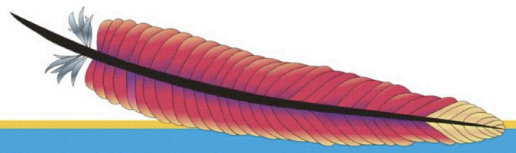
Apache Maven: Best Practices

Wendy Smoak - wsmoak@apache.org

http://people.apache.org/~wsmoak/maven



ApacheCon

Leading the Wave of Open Source

Maven without the PAIN

Maven without the PAIN

- Sometimes unpleasant
- You know it's for your own good!
- Can avoid or alleviate the problems

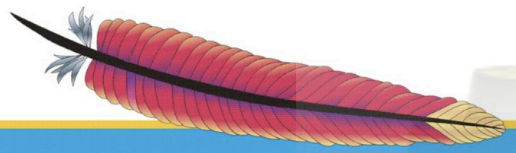
Who?

- Wendy Smoak

  Struts, Tiles, Shale, MyFaces,
  Maven, Continuum, Archiva

- Brett Porter

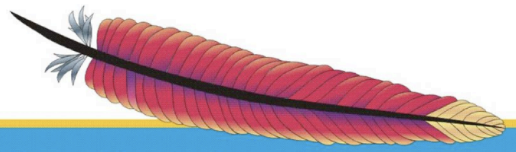
ApacheCon



ApacheCon

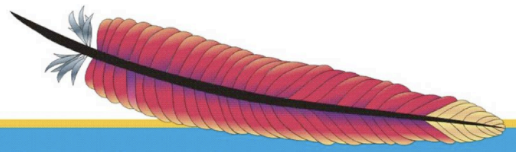
Getting Ready for Maven

Why are you using Maven?

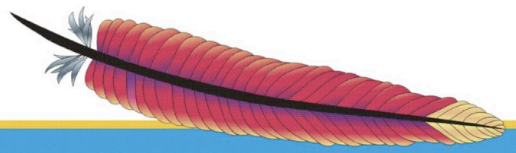


Why are you using Maven?

- Consider this from the beginning
- Right tool for the job
- Ensure you reap the benefits



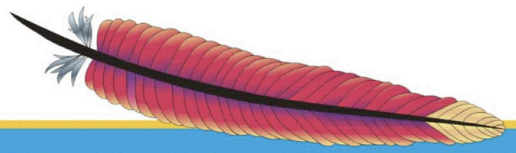
Planning Ahead



ApacheCon

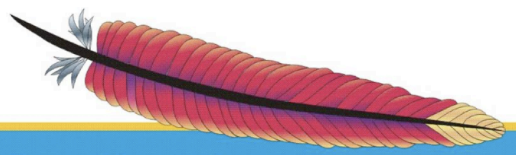
Planning Ahead

Preparing Your Environment



Development Environment

- Maven is best used in the bigger picture
- Plan for the infrastructure you will use

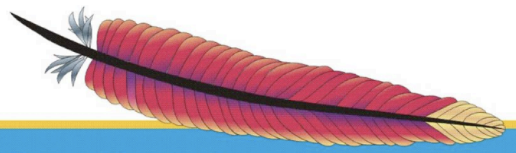


Development Environment

- Maven is best used in the bigger picture
- Plan for the infrastructure you will use

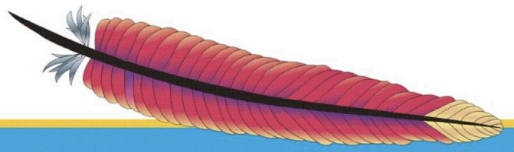
```
<scm>
  <connection>...
  <developerConnection>...
  <url>...
</scm>
```

```
<issueManagement>
  <url>...
  <system>...
</issueManagement>
```



Automated Build Servers

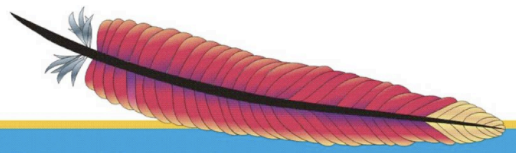
- Maven is intended to be automated
- More than continuous integration



Repository Management

Repository Management

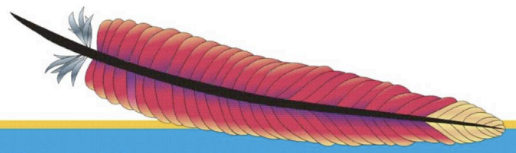
- Centralize storage of artifacts
  - Store the artifacts you build
  - Store third-party artifacts you consume



Repository Servers

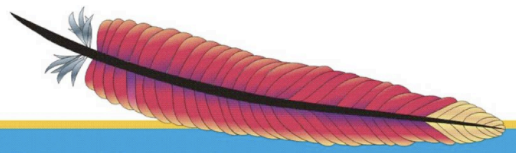
- Can start with simple storage
- Servers can help manage the artifacts

