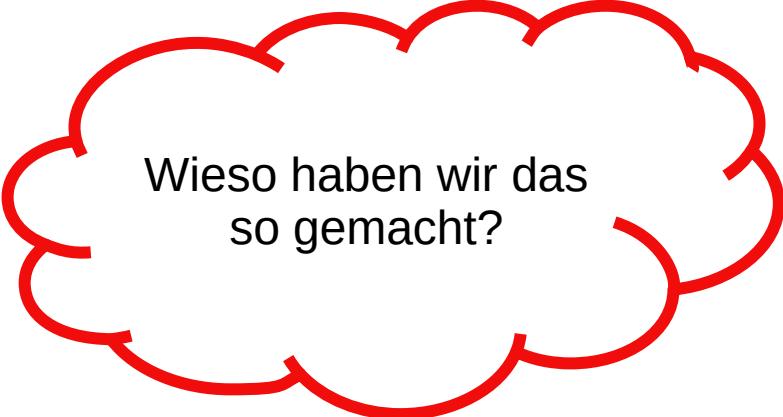
Entspricht die
Implementierung
noch den
Architekturbildern?

Welche
Softwarelizenzen
haben wir im
Einsatz?

Sind wir von
Sicherheitslücken
betroffen?

Onboarding neuer
Devs

Wie bekommen
wir die
Anwendung lokal
zum Laufen?



Wieso haben wir das
so gemacht?

Architektur dokumentieren

1. Einführung und Ziele

- 1.1 Aufgabenstellung
- 1.2 Qualitätsziele
- 1.3 Stakeholder

2. Randbedingungen

- 2.1 Technische Randbedingungen
- 2.2 Organisatorische Randbedingungen
- 2.3 Konventionen

3. Kontextabgrenzung

- 3.1 Fachlicher Kontext
- 3.2 Technischer- oder Verteilungskontext

4. Lösungsstrategie

5. Bausteinsicht

- 5.1 Ebene 1
- 5.2 Ebene 2

6. Laufzeitsicht

- 6.1 Laufzeitszenario 1
- 6.2 Laufzeitszenario 2

7. Verteilungssicht

- 7.1 Infrastruktur Ebene 1
- 7.2 Infrastruktur Ebene 2

8. Querschnittliche Konzepte

- 8.1 Fachliche Struktur und Modelle
- 8.2 Architektur- und Entwurfsmuster
- 8.3 Unter-der-Haube
- 8.4 User Experience

9. Entwurfsentscheidungen

- 9.1 Entwurfsentscheidung 1
- 9.2 Entwurfsentscheidung 2

10. Qualitätsanforderungen

- 10.1 Qualitätsbaum
- 10.2 Qualitätsszenarien

11. Risiken und technische Schulden

12. Glossar



Was ist wenn nix da ist?

Source Code ist immer da.



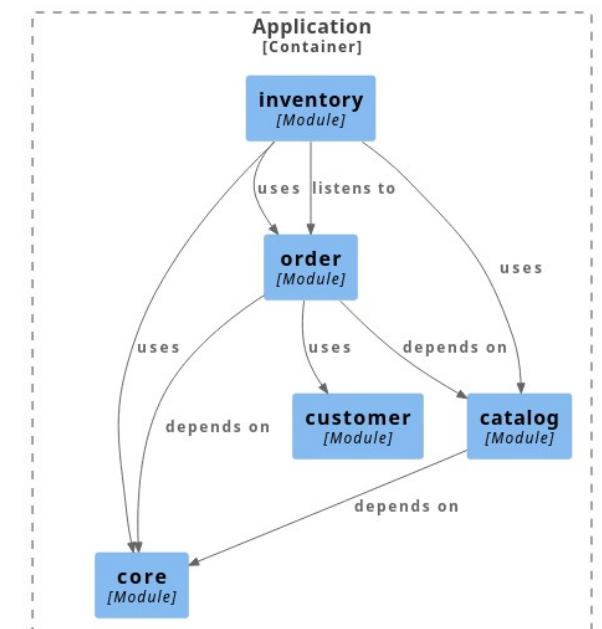
Spring Modulith

@springmodulith

Modulithic Java applications using Spring Boot

- Spring Projekt
- Basiert auf jMolecule
- Spring Boot Erweiterung
- einer der Features:
u.a. Generierung von Diagrammen
basierend auf Annotationen

```
class DocumentationTests {  
  
    ApplicationModules modules = ApplicationModules.of(Application.class);  
  
    @Test  
    void writeDocumentationSnippets() {  
  
        new Documenter(modules)  
            .writeModulesAsPlantUml()  
            .writeIndividualModulesAsPlantUml();  
    }  
}
```



Legend

component

□ container boundary (dashed)

Entscheidungen dokumentieren

ADR (architecture decision records)

<https://adr.github.io/>

[short title of solved problem and solution]

- Status: [proposed | rejected | accepted | deprecated | ... | superseded by [ADR-0005](0005-example.adoc)] <!-- optional -->
- Decision makers: [list everyone involved in the decision] <!-- optional -->
- Date: [YYYY-MM-DD when the decision was last updated] <!-- optional -->

Technical Story: [description | ticket/issue URL] <!-- optional -->

Context and Problem Description

Decision driver <!-- optional -->

- [driver 1, e.g., a force, facing concern, ...]
- [driver 2, e.g., a force, facing concern, ...]
- ... <!-- numbers of drivers can vary -->

Considered Options

- [option 1]
- [option 2]
- [option 3]
- ... <!-- numbers of options can vary -->

Decision

Chosen option: "[option 1]", because [justification. e.g., only option, which meets k.o. criterion decision driver | which resolves force force | ... | comes out best (see below)].

Positive Consequences

- [e.g., improvement of quality attribute satisfaction, follow-up decisions required, ...]
- ...

Negative Consequences

- [e.g., compromising quality attribute, follow-up decisions required, ...]
- ...

Pros and Cons of the Options

[option 1]

[example | description | pointer to more information | ...] <!-- optional -->

- **Good:** because [argument a]
- **Good:** because [argument b]
- **Bad:** because [argument c]
- ... <!-- numbers of pros and cons can vary -->

Links <!-- optional -->

- [Link type] [Link to ADR] <!-- example: Refined by [ADR-0005](0005-example.adoc) -->
- ... <!-- numbers of links can vary -->

```
▼ maintenance-demo [spring-petclinic] ~/dev/workspace/maintenance-demo
  > .idea
  ▼ src
    ▼ main
      ▼ asciidoc
        ▼ technical-decisions
          ▲ 001_ApplicationFramework.adoc
          ▲ decisions.adoc
        ▼ TEMPLATE
          ▲ ADR_TEMPLATE.adoc
  > java
```



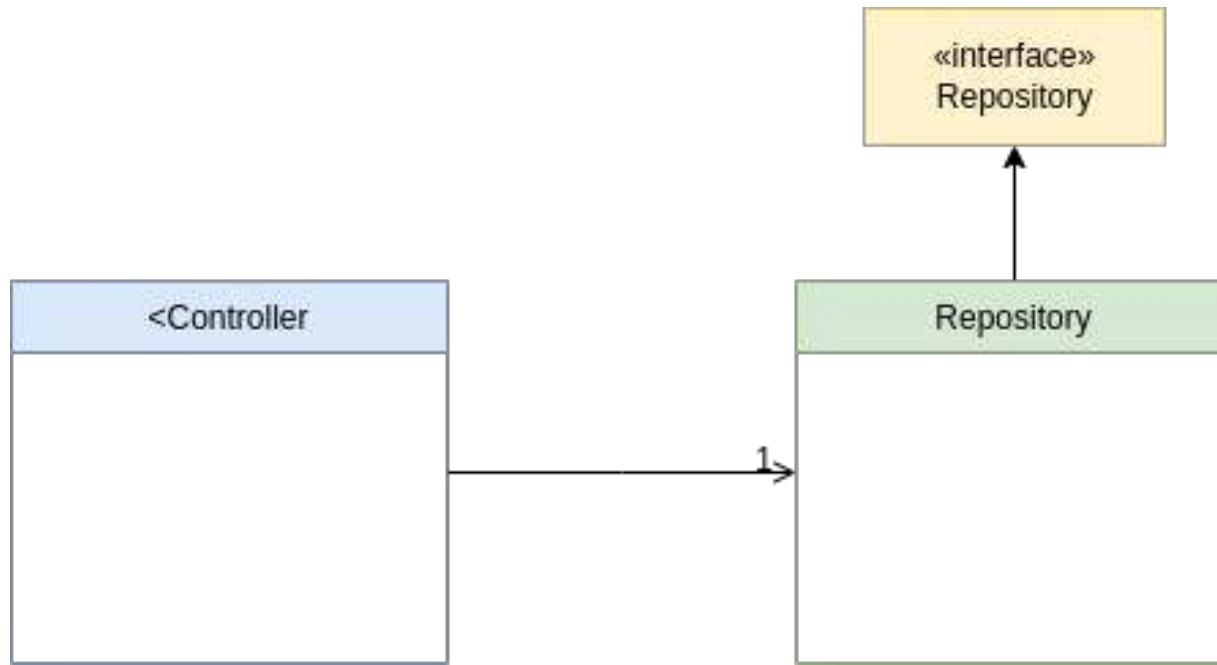
```
<plugin>
  <groupId>org.asciidoctor</groupId>
  <artifactId>asciidoctor-maven-plugin</artifactId>
</plugin>
```

Viele ADR Templates im Netz

- ADR template by Michael Nygard (simple and popular)
- ADR template by Jeff Tyree and Art Akerman (more sophisticated)
- ADR template for Alexandrian pattern (simple with context specifics)
- ADR template for business case (more MBA-oriented, with costs, SWOT, and more opinions)
- ADR template MADR (more Markdown)
- ADR template using Planguage (more quality assurance oriented)



Entspricht die
Implementierung
noch den
Architekturbildern?





