



Apache Maven Project
<http://maven.apache.org>

Michael Greifeneder

maven

Building Software

Small Artifacts – Big Projects

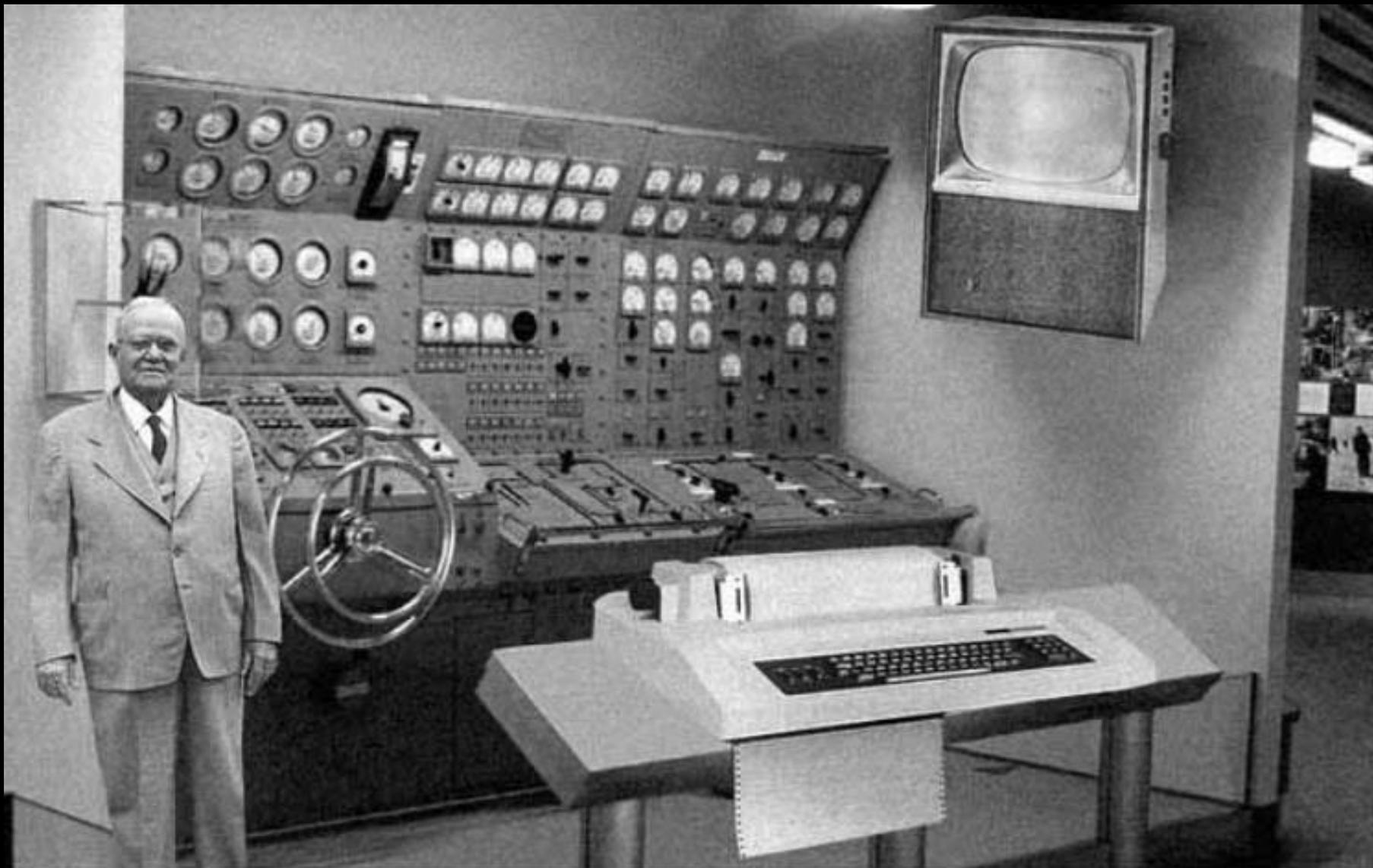
Batch files

Shellscripts

Make

Ant

20



Scientists from the RAND Corporation have created this model to illustrate how a "home computer" could look like in the year 2004. However the needed technology will not be economically feasible for the average home. Also the scientists readily admit that the computer will require not yet invented technology to actually work, but 50 years from now scientific progress is expected to solve these problems. With teletype interface and the Fortran language, the computer will be easy to use.

