

# Что в ROM тебе моем?



Андрей  
Когунь

Croc Code



TWITTER @andrei\_kogun

EMAIL a.kogun@gmail.com

# Что в ROM тебе моем?

Ты jar-а оцени объем!



Андрей  
Когунь

Croc Code

TWITTER [@andrei\\_kogun](#)

EMAIL [a.kogun@gmail.com](mailto:a.kogun@gmail.com)





### Project

Maven Project

Gradle Project

### Language

Java  Kotlin

Groovy

### Spring Boot

3.0.0 (SNAPSHOT)

3.0.0 (M3)

2.7.2 (SNAPSHOT)

2.7.1

2.6.10 (SNAPSHOT)

2.6.9

### Project Metadata

Group

Artifact

Name

Description

### Dependencies

**ADD DEPENDENCIES... CTRL + B**

*No dependency selected*



**GENERATE** CTRL + ⌘

**EXPLORE** CTRL + SPACE

**SHARE...**



**Project**

- Maven Project
- Gradle Project

**Language**

- Java  Kotlin
- Groovy

**Dependencies**

**ADD DEPENDENCIES...** CTRL + B

*No dependencies selected*

# Packaging

- Jar
- War

- Java  18
- 17
- 11
- 8

Name demo

Description Demo project for Spring Boot



**GENERATE** CTRL + G

**EXPLORE** CTRL + SPACE

**SHARE...**



### Project

Maven Project

Gradle Project

### Language

Java  Kotlin

Groovy

### Dependencies

**ADD DEPENDENCIES...** CTRL + B

At least one dependency is required.

## Packaging

Jar

War

## Java

18

17

11

8

Name

Description



**GENERATE** CTRL + G

**EXPLORE** CTRL + SPACE

**SHARE...**

sors-smart-2.4.8.pom

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/net/minidev/accessors-smart/2.4.8/accessors-smart-2.4.8.pom> (10 kB at 9.7 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-core/4.5.1/mockito-core-4.5.1.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-core/4.5.1/mockito-core-4.5.1.pom> (2.5 kB at 2.7 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy/1.12.11/byte-buddy-1.12.11.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy/1.12.11/byte-buddy-1.12.11.pom> (17 kB at 14 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy-parent/1.12.11/byte-buddy-parent-1.12.11.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy-parent/1.12.11/byte-buddy-parent-1.12.11.pom>

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy-agent/1.12.11/byte-buddy-agent-1.12.11.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy-agent/1.12.11/byte-buddy-agent-1.12.11.pom> (11 kB at 9.3 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-junit-jupiter/4.5.1/mockito-junit-jupiter-4.5.1.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-junit-jupiter/4.5.1/mockito-junit-jupiter-4.5.1.pom> (2.3 kB at 2.5 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/springframework/spring-test/5.3.21/spring-test-5.3.21.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/org/springframework/spring-test/5.3.21/spring-test-5.3.21.pom> (2.1 kB at 1.9 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/xmlunit/xmlunit-core/2.9.0/xmlunit-core-2.9.0.pom>

```
>mvn verify
```

sors-smart-2.4.8.pom

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/net/minidev/accessors-smart/2.4.8/accessors-smart-2.4.8.pom> (10 kB at 9.7 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-core/4.5.1/mockito-core-4.5.1.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-core/4.5.1/mockito-core-4.5.1.pom> (2.5 kB at 2.7 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy/1.12.11/byte-buddy-1.12.11.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy/1.12.11/byte-buddy-1.12.11.pom> (17 kB at 14 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy-parent/1.12.11/byte-buddy-parent-1.12.11.pom>

[INFO]

[INFO] BUILD SUCCESS

[INFO]

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/net/bytebuddy/byte-buddy-agent/1.12.11/byte-buddy-agent-1.12.11.pom> (11 kB at 9.3 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-junit-jupiter/4.5.1/mockito-junit-jupiter-4.5.1.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/org/mockito/mockito-junit-jupiter/4.5.1/mockito-junit-jupiter-4.5.1.pom> (2.3 kB at 2.5 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/springframework/spring-test/5.3.21/spring-test-5.3.21.pom>