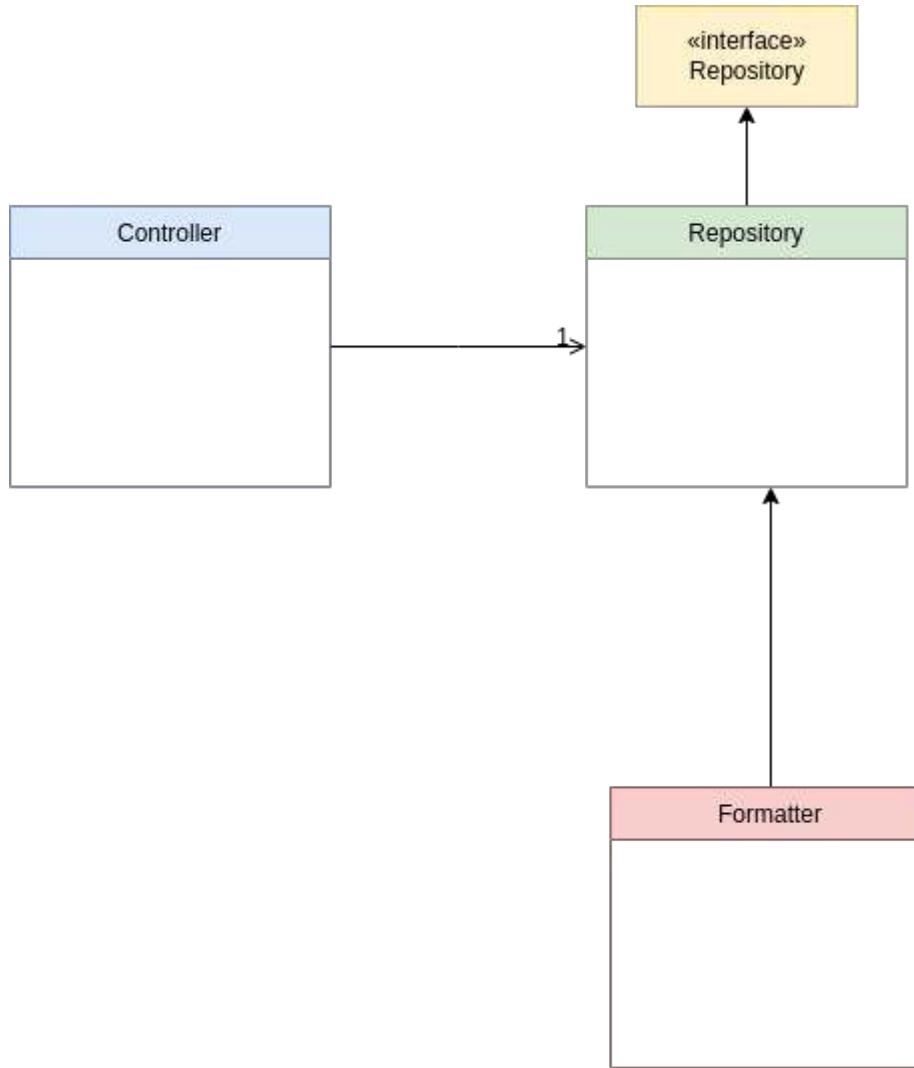


```
@AnalyzeClasses(packages =
{"org.springframework.samples.petclinic.owner", "org.springframework.samples.petclinic.vet"},
    importOptions = ImportOption.DoNotIncludeTests.class)
public class ControllerRepositoryArchTest {

    @ArchTest
    ArchRule relation = classes()
        .that().haveNameMatching(".*Controller")
        .should().accessClassesThat().haveNameMatching(".*Repository");

    @ArchTest
    ArchRule controllerAnnotation = classes()
        .that().haveNameMatching(".*Controller")
        .should().beAnnotatedWith("org.springframework.stereotype.Controller");

    @ArchTest
    ArchRule repositoryDefinition = classes()
        .that().haveNameMatching(".*Repository")
        .should().beAnnotatedWith("org.springframework.stereotype.Repository")
        .orShould().beAssignableTo("org.springframework.data.repository.Repository");
}
```





```
@AnalyzeClasses(importOptions = ImportOption.DoNotIncludeTests.class)
public class ControllerRepositoryFreezeArchTest {

    @ArchTest
    ArchRule relation = FreezingArchRule.freeze(classes()
        .that().haveNameMatching(".*Repository")
        .should().onlyBeAccessed().byClassesThat().haveNameMatching(".*Controller"));
}
```



```
// archunit.properties  
  
freeze.store.default.allowStoreCreation = true  
archRule.failOnEmptyShould = false
```



Welche
Softwarelizenzen
haben wir im
Einsatz?



Welche
Abhängigkeiten
haben wir im
Einsatz?

„Schnelle Küche“

-

mit Maven Boardmitteln

```
mvn dependency:analyze-report
```

Abhängigkeitsanalyse

Verwendete und deklarierte Abhängigkeiten

GroupId	ArtifactId	Version	Scope	Classifier	Type	Optional
javax.cache	cache-api	1.1.1	compile	jar	false	
com.tngtech.archunit	archunit	1.0.1	test	jar	false	

Verwendete aber undeklärte Abhängigkeiten

Unverwendete aber deklarierte Abhängigkeiten

GroupId	ArtifactId	Vers	GroupId	ArtifactId	Version	Scope	Classifier	Type	Optional
org.springframework	spring-test	5.2.0	org.springframework.boot	spring-boot-starter-actuator	2.2.0.RELEASE	compile		jar	false
org.springframework	spring-core	5.2.0	org.springframework.boot	spring-boot-starter-cache	2.2.0.RELEASE	compile		jar	false
org.springframework.boot	spring-boot	2.2.0	org.springframework.boot	spring-boot-starter-data-jpa	2.2.0.RELEASE	compile		jar	false
org.springframework	spring-webmvc	5.2.0	org.springframework.boot	spring-boot-starter-web	2.2.0.RELEASE	compile		jar	false
org.springframework.data	spring-data-commons	2.2.0	org.springframework.boot	spring-boot-starter-thymeleaf	2.2.0.RELEASE	compile		jar	false
org.springframework	spring-context	5.2.0	org.springframework.boot	spring-boot-starter-test	2.2.0.RELEASE	test		jar	false
org.springframework.boot	spring-boot-test-autoconfigure	2.2.0	org.springframework.boot	spring-boot-starter-thymeleaf	2.2.0.RELEASE	compile		jar	false
org.springframework	spring-web	5.2.0	org.springframework.boot	spring-boot-starter-test	2.2.0.RELEASE	test		jar	false
jakarta.validation	jakarta.validation-api	2.0.1	org.hsqldb	hsqldb	2.5.0	runtime		jar	false
org.hamcrest	hamcrest	2.1	mysql	mysql-connector-java	8.0.18	runtime		jar	false
org.assertj	assertj-core	3.13.	org.ehcache	ehcache	3.8.1	compile		jar	false
org.springframework.boot	spring-boot-autoconfigure	2.2.0	org.webjars	webjars-locator-core	0.41	compile		jar	false
org.springframework.data	spring-data-jpa	2.2.0	org.webjars	jquery	2.2.4	compile		jar	false
junit	junit	4.12	org.webjars	jquery-ui	1.11.4	compile		jar	false
org.springframework.boot	spring-boot-test	2.2.0	org.webjars	bootstrap	3.3.6	compile		jar	false
org.mockito	mockito-core	3.1.0	org.springframework.boot	spring-boot-devtools	2.2.0.RELEASE	compile		jar	false
jakarta.persistence	jakarta.persistence-api	2.2.3	com.tngtech.archunit	archunit-junit5	1.0.1	test		jar	false
org.springframework	spring-beans	5.2.0							
com.tngtech.archunit	archunit-junit5-api	1.0.1							
jakarta.xml.bind	jakarta.xml.bind-api	2.3.2							
org.springframework	spring-tx	5.2.0							
org.springframework	spring-orm	5.2.0							

```
mvn license:license-list
```

[INFO] Available licenses :

- * agpl_v3 : GNU Affero General Public License (AGPL) version 3.0
- * apache_v2 : Apache License version 2.0
- * bsd_2 : BSD 2-Clause License
- * bsd_3 : BSD 3-Clause License
- * cddl_v1 : COMMON DEVELOPMENT AND DISTRIBUTION LICENSE (CDDL) Version 1.0
- * epl_only_v1 : Eclipse Public License - v 1.0
- * epl_only_v2 : Eclipse Public License - v 2.0
- * epl_v1 : Eclipse Public + Distribution License - v 1.0
- * epl_v2 : Eclipse Public License - v 2.0 with Secondary License
- * eupl_v1_1 : European Union Public License v1.1
- * fdl_v1_3 : GNU Free Documentation License (FDL) version 1.3
- * gpl_v1 : GNU General Public License (GPL) version 1.0
- * gpl_v2 : GNU General Public License (GPL) version 2.0
- * gpl_v3 : GNU General Public License (GPL) version 3.0
- * lgpl_v2_1 : GNU General Lesser Public License (LGPL) version 2.1
- * lgpl_v3 : GNU General Lesser Public License (LGPL) version 3.0
- * mit : MIT-License

[INFO]

[INFO] BUILD SUCCESS

[INFO]

[INFO] Total time: 6.328 s

[INFO] Finished at: 2022-12-12T19:42:48+01:00

[INFO]

```
mvn license:third-party-report
```

Overview

This report summarizes all third-party dependencies of the project.

# of total dependencies	121
# of dependencies with license declared in their pom	121
# of dependencies with no license in their pom, but filled in the third-party missing file	0
# of dependencies with no license at all	0

Third-parties list

Status	GroupId:ArtifactId:Version	Scope	Classifier	Type	License(s)
antlr:antlr:2.7.7		compile	jar	BSD License	
ch.qos.logback:logback-classic:1.2.3		compile	jar	Eclipse Public License - v 1.0 GNU Lesser General Public License	
ch.qos.logback:logback-core:1.2.3		compile	jar	Eclipse Public License - v 1.0 GNU Lesser General Public License	
com.fasterxml.jackson.core:jackson-annotations:2.10.0		compile		bundle The Apache Software License, Version 2.0	
com.fasterxml.jackson.core:jackson-core:2.10.0		compile		bundle The Apache Software License, Version 2.0	
com.fasterxml.jackson.core:jackson-databind:2.10.0		compile		bundle The Apache Software License, Version 2.0	
com.fasterxml.jackson.datatype:jackson-datatype-jdk8:2.10.0		compile		bundle The Apache Software License, Version 2.0	
com.fasterxml.jackson.datatype:jackson-datatype-jsr310:2.10.0		compile		bundle The Apache Software License, Version 2.0	
com.fasterxml.jackson.module:jackson-module-parameter-names:2.10.0		compile		bundle The Apache Software License, Version 2.0	
com.jayway.jsonpath:json-path:2.4.0		test		jar The Apache Software License, Version 2.0	
com.sun.istack:istack-commons-runtime:3.0.8		compile		jar Eclipse Distribution License - v 1.0	
com.sun.xml.fastinfoset:FastInfoset:1.2.16		compile		jar Apache License, Version 2.0 Eclipse Distribution License - v 1.0	
com.tngtech.archunit:archunit:1.0.1		test		jar BSD	
com.tngtech.archunit:archunit-junit5:1.0.1		test		jar The Apache Software License, Version 2.0	
com.tngtech.archunit:archunit-junit5-api:1.0.1		test		jar The Apache Software License, Version 2.0	
com.tngtech.archunit:archunit-junit5-engine:1.0.1		test		jar The Apache Software License, Version 2.0	
com.tngtech.archunit:archunit-junit5-engine-api:1.0.1		test		jar The Apache Software License, Version 2.0	
com.vaadin.external.google:android-json:0.0.20131108.vaadin1		test		jar Apache License 2.0	
com.zaxxer:HikariCP:3.4.1		compile		bundle The Apache Software License, Version 2.0	

Neuer Silvet Bullet?