Inhalt

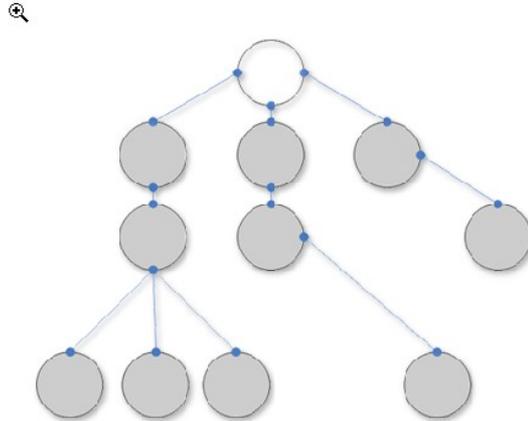
- Schnelle Wiederholung
- Features
- Plugins
- Report
- Zusammenfassung

Quick Review

A build tool!

```
C:\WINDOWS\system32\cmd.exe
Downloading: http://repo1.maven.org/maven2/org/apache/maven/wagon/wagon-1.0-alpha-4/wagon-1.0-alpha-4.pom
3K downloaded
Downloading: http://repo1.maven.org/maven2/org/apache/maven/wagon/wagon-provider-api-1.0-alpha-4/wagon-provider-api-1.0-alpha-4.jar
45K downloaded
Downloading: http://repo1.maven.org/maven2/org/apache/maven/maven-artifact-manager/2.0-alpha-3/maven-artifact-manager-2.0-alpha-3.jar
32K downloaded
[INFO] [install:install]
[INFO] Installing C:\my-app\target\my-app-1.0-SNAPSHOT.jar to C:\Documents and Settings\Administrator\TOSHIBA\m2\repository\com\mycompany\app\my-app-1.0-SNAPSHOT\my-app-1.0-SNAPSHOT.jar
[INFO] -----
[INFO] BUILD SUCCESSFUL
[INFO] -----
[INFO] Total time: 47 seconds
[INFO] Finished at: Fri Jun 24 16:24:10 PDT 2005
[INFO] Final Memory: 2M/5M
[INFO] -----
C:\my-app>
```

A dependency management tool!



A documentation tool!

