All you need to know about Maven 4

About me

Matthias Bünger

- 🧑‍💻 Software engineer
- 🧑‍🔧 Maven committer
- @bukama@mastodon.social
- 🦋 @bukama.bsky.social

What Maven version are you using?

Agenda

- Motivation
- Major changes Maven 4
- Improved user experience
- Migrate projects to Maven 4

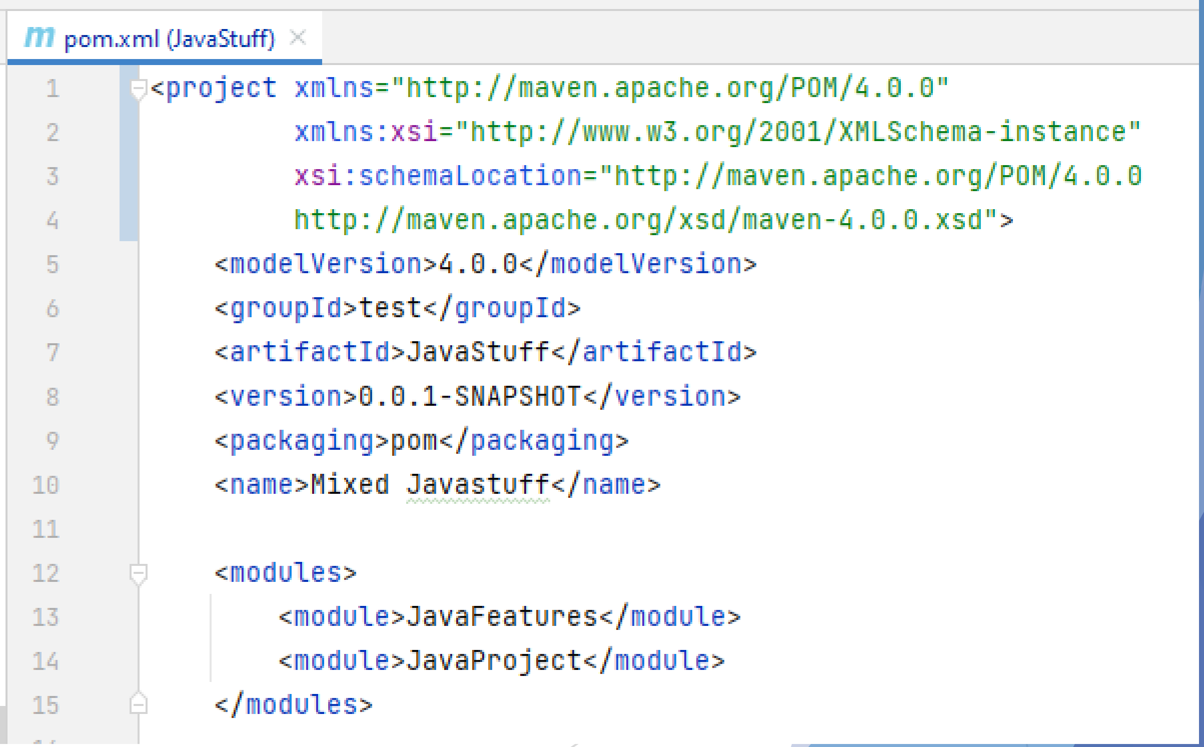
Motivation

Current state of Maven

- 20+ years old (Maven 1.0 July 2004)
- Consists of ~100 artifacts (plugins + components)
  - Not counting ~6.000 community plugins
- Maven 3.0 released 2010-10-08
  - Current version 3.9.11 from 2025-07-12

Maven`s POM (Model version 4.0.0)

- XML-Schema compatible since Maven 2 (2005)
- Build and usages information
  - Modules
  - Dependencies
  - Plugins (and their configuration)
  - Repositories, meta-data, etc.
- Deployed to Maven central