-

SBOM



High-Level Object Model

Metadata	Supplier	Authors	Component
	Manufacturer	Tools	
Components	Application	Framework	Operating System
	Library	Container	Device
Services	Provider	Data	
	Endpoints	Trust Boundary	
Dependencies	Components	Services	
Compositions	Completeness of:		
	Components	Services	Dependencies
Vulnerabilities	Details	Source	Exploitability
	Advisories	Risk Ratings	
Extensions	Properties	Per Organization	Per Team
	Formal Taxonomy	Per Industry	...

License Compliance

```
● ● ●

<?xml version="1.0" encoding="UTF-8"?>
<bom xmlns="http://cyclonedx.org/schema/bom/1.4"
      serialNumber="urn:uuid:3e671687-395b-41f5-a30f-a58921a69b79"
      version="1">
  <components>
    <component type="library">
      <group>org.acme</group>
      <name>card-verifier</name>
      <version>1.0.2</version>
      <licenses>
        <!-- This component has an SPDX license expression -->
        <expression>EPL-2.0 OR GPL-2.0 WITH Classpath-exception-2.0</expression>
      </licenses>
    </component>
    <component type="library">
      <group>com.example</group>
      <name>util</name>
      <version>2.0.0</version>
      <licenses>
        <license>
          <!-- This component has a license name
              with optional text -->
          <name>Example, Inc. Commercial License</name>
          <text content-type="text/plain">
mU=</text>
          </license>
        </licenses>
      </component>
    </components>
  </bom>
```

Dependency Graph

```
● ● ●

<?xml version="1.0" encoding="UTF-8"?>
<bom xmlns="http://cyclonedx.org/schema/bom/1.4"
      serialNumber="urn:uuid:3e671687-395b-41f5-a30f-a58921a69b79"
      version="1">
  <metadata>
    <!-- The component for which this BOM describes -->
    <component type="application" bom-ref="acme-app">
      <name>Acme Application</name>
      <version>9.1.1</version>
    </component>
  </metadata>
  <components>
    <component type="framework" bom-ref="pkg:maven/org.acme/web-framework@1.0.0">
      <group>org.acme</group>
      <name>web-framework</name>
      <version>1.0.0</version>
      <purl>pkg:maven/org.acme/web-framework@1.0.0</purl>
    </component>
    <component type="library" bom-ref="pkg:maven/org.acme/persistence@3.1.0">
      <group>org.acme</group>
      <name>persistence</name>
      <version>3.1.0</version>
      <purl>pkg:maven/org.acme/persistence@3.1.0</purl>
    </component>
    <component type="library" bom-ref="pkg:maven/org.acme/common-util@3.0.0">
      <group>org.acme</group>
      <name>common-util</name>
      <version>3.0.0</version>
      <purl>pkg:maven/org.acme/common-util@3.0.0</purl>
    </component>
  </components>
  <dependencies>
```

Dependency Graph

```
</components>
<dependencies>
    <!-- Direct dependencies of the main application -->
    <dependency ref="acme-app">
        <dependency ref="pkg:maven/org.acme/web-framework@1.0.0"/>
        <dependency ref="pkg:maven/org.acme/persistence@3.1.0"/>
    </dependency>
    <!-- All other dependency relationships -->
    <dependency ref="pkg:maven/org.acme/web-framework@1.0.0">
        <dependency ref="pkg:maven/org.acme/common-util@3.0.0"/>
    </dependency>
    <dependency ref="pkg:maven/org.acme/persistence@3.1.0">
        <dependency ref="pkg:maven/org.acme/common-util@3.0.0"/>
    </dependency>
    <dependency ref="pkg:maven/org.acme/common-util@3.0.0"/>
</dependencies>
</bom>
```

Tool Center

Open source tools that support CycloneDX

All tools 165

Open Source 118

Proprietary 49

Build Integration 65

Analysis 58

Author 2

Github Action 14

Transform 9

Library 10

Signing / Notary 7

Distribute 5

opensource build-integration

Audits

Sonatype

Audits an NPM package.json file to identify known vulnerabilities

Forks 50 Stars 189

opensource distribute

BOM Repository Server

CycloneDX

A lightweight repository server used to publish, manage, and distribute CycloneDX SBOMs

Forks 8 Stars 48

opensource analysis

Checkov

Checkov

Prevent cloud misconfigurations during build-time for Terraform, Cloudformation, Kubernetes, Serverless framework and other infrastructure-as-code-languages with Checkov by Bridgecrew. Can output to CycloneDX.

Forks 768 Stars 5k

opensource build-integration

Chelsea

Sonatype

Dependency vulnerability auditor for Ruby

Forks 9 Stars 10

opensource signing-notary

CodeNotary CAS

CodeNotary

CAS is an open source attestation service for the community. Notarize and authorize files, directories, git repos and Build SBOMs of containers. CAS natively supports CycloneDX SBOMs.

Forks 23 Stars 168

opensource signing-notary github-action

Codenotary CAS Authenticate Docker Image and SBOM

Codenotary

A GitHub Action which authenticates notarized Docker images and SBOMs.

Forks 0 Stars 0

opensource signing-notary github-action

Codenotary CAS Notarize Docker Image and SBOM

Codenotary

A GitHub Action which notarizes and creates an SBOM for Docker images.

Forks 1 Stars 1

opensource signing-notary

Codenotary vcn

Codenotary

Protects an organization's software development pipeline from supply chain attacks. Codenotary natively supports CycloneDX SBOMs.

Forks 23 Stars 168

opensource signing-notary

Cosign

Sigstore

Container Signing, Verification and Storage in an OCI registry, including CycloneDX SBOMs

Forks 336 Stars 2.8k

```
mvn cyclonedx:makeAggregateBom
```

```
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.springframework.samples:spring-petclinic >-----
[INFO] Building petclinic 2.2.0.BUILD-SNAPSHOT
[INFO] ----- [ jar ] -----
[INFO]
[INFO] --- cyclonedx-maven-plugin:2.7.3:makeAggregateBom (default-cli) @ spring-petclinic ---
[INFO] CycloneDX: Parameters
[INFO] -----
[INFO] schemaVersion      : 1.4
[INFO] includeBomSerialNumber : true
[INFO] includeCompileScope   : true
[INFO] includeProvidedScope  : true
[INFO] includeRuntimeScope   : true
[INFO] includeTestScope      : false
[INFO] includeSystemScope    : true
[INFO] includeLicenseText    : false
[INFO] outputReactorProjects : true
[INFO] outputFormat         : all
[INFO] outputName           : bom
[INFO] -----
[INFO] CycloneDX: Creating BOM
[INFO] CycloneDX: Writing BOM (XML): /home/sparsick/dev/workspace/maintenance-demo/target/bom.xml
[INFO] CycloneDX: Validating BOM (XML): /home/sparsick/dev/workspace/maintenance-demo/target/bom.xml
[INFO] CycloneDX: Writing BOM (JSON): /home/sparsick/dev/workspace/maintenance-demo/target/bom.json
[INFO] CycloneDX: Validating BOM (JSON): /home/sparsick/dev/workspace/maintenance-demo/target/bom.json
[WARNING] Unknown keyword additionalItems - you should define your own Meta Schema. If the keyword is irrelevant for validation, just use a NonValidationKeyword
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 3.349 s
[INFO] Finished at: 2022-12-12T19:54:08+01:00
[INFO] -----
```



Wie können wir
am besten up to
date bleiben?



Dependabot

Automated dependency updates built into GitHub

593 followers United States of America https://github.com/features/security support@github.com Verified

Overview

Repositories 23

Projects

Packages

People 11

Pinned

dependabot-core Public

Dependabot's update PR creation logic. Feedback: https://github.com/orgs/community/discussions/categories/code-security

Ruby 3.1k 739

Repositories

Find a repository...

Type ▾

Language ▾

Sort ▾

dependabot-core Public

Dependabot's update PR creation logic. Feedback: https://github.com/orgs/community/discussions/categories/code-security



Ruby

3,060

739

692 (6 issues need help)

88

Updated 5 hours ago

dependabot-actions-workflow Public

Example workflow for updating Dependabot pull requests



Ruby

21

16

0

2

Updated 9 hours ago

dependabot-script Public

A simple script that demonstrates how to use Dependabot Core



Ruby

420

MIT

234

54

10

Updated 3 days ago

cli

Public

A tool for testing and debugging Dependabot update jobs.



Go

22

MIT

2

3

2

Updated 6 days ago



MEND
Renovate



Supported Platforms

Renovate works on these platforms:

- GitHub (.com and Enterprise Server)
- GitLab (.com and CE/EE)
- Bitbucket Cloud
- Bitbucket Server
- Azure DevOps
- AWS CodeCommit
- Gitea

Label issues and pull requests for new contributors

[Dismiss](#)

Now, GitHub will help potential first-time contributors [discover issues](#) labeled with [good first issue](#)

[Go to Labels](#)[Filters ▾](#) Q is:pr is:open[Labels 7](#)[Milestones 0](#)[New pull request](#) 13 Open ✓ 149 Closed

Author ▾

Label ▾

Projects ▾

Milestones ▾

Reviews ▾

Assignee ▾

Sort ▾

Update dependency org.apache.maven.plugins:maven-project-info-reports-plugin to v3.4.1 ✓

#310 opened 25 days ago by renovate bot 1 task

1

Update dependency org.apache.maven.plugins:maven-jxr-plugin to v3.3.0 ✓

#309 opened 26 days ago by renovate bot 1 task

Update dependency org.apache.maven.plugins:maven-javadoc-plugin to v3.4.1 ✓

#308 opened 26 days ago by renovate bot 1 task

1

Update dependency org.eclipse.platform:org.eclipse.osgi to v3.18.200 ✓

#292 opened on May 31 by renovate bot 1 task

1

Update dependency org.codehaus.plexus:plexus-component-metadata to v2.1.1 ✓

#291 opened on May 31 by renovate bot 1 task

1

Update dependency org.codehaus.plexus:plexus-component-annotations to v2.1.1 ✓

#290 opened on May 31 by renovate bot 1 task

1

Update dependency org.apache.maven.plugins:maven-plugin-plugin to v3.7.0 ✓

#284 opened on Dec 16, 2021 by renovate bot 1 task

1

Update dependency org.apache.maven.plugin-tools:maven-plugin-annotations to v3.7.0 ✓

#265 opened on Apr 26, 2021 by renovate bot 1 task

1

Update dependency org.apache.maven.plugins:maven-project-info-reports-plugin to v3.4.1 #310

[Edit](#)[Code](#) ▾[Gitpod](#)[Open](#)renovate wants to merge 1 commit into `master` from `renovate/org.apache.maven.plugins-maven-project-info-reports-plugin-3.x`[Conversation 0](#)[-o Commits 1](#)[Checks 8](#)[Files changed 1](#)[+1 -1](#)

renovate (bot) commented 25 days ago

[Contributor](#)

...