Downloaded from repo: <https://repo.croc.ru/artifactory/aspur/org/springframework/spring-test/5.3.21/spring-test-5.3.21.pom> (2.1 kB at 1.9 kB/s)

Downloading from repo: <https://repo.croc.ru/artifactory/aspur/org/xmlunit/xmlunit-core/2.9.0/xmlunit-core-2.9.0.pom>

Все было хорошо пока не появился





Все было хорошо пока не появился



**BUILD FAILURE**

## # КАК ВЫГЛЯДИТ ПРОБЛЕМА?

- Я что-то сделал и получил неожиданный результат (оно сломалось)

## # КАК ВЫГЛЯДИТ ПРОБЛЕМА?

- Я что-то сделал и получил неожиданный результат (оно сломалось)

Ты: `java.version 8 -> 11` , оно: `NullPointerException`

```
at org.apache.maven.lifecycle.internal.MojoExecutor.execute (MojoExecutor.java:213)
at org.apache.maven.lifecycle.internal.MojoExecutor.execute (MojoExecutor.java:154)
at org.apache.maven.lifecycle.internal.MojoExecutor.execute (MojoExecutor.java:146)
at org.apache.maven.lifecycle.internal.LifecycleModuleBuilder.buildProject (LifecycleModuleBuilder.java:117)
at org.apache.maven.lifecycle.internal.LifecycleModuleBuilder.buildProject (LifecycleModuleBuilder.java:81)
at org.apache.maven.lifecycle.internal.builder.singlethreaded.SingleThreadedBuilder.build
(SingleThreadedBuilder.java:56)
at org.apache.maven.lifecycle.internal.LifecycleStarter.execute (LifecycleStarter.java:128)
at org.apache.maven.DefaultMaven.doExecute (DefaultMaven.java:305)
at org.apache.maven.DefaultMaven.doExecute (DefaultMaven.java:192)
at org.apache.maven.DefaultMaven.execute (DefaultMaven.java:105)
```

## # КАК ВЫГЛЯДИТ ПРОБЛЕМА?

- Я что-то сделал и получил неожиданный результат (оно сломалось)

Ты: java.version 8 -> 11 , оно: NullPointerException

<https://bugs.openjdk.org/browse/JDK-8216202>

A vibrant, cartoon-style illustration of a tropical beach. In the center, a tall, colorful character stands on the sand. The character has a pink, wide-mouthed head with two red antennae-like protrusions. Its neck is a vertical stack of various colored blocks: grey, red, yellow, grey, red, yellow, and pink. The body is orange with pink joints. The background features a bright blue sky with white clouds, a clear blue ocean, and several green palm trees on a sandy beach. A small seashell is visible on the sand in the lower left.

Перепишем на  
Gradle?

## # КАК ВЫГЛЯДИТ ПРОБЛЕМА?

- Я что-то сделал и получил неожиданный результат (оно сломалось)
- Я что-то сделал, а оно не работает



Перепишем на  
Gradle?

## # КАК ВЫГЛЯДИТ ПРОБЛЕМА?

- Я что-то сделал и получил неожиданный результат (оно сломалось)
- Я что-то сделал, а оно не работает
- Я ничего не делал, а оно сломалось





Перепишем на  
Gradle?

