

# Apache Maven Version 4.0.0 - alpha X



**Maven**<sup>TM</sup>



**Maven**

Blog: [blog.soebes.io](http://blog.soebes.io)  
Twitter: @khmarbaise  
Email: [maven4@soebes.de](mailto:maven4@soebes.de)  
Dipl.Ing.(FH) Karl Heinz Marbaise

**SO=BeS**  
SoftwareEntwicklung ▶ Beratung ▶ Schulung

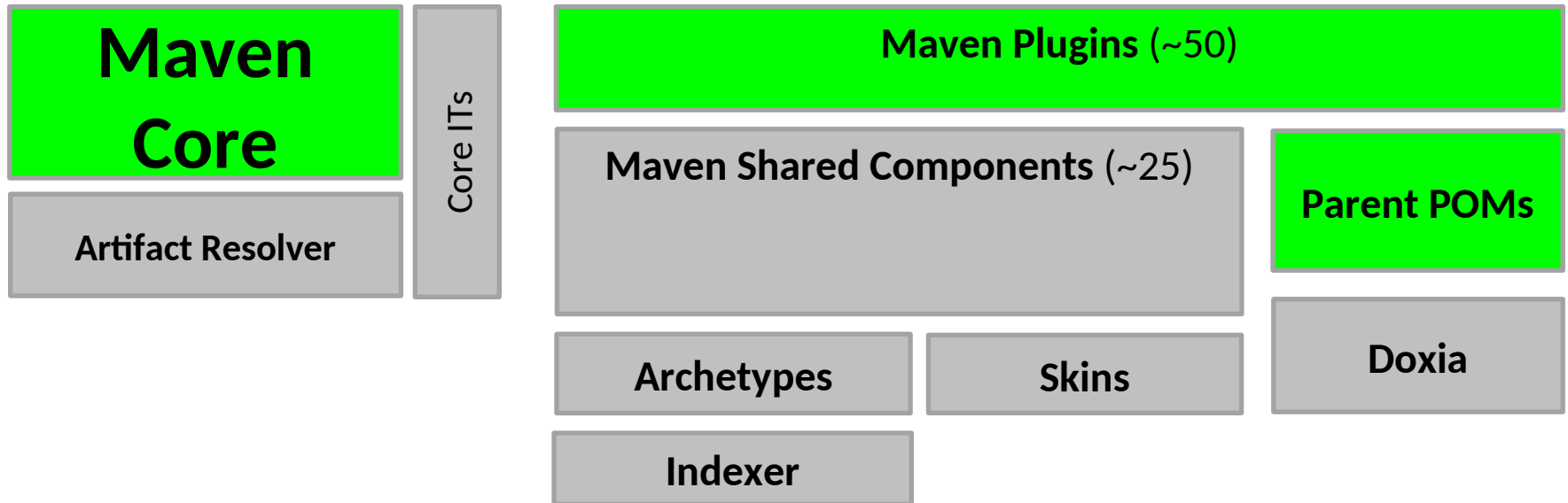
# About me

- Apache Maven Committer
  - since Dec. 2013
- Apache Maven PMC Member
  - since August 2014
- Apache Software Foundation Member
  - since April 2017
- Apache Maven Project Chairman
  - Since April 2022

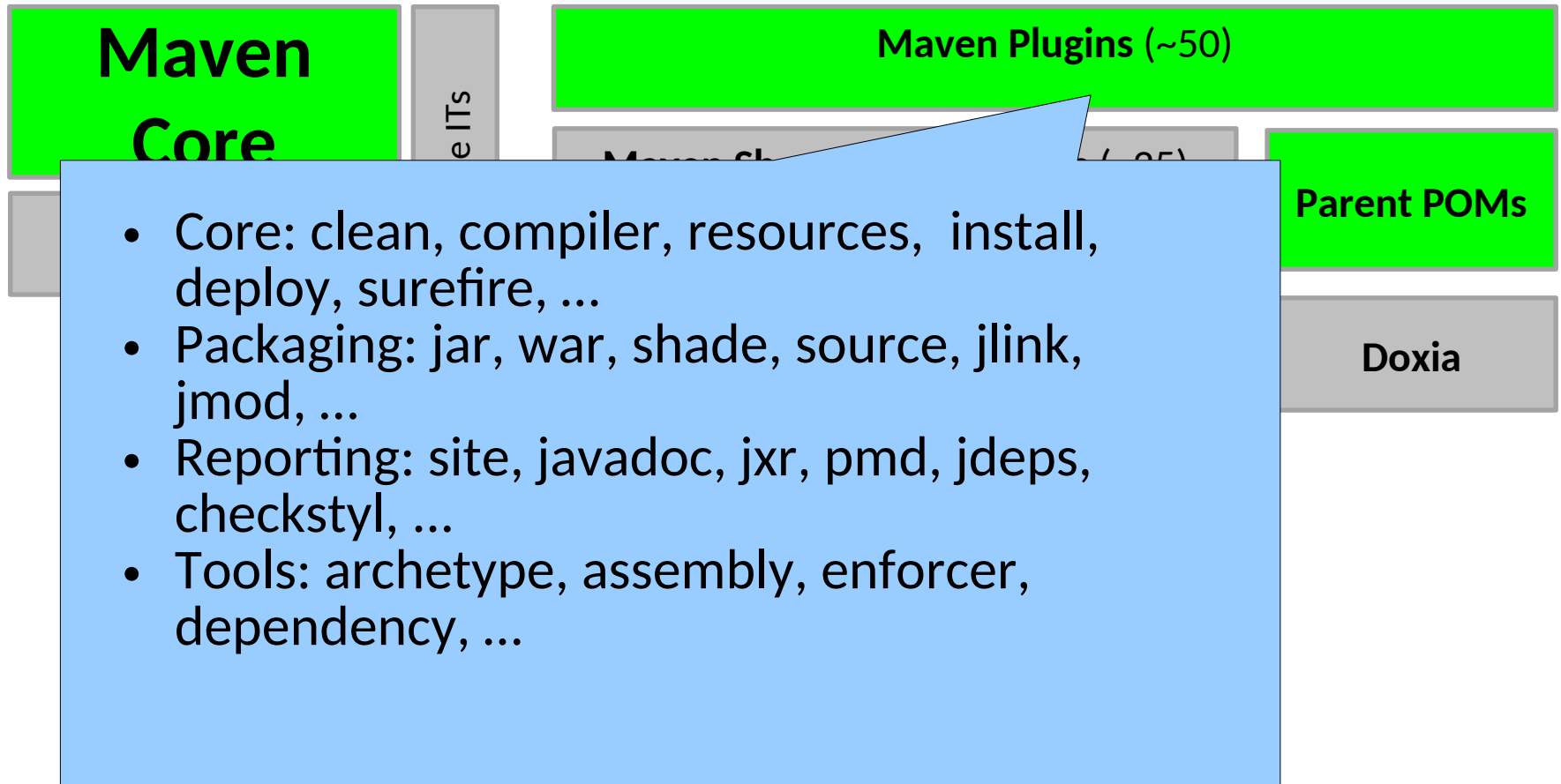
# Agenda

- Apache Maven Ecosystem
- Maven Ecosystem
- Architecture
- Consumer/Build POM
- Reactor Enhancements
- Timeline
- JDK Requirement