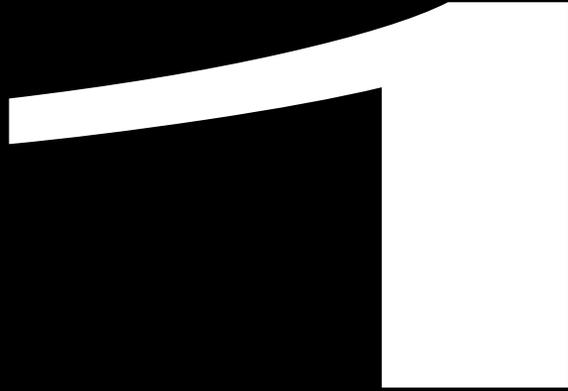


100000

Klassen

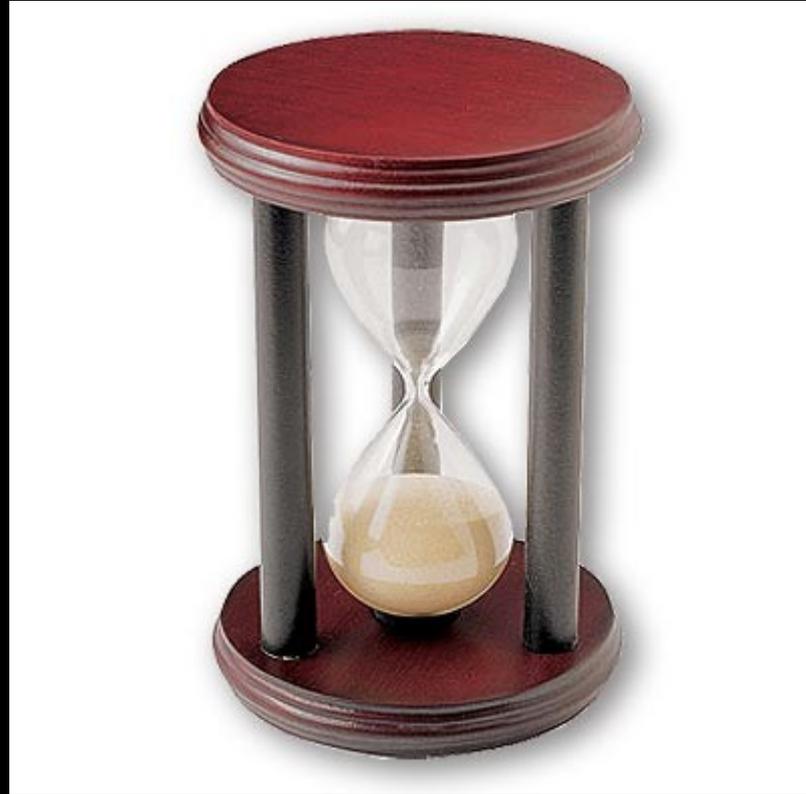
100 - 1000

Packages



Projekt







<http://www.thecoolstgifts.com/wp-content/naughty-knot.jpg>

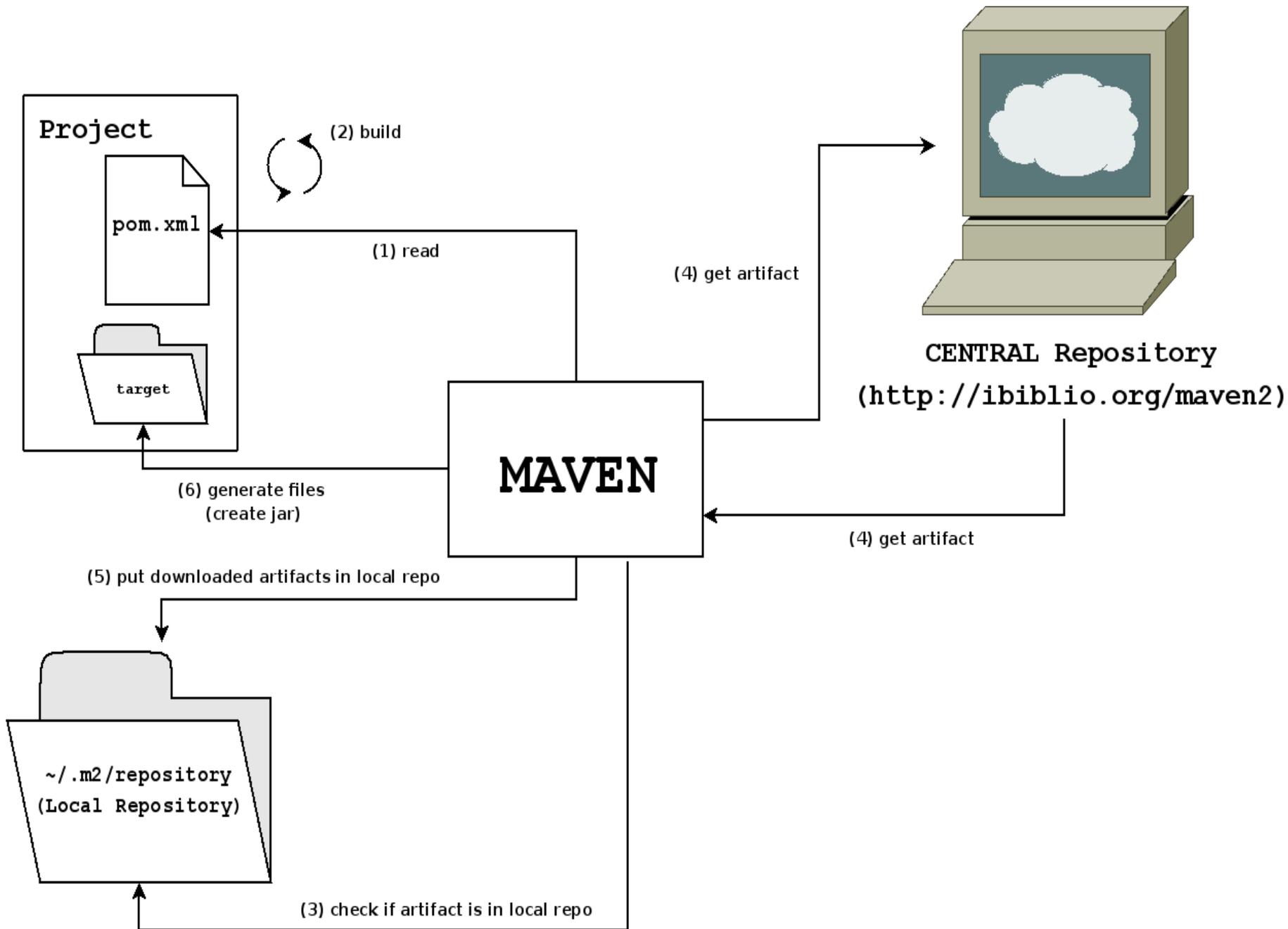
Quick Review

- Projekt Management Tool
 - Jedes Projekt erzeugt EIN “Artifakt”
 - Dependencies
 - Multi-Module Build
- Project Object Model – pom.xml
 - Deklarativ
 - Convention over Configuration



STANDARDS

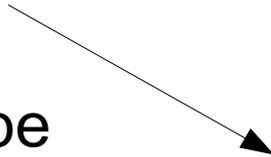
THEY ARE GREAT. EVERYONE SHOULD HAVE THEIR OWN.