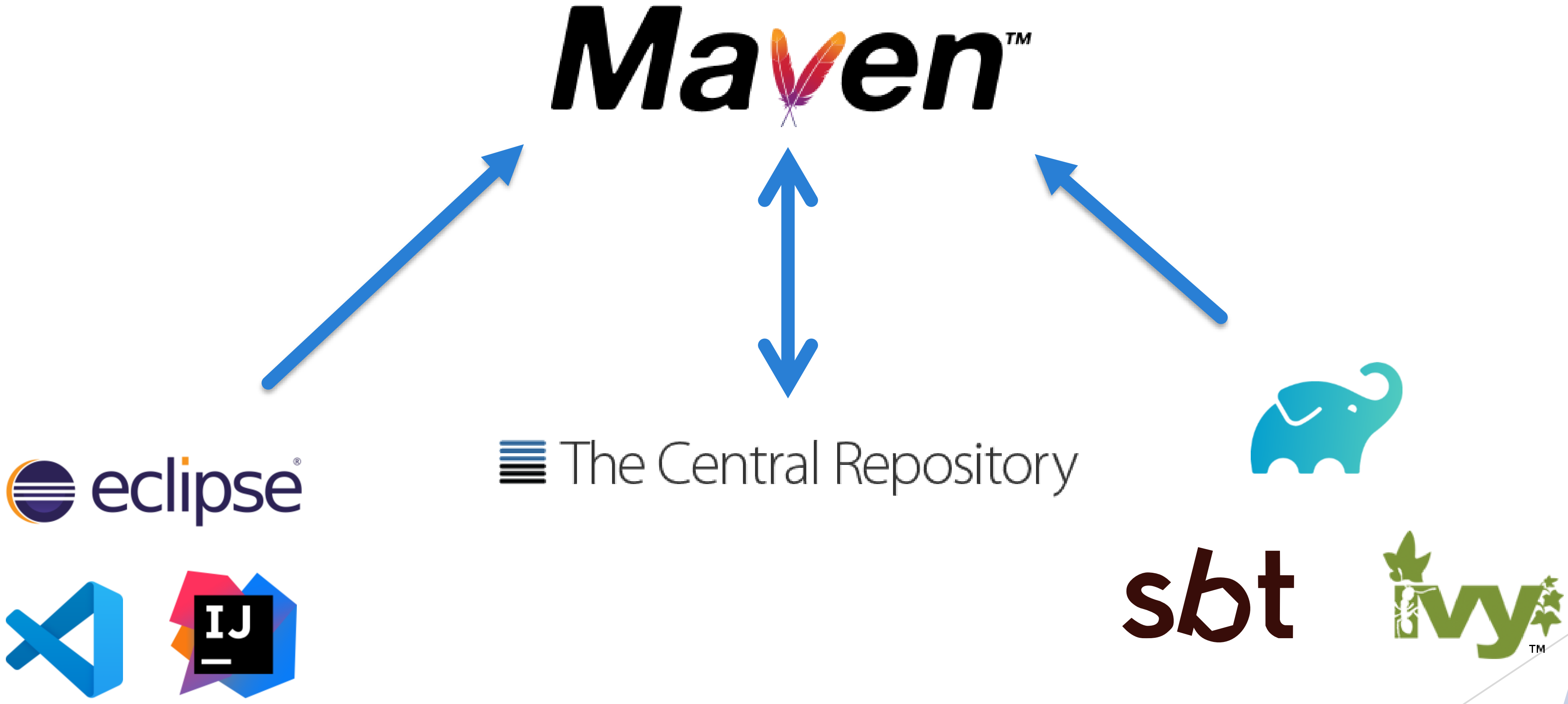


```xml
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
         http://maven.apache.org/xsd/maven-4.0.0.xsd">
    <modelVersion>4.0.0</modelVersion>
    <groupId>test</groupId>
    <artifactId>JavaStuff</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <packaging>pom</packaging>
    <name>Mixed Javastuff</name>

    <modules>
        <module>JavaFeatures</module>
        <module>JavaProject</module>
    </modules>
```

Maven wants to improve

- Changing the POM
- Get rid of some very old, redundant and problematic (e.g. mutable) code and dependencies of Maven 3 and plugin API with unclear API boundaries
  - Many plugins still use Maven 2 (and older) methods
- Make use of more modern Java versions

More than a tool



Maven™

eclipse®

The Central Repository

sbt

ivy™

> “With the Maven build schema preserved in amber, we can’t evolve much: we’ll stay forever with Maven 3 minor releases, unable to implement improvements that we imagine will require seriously updating the POM schema…”

Hervé Boutemy

https://www.javaadvent.com/2021/12/from-maven-3-to-maven-5.html

Image created with Freepik

Building bridges

- Future Maven versions could have different POM schemas

- Maven 4 will clear the way

Image: https://unsplash.com/photos/brown-wooden-bridge-in-forest-during-daytime-R1NnjcY2aDI

Major changes in Maven 4

Disclaimer: Still RC (4) – things may change – but they got quite stable now!