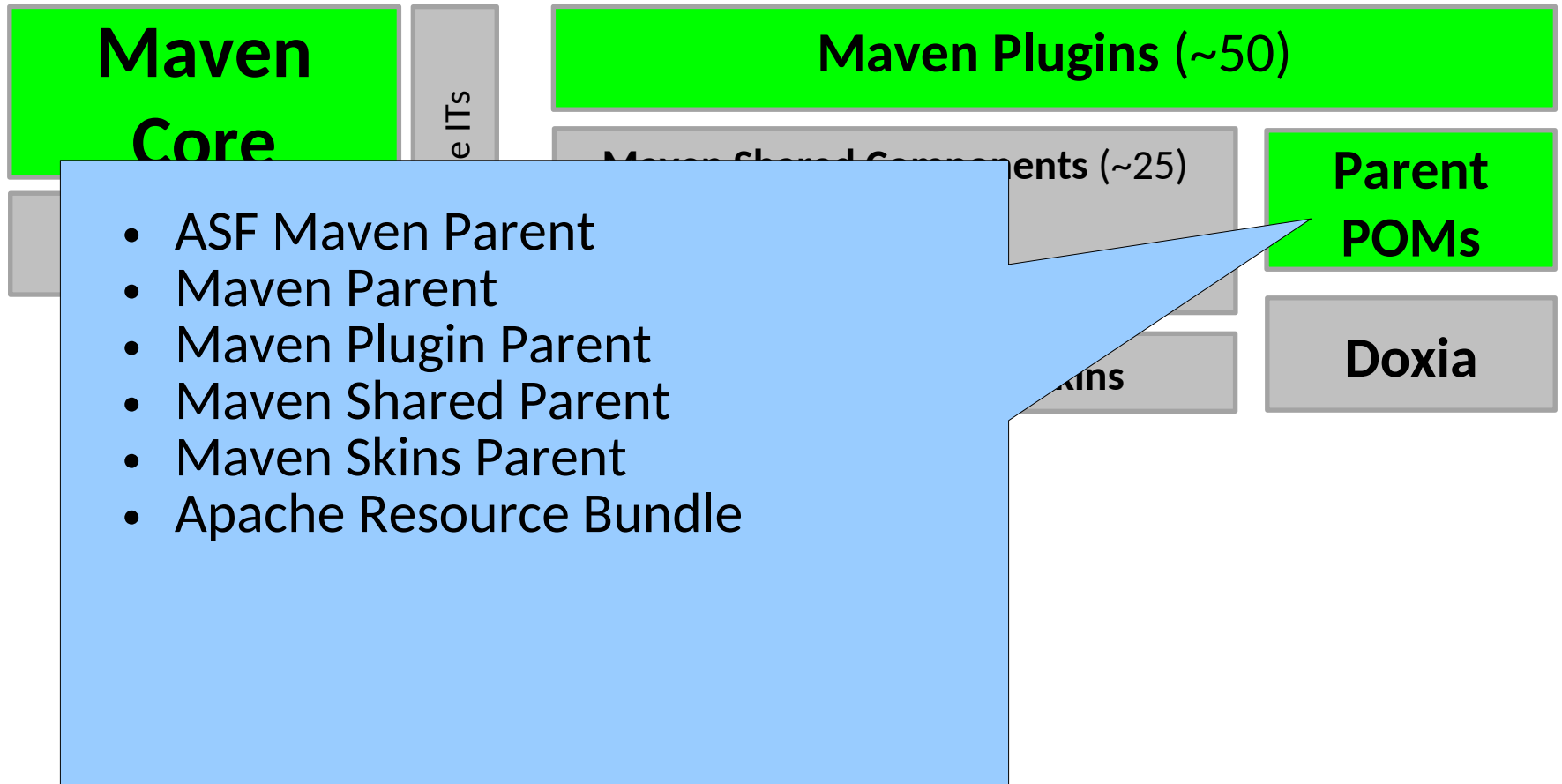
# Apache Maven Ecosystem



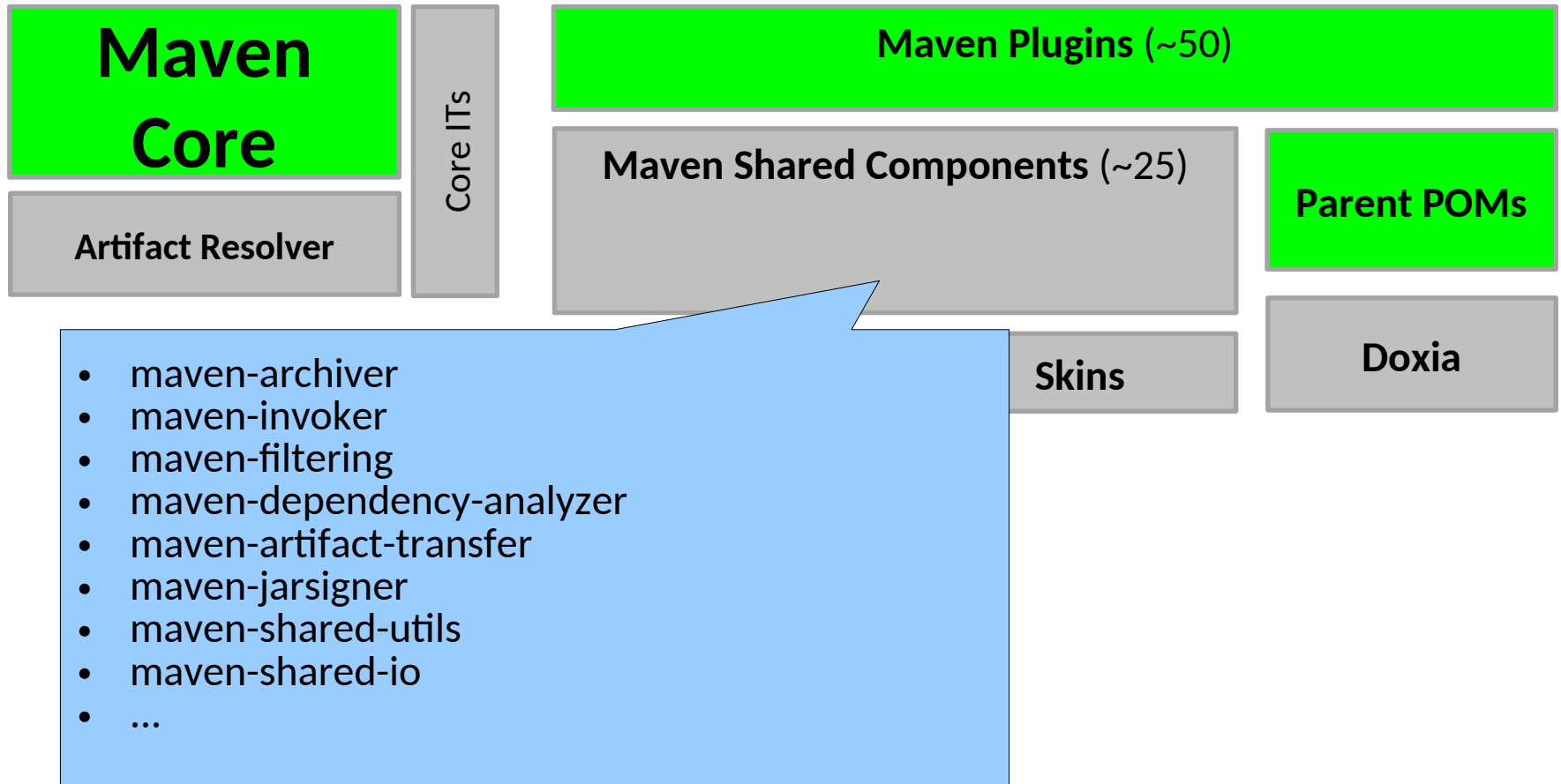
# Apache Maven Ecosystem



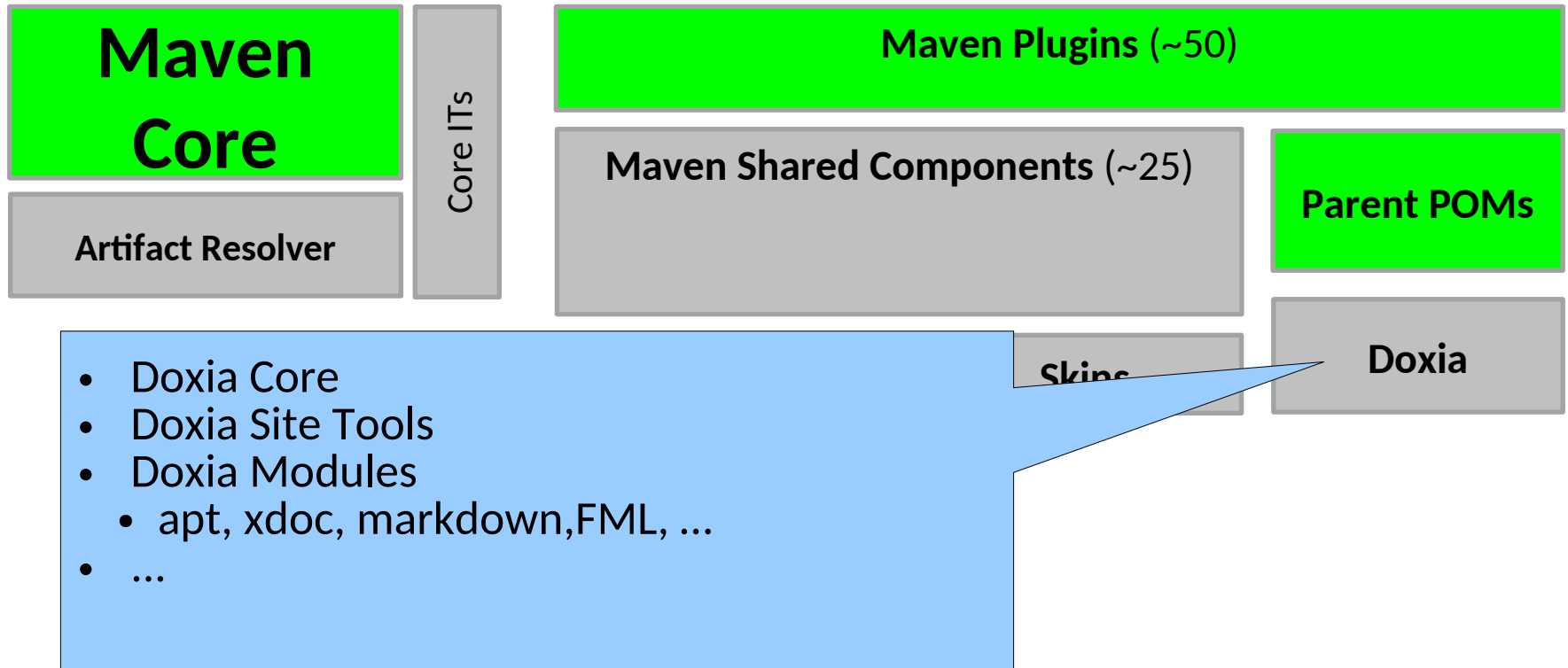
# Apache Maven Ecosystem



# Apache Maven Ecosystem

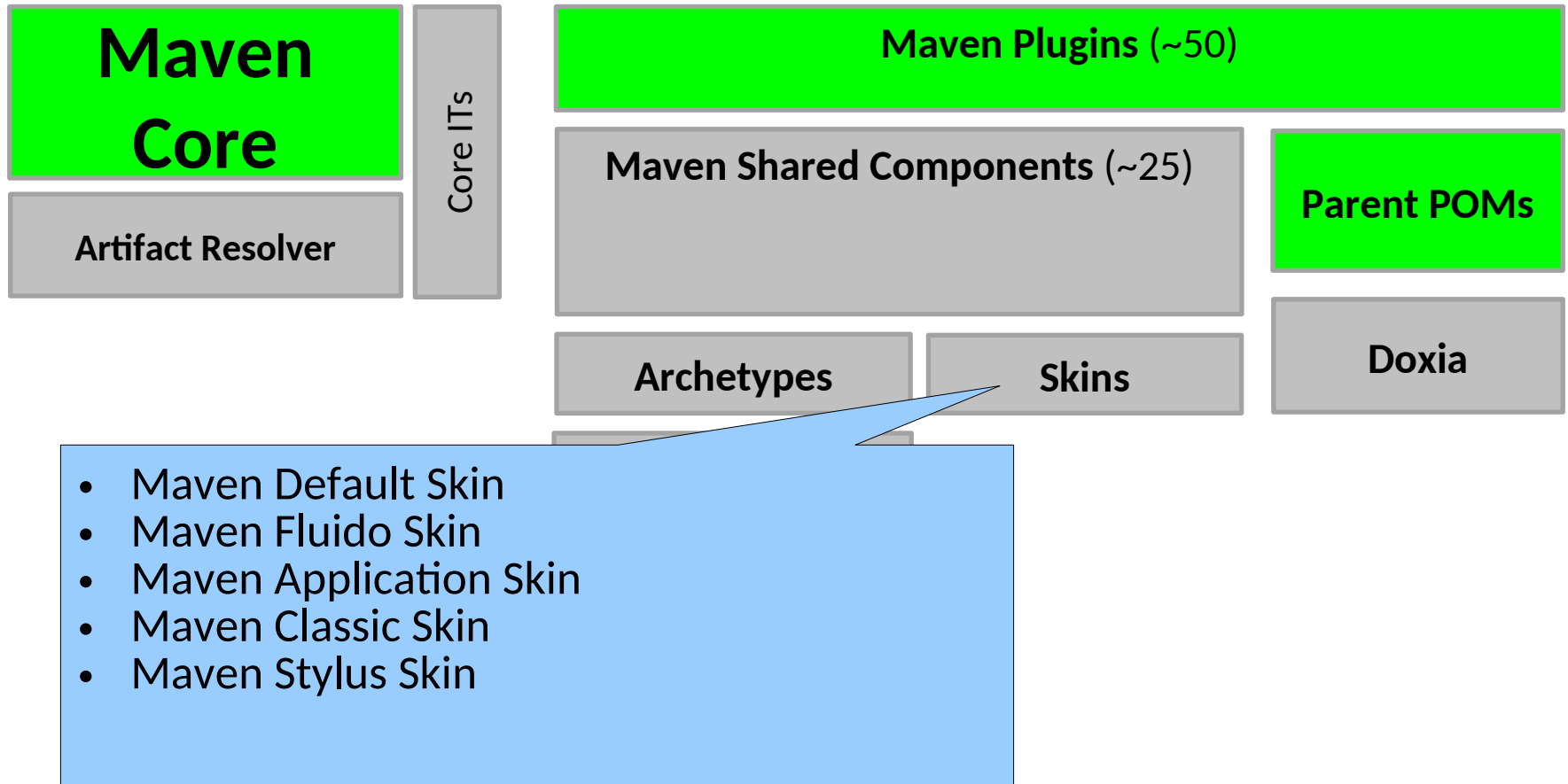