Nexus



Using Repositories

```
<repository>
  <id>apache-snapshots</id>
  <name>Apache Snapshots Repository</name>
  <url>http://people.apache.org/repo/m2-snapshot-repository</url>
  <releases>
    <enabled>false</enabled>
  </releases>
</repository>
```



Using Repositories

```
<repository>
  <id>apache-snapshots</id>
  <name>Apache Snapshots Repository</name>
  <url>http://people.apache.org/repo/m2-snapshot-repository</url>
  <releases>
    <enabled>false</enabled>
  </releases>
</repository>

<repository>
  <id>java-net-m1</id>
  <name>Java.net Repository</name>
  <url>http://download.java.net/maven/1/</url>
  <layout>legacy</layout>
  <snapshots>
    <enabled>false</enabled>
  </snapshots>
</repository>
```

Using Repositories

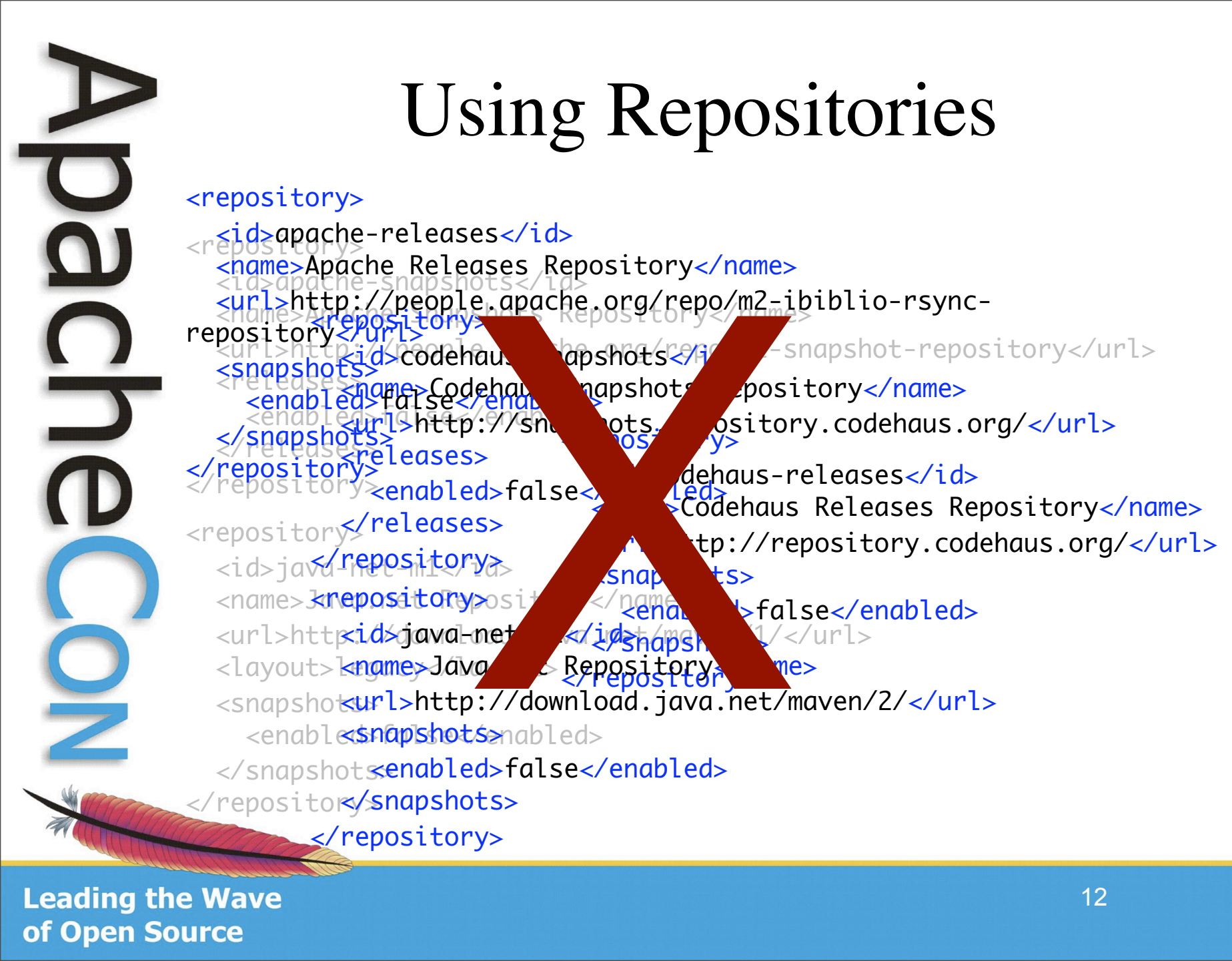
```
<repository>
  <id>apache-snapshots</id>
  <name>Apache Snapshots Repository</name>
  <url>http://people.apache.org/repo/m2-snapshot-repository</url>
  <releases>
    <enabled>false</enabled>
  </releases>
</repository>

<repository>
  <id>java-net-m1</id>
  <name>Java.net Repository</name>
  <url>http://download.java.net/maven/1/</url>
  <layout>legacy</layout>
  <snapshots>
    <enabled>false</enabled>
  </snapshots>
</repository>
```