Required Java Version

- In March 2024: Vote passed for lifting Maven 4 to $\boxed{1}$ $\boxed{7}$
  - First Java 17 version 4.0.0-alpha 13 (2024-03-06)

Overview: https://maven.apache.org/docs/history.html

Required Java Version

TO RUN MAVEN ⚠

(not your app)

Code updates

- Continously updated to Java 8+
  - Use new Java features / improvements
  - First Java 8 version: 3.9.0 (2023-01-31)

- Updated default version of Maven plugins and dependencies

API cleanup

- Remove old Maven 1/2 [Plugin] APIs
  - Warnings in 3.9 / Failures in 4.0
  - Now: Immutable (plugin) model
- "Plexus" dependency resolver removed
  - Deprecated since Maven 3.2 (2010)

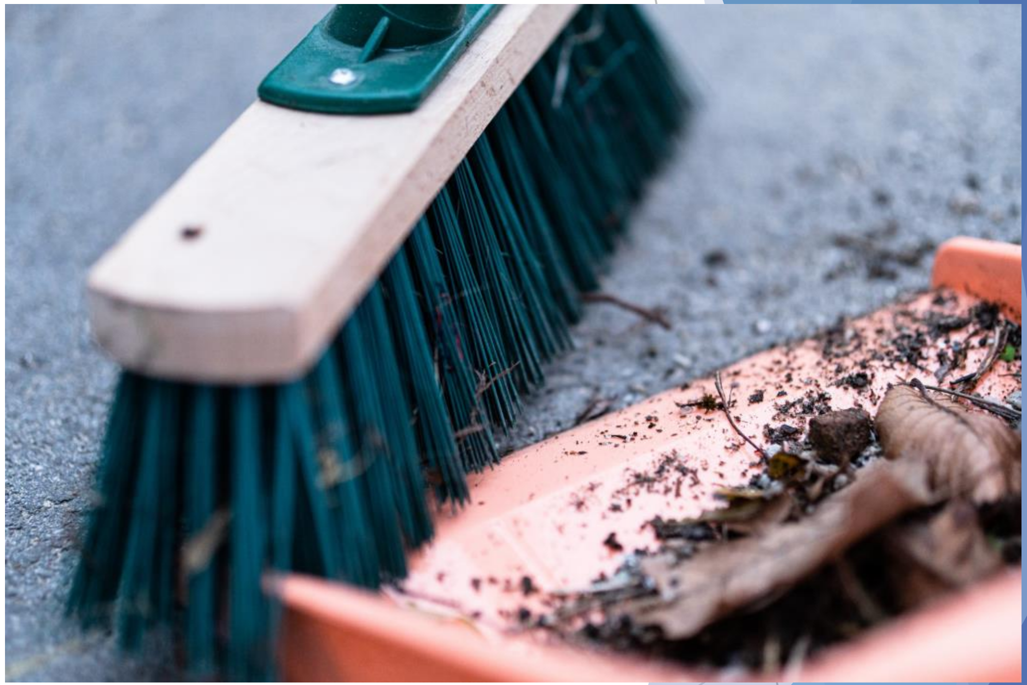
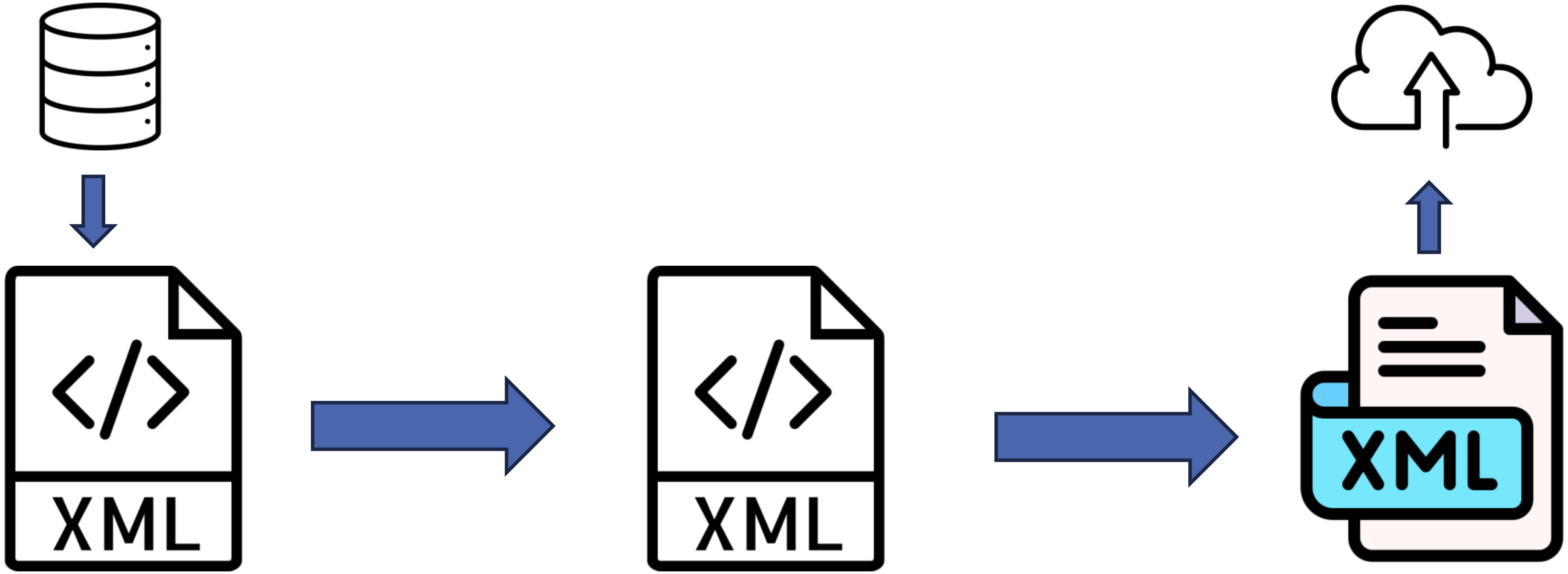