# Apache Maven Ecosystem

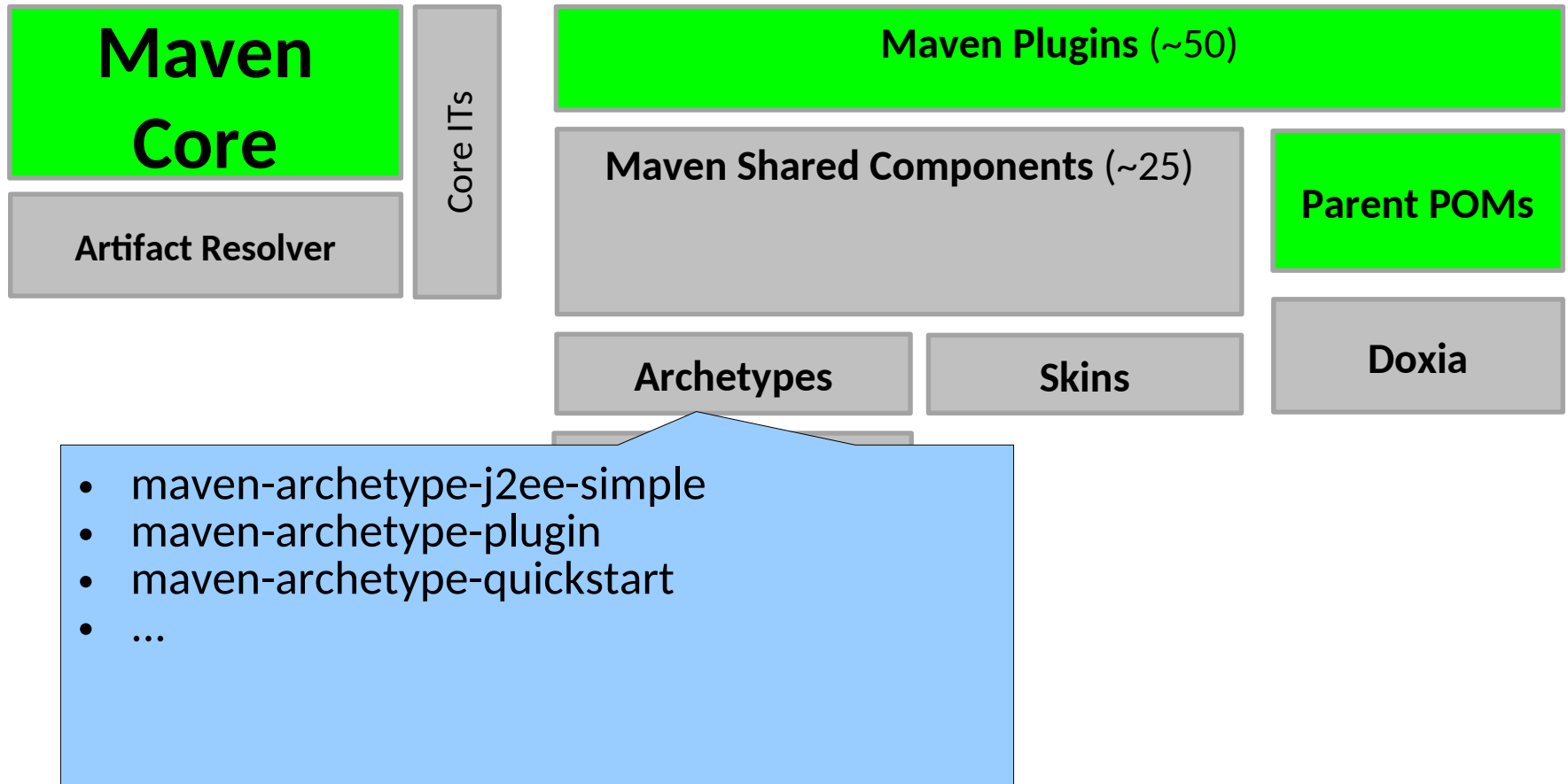




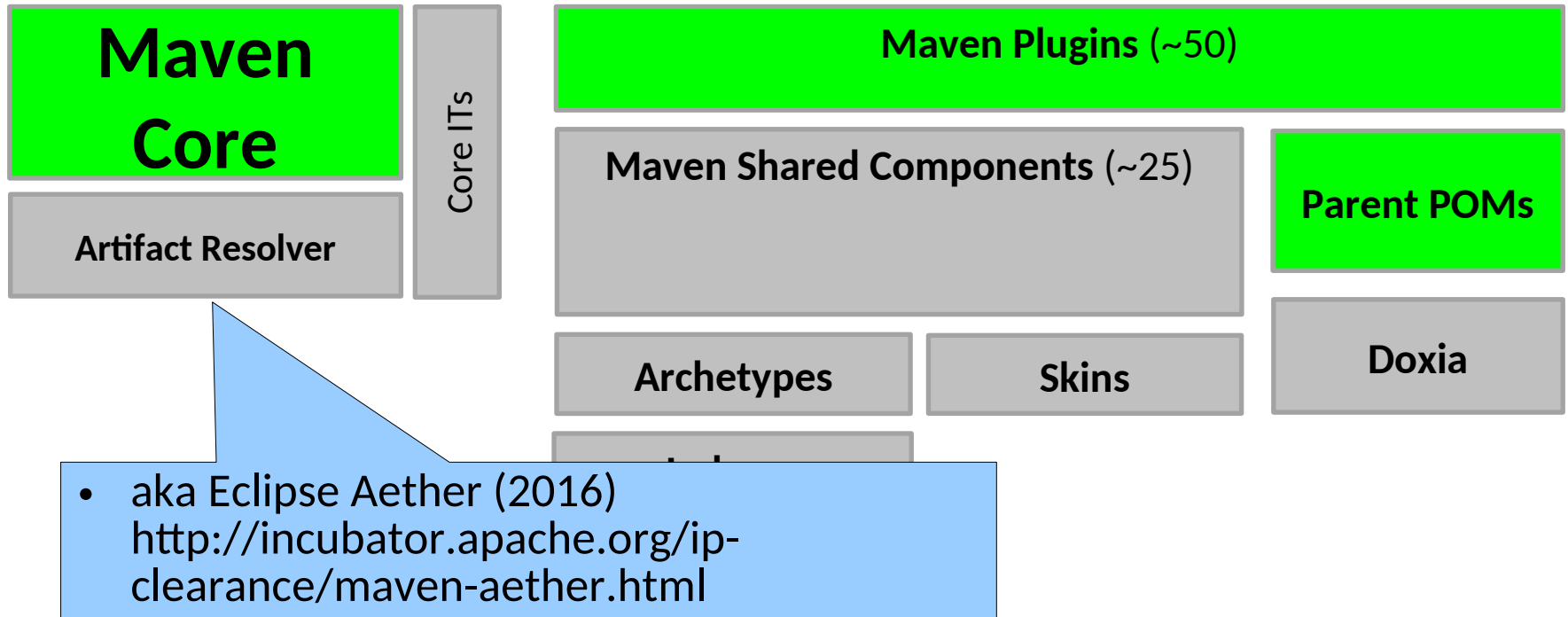
# Apache Maven Ecosystem



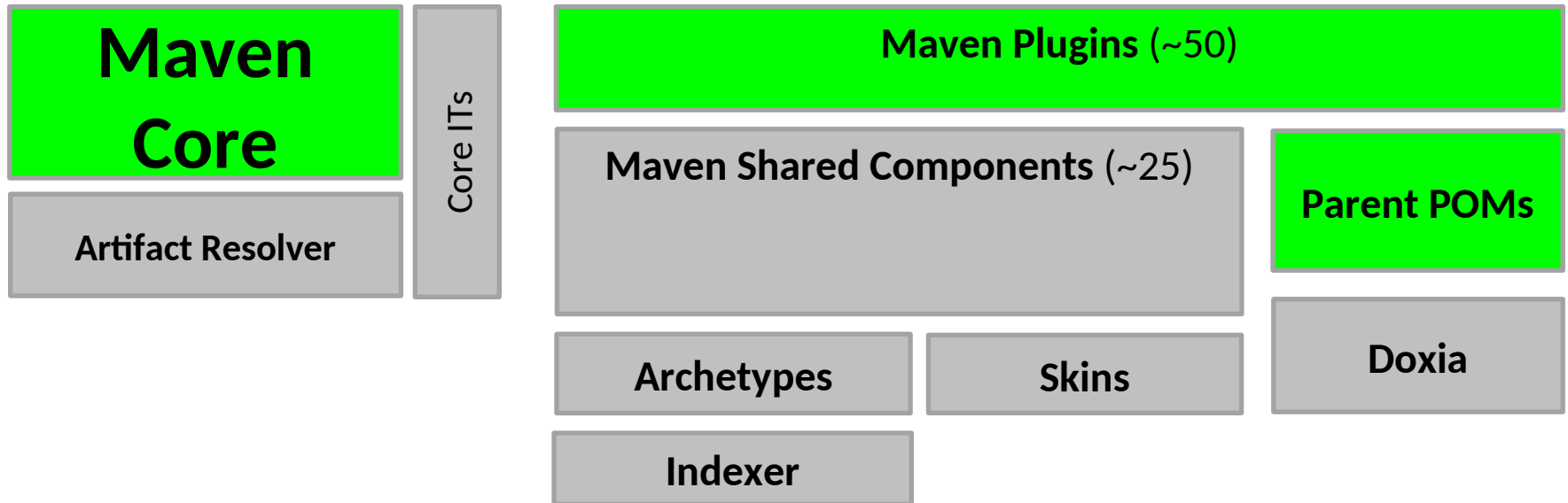
# Apache Maven Ecosystem



# Apache Maven Ecosystem



# Apache Maven Ecosystem



# Maven Ecosystem

- <https://github.com/codehaus-plexus>
  - plexus-classworlds
  - plexus-modello
  - plexus-utils
  - plexus-archiver
  - plexus-io
  - plexus-languages
  - Plexus-interpolation
  - ...

Plugins (~50)

Components (~25)

Parent POMs

Skins

Doxia

Plexus Components (~13)

MojoHaus Plugins (~80)

# Maven Ecosystem

- <https://github.com/mojohaus/>
  - mrm-maven-plugin
  - versions-maven-plugin
  - build-helper-maven-plugin
  - buildnumber-maven-plugin
  - animal-sniffer-maven-plugin
  - appassembler-maven-plugin
  - jaxb2-maven-plugin
  - flatten-maven-plugin
  - exec-maven-plugin
  - rpm-maven-plugin
  - templating-maven-plugin
  - ...