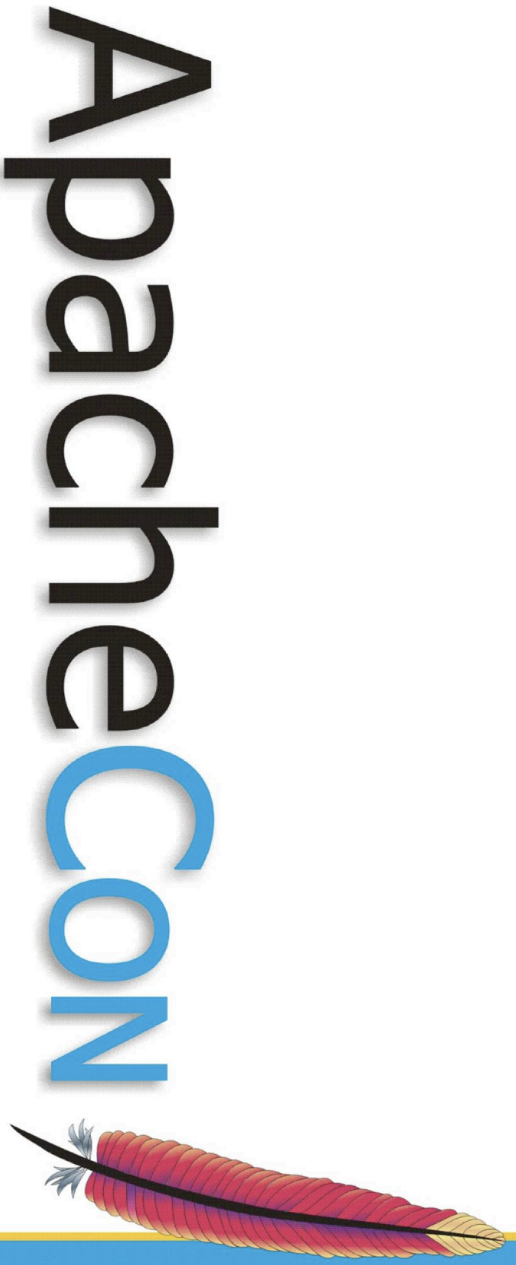
Using Repositories

```
<repository>
  <id>apache-releases</id>
  <name>Apache Releases Repository</name>
  <url>http://people.apache.org/repo/m2-ibiblio-rsync-
repository</url>
  <snapshots>
    <enabled>false</enabled>
  </snapshots>
</repository>
```

```
<repository>
  <id>java-net</id>
  <name>Java.net Repository</name>
  <url>http://download.java.net/maven/2/</url>
  <snapshots>
    <enabled>false</enabled>
  </snapshots>
</repository>
```

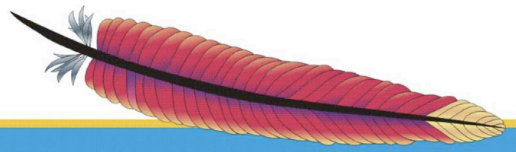


Using Repositories



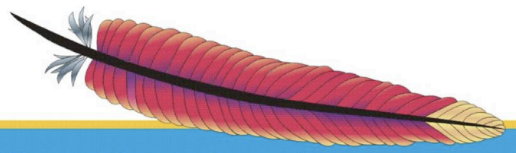
Using Repositories

- Minimize the number of repositories



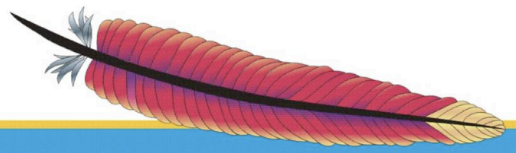
Using Repositories

- Minimize the number of repositories
- Only in the POM if you redistribute



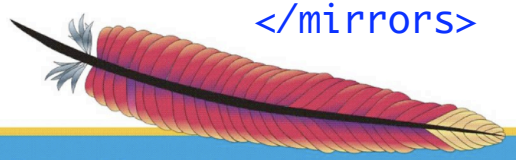
Using Repositories

- Minimize the number of repositories
- Only in the POM if you redistribute
- Use repository manager to centralize



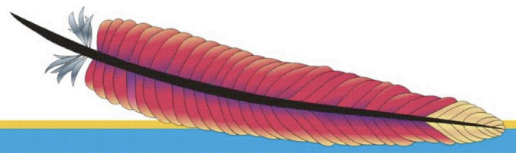
Locking Down

```
<mirrors>
  <mirror>
    <id>archiva-snapshots-apache</id>
    <url>http://reposerver/archiva/repository/snapshots/</url>
    <mirrorOf>apache.snapshots</mirrorOf>
  </mirror>
  <mirror>
    <id>archiva-snapshots-codehaus</id>
    <url>http://reposerver/archiva/repository/snapshots/</url>
    <mirrorOf>codehaus.snapshots</mirrorOf>
  </mirror>
  ...
  <mirror>
    <id>archiva-default</id>
    <url>http://reposerver/archiva/repository/internal/</url>
    <mirrorOf>*</mirrorOf>
  </mirror>
</mirrors>
```