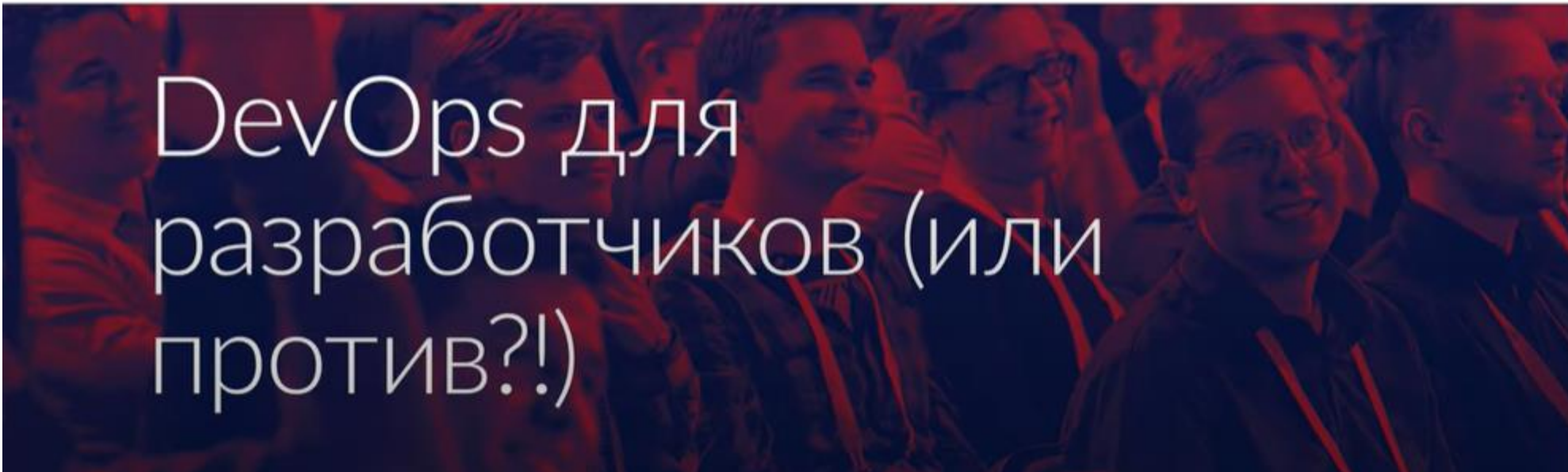
## # ЧТО ДЕЛАТЬ?

- Я главный по мавену, сам все починю тут
- Починит падаван, у нас любой с таким справится
- Починит DevOps / я DevOps, я починю

# КАЖЕТСЯ, БАРУХ БЫЛ ПРАВ

Joker<?>  
2019

Барух Садогурский  
JFrog

DevOps для  
разработчиков (или  
против?!) 



<https://www.youtube.com/watch?v=z-GEbyXKetE>



Перепишем  
РОМ!



Перепишем  
РОМ!

## # О СЕБЕ

### **Андрей Когунь — программист**

- 15+ лет эксплуатирую Maven (или он меня)



## # НАЧИНАЕМ РАЗБИРАТЬСЯ

- Maven – это Java программа и все плагины - тоже
- Можно подебажить, а можно:
- **`mvn -X -e`**

# ДЛЯ НАЧАЛА НАДО РАЗОБРАТЬСЯ С ЗАВИСИМОСТЯМИ

- <https://habr.com/ru/company/jugru/blog/191246/>



jbaruch 25 августа 2013 в 23:47

## Разрешение конфликтов в транзитивных зависимостях — Хороший, Плохой, Злой

Блог компании JUG Ru Group, Java\*



+40



35K



97





## # ДЛЯ НАЧАЛА НАДО РАЗОБРАТЬСЯ С ЗАВИСИМОСТЯМИ

- <https://habr.com/ru/company/jugru/blog/191246/>



jbaruch 25 августа 2013 в 23:47

Разрешение конфликтов в транзитивных зависимостях —  
Хороший, Плохой, Злой

Блог компании JUG Ru Group, Java\*

◆ +40 👁 35K 📄 97 ➡

- **mvn help:effective-pom**
- **mvn dependency:tree**
- **mvn dependency:display-ancestors**

# ДЛЯ НАЧАЛА НАДО РАЗОБРАТЬСЯ С ЗАВИСИМОСТЯМИ

- `properties` – добавляй новые в конец
- `dependencyManagement` – добавляй новые в начало
- `dependencyManagement` – используй `scope import`
- `dependency` – лучше без указания версий совсем

## # ЗАВИСИМОСТИ И ВЕРСИИ БЫВАЮТ И У ПЛАГИНОВ

- **mvn dependency:resolve-plugins**
- **mvn dependency:tree -f some-maven-plugin-x.x.x.pom**