Image: https://unsplash.com/photos/green-and-white-brush-on-white-wall-5Mx-o-H8sqY

Introduction of Consumer POM

- Based on POM v4
  - Consumer POM with POM v4.0
  - Build POM with new POM v4.1
- Consumer POM is generated during build
- Does not contain all information of Build POM

From Build POM to Consumer POM



Build-POM

(in repository, 4.1.0)

Enriched POM

(internal during build)

Consumer POM

(Reduced to 4.0.0)

Build and Consumer POM

	Build POM	Consumer POM *
POM version	4.1.0	4.0.0
Dependencies - Full parent / subproject - 3rd parties	 ❌ ✅	 ❌ ✅
Properties	✅	❌
Plugin configuration	✅	❌
Repository information	✅	✅
Profiles	✅	✅
Project information / Environment settings	✅	✅
Deployment to Remote	✅	✅

* Not for type „pom“

Modules are now called „subprojects“

- Renamed to align better with “(Maven-)Project”
- Requires POM version 4.1.0
- Old “modules” tag is deprecated, but still available

New packaging type: bom

- New feature in Maven 4 for Bill of Materials
- Requires POM version 4.1.0
- Automatically handles projects versions when creating a BOM file

- Example by Karl Heinz Marbaise (at IntelliJ IDEA Conf 2024)
  - https://www.youtube.com/watch?v=ZD_YxTmQ16Q&t=16710s

Alternate POM syntaxes

- Model Parser SPI allows custom syntaxes
  - https://issues.apache.org/jira/browse/MNG-7836

- „Mason“ extension offers YAML, JSON5, TOML, HOCON
  - https://github.com/maveniverse/mason
  - By Tamas Cservenak (Maven PMC) and Guillaume Nodet (Maven PMC)