Settings and Installation

- Three levels of configuration
  - project (`pom.xml`)
  - user (`~/.m2/settings.xml`)
  - installation (`<maven>/conf/settings.xml`)



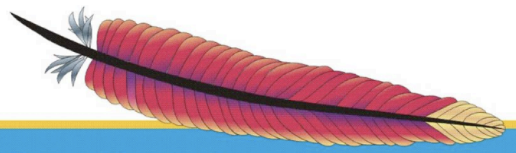

ApacheCon
Keeping it Simple

Keeping it Simple

Creating New Projects

Archetypes

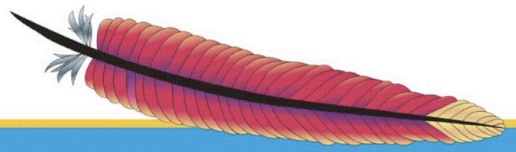
- Create new projects quickly
- Standard project layouts
- Include organizational POM
- Facilitates consistency



Writing the POM

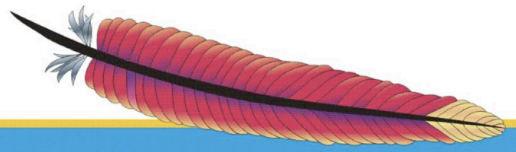
Writing the POM

- Write the build like you write code



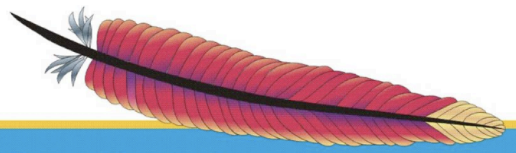
Writing the POM

- Write the build like you write code
- Utilize conventions



Writing the POM

- Write the build like you write code
- Utilize conventions
- Use multiple modules

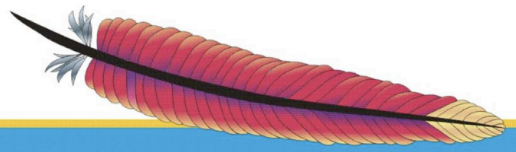


Inheritance

Inheritance

- Multi-module inheritance

```
<parent>
  <groupId>org.apache.maven</groupId>
  <artifactId>maven</artifactId>
  <version>2.1-SNAPSHOT</version>
</parent>
<artifactId>maven-core</artifactId>
```



Inheritance

- Multi-module inheritance

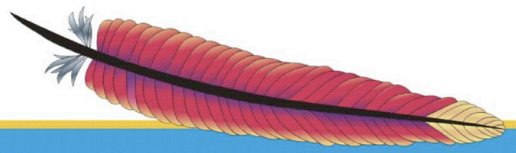
```
<parent>
  <groupId>org.apache.maven</groupId>
  <artifactId>maven</artifactId>
  <version>2.1-SNAPSHOT</version>
</parent>
<artifactId>maven-core</artifactId>
```

- Organisational POM hierarchy

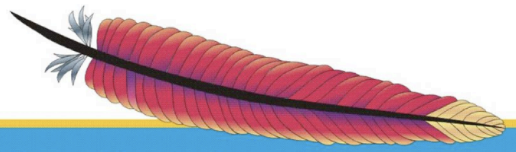
```
<parent>
  <groupId>org.apache</groupId>
  <artifactId>apache</artifactId>
  <version>3</version>
</parent>
<groupId>org.apache.maven</groupId>
<artifactId>maven</artifactId>
<version>2.1-SNAPSHOT</version>
```

Dependencies

- Specify only what you need
- Specify scope
- Use `dependencyManagement` to:
  - coerce Maven to use a particular version
  - enforce consistency within a project