Archetypen



Archetype



```
mvn archetype:create ...
```



My Artifact

```
|-- pom.xml
```

```
`-- src
```

```
|-- main
```

```
|   |-- java
```

```
|   `-- resources
```

```
|-- test
```

```
|   |-- java
```

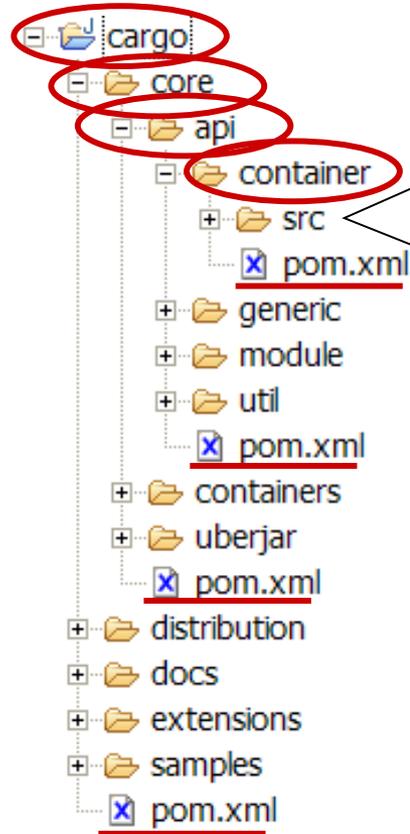
```
|   `-- resources
```

```
`-- site
```