## # ОТКРЫТЫЕ ВЕРСИИ – ПРЯМО ЗЛО

- скачает обязательно все
- обратная совместимость – это сложно

# ЛУЧШЕ ПРЯМО ВСЕ ВЕРСИИ ЗАФИКСИРОВАТЬ

- `mvnw` – как `gradle wrapper`, только `maven`

## # ЛУЧШЕ ПРЯМО ВСЕ ВЕРСИИ ЗАФИКСИРОВАТЬ

- mvnw – как gradle wrapper, только maven
- maven-frontend-plugin

```
<execution>
  <id>install node and npm</id>
  <goals>
    <goal>install-node-and-npm</goal>
  </goals>
  <configuration>
    <nodeVersion>${frontend-maven-plugin.nodeVersion}</nodeVersion>
    <npmVersion>${frontend-maven-plugin.npmVersion}</npmVersion>
  </configuration>
</execution>
```

# ОК, MAVEN, ВЫПОЛНИ СНАЧАЛА ЭТУ ЗАДАЧУ, А ПОТОМ ЭТУ

- executions
- phase
- goals

```
<plugin>
  <groupId>org.codehaus.mojo</groupId>
  <artifactId>build-helper-maven-plugin</artifactId>
  <executions>
    <execution>
      <id>build-wc-props</id>
      <phase>initialize</phase>
      <goals>
        <goal>bsh-property</goal>
      </goals>
      ...
    </execution>
    <execution>
      <id>parse-version</id>
      <phase>validate</phase>
      <goals>
        <goal>parse-version</goal>
      </goals>
      ...
    </execution>
  </executions>
</plugin>
```

# НАГЕНЕРИЛ - УБЕРИ ЗА СОБОЙ

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-clean-plugin</artifactId>
  <configuration>
    <filesets>
      <fileset>
        <directory>.</directory>
        <includes>
          <include>src/main/generated/**</include>
        </includes>
        <followSymlinks>>false</followSymlinks>
      </fileset>
    </filesets>
  </configuration>
```



## # ФИЛЬТРУЙ РЕСУРС

```
<resources>
  <resource>
    <directory>src/main/resources</directory>
    <excludes>
      <exclude>**/*.md</exclude>
    </excludes>
  </resource>
</resources>
```

## # СТАЛО ПЛОХО – ВСПОМИНАЕМ О РОДИТЕЛЯХ

- Когда написал 100500 строк xml-я и загрузил
- Уноси все в pluginManagement
- И в parent POM

- Профили!

```
<profiles>
  <profile>
    <id>setAlternativeBuildDirectory</id>
    <activation>
      <property>
        <name>alternative.build.directory</name>
      </property>
    </activation>
    <build>
      <directory>${alternative.build.directory}</directory>
    </build>
  </profile>
</profiles>
```

- Профили!

Один основной, остальные активируются по условию, либо явно: **`mvn -Pdev`**

- **`mvn help:active-profiles`**

## # ОПТИМИЗИРУЕМ / УСКОРЯЕМ

- Профили!
- -nsu

## # ОПТИМИЗИРУЕМ / УСКОРЯЕМ

- Профили!
- -nsu
- -Dskip\*

## # ОТКЛЮЧАЙ ЭТО

```
<execution>
  <id>[wc] grunt build prod</id>
  <goals>
    <goal>grunt</goal>
  </goals>
  <phase>process-classes</phase>
  <configuration>
    <skip>${buildWCInDev}</skip>
    <arguments>build:prod --target=${project.build.directory}</arguments>
  </configuration>
</execution>
```

## # ОТКЛЮЧАЙ ЭТО

```
<execution>
  <id>[wc] grunt build</id>
  <goals>
    <goal>grunt</goal>
  </goals>
  <phase>process-classes</phase>
  <configuration>
    <skip>${buildWCInProd}</skip>
    <arguments>build --target=${project.build.directory}</arguments>
  </configuration>
</execution>
```



## # ОТКЛЮЧАЙ ЭТО