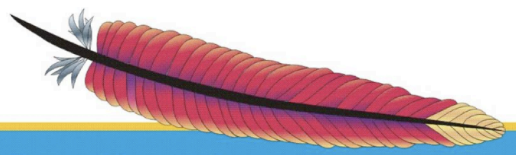


Build Pipeline



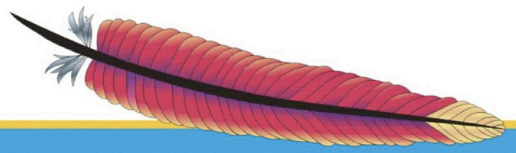
Build Pipeline

- Depends on your team



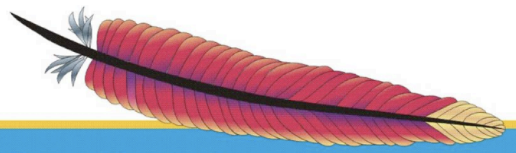
Build Pipeline

- Depends on your team
- Use profiles for controlling complexity



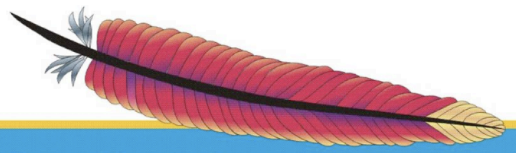
Build Pipeline

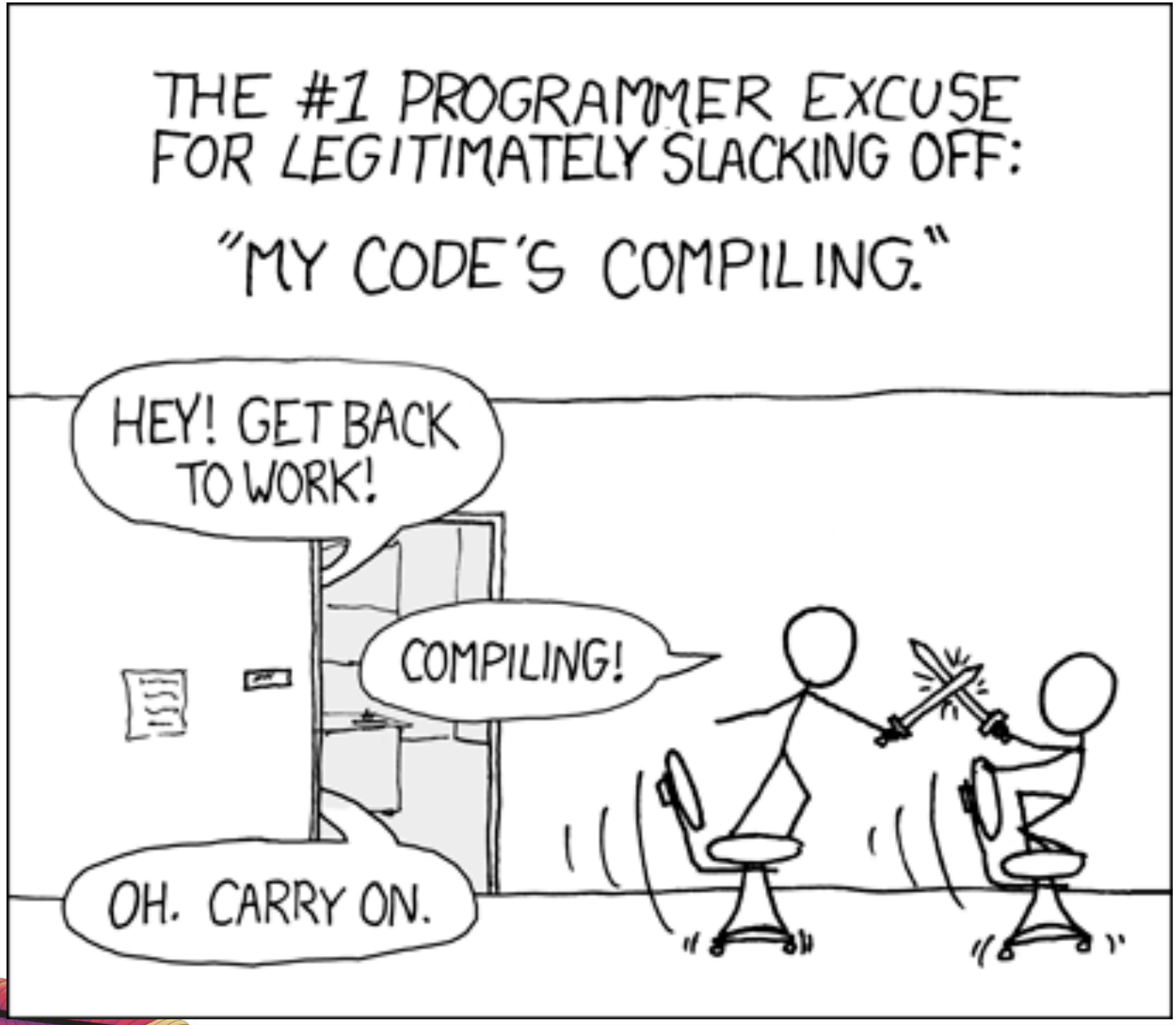
- Depends on your team
- Use profiles for controlling complexity
- Applies to testing as well



Build Pipeline

- Depends on your team
- Use profiles for controlling complexity
- Applies to testing as well
- Key is to keep the build fast

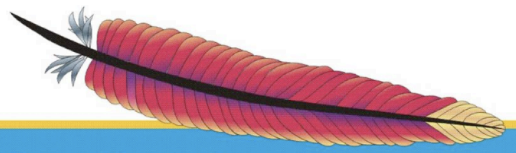



THE #1 PROGRAMMER EXCUSE FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S COMPILING."
HEY! GET BACK TO WORK!
COMPILING!
OH. CARRY ON.