Resultat

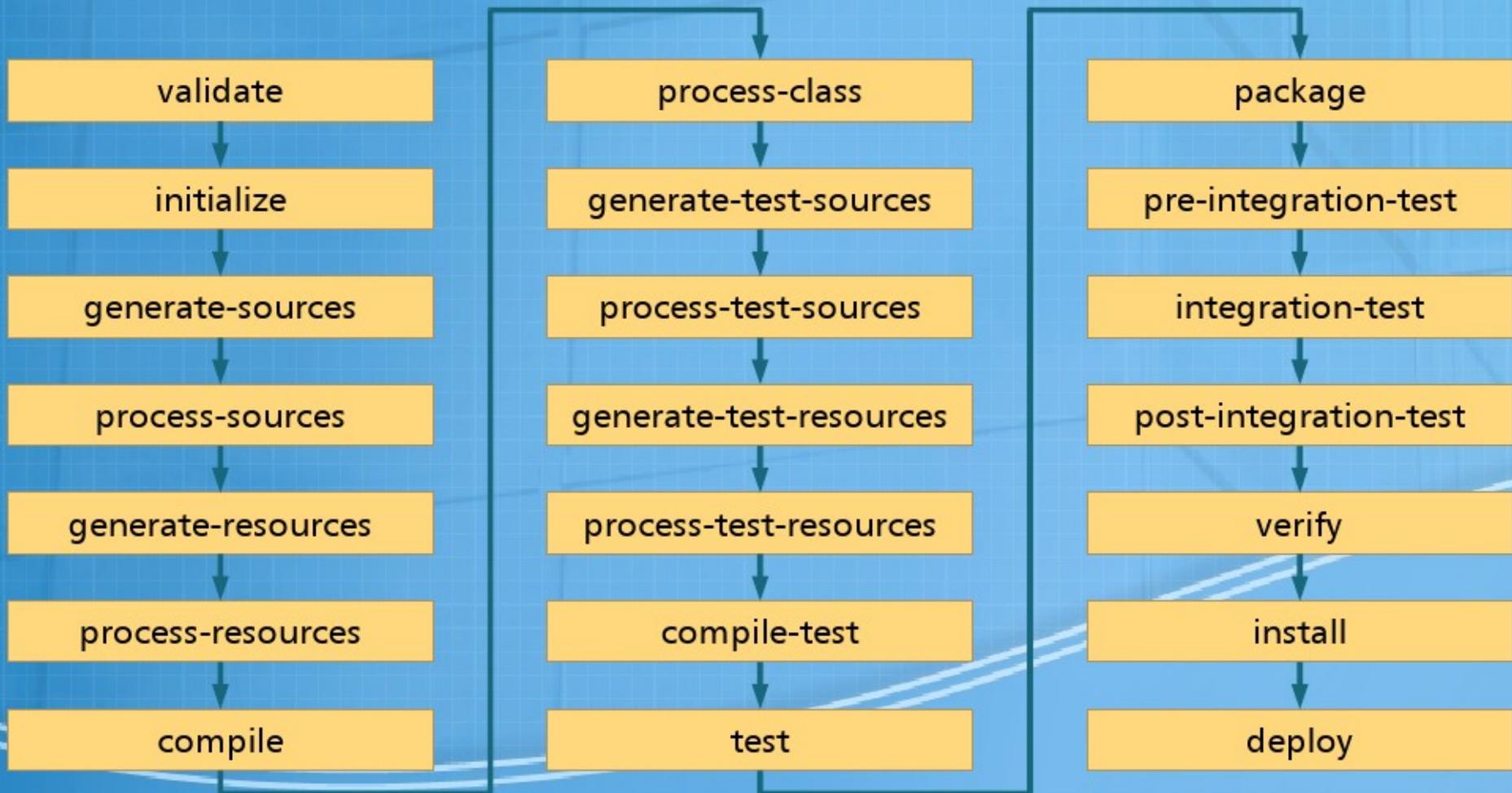
4 nested projects

Other projects

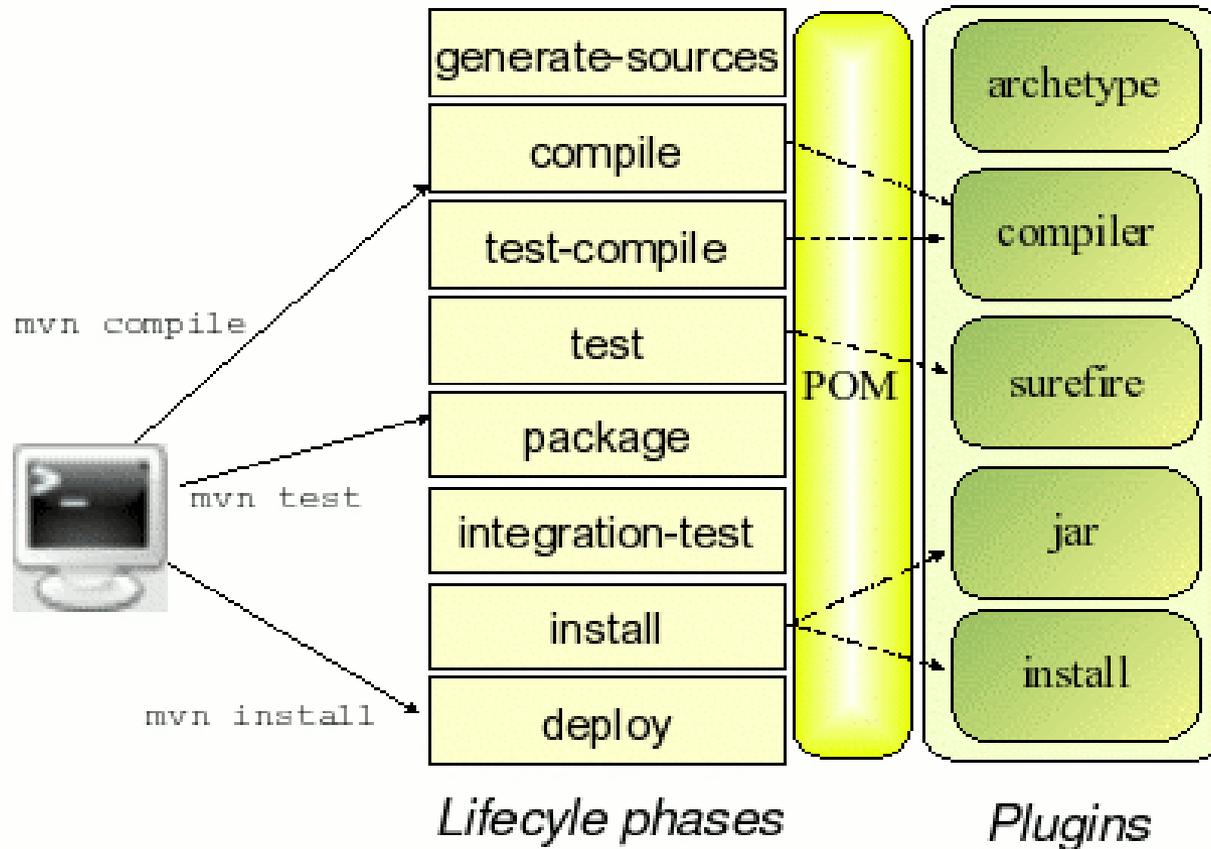


- src/
 - main/
 - java/
 - resources/
 - webapp/
 - application/
 - groovy/
 - test/
 - java/
 - resources/
 - cactus/
 - site/

Build Lifecycle Phases



Die Wichtigsten



Sonstige

- `mvn eclipse:eclipse`
- `mvn idea:idea`
- `Mvn jetty:run`
 - `mvn org.mortbay.jetty:maven-jetty-plugin:run`
- `mvn site`

Site

Project Information

This document provides an overview of the various documents and links that are part of this project's general information. All of this content is automatically generated by [Maven](#) on behalf of the project.

Overview

Document	Description
About	There is currently no description associated with this project.
Dependencies	This document lists the projects dependencies and provides information on each dependency.
Issue Tracking	This is a link to the issue management system for this project. Issues (bugs, features, change requests) can be created and queried using this link.
Project License	This is a link to the definitions of project licenses.
Project Summary	This document lists other related information of this project
Project Team	This document provides information on the members of this project. These are the individuals who have contributed to the project in one form or another.
Source Repository	This is a link to the online source repository that can be viewed via a web browser.

Reports

Generated Reports

This document provides an overview of the various reports that are automatically generated by [Maven](#). Each report is briefly described below.

Overview

Document	Description
CPD Report	Duplicate code detection.
JavaDocs	JavaDoc API documentation.
Maven Surefire Report	Report on the test results of the project.
PMD Report	Verification of coding rules.
Source Xref	HTML based, cross-reference version of Java source code.
Tag List	Report on various tags found in the code.
Test Source Xref	HTML based, cross-reference version of Java test source code.

PMD

PMD Results

The following document contains the results of [PMD 3.9](#).