Improved user experience

Minor changes

Reactor improvements

- Resuming in projects with subprojects **much** better
  - “—also-make” together with “—resume-from”
  - Also detecting modules build successful when building in parallel
- Aware of subfolder build
- Better support for multiple subprojects which are multiple levels deep

Summary: https://maarten.mulders.it/2020/11/whats-new-in-maven-4/

A project-local repository

- Only contains the produced artifacts of the project, not the consumed ones
- Prevents „polluting“ the user repository before releasing
  - Less side effects caused by „not released“ (installed) artifacts

MVN
CLEAN
INSTALL

Image: https://github.com/aalmiray/mvn-clean-install

Choose wisely

- „mvn verify“ is enough for **regular** builds
- There **are** cases to use „install“
- „Clean“ removes everything
  - No caching possible
- About repository handling also see
  - „Never say never“ by Tamas Cservenak (Maven PMC)

Image: https://github.com/aalmiray/mvn-clean-install

Clear definition of project root / properties

Name	Scope	Definition	Always?
`project.rootDirectory`	project	`.mvn`-folder or `root`-attribute in pom	No
`session.topDirectory`	session	current directory or `--file` argument	Yes
`session.rootDirectory`	session	`.mvn`-folder or `root`-attribute in pom for the topDirectory project	No

Summary:

https://issues.apache.org/jira/browse/MNG-7038

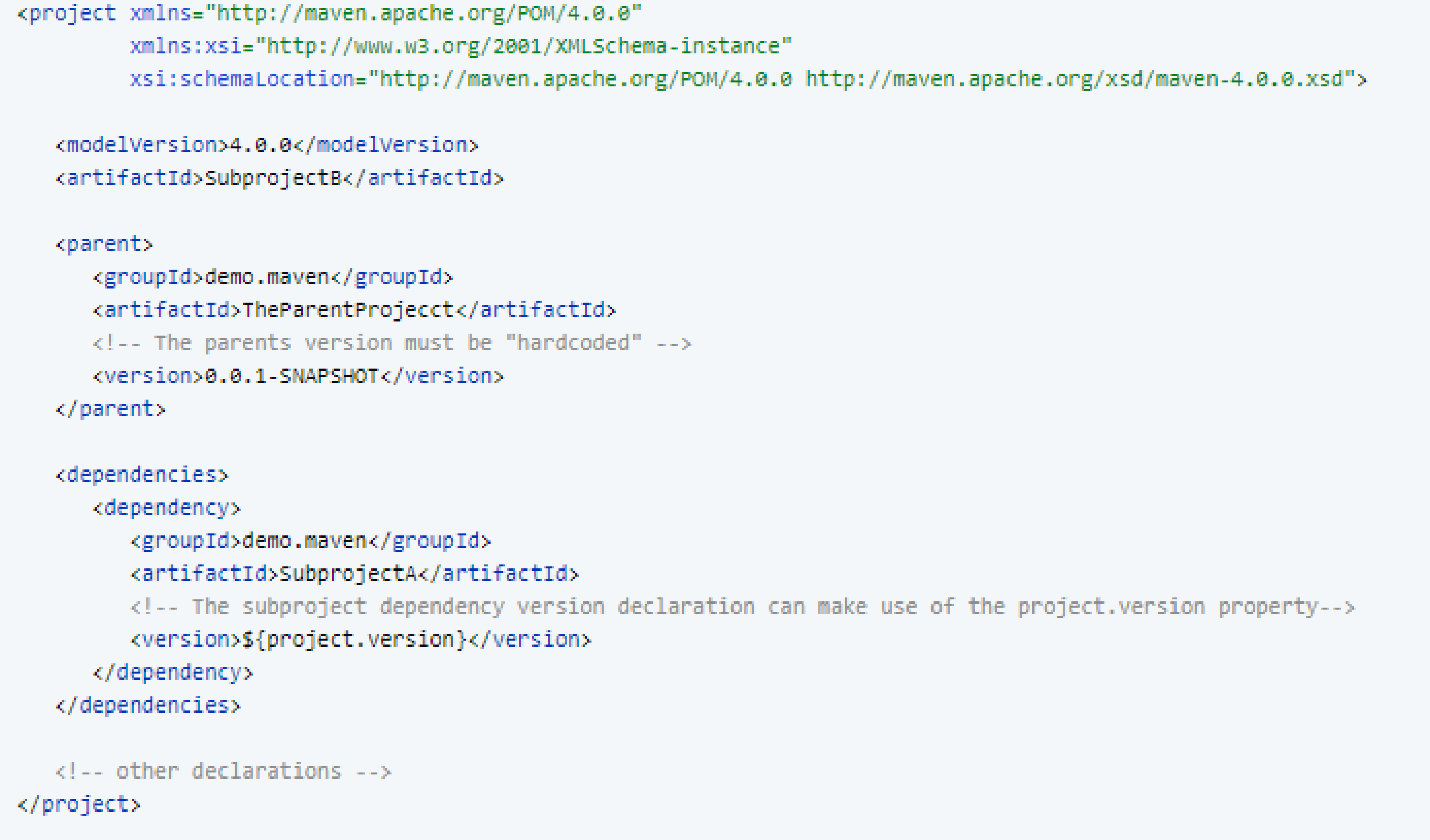
https://github.com/apache/maven/pull/1061