Scripting

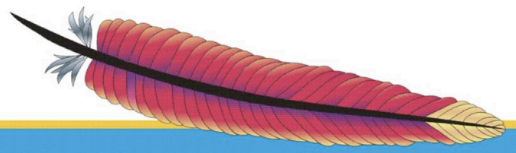
Scripting

- Maven is declarative by design



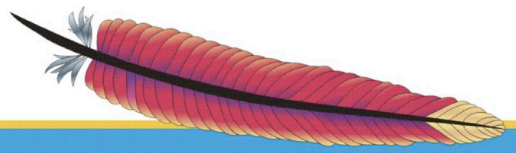
Scripting

- Maven is declarative by design
- Integrate scripting if necessary



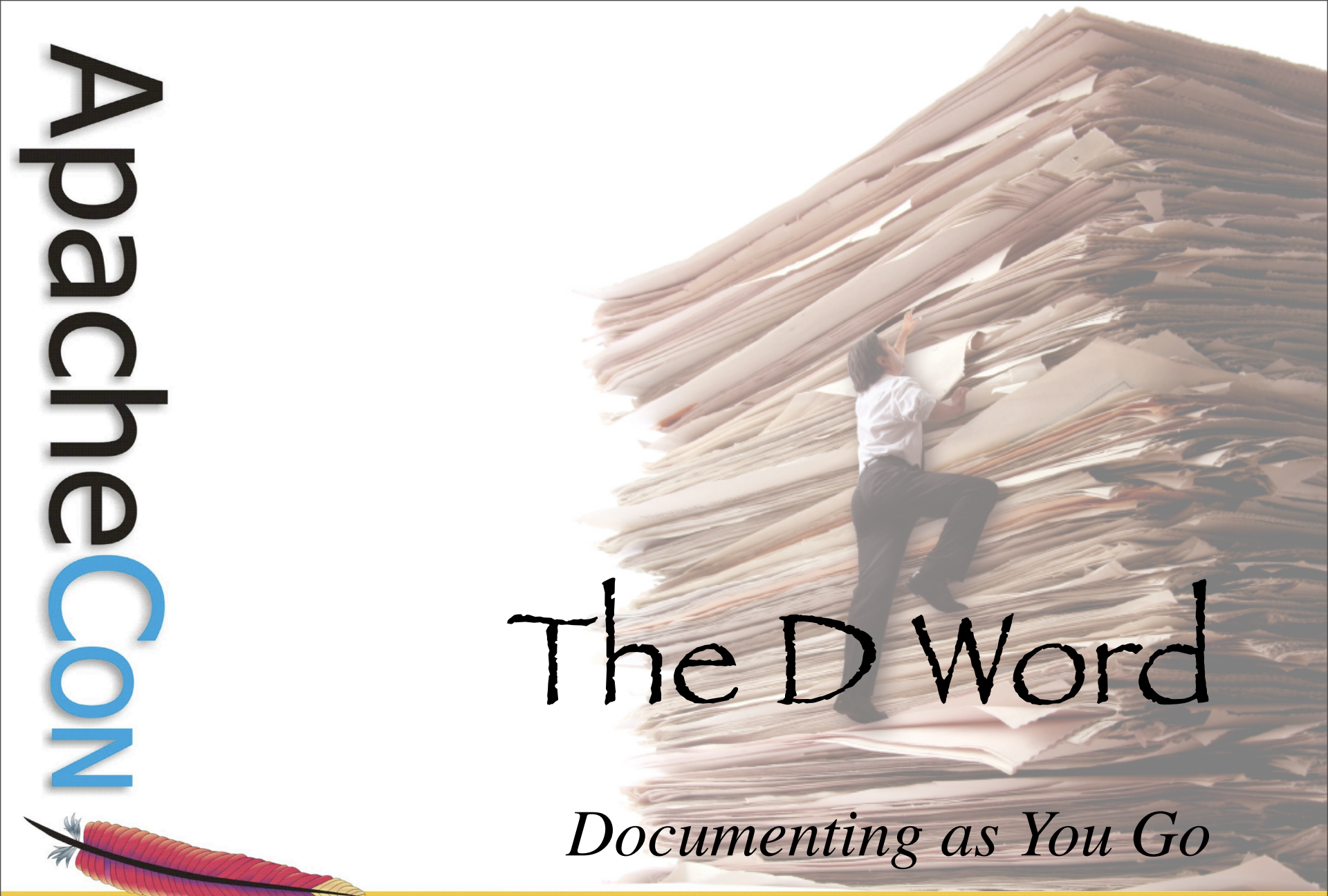
Scripting

- Maven is declarative by design
- Integrate scripting if necessary
- Consider writing plugins