This PR contains the following updates:

Package	Change	Age	Adoption	Passing	Confidence
org.apache.maven.plugins:maven-project-info-reports-plugin	3.1.1 -> 3.4.1	124d	32%	96%	high

Configuration

Schedule: Branch creation - At any time (no schedule defined), Automerger - At any time (no schedule defined).

Automerger: Disabled by config. Please merge this manually once you are satisfied.

Rebasing: Whenever PR becomes conflicted, or you tick the rebase/retry checkbox.

Ignore: Close this PR and you won't be reminded about this update again.

If you want to rebase/retry this PR, check this box

This PR has been generated by Mend Renovate. View repository job log [here](#).

-o Update dependency org.apache.maven.plugins:maven-project-info-reports... ...

[Verified](#)

✓ 13bcd79

[Unsubscribe](#)

You're receiving notifications because you're watching this repository.

Reviewers



Suggestions

[Request](#)

Still in progress? Convert to draft

Assignees



No one—assign yourself

Labels



None yet

Projects



None yet

Milestone



No milestone

Development



Successfully merging this pull request may close these issues.

None yet

Notifications

[Customize](#)[Unsubscribe](#)

Dependency Dashboard #278

[Edit](#)[New issue](#)[Gitpod](#)[Open](#)[29 tasks](#)

renovate (bot) opened this issue on Sep 14, 2021 · 0 comments



renovate (bot) commented on Sep 14, 2021 · edited

[Contributor](#)

...

This issue lists Renovate updates and detected dependencies. Read the [Dependency Dashboard](#) docs to learn more.

Rate-Limited

These updates are currently rate-limited. Click on a checkbox below to force their creation now.

- Update dependency org.apache.maven.plugins:maven-site-plugin to v3.12.1
- Update dependency org.apache.maven.wagon:wagon-ftp to v3.5.2
- Update dependency org.apache.maven.wagon:wagon-ssh to v3.5.2
- Update dependency org.apache.rat:apache-rat-plugin to v0.15
- Update dependency org.assertj:assertj-core to v3.23.1
- Update dependency org.twdata.maven:mojo-executor to v2.4.0
- Update maven.version to v3.8.6 (org.apache.maven:maven-core , org.apache.maven:maven-artifact , org.apache.maven:maven-plugin-api , org.apache.maven:maven-model , org.apache.maven:maven-settings , org.apache.maven:maven-compat)
- Update release-drafter/release-drafter action to v5.21.1
- Update actions/cache action to v3
- Update actions/checkout action to v3
- Update actions/setup-java action to v3
- Update codecov/codecov-action action to v3
- Update dependency biz.aQute.bnd:biz.aQute.bndlib to v6
- Update dependency org.apache.maven.plugins:maven-gpg-plugin to v3
- Update dependency org.mockito:mockito-inline to v4
- Update tycho.version to v3 (major) (org.eclipse.tycho:sisu-equinox-launching , org.eclipse.tycho:tycho-core)
- Create all rate-limited PRs at once 🔑

Assignees

No one—assign yourself



Labels

None yet



Projects

None yet



Milestone

No milestone



Development

[Create a branch](#) for this issue or link a pull request.

Notifications

[Customize](#)[🔕 Unsubscribe](#)

You're receiving notifications because you're watching this repository.

0 participants

[🔒 Lock conversation](#)[📌 Pin issue ⓘ](#)[→ Transfer issue](#)

Self Hosting

- Fertige GitHub Action und GitLab Runner
- Docker Image
- Mend Renovate On-Premises
- Mend Remediate (Kommerziell)



```
# Renovate Global Config
$ cat ./renovate/config.js
module.exports = {
  repositories: ["sparsick/maintenance-demo"],
  dryRun : "true",
  prCommitsPerRunLimit : "5"
};

# Renovate Per Repository Config
$ cat renovate.json
{
  "extends": [
    "config:base"
  ]
}
```

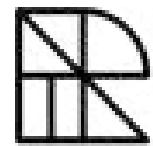


```
$ docker run --rm -v "$PWD/.renovate/config.js:/usr/src/app/config.js" -e RENOVATE_TOKEN=apitoken renovate/renovate

INFO: Repository started (repository=sparsick/maintenance-demo)
      "renovateVersion": "34.55.0"
INFO: Dependency extraction complete (repository=sparsick/maintenance-demo, baseBranch=master)
      "stats": {
        "managers": {
          "docker-compose": {"fileCount": 1, "depCount": 1},
          "maven": {"fileCount": 1, "depCount": 16}
        },
        "total": {"fileCount": 2, "depCount": 17}
      }
INFO: DRY-RUN: Would commit files to branch renovate/org.asciidoc-asciidocj-diagram-1.x (repository=sparsick/maintenance-demo,
branch=renovate/org.asciidoc-asciidocj-diagram-1.x)
...
INFO: DRY-RUN: Would ensure Dependency Dashboard (repository=sparsick/maintenance-demo)
      "title": "Dependency Dashboard"
INFO: DRY-RUN: Would save repository cache. (repository=sparsick/maintenance-demo)
INFO: Repository finished (repository=sparsick/maintenance-demo)
      "cloned": true,
      "durationMs": 52013
```

Was ist bei Updates, wo ich viel Code anpassen
muss/will?

Java 8 → Java 11
JUnit4 → JUnit5



Rewrite

Large-Scale Automated Source Code Refactoring



```
<plugin>
    <groupId>org.openrewrite.maven</groupId>
    <artifactId>rewrite-maven-plugin</artifactId>
    <version>4.38.0</version>
    <configuration>
        <activeRecipes>
            <recipe>org.openrewrite.java.migrate.Java8toJava11</recipe>
        </activeRecipes>
    </configuration>
    <dependencies>
        <dependency>
            <groupId>org.openrewrite.recipe</groupId>
            <artifactId>rewrite-migrate-java</artifactId>
            <version>1.13.0</version>
        </dependency>
    </dependencies>
</plugin>
```

```
$ mvn rewrite:discover

[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.springframework.samples:spring-petclinic >-----
[INFO] Building petclinic 2.2.0.BUILD-SNAPSHOT
[INFO]           [ jar ]
[INFO]
[INFO] --- rewrite-maven-plugin:4.38.0:discover (default-cli) @ spring-petclinic ---
[INFO] Available Recipes:
[INFO]   ...
[INFO]   org.openrewrite.java.cleanup.AddSerialVersionUIDToSerializable
[INFO]   org.openrewrite.java.cleanup.AtomicPrimitiveEqualsUsesGet
[INFO]   org.openrewrite.java.format.EmptyNewlineAtEndOfFile
[INFO]   org.openrewrite.java.format.NormalizeTabsOrSpaces
[INFO]   org.openrewrite.java.migrate.AddSuppressionForIllegalReflectionWarningsPlugin
[INFO]   org.openrewrite.java.migrate.Java8toJavall
[INFO]   org.openrewrite.java.migrate.UseJavaUtilBase64
[INFO]   org.openrewrite.java.migrate.apache.commons.codec.ApacheBase64ToJavaBase64
[INFO]   org.openrewrite.java.migrate.guava.NoGuavaCreateTempDir
[INFO]   ...
[INFO] Available Styles:
[INFO]   org.openrewrite.java.IntelliJ
[INFO]   com.netflix.eureka.Style
[INFO]   org.openrewrite.java.GoogleJavaFormat
[INFO]   org.openrewrite.java.SpringFormat
[INFO]   com.netflix.genie.Style
[INFO]
[INFO] Active Styles:
[INFO]
[INFO] Active Recipes:
[INFO]   org.openrewrite.java.migrate.Java8toJavall
[INFO]
[INFO] Found 416 available recipes and 5 available styles.
[INFO] Configured with 1 active recipes and 0 active styles.
[INFO]
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 4.780 s
[INFO] Finished at: 2022-12-13T10:47:34+01:00
[INFO] -----
```