Several interal properties are deprecated or removed!

Improvements for projects with subprojects

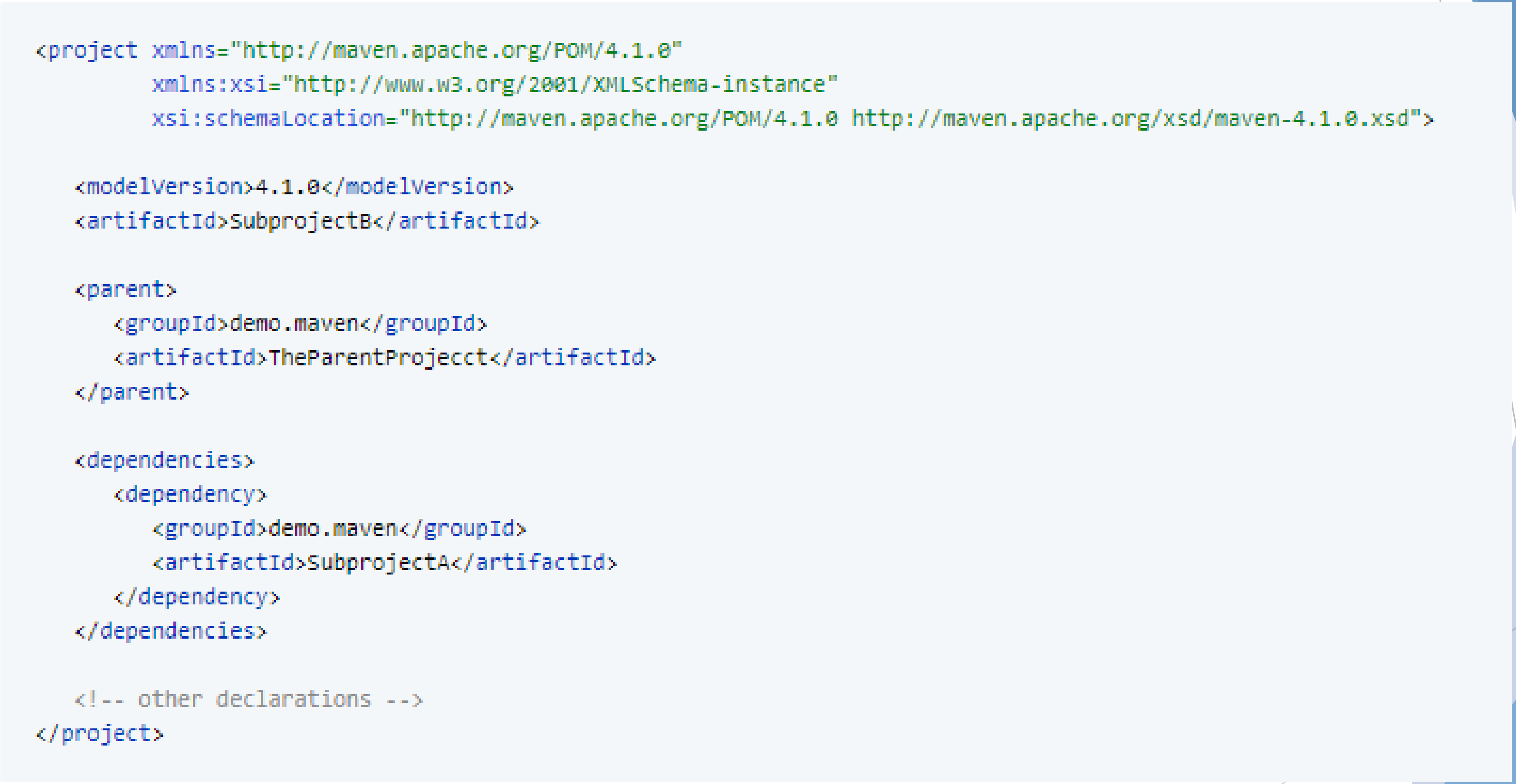
- Consistent timestamps for all subprojects builds in a project
- Install / Deploy all or none
  - Default value of “installAtEnd” / “deployAtEnd” changed to “true”
- Automatic versioning 🥰
  - No parent version needed anymore
  - No versions for project-own subprojects needed anymore