**Maven Plugins (~50)**

**Components (~25)**

**Parent POMs**

**Skins**

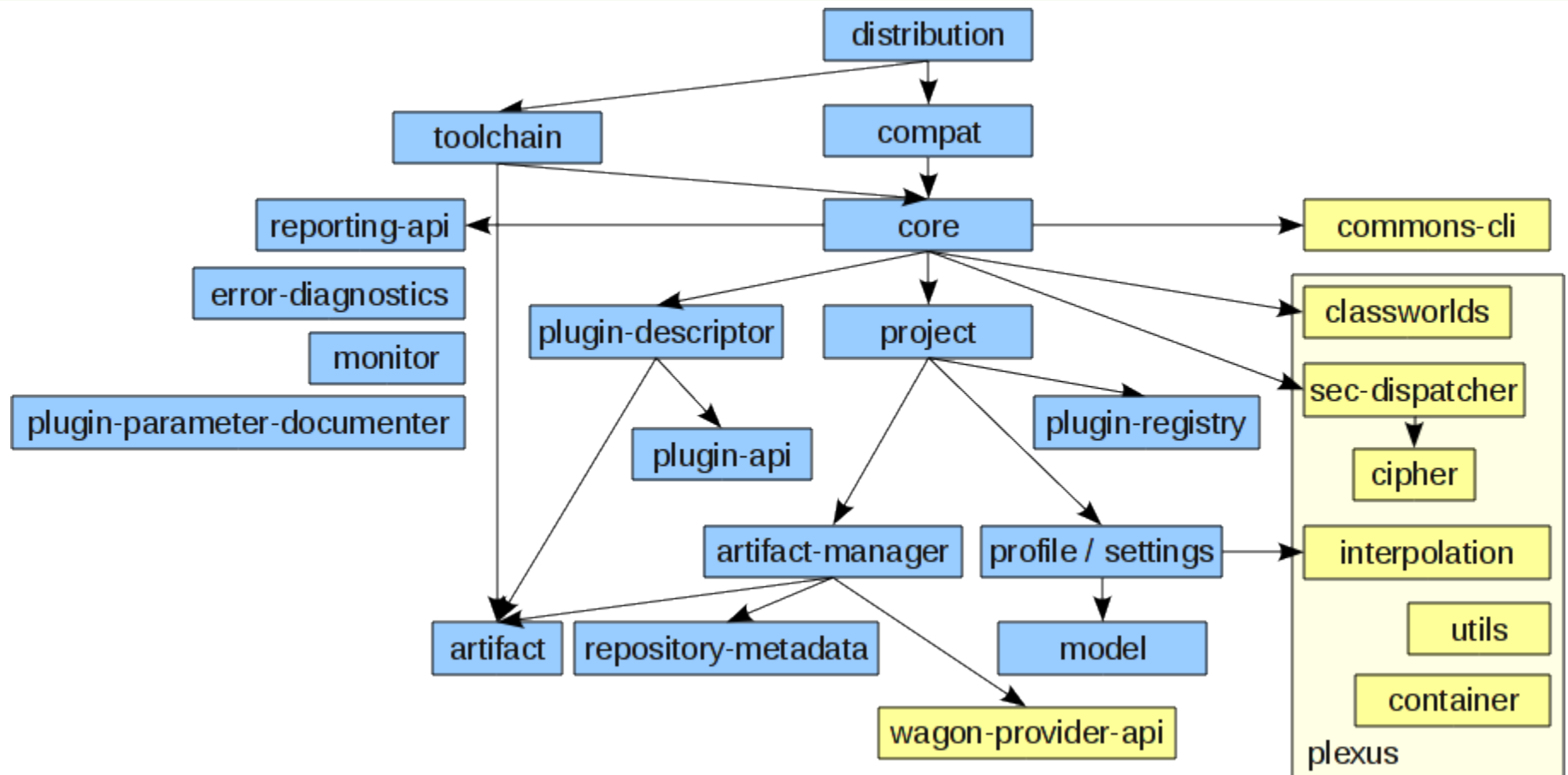
**Doxia**

**Plexus Components (~13)**

**MojoHaus Plugins (~80)**

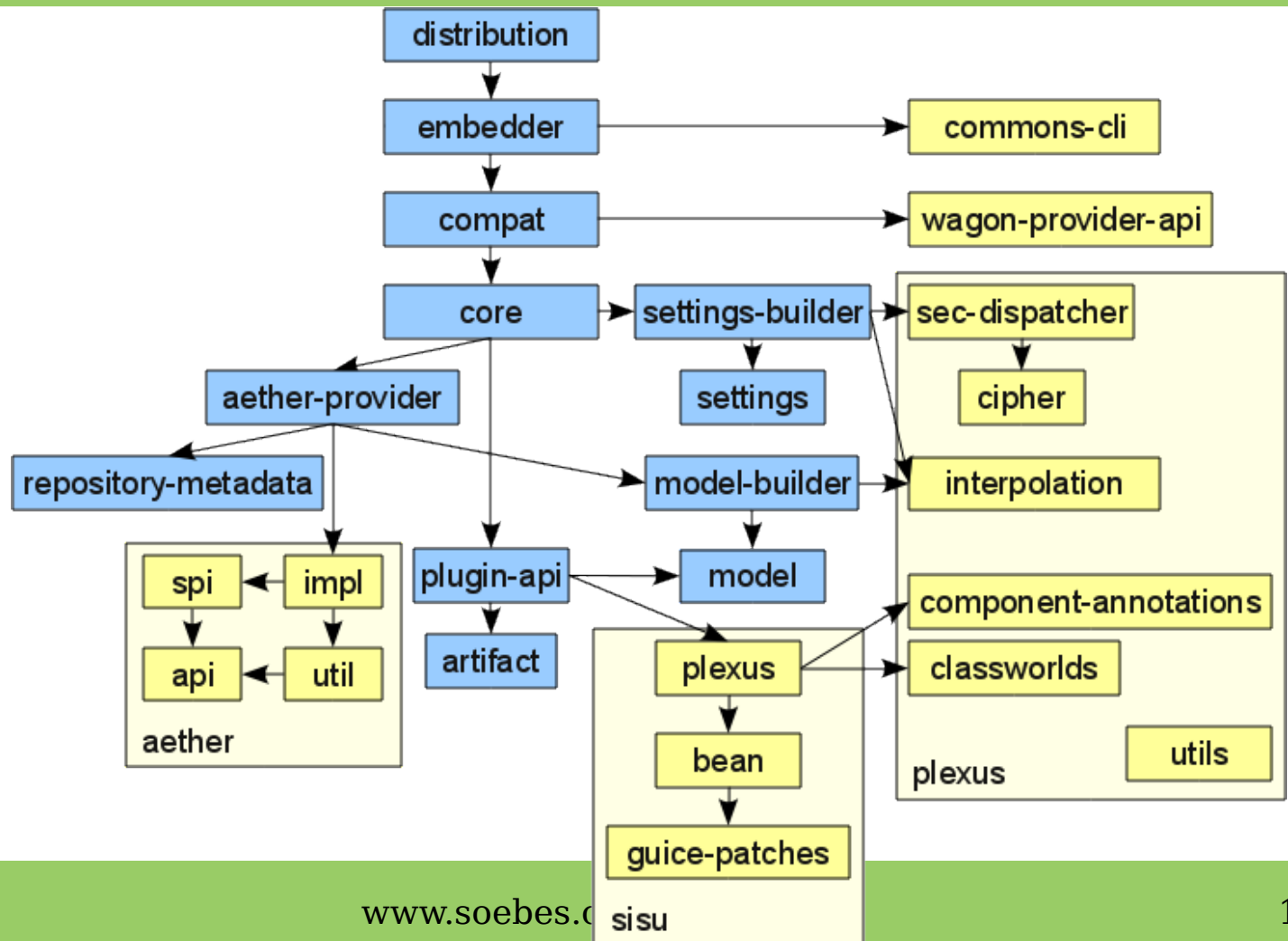
# Architecture

## Maven 2 - 2005



# Architecture

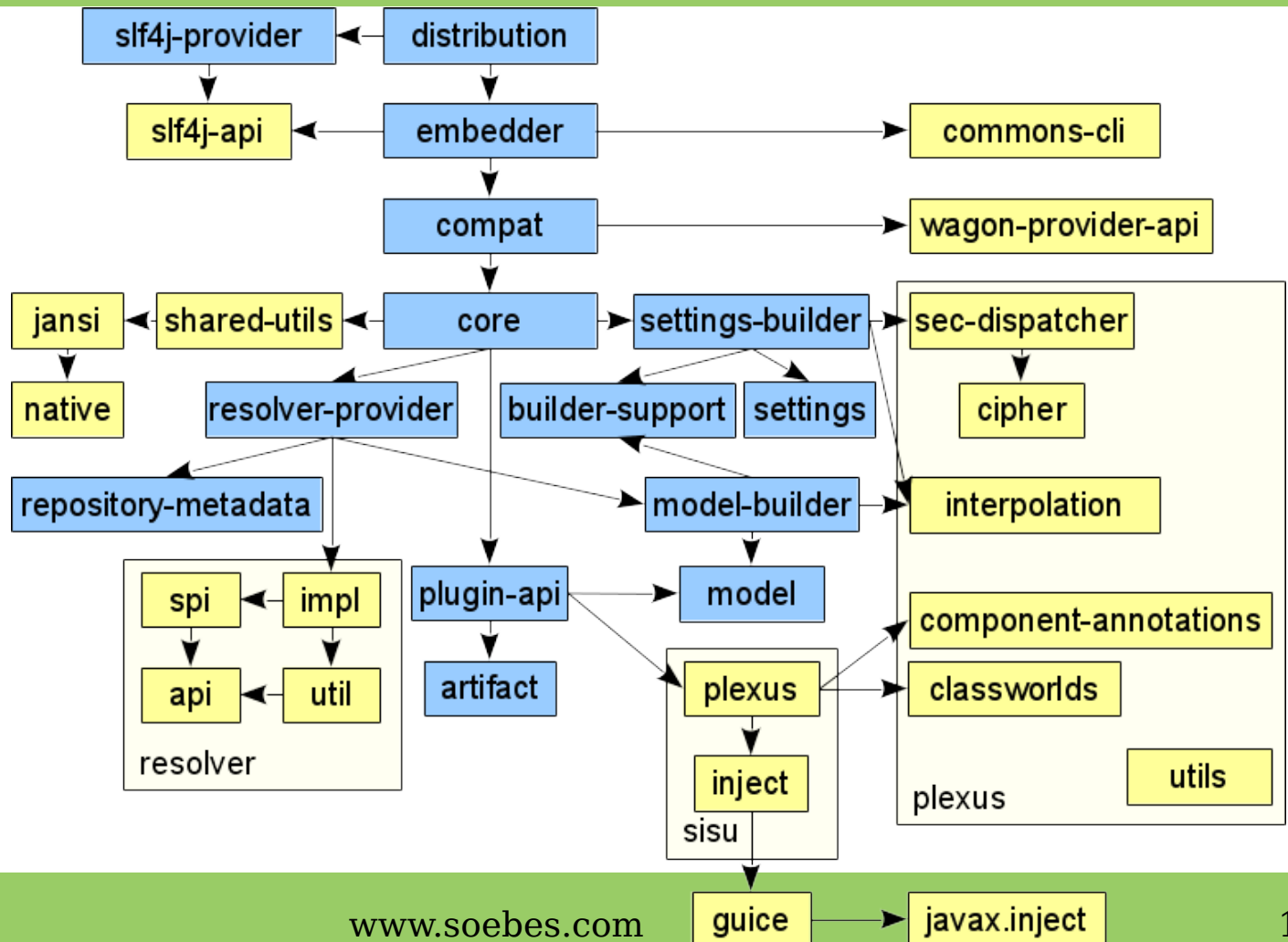
## Maven 3 - 2010





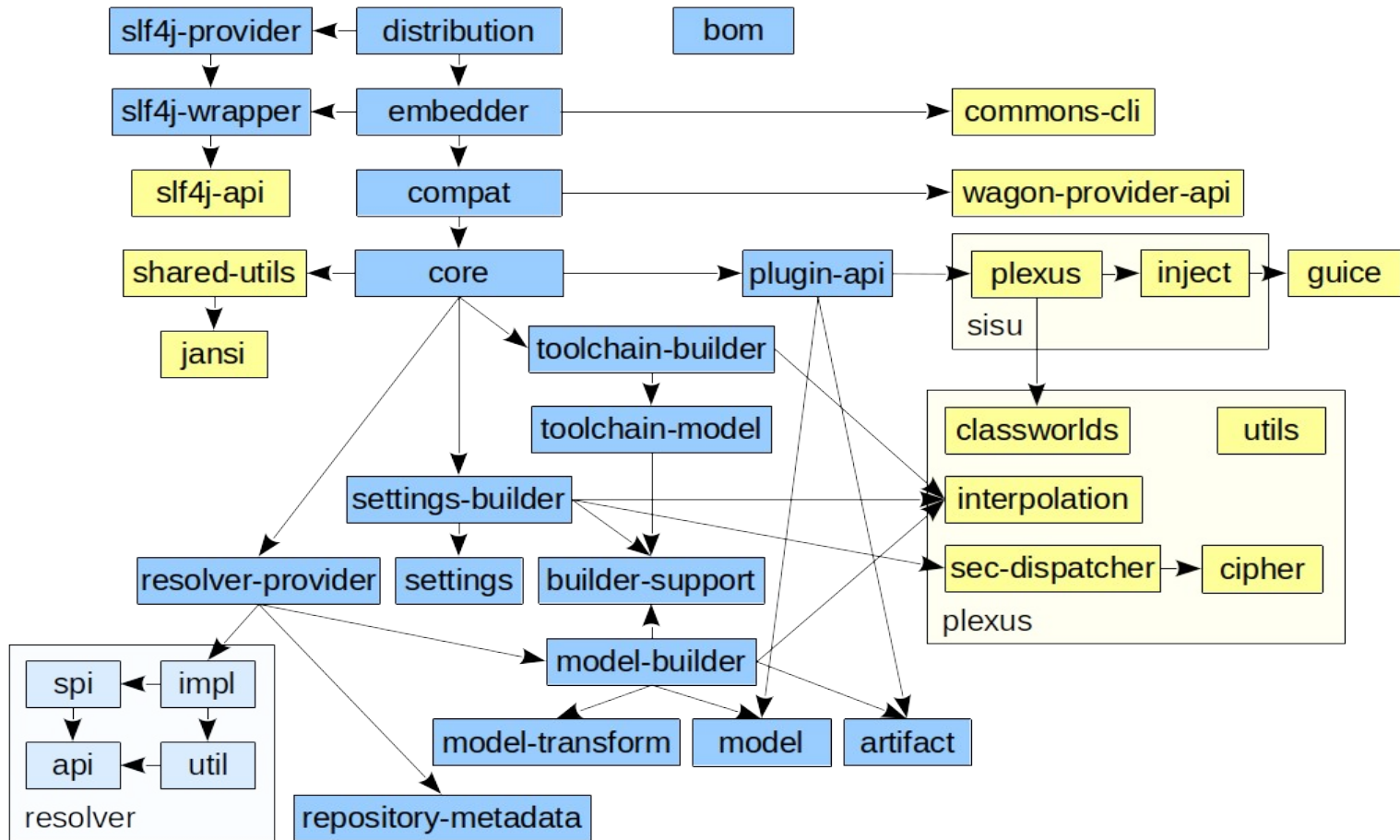
# Architecture

## Maven 3.5+ - 2017



# Architecture

## Maven 4.X - 2020-2024



# Consumer / Build POM

## POM V4

- ~~Maven 1 started with the story~~  
~~modelVersion 3.0.0~~
- Maven 2/3 has continued the story with  
modelVersion 4.0.0

```
<project ...>  
  <modelVersion>4.0.0</modelVersion>  
  ...  
</project>
```

# Consumer / Build POM

## POM V4

- The pom declarares the following information:
  - dependencies, dependencyManagement
  - plugins, pluginManagement
  - profiles, repositories
  - build, reporting
  - distributionManagement
  - etc.

# Consumer / Build POM

## POM V4

- The function of a POM:
  1. Serves as declarative description of the build process for a project.
  2. Serves as a description of the project dependency graph.

# Consumer / Build POM POM V4

- Changing the POM format?
  - The POM is consumed by:
    - Apache BuildR, Gradle, Apache Ivy, sbt
    - by IDE's
    - and many other tools
  - as a description of dependencies.

# Consumer / Build POM POM V4

- Requirements to change the POM format:
  - All tools would needed to be changed.
  - We have to ***fork*** Central Repository.