ApacheCon

The D Word

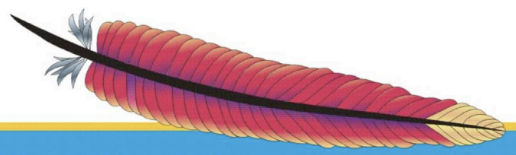


The D Word

Documenting as You Go

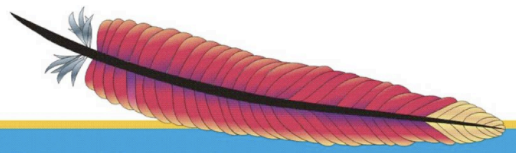
Documentation

- Document the build
- Developer documentation
  - separate from product documentation
  - keep it together with reports

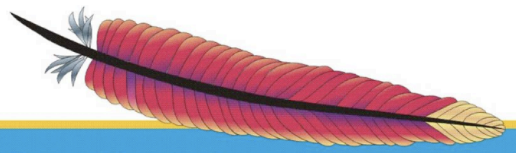


Sites and Reports

- A whole other topic of best practices!
- But apply the same principles
  - set up what you'll actually use
  - set up enforcement checks, not just reports



Portability



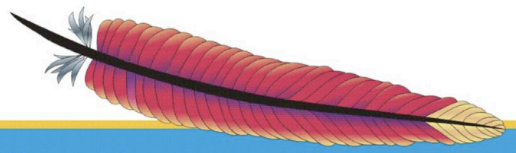
Portability

Playing well with others

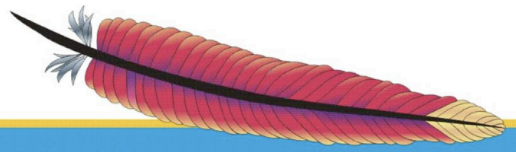
The Goal

The Goal

- When a new developer builds the project
  - it works first go
  - ... and it keeps working

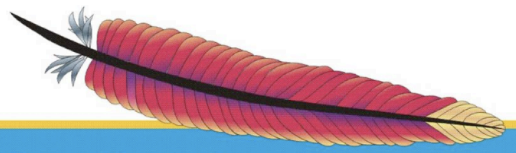


ApacheCON



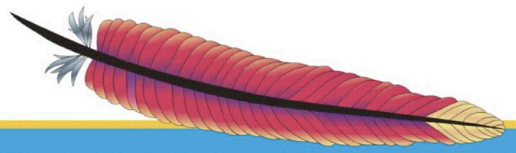
maven.test.skip

is evil



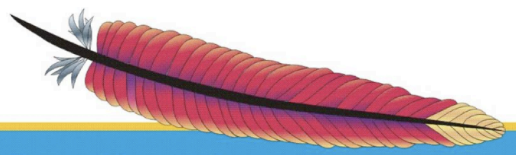
Hard Coding

- Don't hard code **paths**
- Don't hard code **databases**
- Don't hard code **properties**
- Don't do it in the **tests** either



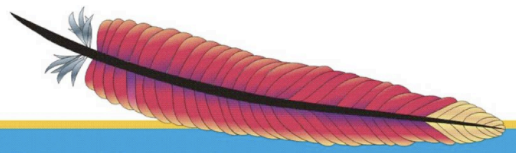
Profiles

- Very useful - but don't abuse them
- Document them all
- Avoid depending on the environment



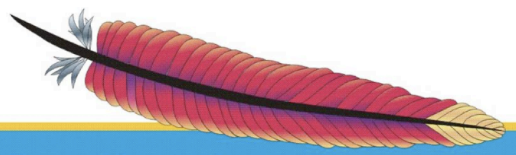
Portable Artifacts

```
<bean
 id="dataSource"
 class="org.springframework.jdbc.datasource.DriverManagerDataSource">
  <property name="driverClassName" value="org.hsqldb.jdbcDriver" />
  <property name="url" value="jdbc:hsqldb:database" />
  <property name="username" value="sa" />
  <property name="password" value="" />
</bean>
```

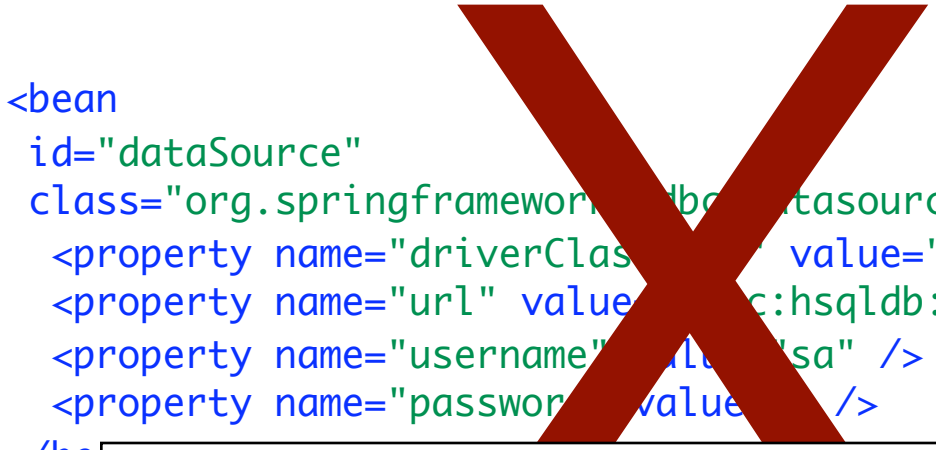


Portable Artifacts

```
<bean
 id="dataSource"
 class="org.springframework.jdbc.datasource.DriverManagerDataSource">
  <property name="driverClassName" value="org.hsqldb.jdbcDriver" />
  <property name="url" value="jdbc:hsqldb:database" />
  <property name="username" value="sa" />
  <property name="password" value="" />
</bean>
```