Beispiel: Java 8 nach Java 11

```
<plugin>
    <groupId>org.openrewrite.maven</groupId>
    <artifactId>rewrite-maven-plugin</artifactId>
    <version>4.38.0</version>
    <configuration>
        <activeRecipes>
            <recipe>org.openrewrite.java.migrate.Java8toJava11</recipe>
        </activeRecipes>
    </configuration>
    <dependencies>
        <dependency>
            <groupId>org.openrewrite.recipe</groupId>
            <artifactId>rewrite-migrate-java</artifactId>
            <version>1.13.0</version>
        </dependency>
    </dependencies>
</plugin>
```

```
$ mvn rewrite:run
[INFO] --- rewrite-maven-plugin:4.38.0:run (default-cli) @ spring-petclinic ---
[INFO] Using active recipe(s) [org.openrewrite.java.migrate.Java8toJava11]
[INFO] Using active styles(s) []
[INFO] Validating active recipes...
[INFO] Project [petclinic] Resolving Poms...
[INFO] Project [petclinic] Parsing Source Files
[INFO] Running recipe(s)...
[WARNING] Changes have been made to pom.xml by:
[WARNING]     org.openrewrite.java.migrate.Java8toJava11
[WARNING]         org.openrewrite.java.migrate.javax.AddJaxbDependencies
[WARNING]             org.openrewrite.java.migrate.javax.AddJaxbRuntime: {runtime=sun}
[WARNING]         org.openrewrite.java.migrate.AddJDeprScanPlugin: {release=}
[WARNING]         org.openrewrite.java.migrate.wro4j.UpgradeWro4jMavenPluginVersion
[WARNING]             org.openrewrite.maven.UpgradePluginVersion: {groupId=ro.isdc.wro4j,
artifactId=wro4j-maven-plugin, newVersion=1.10.1}
[WARNING]         org.openrewrite.java.migrate.jacoco.UpgradeJaCoCoMavenPluginVersion
[WARNING]             org.openrewrite.maven.UpgradePluginVersion: {groupId=org.jacoco,
artifactId=jacoco-maven-plugin, newVersion=0.8.8}
[WARNING]         org.openrewrite.java.migrate.JavaVersion11
[WARNING]             org.openrewrite.maven.ChangePropertyValue: {key=java.version, newValue=11,
addIfMissing=false}
[WARNING] Please review and commit the results.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  34.463 s
[INFO] Finished at: 2022-12-13T11:07:24+01:00
[INFO] -----
```

```
$ git diff
diff --git a/pom.xml b/pom.xml
index 75e36f9..ff909b4 100644
--- a/pom.xml
+++ b/pom.xml
@@ -17,7 +17,7 @@
<properties>

    <!-- Generic properties -->
-    <java.version>1.8</java.version>
+    <java.version>11</java.version>

@@ -25,13 +25,13 @@
-    <wro4j.version>1.8.0</wro4j.version>
+    <wro4j.version>1.10.1</wro4j.version>

-    <jacoco.version>0.8.2</jacoco.version>
+    <jacoco.version>0.8.8</jacoco.version>

@@ -91,6 +91,24 @@
        <groupId>org.webjars</groupId>
        <artifactId>webjars-locator-core</artifactId>
    </dependency>
+    <dependency>
+        <groupId>com.sun.xml.bind</groupId>
+        <artifactId>jaxb-impl</artifactId>
+        <version>2.3.2</version>
+        <scope>provided</scope>
+    </dependency>
+    <dependency>
+        <groupId>com.sun.xml.bind</groupId>
+        <artifactId>jaxb-impl</artifactId>
+        <version>2.3.2</version>
+        <scope>provided</scope>
+    </dependency>
+    <dependency>
+        <groupId>com.sun.xml.bind</groupId>
+        <artifactId>jaxb-impl</artifactId>
+        <version>2.3.2</version>
+        <scope>provided</scope>
+    </dependency>
    <dependency>
        <groupId>org.webjars</groupId>
        <artifactId>jquery</artifactId>
@@ -230,89 +248,97 @@
```

Beispiel: Java 11 nach Java 17



```
$ mvn rewrite:discover | grep Java17
[INFO]          org.openrewrite.java.migrate.UpgradeJava17
```



```
$ mvn rewrite:run

[INFO] ---- rewrite-maven-plugin:4.38.0:run (default-cli) @ spring-petclinic ---
[INFO] Using active recipe(s) [org.openrewrite.java.migrate.UpgradeJava17]
[INFO] Using active styles(s) []
[INFO] Validating active recipes...
[INFO] Project [petclinic] Resolving Poms...
[INFO] Project [petclinic] Parsing Source Files
[INFO] Running recipe(s)...
[WARNING] Changes have been made to pom.xml by:
[WARNING]   org.openrewrite.java.migrate.UpgradeJava17
[WARNING]     org.openrewrite.java.migrate.Java8toJava11
[WARNING]       org.openrewrite.java.migrate.javax.AddJaxbDependencies
[WARNING]         org.openrewrite.java.migrate.javax.AddJaxbRuntime: {runtime=sun}
[WARNING]       org.openrewrite.java.migrate.wro4j.UpgradeWro4jMavenPluginVersion
[WARNING]         org.openrewrite.maven.UpgradePluginVersion: {groupId=ro.isdc.wro4j, artifactId=wro4j-maven-plugin,
newVersion=1.10.1}
[WARNING]       org.openrewrite.java.migrate.JavaVersion11
[WARNING]         org.openrewrite.maven.ChangePropertyValue: {key=java.version, newValue=11, addIfMissing=false}
[WARNING]     org.openrewrite.java.migrate.JavaVersion17
[WARNING]       org.openrewrite.maven.ChangePropertyValue: {key=java.version, newValue=17, addIfMissing=false}
[WARNING] Changes have been made to src/test/java/org/springframework/samples/petclinic/owner/OwnerControllerTests.java by:
[WARNING]   org.openrewrite.java.migrate.UpgradeJava17
[WARNING]     org.openrewrite.java.migrate.Java8toJava11
[WARNING]       org.openrewrite.java.migrate.util.JavaUtilAPIs
[WARNING]         org.openrewrite.java.migrate.util.MigrateCollectionsSingletonList
[WARNING]         org.openrewrite.java.migrate.util.MigrateCollectionsSingletonSet
[WARNING] Please review and commit the results.
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time:  32.266 s
[INFO] Finished at: 2022-12-13T11:46:21+01:00
[INFO] -----
```

```
$ git diff

diff --git a/pom.xml b/pom.xml
index cd0f967..c7921ac 100644
--- a/pom.xml
+++ b/pom.xml
@@ -17,7 +17,7 @@
-   <java.version>11</java.version>
+   <java.version>17</java.version>

@@ -25,7 +25,7 @@
-   <wro4j.version>1.8.0</wro4j.version>
+   <wro4j.version>1.10.1</wro4j.version>

@@ -97,6 +97,24 @@
+   <dependency>
+     <groupId>com.sun.xml.bind</groupId>
+     <artifactId>jaxb-impl</artifactId>
+     <version>2.3.2</version>
+     <scope>provided</scope>
+   </dependency>
+   <dependency>
+     <groupId>com.sun.xml.bind</groupId>
+     <artifactId>jaxb-impl</artifactId>
+     <version>2.3.2</version>
+     <scope>provided</scope>
+   </dependency>
+   <dependency>
+     <groupId>com.sun.xml.bind</groupId>
+     <artifactId>jaxb-impl</artifactId>
+     <version>2.3.2</version>
+     <scope>provided</scope>
+   </dependency>
+   <dependency>
```

diff --git a/src/test/java/org/springframework/samples/petclinic/owner/OwnerControllerTests.java b/src/test/java/org/springframework/samples/petclinic/owner/OwnerControllerTests.java

```
index 5084f42..4855b19 100644
--- a/src/test/java/org/springframework/samples/petclinic/owner/OwnerControllerTests.java
+++ b/src/test/java/org/springframework/samples/petclinic/owner/OwnerControllerTests.java
@@ -17,8 +17,8 @@
 package org.springframework.samples.petclinic.owner;

 import java.time.LocalDate;
-import java.util.Collections;
 import java.util.List;
+import java.util.Set;

@@ -84,11 +84,11 @@ public class OwnerControllerTests {
-    george.setPetsInternal(Collections.singleton(max));
+    george.setPetsInternal(Set.of(max));

-    given(this.visits.findByPetId(max.getId())).willReturn(Collections.singletonList(visit));
+    given(this.visits.findByPetId(max.getId())).willReturn(List.of(visit));
```



```
$ sdk use java 17-open
$ mvn clean verify

...
[INFO] Running org.springframework.samples.petclinic.service.ClinicServiceTests
[ERROR] Tests run: 1, Failures: 0, Errors: 1, Skipped: 0, Time elapsed: 0 s <<< FAILURE! - in
org.springframework.samples.petclinic.service.ClinicServiceTests
[ERROR] initializationError  Time elapsed: 0 s  <<< ERROR!
org.springframework.beans.factory.BeanDefinitionStoreException: Failed to read candidate component class: file [/home/sparsick
/dev/workspace/maintenance-demo/target/test-classes/org/springframework/samples/petclinic/service/ClinicServiceTests.class]; nested
exception is org.springframework.core.NestedIOException: ASM ClassReader failed to parse class file - probably due to a new Java
class file version that isn't supported yet: file [/home/sparsick/dev/workspace/maintenance-demo/target/test-classes
/org/springframework/samples/petclinic/service/ClinicServiceTests.class]; nested exception is java.lang.IllegalArgumentException:
Unsupported class file major version 61
Caused by: org.springframework.core.NestedIOException: ASM ClassReader failed to parse class file - probably due to a new Java class
file version that isn't supported yet: file [/home/sparsick/dev/workspace/maintenance-demo/target/test-classes/org/springframework
/samples/petclinic/service/ClinicServiceTests.class]; nested exception is java.lang.IllegalArgumentException: Unsupported class file
major version 61
Caused by: java.lang.IllegalArgumentException: Unsupported class file major version 61

[INFO] Running org.springframework.samples.petclinic.PetclinicIntegrationTests
[ERROR] Tests run: 1, Failures: 0, Errors: 1, Skipped: 0, Time elapsed: 0 s <<< FAILURE! - in
org.springframework.samples.petclinic.PetclinicIntegrationTests
[ERROR] initializationError  Time elapsed: 0 s  <<< ERROR!
org.springframework.beans.factory.BeanDefinitionStoreException: Failed to read candidate component class: file [/home/sparsick
/dev/workspace/maintenance-demo/target/test-classes/org/springframework/samples/petclinic/PetclinicIntegrationTests.class]; nested
exception is org.springframework.core.NestedIOException: ASM ClassReader failed to parse class file - probably due to a new Java
class file version that isn't supported yet: file [/home/sparsick/dev/workspace/maintenance-demo/target/test-classes
/org/springframework/samples/petclinic/PetclinicIntegrationTests.class]; nested exception is java.lang.IllegalArgumentException:
Unsupported class file major version 61
Caused by: org.springframework.core.NestedIOException: ASM ClassReader failed to parse class file - probably due to a new Java class
file version that isn't supported yet: file [/home/sparsick/dev/workspace/maintenance-demo/target/test-classes/org/springframework
/samples/petclinic/PetclinicIntegrationTests.class]; nested exception is java.lang.IllegalArgumentException: Unsupported class file
major version 61
Caused by: java.lang.IllegalArgumentException: Unsupported class file major version 61

...
```