Files

[com/App.java](#)

Violation	Line
All methods are static. Consider using Singleton instead. Alternatively, you could add a private constructor or make the class abstract to silence this warning.	1 - 13
Each class should declare at least one constructor	7 - 13

Source Xref

All Classes

Packages

com

com

Classes

App

[View Javadoc](#)

```
1 package com;
2
3 /**
4  * Hello world!
5  *
6  */
7 public class App
8 {
9     public static void main( String[] args )
10    {
11        System.out.println( "Hello World!" );
12    }
13 }
```

This page was automatically generated by [Maven](#)

[Overview](#)
[All Classes](#)

All Packages

- [org.jboss](#) (23.4%)
- [org.jboss.ant](#) (0%)
- [org.jboss.ant.taskdefs](#) (89.9%)
- [org.jboss.boot.servlets](#) (0%)
- [org.jboss.crvnto](#) (13.4%)

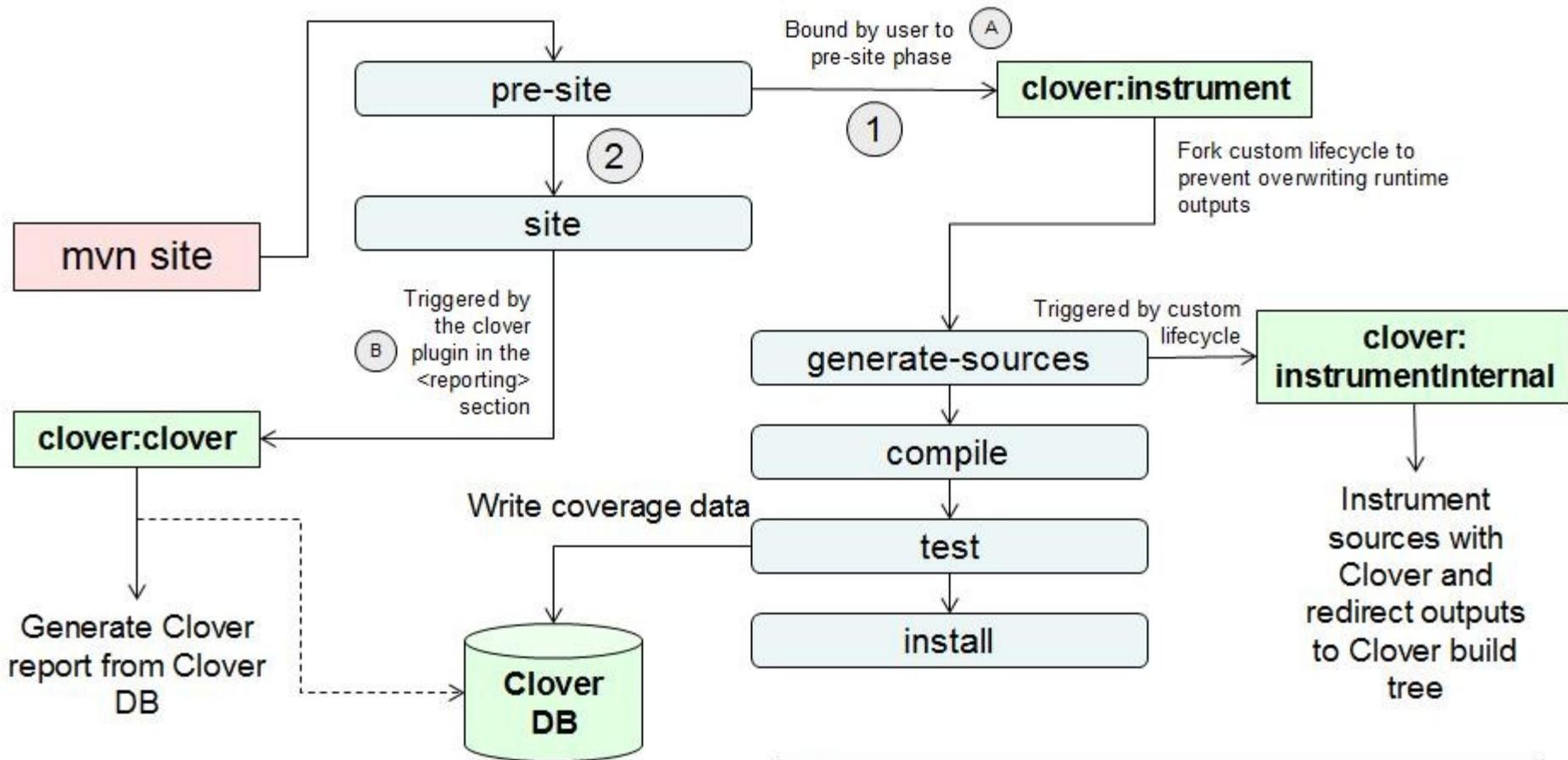
All Classes

- [ABTest](#) (97.1%)
- [ABTest](#) (94.4%)
- [ABTest](#) (95.9%)
- [ABTest](#) (83.2%)
- [ABTest](#) (41.8%)
- [ABTest](#) (82.5%)