Subproject declaration (Maven 3)



```
<project xmlns="http://maven.apache.org/POM/4.0.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

   <modelVersion>4.0.0</modelVersion>
   <artifactId>SubprojectB</artifactId>

   <parent>
      <groupId>demo.maven</groupId>
      <artifactId>TheParentProjecct</artifactId>
      <!-- The parents version must be "hardcoded" -->
      <version>0.0.1-SNAPSHOT</version>
   </parent>

   <dependencies>
      <dependency>
         <groupId>demo.maven</groupId>
         <artifactId>SubprojectA</artifactId>
         <!-- The subproject dependency version declaration can make use of the project.version property-->
         <version>${project.version}</version>
      </dependency>
   </dependencies>

   <!-- other declarations -->
</project>
```

Subproject declaration (Maven 4)



```xml
<project xmlns="http://maven.apache.org/POM/4.1.0"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://maven.apache.org/POM/4.1.0 http://maven.apache.org/xsd/maven-4.1.0.xsd">

   <modelVersion>4.1.0</modelVersion>
   <artifactId>SubprojectB</artifactId>

   <parent>
      <groupId>demo.maven</groupId>
      <artifactId>TheParentProjecct</artifactId>
   </parent>

   <dependencies>
      <dependency>
         <groupId>demo.maven</groupId>
         <artifactId>SubprojectA</artifactId>
      </dependency>
   </dependencies>

   <!-- other declarations -->
</project>
```

Build improvements

- CI-friendly build variables (e.g. “${revision}”)
  - First approach in 3.5.0 (back then in combination with flatten-plugin)
  - https://blog.soebes.io/posts/2024/03/2024-03-31-maven-4-part-i/
- „Fail on severity“-Parameter
  - Breaking builds on messages with given severity, e.g. „-fos WARN“
    - Default in Maven 3: Errors do not break build, only exceptions do!
  - Improves project maintenance
  - Some of the current warnings will break your build in Maven 4!

Changed lifecycle

- Lifecycle is now a tree of phases not a list
    - Allows execution of phases without others, e.g.„deploy“ without „install“
    - Some phases still have constraints, e.g. „compile“ after „sources“
    - Default lifecycle stays the same
- Each phase now has a „before:“ and a „after:“ phase
- New „all“ and „each“ phases on project level

Optional profiles

- Using a nonexistent profile (still) breaks the build
- Using an optional nonexistent profile does not
    - Logs [INFO] instead (at start and end)
- Definition by using „?“, e.g. „-P?release

- https://issues.apache.org/jira/browse/MNG-7051

Recap: Maven 4

- Minimum Java 17 with lots of code / API cleanup
- Separation of Build & Consumer POM
- A lot of UX improvements
  - Reactor
  - New lifecycle phases
  - Automatic versioning
  - Official properties for project root
- And more…

https://maven.apache.org/whatsnewinmaven4.html

Migrate projects to Maven 4

1. Successfull build with Maven 3.9 and up-to-date-3.x plugins
2. Prepare build environment for Java 17
3. Use Maven 4 (and maybe fix errors)
4. Optional:
   1. Use final 4.x-plugins
   2. Use Maven 4 features

- Full guide at: https://maven.apache.org/guides/mini/guide-migration-to-mvn4.html

(At best) No manual work needed!