Beispiel: JUnit4 nach JUnit5



```
<plugin>
  <groupId>org.openrewrite.maven</groupId>
  <artifactId>rewrite-maven-plugin</artifactId>
  <version>4.38.0</version>
  <configuration>
    <activeRecipes>
      <recipe>org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration</recipe>
    </activeRecipes>
  </configuration>
  <dependencies>
    <dependency>
      <groupId>org.openrewrite.recipe</groupId>
      <artifactId>rewrite-spring</artifactId>
      <version>4.29.0</version>
    </dependency>
  </dependencies>
</plugin>
```



```
$ mvn rewrite:run
[INFO] Using active recipe(s) [org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration]
[INFO] Using active styles(s) []
[INFO] Validating active recipes...
[INFO] Project [petclinic] Resolving Poms...
[INFO] Project [petclinic] Parsing Source Files
[INFO] Running recipe(s)...
[WARNING] Changes have been made to pom.xml by:
[WARNING]   org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration
[WARNING]     org.openrewrite.java.testing.junit5.JUnit4to5Migration
[WARNING]       org.openrewrite.maven.ExcludeDependency: {groupId=junit, artifactId=junit}
[WARNING]       org.openrewrite.maven.ExcludeDependency: {groupId=org.junit.vintage, artifactId=junit-vintage-engine}
[WARNING] Changes have been made to src/test/java/org/springframework/samples/petclinic/owner/OwnerControllerTests.java by:
[WARNING]   org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration
[WARNING]     org.openrewrite.java.testing.junit5.JUnit4to5Migration
[WARNING]       org.openrewrite.java.testing.junit5.UpdateBeforeAfterAnnotations
[WARNING]       org.openrewrite.java.testing.junit5.UpdateTestAnnotation
[WARNING]     org.openrewrite.java.spring.boot2.UnnecessarySpringRunWith
[WARNING]       org.openrewrite.java.testing.junit5.RunnerToExtension: {runners=
[org.springframework.test.context.junit4.SpringRunner, org.springframework.test.context.junit4.SpringJUnit4ClassRunner],
extension=org.springframework.test.context.junit.jupiter.SpringExtension}
[WARNING]       org.openrewrite.java.spring.boot2.UnnecessarySpringExtension
[WARNING] Changes have been made to src/test/java/org/springframework/samples/petclinic/owner/VisitControllerTests.java by:
[WARNING]   org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration
[WARNING]     org.openrewrite.java.testing.junit5.JUnit4to5Migration
[WARNING]       org.openrewrite.java.testing.junit5.UpdateBeforeAfterAnnotations
[WARNING]       org.openrewrite.java.testing.junit5.UpdateTestAnnotation
[WARNING]     org.openrewrite.java.spring.boot2.UnnecessarySpringRunWith
[WARNING]       org.openrewrite.java.testing.junit5.RunnerToExtension: {runners=
[org.springframework.test.context.junit4.SpringRunner, org.springframework.test.context.junit4.SpringJUnit4ClassRunner],
extension=org.springframework.test.context.junit.jupiter.SpringExtension}
[WARNING]       org.openrewrite.java.spring.boot2.UnnecessarySpringExtension
```

```
[INFO] --- org.springframework:spring-framework-test-context:junit5-jupiter:spring-framework-extens...  
[WARNING]      org.openrewrite.java.spring.boot2.UnnecessarySpringExtension  
[WARNING] Changes have been made to src/test/java/org/springframework/samples/petclinic/vet/VetControllerTests.java by:  
[WARNING]      org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration  
[WARNING]      org.openrewrite.java.testing.junit5.JUnit4to5Migration  
[WARNING]          org.openrewrite.java.testing.junit5.UpdateBeforeAfterAnnotations  
[WARNING]          org.openrewrite.java.testing.junit5.UpdateTestAnnotation  
[WARNING]      org.openrewrite.java.spring.boot2.UnnecessarySpringRunWith  
[WARNING]          org.openrewrite.java.testing.junit5.RunnerToExtension: {runners=  
[org.springframework.test.context.junit4.SpringRunner, org.springframework.test.context.junit4.SpringJUnit4ClassRunner],  
extension=org.springframework.test.context.junit.jupiter.SpringExtension}  
[WARNING]      org.openrewrite.java.spring.boot2.UnnecessarySpringExtension  
[WARNING] Changes have been made to src/test/java/org/springframework/samples/petclinic/vet/VetTests.java by:  
[WARNING]      org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration  
[WARNING]      org.openrewrite.java.testing.junit5.JUnit4to5Migration  
[WARNING]          org.openrewrite.java.testing.junit5.UpdateTestAnnotation  
[WARNING] Changes have been made to src/test/java/org/springframework/samples/petclinic/service/ClinicServiceTests.java by:  
[WARNING]      org.openrewrite.java.spring.boot2.SpringBoot2JUnit4to5Migration  
[WARNING]      org.openrewrite.java.testing.junit5.JUnit4to5Migration  
[WARNING]          org.openrewrite.java.testing.junit5.UpdateTestAnnotation  
[WARNING]      org.openrewrite.java.spring.boot2.UnnecessarySpringRunWith  
[WARNING]          org.openrewrite.java.testing.junit5.RunnerToExtension: {runners=  
[org.springframework.test.context.junit4.SpringRunner, org.springframework.test.context.junit4.SpringJUnit4ClassRunner],  
extension=org.springframework.test.context.junit.jupiter.SpringExtension}  
[WARNING]      org.openrewrite.java.spring.boot2.UnnecessarySpringExtension  
[WARNING] Please review and commit the results.  
[INFO] -----  
[INFO] BUILD SUCCESS  
[INFO] -----  
[INFO] Total time:  36.321 s  
[INFO] Finished at: 2022-12-13T12:17:55+01:00  
[INFO] -----
```



```
$ git diff
```

```
diff --git a/pom.xml b/pom.xml
index cd0f967..c8db6dc 100644
--- a/pom.xml
+++ b/pom.xml
@@ -62,6 +62,16 @@
     <groupId>org.springframework.boot</groupId>
     <artifactId>spring-boot-starter-test</artifactId>
     <scope>test</scope>
+    <exclusions>
+        <exclusion>
+            <groupId>junit</groupId>
+            <artifactId>junit</artifactId>
+        </exclusion>
+        <exclusion>
+            <groupId>org.junit.vintage</groupId>
+            <artifactId>junit-vintage-engine</artifactId>
+        </exclusion>
+    </exclusions>
</dependency>
```

```
diff --git a/src/test/java/org/springframework/samples/petclinic/PetclinicIntegrationTests.java b/src/test/java/org/springframework/samples/petclinic/PetclinicIntegrationTests.java
index 669dbdd..1db15a8 100644
--- a/src/test/java/org/springframework/samples/petclinic/PetclinicIntegrationTests.java
+++ b/src/test/java/org/springframework/samples/petclinic/PetclinicIntegrationTests.java
@@ -16,15 +16,12 @@
package org.springframework.samples.petclinic;

-import org.junit.Test;
-import org.junit.runner.RunWith;
+import org.junit.jupiter.api.Test;

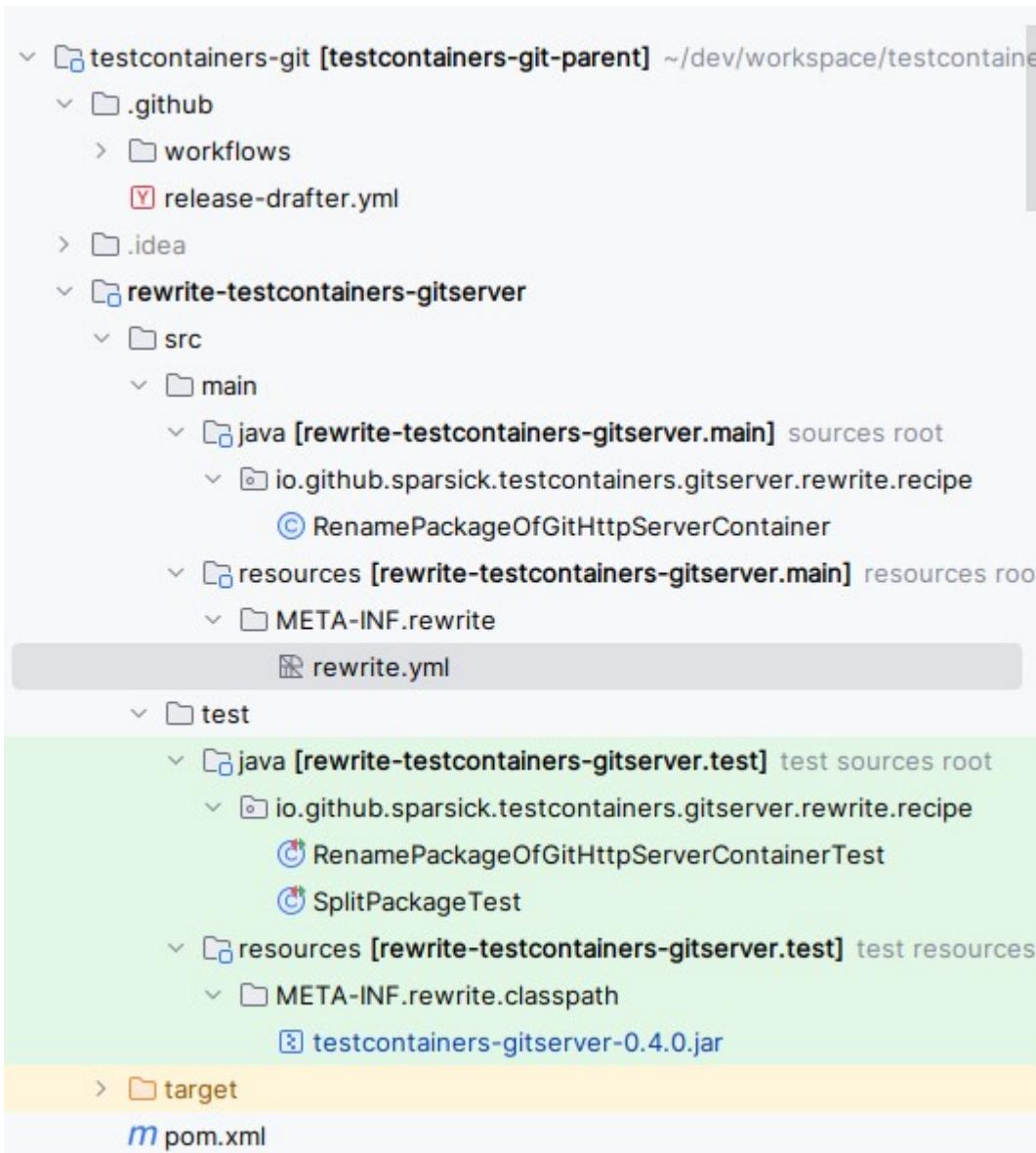
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.samples.petclinic.vet.VetRepository;
-import org.springframework.test.context.junit4.SpringRunner;

-@RunWith(SpringRunner.class)
@SpringBootTest
public class PetclinicIntegrationTests {

@@ -32,7 +29,7 @@ public class PetclinicIntegrationTests {
    private VetRepository vets;

    @Test
-   public void testfindAll() throws Exception {
+   void testfindAll() throws Exception {
        vets.findAll();
        vets.findAll(); // served from cache
    }
}
```