Clover

Package	Conditionals	Statements	Methods	TOTAL	
org.jboss.ejb.plugins.cmp.ejbql	24.5%	37.6%	53.6%	34.5%	

Classes	Conditionals	Statements	Methods	TOTAL	
JBossQLParser	27.5%	41.4%	69.2%	38%	
EJBQLParser	11.1%	25%	41.5%	21.2%	
JBossQLParserTokenManager	40.9%	50.5%	70.6%	48.1%	
EJBQLParserTokenManager	35.7%	48.2%	73.5%	45.1%	
ASCII UCodeESC CharStream	28.1%	35.1%	59.3%	35.8%	
BasicVisitor	30%	28.8%	28.3%	28.9%	
ASTPath	50%	55%	56%	54.7%	
JJTJBossQLParserState	50%	65.1%	58.3%	61.5%	
JJTEJBQLParserState	50%	65.1%	58.3%	61.5%	
BlockStringBuffer	50%	20%	26.1%	23.4%	
IdentifierManager	50%	73.7%	77.8%	70.6%	



pom.xml

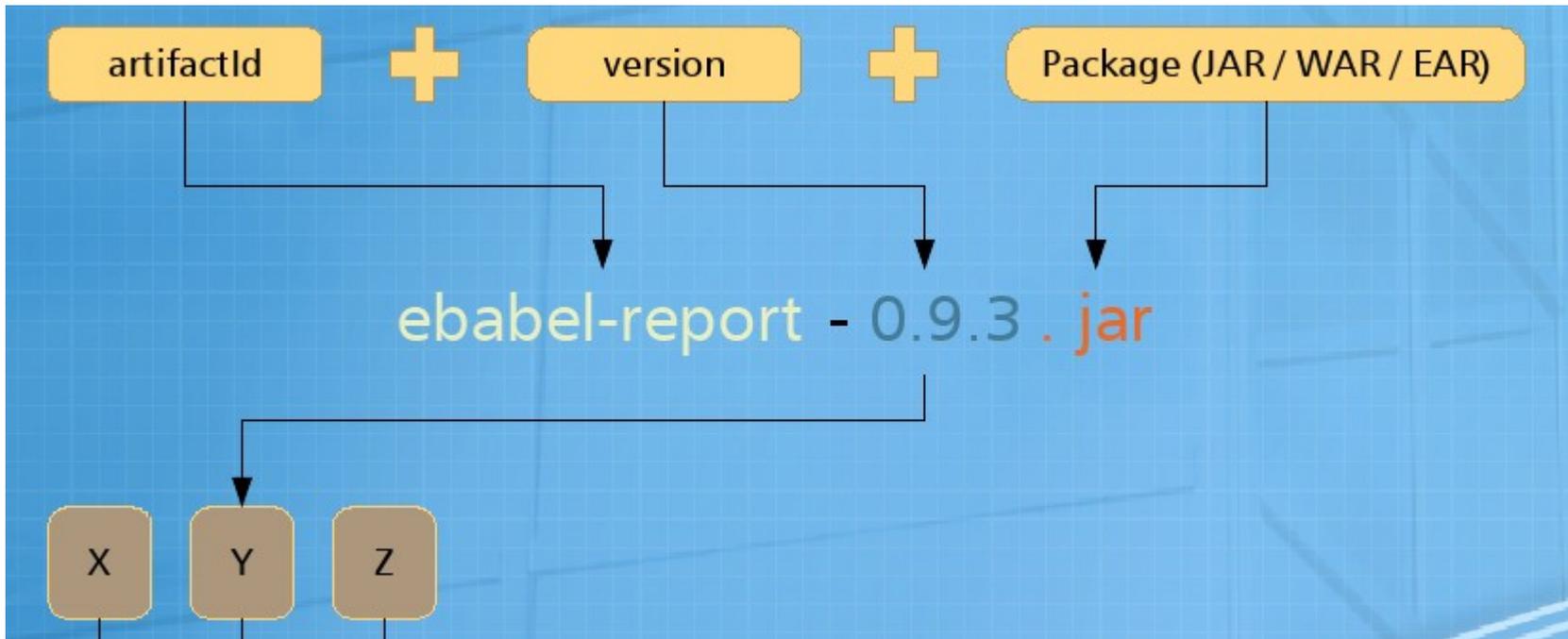
```
<reporting>
  <plugins>
    <plugin>
      <artifactId>maven-clover-plugin</artifactId>
      <configuration>
        [...]
      </configuration>
    </plugin>
    [...]
  </plugins>
</reporting>
```

(B)

```
<build>
  <plugins>
    <plugin>
      <artifactId>maven-clover-plugin</artifactId>
      <executions>
        <execution>
          <execution>
            <id>site</id>
            <phase>pre-site</phase>
            <goals>
              <goal>instrument</goal>
            </goals>
          </execution>
        </executions>
      </plugin>
    [...]
  </build>
```