- Built-in migration tool available!
- Can update to
  - Maven 3 compatible (Model 4.0.0)
  - Maven 4 only (Model 4.1.0)

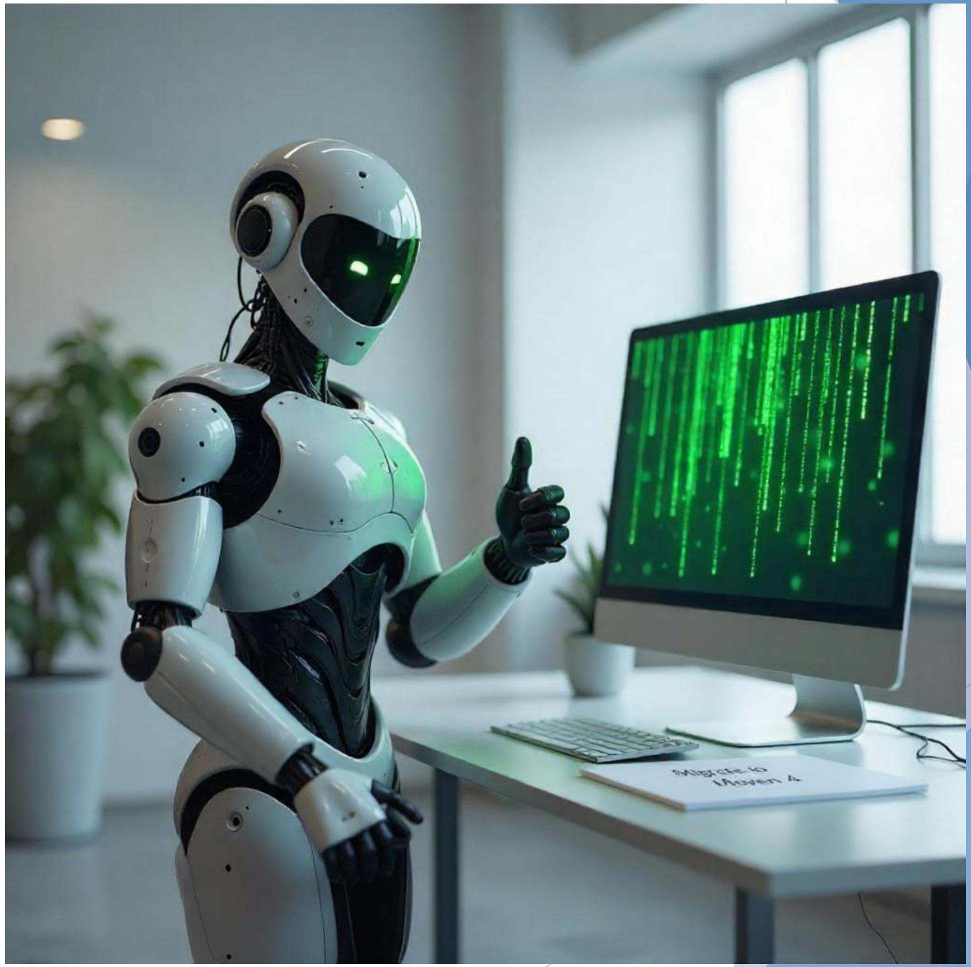


Image created with Freepik

What can/should one do?

- Update to latest Maven 3.9.x
- Prepare build environment to run Maven on Java 17
- Test out Maven 4 Release Candidate
- Give feedback to Apache Maven team

- ✉ users@maven.apache.org
- 🌍 https://maven.apache.org/

Image: https://pixabay.com/de/illustrations/feedback-wei%C3%9Fe-m%C3%A4nnchen-3d-model-1889007/