Und wenn ich doch was eigene Rezepte
schreiben möchte?





```
---  
type: specs.openrewrite.org/v1beta/recipe  
name:  
io.github.sparsick.testcontainers.gitserver.rewrite.recipe.RenamePackageOfBasicAuthenticationCredentials  
displayName: Rename package of BasicAuthenticationCredentials  
recipeList:  
- org.openrewrite.java.ChangeType:  
  oldFullyQualifiedTypeName:  
    com.github.sparsick.testcontainers.gitserver.BasicAuthenticationCredentials  
    newFullyQualifiedTypeName:  
      com.github.sparsick.testcontainers.gitserver.http.BasicAuthenticationCredentials  
      ignoreDefinition: null  
  
---  
type: specs.openrewrite.org/v1beta/recipe  
name: io.github.sparsick.testcontainers.gitserver.rewrite.recipe.RenamePackageOfGitServerContainer  
displayName: Rename package of GitServerContainer  
recipeList:  
- org.openrewrite.java.ChangeType:  
  oldFullyQualifiedTypeName: com.github.sparsick.testcontainers.gitserver.GitServerContainer  
  newFullyQualifiedTypeName: com.github.sparsick.testcontainers.gitserver.plain.GitServerContainer  
  ignoreDefinition: null
```

```
public class RenamePackageOfGitHttpServerContainer extends Recipe {
    @JsonCreator
    public RenamePackageOfGitHttpServerContainer() {
    }

    @Override
    public String getDisplayName() {
        return "Rename import for GitHttpContainer";
    }

    @Override
    public String getDescription() {
        return "Rename import for GitHttpContainer.";
    }

    @Override
    public TreeVisitor<?, ExecutionContext> getVisitor() {
        return Preconditions.checkNotNull(Preconditions.or(
            new UsesType<>("com.github.sparsick.testcontainers.gitserver.GitHttpServerContainer",
false),
            new FindImports("com.github.sparsick.testcontainers.gitserver.GitHttpServerContainer",
null).getVisitor(),
            new RenamePackageOfGitHttpContainerVisitor()));
    }

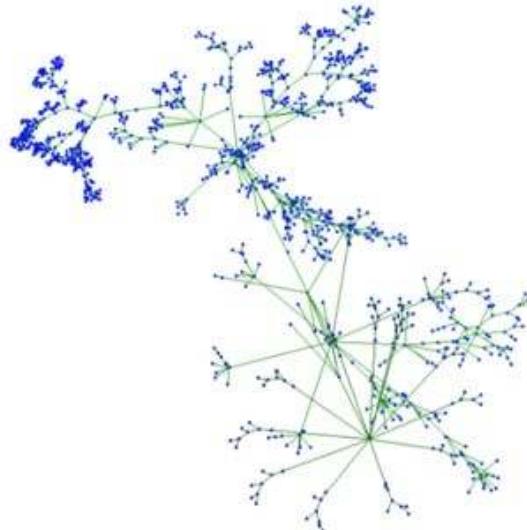
    private class RenamePackageOfGitHttpContainerVisitor extends JavaIsoVisitor<ExecutionContext> {
        @Override
        public J.CompilationUnit visitCompilationUnit(J.CompilationUnit cu, ExecutionContext ctx) {
            J.CompilationUnit compilationUnit = super.visitCompilationUnit(cu, ctx);
            Set<NameTree> nameTreeSet =
compilationUnit.findType("com.github.sparsick.testcontainers.gitserver.GitHttpServerContainer");
            if (!nameTreeSet.isEmpty()) {
                compilationUnit = (J.CompilationUnit) new
ChangeType("com.github.sparsick.testcontainers.gitserver.GitHttpServerContainer",
"com.github.sparsick.testcontainers.gitserver.http.GitHttpServerContainer", true)
                    .getVisitor().visitNonNull(compilationUnit, ctx);
            }
            return compilationUnit;
        }
    }
}
```

Lossless Semantic Tree (LST)

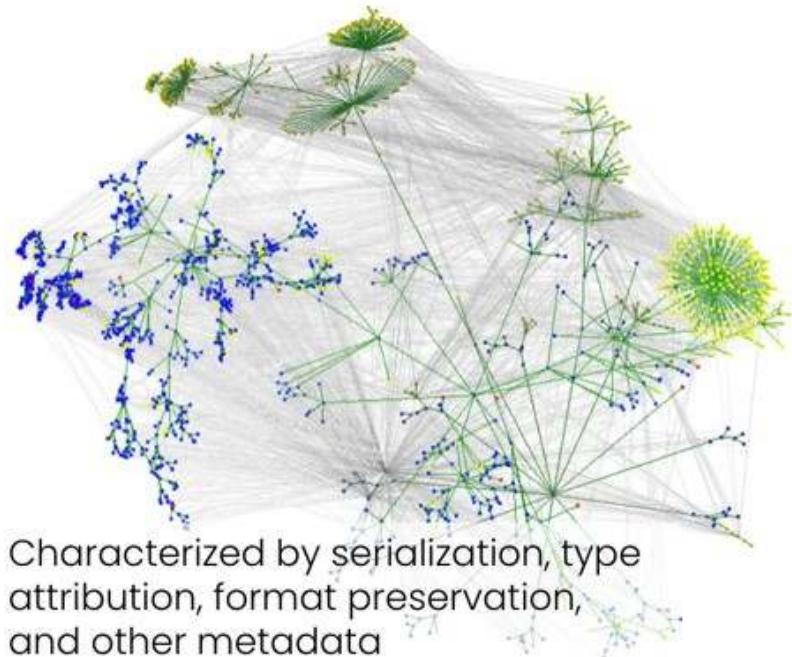
Text-based search

```
void m() {  
    if(c > 2) {  
        x = c;  
    }  
    while(c < 10) {  
        x += p();  
        c++;  
    }  
}
```

Abstract Syntax Tree (AST)



Lossless Semantic Tree (LST)

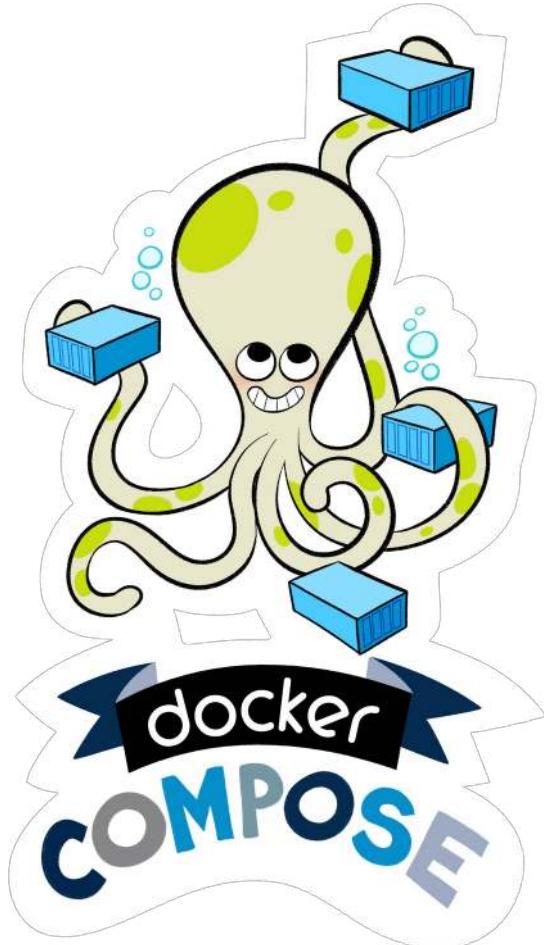


Characterized by serialization, type attribution, format preservation, and other metadata

+ another 600,000 nodes that didn't fit in this graphic



Wie bekommen
wir die
Anwendung lokal
zum Laufen?



Spring Boot Maven Plugin



```
version: "3.9"
services:
  database:
    image: mongo:4.2.21
    restart: always
    ports:
      - 27017:27017
    environment:
      MONGO_INITDB_ROOT_USERNAME: root
      MONGO_INITDB_ROOT_PASSWORD: root123
    volumes:
      - ./local-env/:/dockerentrypoint-initdb.d/
```



```
<plugin>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-maven-plugin</artifactId>
  <configuration>
    <layers>
      <enabled>true</enabled>
    </layers>
    <environmentVariables>
      <MONGODB_ENABLED>true</MONGODB_ENABLED>
      <MONGODB_URI>mongodb://test:test123@localhost/test</MONGODB_URI>
    </environmentVariables>
  </configuration>
</plugin>
```

Demo

Spring Boot Docker Compose Support

```
<!-- Since Spring Boot 3.1 -->
<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-docker-compose</artifactId>
        <optional>true</optional>
    </dependency>
</dependencies>
<build>
    <plugins>
        <plugin>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-maven-plugin</artifactId>
            <configuration>
                <layers>
                    <enabled>true</enabled>
                </layers>
            </configuration>
        </plugin>
    </plugins>
</build>
```