# Consumer / Build POM

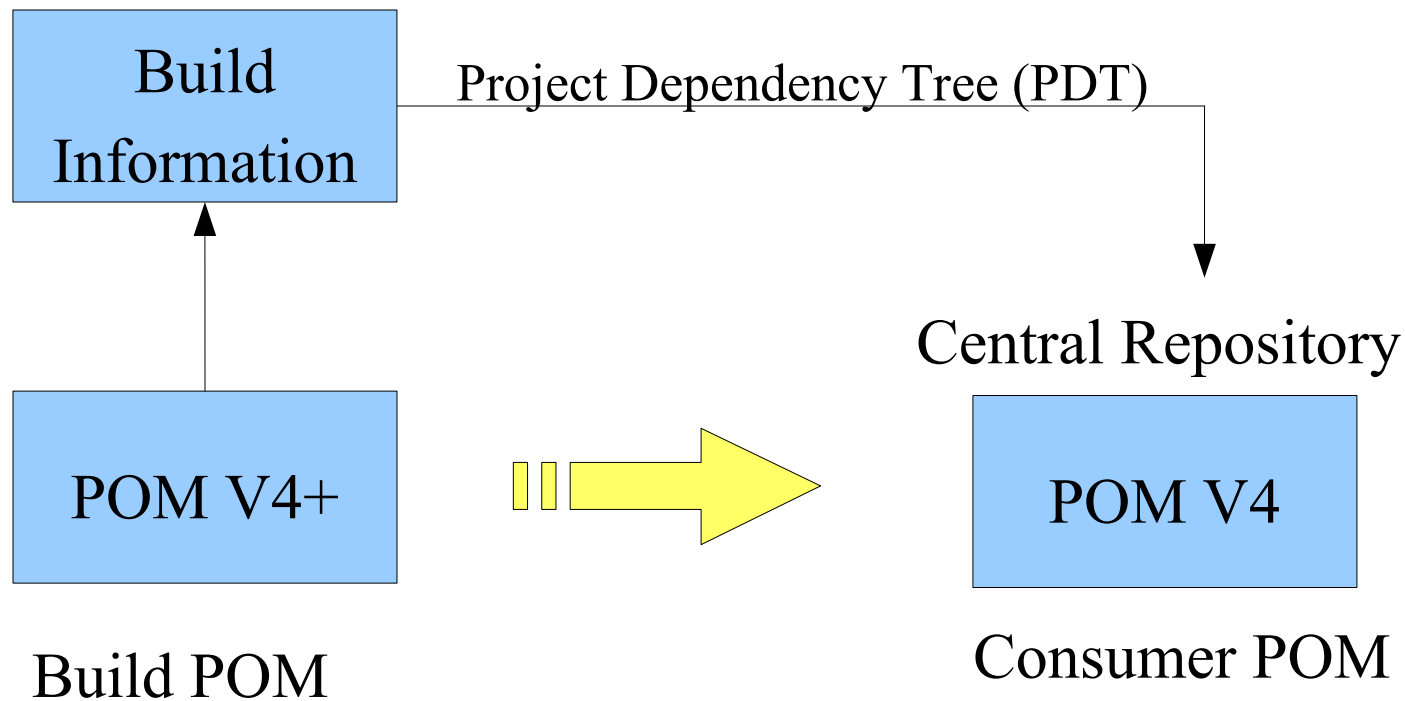
## POM V4

- Requirements to change the POM format:
  - All tools would needed to be changed.
  - We have to *fork* Central Repository.





# Consumer / Build POM POM V4+



# Consumer / Build POM

## POM V4

- POM V4 a.k.a. *Consumer POM*
  - Contains only dependencies (+ dependencyManagement)
  - Remove all properties / build / modules / parent / ....
  - See flatten-maven-plugin
- <https://cwiki.apache.org/confluence/display/MAVEN/Build+vs+Consumer+POM>

DEMO

# POM V4+ → Maven 4.X

- Core must be able to read:
  - POM V4
  - POM V4+
- Core must be able to produce POM V4

# Timeline

- Maven 3.9.X (3.9.0...3.9.6 in 2023)
  - Bug fixing and improvements
  - Moved away from Wagon to Maven Resolver (aka native HTTP).
  - Plugin API changes with some hints as preparation for Maven 4.X  
You might have seen some of them!
- <https://maven.apache.org/docs/history.html>
- <https://maven.apache.org/guides/mini/guide-resolver-transport.html>

# Timeline

- Maven 3.9.X (3.9.0...3.9.6 in 2023)
  - Since Maven 2 ancient version of plexus-utils was injected. Not done anymore (might break something!).
  - No bootstrap of RepositorySystem (via deprecated ServiceLocator!)
  - Support for „mvn pluginPrefix:version:goal” invocation

<https://maven.apache.org/docs/history.html>

# Timeline

- Maven 3.9.2+
  - Use the Force Luke:
    - Create an „.mvn“ directory.
  - New properties:
    - *session.topDirectory* - reactor top
    - *session.rootDirectory* - project root (where „.mvn“ required!).

<https://maven.apache.org/docs/3.9.2/release-notes.html>

# Timeline

- Maven 4 – 4.0.0-alpha-7 (June 2023)
  - Build/Consumer POM
  - A lot of reactor improvements
  - Plugin API Changes
    - Some of them are breaking changes!  
See 3.9.X
  - Identify root location of multi module build („.mvn/“ ~~or POM configuration~~).

<https://cwiki.apache.org/confluence/display/MAVEN/Build+vs+Consumer+POM>



# Timeline

- Maven 4 – 4.0.0-alpha-8 (20. Oct 2023)
  - Switch to Woodstox XML
  - Introduce Model Version 4.1.0
    - „root“ element in version 4.1.0
  - Glob Pattern for exclusions

```
<excludes>  
  <exclude>  
    <groupId>com.jdom.*</groupId>  
    <artifactId>*</artifactId>  
  </exclude>  
</excludes>
```

# Timeline

- Maven 4 – 4.0.0-alpha-8 (20. Oct 2023)
  - „BOM“ Packaging, Trim down consumer POM
  - Core Extension in .m2
  - Alternate POM Syntax possible
    - MNG-7836

# Timeline

- Maven 4 – 4.0.0-alpha-8 (20. Oct 2023)
  - POM Goal Ordering (MNG-7804, Model 4.1.0):

```
<execution>
  <id>400. This should be fourth</id>
  <goals>
    <goal>run</goal>
  </goals>
  <phase>clean</phase>
  <priority>400</priority>
</execution>
```

# Timeline

- Maven 4 – 4.0.0-alpha-8 (20. Oct 2023)
  - Support configuration in extensions.xml (MNG-7891) (basic).
  - support for exclusions when importing BOMs

# Timeline

- Maven 4 – 4.0.0-alpha-10 (11.Dec 2023)
  - Fixed broken javadoc:jar
  - Resolver fixes.

# Timeline

- Maven 4 – 4.0.0-alpha-12 (12.Jan 2024)
  - Resolver 2.0.0-alpha-6
  - Collection filtering
  - Resolvers transitive dependency manager.
  - No resolving of projects outside the reactor.

# Timeline

- Maven Cache Extension

- Incremental builds works on the modified part of the project graph part only
- Subtree support for multimodule projects to work on the part of the codebase in isolation
- Version normalization to support project version agnostic caches
- Project state restoration (partial) to avoid expensive tasks (code generation and similar)

<https://maven.apache.org/extensions/maven-build-cache-extension/>

# Timeline

- Incremental Build Prototype!
  - Only build what has changed without cache.
  - ...

<https://github.com/khmarbaise/incremental-module-builder>



# Minimum JDK Runtime Requirements

- Maven 3.8.X - JDK 7
- Maven 3.9.6 - JDK 8
- Maven 4.0.0-alpha-12
  - Runtime: JDK 8
  - Buildtime JDK11+

<https://maven.apache.org/docs/history.html>

# Performance Measurement

- Setup for project exist, several executions
  - Different JDK's (six!)
  - Different Maven versions(20!)
  - Each test runs 10 times
  - $20 \times 6 \times 10 = 1200$  runs
  - ...

<https://github.com/khmarbaise/maven-performance-testing>

# Performance Measurement

- Setup for project exist, several executions (prototype)
  - Current scenario
    - Multi module build with different number of children (10...10.000)..

<https://github.com/khmarbaise/maven-performance-testing>

# Questions?

[khmarbaise@apache.org](mailto:khmarbaise@apache.org)

Thank you for your attention.