```
<execution>
  <id>[wc] install npm dependencies</id>
  <goals>
    <goal>npm</goal>
  </goals>
  <configuration>
    <workingDirectory>wc</workingDirectory>
    <arguments>install --omit=optional</arguments>
    <skip>${backEndOnly}</skip>
  </configuration>
</execution>
```

## # ОТКЛЮЧАЙ ЭТО - ИЗИ

```
<goals>
  <goal>bsh-property</goal>
</goals>
<configuration>
  <source>
    buildWCInProdValue = session.getUserProperties().getProperty("buildWCInProd");
    buildWCInProdValue = buildWCInProdValue == null ? false : Boolean.parseBoolean(buildWCInProdValue);
    buildWCInDevValue = session.getUserProperties().getProperty("buildWCInDev");
    buildWCInDevValue = buildWCInDevValue == null ? true : Boolean.parseBoolean(buildWCInDevValue);
    backEndOnlyValue = session.getUserProperties().getProperty("backEndOnly");
    backEndOnlyValue = backEndOnlyValue == null ? false : Boolean.parseBoolean(backEndOnlyValue);
    buildWCInProd = buildWCInProdValue || backEndOnlyValue;
    buildWCInDev = buildWCInDevValue || backEndOnlyValue;
    print("backEndOnly: " + backEndOnlyValue + " -&gt; " + "buildWCInDev: " + buildWCInDev +
    ", buildWCInProd: " + buildWCInProd);
  </source>
  <properties>
    <property>buildWCInProd</property>
    <property>buildWCInDev</property>
  </properties>
</configuration>
```

# НЕ НРАВИТСЯ ЭТОТ ПЛАГИН – СДЕЛАЙ СВОЙ!

- Массовые операции
- Генерация нового кода
- Можно вообще собрать отдельный тул

# НЕ НРАВИТСЯ ЭТОТ ПЛАГИН – СДЕЛАЙ СВОЙ!  
НАПРИМЕР

- Массовые операции
- [jxfw/BirtReportModifier.java at master · croc-code/jxfw · GitHub](#)
- Генерация нового кода
- [jxfw/wcgen-maven-plugin at master · croc-code/jxfw · GitHub](#)
- Можно вообще собрать отдельный тул
- [jxfw/jxfw-cli at master · croc-code/jxfw · GitHub](#)

## # ДЛЯ ЧЕГО НАМ ARCHETYPE?

- Проверка не тестами, но полноценным запуском

## # ГДЕ ЖИВУТ АРТЕФАКТЫ?

- .m2
- [repo.private.com](https://repo.private.com)
- Maven Central / [maven.pkg.github.com](https://maven.pkg.github.com)

# И КАК ТУДА ПОПАСТЬ?

- `mvn -s .m2/settings.xml`

```
<mirrors>
  <mirror>
    <mirrorOf>*</mirrorOf>
    <id>repo</id>
    <url>https://repo.croc.ru/artifactory/ctp.jxfw</url>
  </mirror>
</mirrors>
```

# И КАК ТУДА ПОПАСТЬ?

- `mvn -s .m2/settings.xml`

```
<servers>
  <server>
    <id>archetype</id>
    <username>akogun</username>
    <password>password</password>
  </server>
  <server>
    <id>eclipse-repo</id>
    <username>akogun</username>
    <password>password</password>
  </server>
  <server>
    <id>repo</id>
    <username>akogun</username>
    <password>password</password>
  </server>
</servers>
```



# И КАК ТУДА ПОПАСТЬ?

- `mvn -s .m2/settings.xml`

```
<repository>
  ...
  <snapshots>
    <enabled>true</enabled>
  </snapshots>
</repository>
...
<pluginRepository>
  ...
  <snapshots>
    <enabled>true</enabled>
  </snapshots>
</pluginRepository>
```

## # ЧТО МЫ УЗНАЛИ

- Многомодульные сборки – make sense
- Локальный репозиторий – твой бро
- Maven – жить можно\*
- \*если научиться пользоваться

# Что в ROM тебе моем?

Q & A



Андрей  
Когунь

Croc Code



TWITTER @andrei\_kogun

EMAIL a.kogun@gmail.com