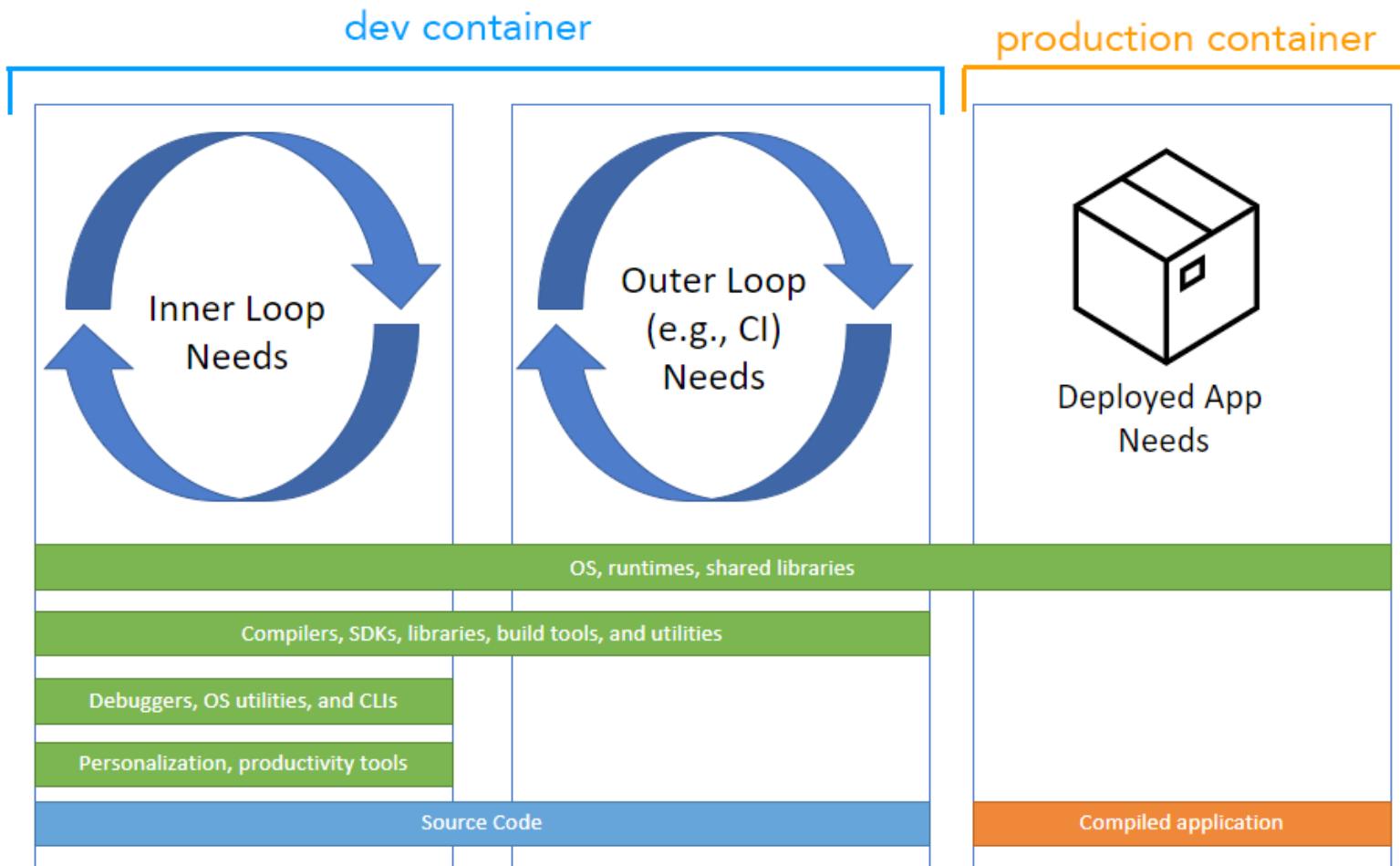
```
# application.properties
# default: compose.yml
spring.docker.compose.file=docker-compose.yml
```

Demo

New Kids on the Block



Idee



Demo



devenv.nix

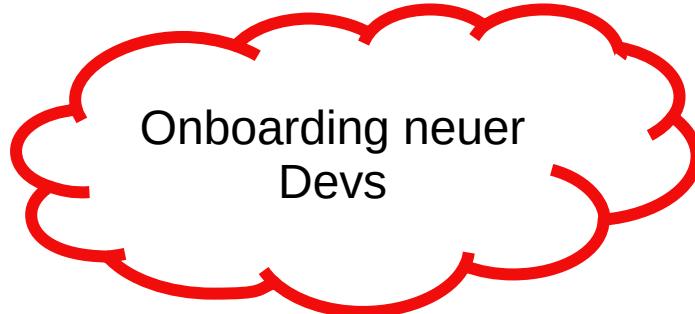
```
{ pkgs, config, ... }: {
  env.GREET = "determinism";

  packages = [
    pkgs.ncdu
  ];

  enterShell = ''
    echo hello ${config.env.GREET}
    ncdlu --version
  '';
}

$ devenv shell
hello determinism
ncdlu 2.3
```

(devenv) \$



Onboarding neuer
Devs



Get started now!

Go on, paste and run the following in a terminal:

```
curl -s "https://get.sdkman.io" | bash
```

The Software Development Kit Manager

SDKMAN! is a tool for managing parallel versions of multiple **Software Development Kits** on most Unix based systems. It provides a convenient Command Line Interface (CLI) and API for installing, switching, removing and listing Candidates. Formerly known as **GVM** the **Groovy enVironment Manager**, it was inspired by the very useful [RVM](#) and [rbenv](#) tools, used at large by the Ruby community.



By Developers, for Developers

Making life easier. No more trawling download pages, extracting archives, messing with `_HOME` and `PATH` environment variables.



Multi-platform

Runs on any UNIX based platforms: macOS, Linux, Cygwin, Solaris and FreeBSD.



Java all the way down

Install Software Development Kits for the JVM such as Java, Groovy, Scala, Kotlin and Ceylon, Ant, Gradle, Grails, Maven, SBT, Spark, Spring Boot, Vert.x and many others also supported.



APIs

New Clients can easily be written by consuming our open Broker REST API. Vendors can publish and announce their own releases through a secure Vendor API.



Lightweight

Written in [bash](#) and only requires [curl](#) and [zip/unzip](#) to be present on your system. Even works with [ZSH](#) too.



Open Source

Backed by a global community of developers.
Licensed under [Apache 2.0](#).



```
$ sdk install java 17-open
$ sdk install maven 3.6.3
$ sdk install maven 3.8.3
$ sdk use java 17-open
$ sdk use maven 3.8.3
$ sdk env init
$ cat .sdkmanrc
# Enable auto-env through the sdkman_auto_env config
# Add key=value pairs of SDKs to use below
java=17-open
maven=3.8.3
```



EditorConfig

[What is EditorConfig?](#)

[Example File](#)

[File Location](#)

[File Format Details](#)

[No Plugin Necessary](#)

[Download a Plugin](#)

[Contributing](#)

[Blog](#)

[Project Page
on GitHub](#)



[Follow Us
on Twitter](#)



[Tweet](#)

What is EditorConfig?

EditorConfig helps maintain consistent coding styles for multiple developers working on the same project across various editors and IDEs. The EditorConfig project consists of a **file format** for defining coding styles and a collection of **text editor plugins** that enable editors to read the file format and adhere to defined styles. EditorConfig files are easily readable and they work nicely with version control systems.

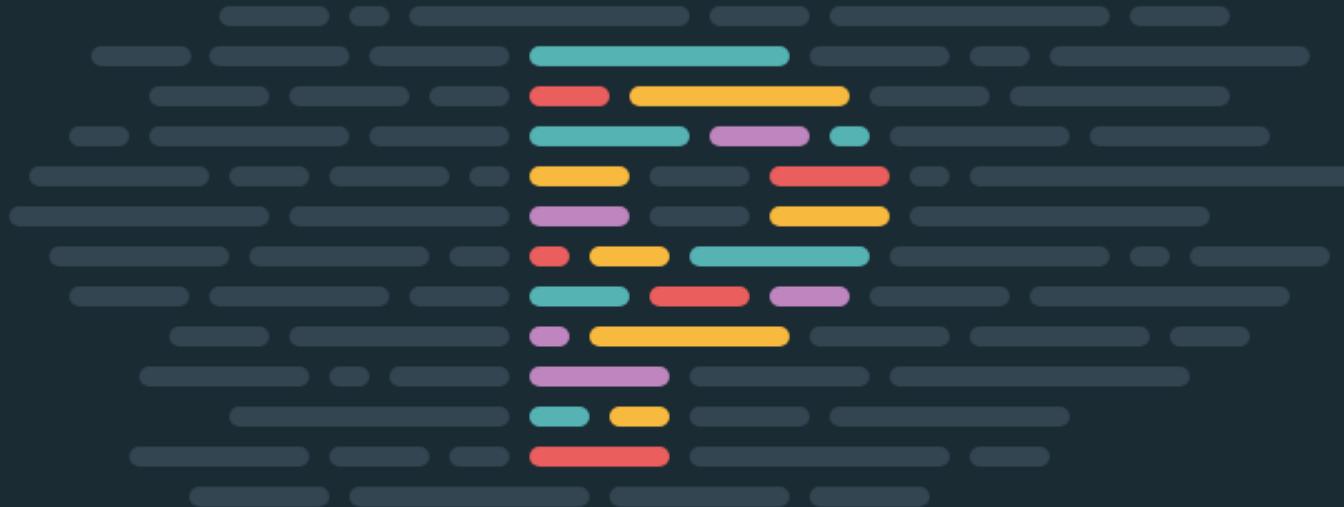
What's an EditorConfig file look like?

(A [formal specification of EditorConfig](#) is also available.)

Example file

Below is an example .editorconfig file setting end-of-line and indentation styles for Python and JavaScript files.

```
# EditorConfig is awesome:  
https://EditorConfig.org  
  
# top-most EditorConfig file  
root = true  
  
# Unix-style newlines with a newline ending  
every file  
[*]  
end_of_line = lf  
insert_final_newline = true
```



TRY IT ONLINE

INSTALL PRETTIER

google-java-format

`google-java-format` is a program that reformats Java source code to comply with [Google Java Style](#).

Fazit

Bei Greenfield Projekte

Automatisierung
(was man automatisieren kann)
von Beginn an

Bei Brownfield Projekte

Guckt was die größten Schmerzen bereiten und versucht diese weg zu automatisieren.

Ähm, und was ist mit AI? 😎

Code maintenance



Second

Fragen?

mail@sandra-parsick.de

@SandraParsick

@sparsick@mastodon.social

<https://github.com/sparsick/maintenance-talk>

Referenzen

- <https://github.com/spring-projects/spring-petclinic>
- <https://spring-petclinic.github.io/>
- <https://www.arc42.de/overview/>
- <https://adr.github.io/>
- <https://www.unusual.vc/post/ai-software-development>

Bildnachweise

- https://pxhere.com/de/photo/970825?utm_content=shareClip&utm_medium=referral&utm_source=pxhere
- https://pxhere.com/de/photo/867723?utm_content=shareClip&utm_medium=referral&utm_source=pxhere
- https://pxhere.com/de/photo/1409158?utm_content=shareClip&utm_medium=referral&utm_source=pxhere
- <https://www.pexels.com/de-de/foto/hausrenovierung-3990359/>
- <https://www.pexels.com/de-de/foto/mann-bucher-schreibtisch-laptop-9572724/>
- <https://unsplash.com/de/fotos/eine-holzbank-mitten-im-wald-LRPcFbTs848>
- <https://pixabay.com/users/tama66-1032521>
- <https://pixabay.com/users/anncapictures-1564471/>
- <https://pixabay.com/users/tumisu-148124>