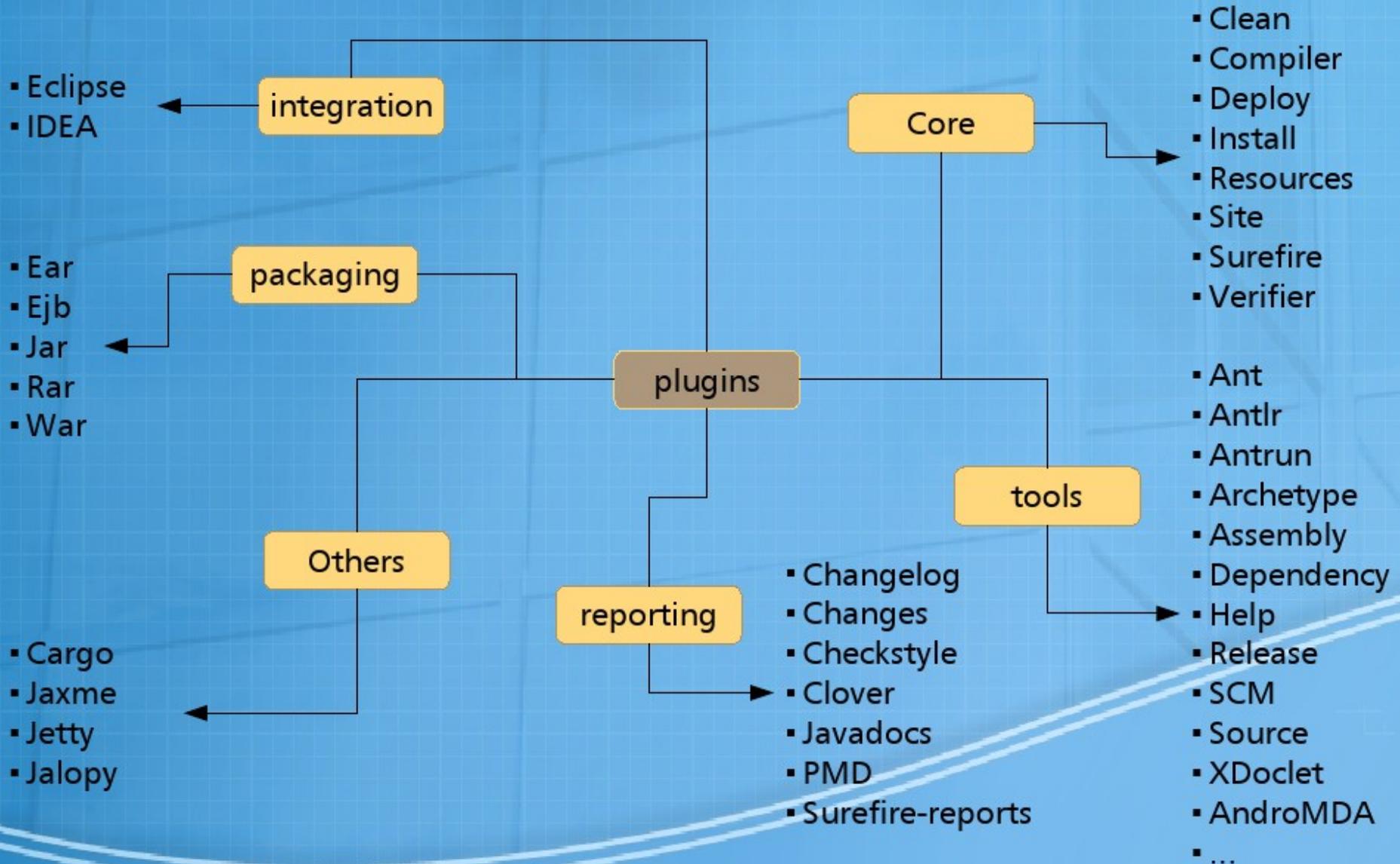
(A)

Versionierung



- X...Major
- Y...Minor
- Z...Revision-BuildNumber

Plugins



<http://maven.apache.org/plugins/index.html>

http://www.manuelrecena.com/docs/maven_061106.pdf

Zusammenfassung

- Vieles gibt es “Out Of The Box”
- Flexibel durch Plugins
- Komplizierter als Ant
- Zwingt zum Arbeiten in “Maven Way”

ENDE

Danke für die Aufmerksamkeit!

???

Fragen

???

Diskussion

Links

- <http://maven.apache.org/>
- <https://www6.software.ibm.com/developerworks/education/j-mavenv2>
- <http://www.mergere.com/better-build-maven>
- http://www.sonatype.com/community/definitive_guide.html
- http://www.manuelrecena.com/docs/maven_061106.pdf
- <http://www.jfrog.org/sites/dep-analyzer/latest/>