Portable Artifacts

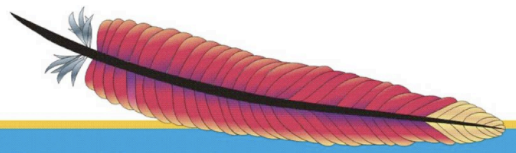


```
<bean
 id="dataSource"
 class="org.springframework...tasource.DriverManagerDataSource">
  <property name="driverClas..." value="org.hsqldb.jdbcDriver" />
  <property name="url" value...c:hsqldb:database" />
  <property name="username"...sa" />
  <property name="passwor...value... />
</be...
```

```
value="org.hsqldb.jdbcDriver"
value="jdbc:hsqldb:database"
```

Portable Artifacts

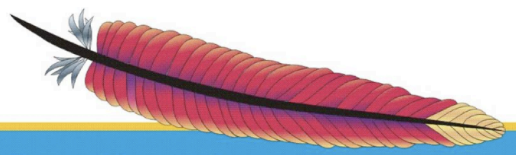
- Artifacts in repository must be unique
- Either by classifier, or being portable
- Recommend externalising configuration
  - database
  - target environment
  - properties



Resource Filtering

Resource Filtering

- Use with great care!

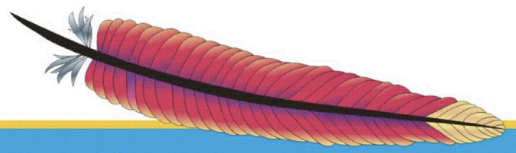


Resource Filtering

- Use with great care!
- Centralization, not substitution

✓ `google.analytics.code=UA-1234567-1`

X `database.password=reallysecretpword`

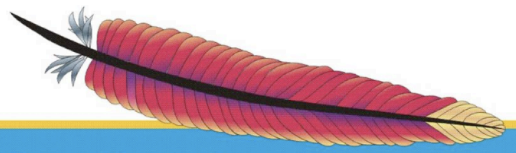


Resource Filtering

- Use with great care!
- Centralization, not substitution

  ✓ `google.analytics.code=UA-1234567-1`

  X `database.password=reallysecretpword`
- Useful for once-off alterations

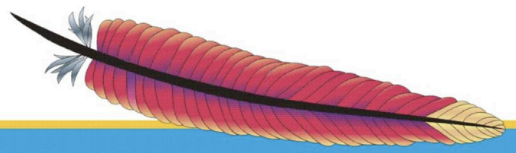


Resource Filtering

- Use with great care!
- Centralization, not substitution

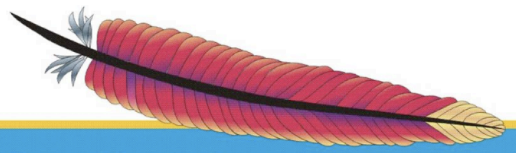
  ✓ `google.analytics.code=UA-1234567-1`

  X `database.password=reallysecretpword`
- Useful for once-off alterations
- Consider externalizing configuration



Shared Resources

- Don’t duplicate resources across projects
- ZIP and put in the repository
- Use the dependency plugin to retrieve



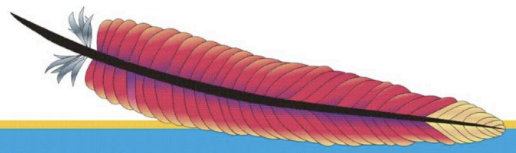

Reproducibility


ApacheCon
Reproducibility
Ensuring Reproducible Builds
Leading the Wave
of Open Source
35

Reproducibility

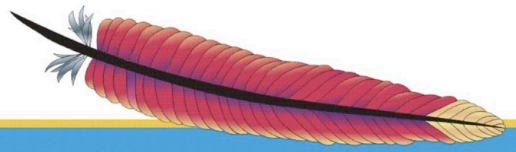
Reproducibility

- Important for releases



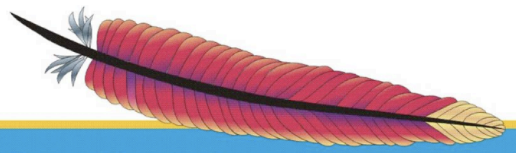
Reproducibility

- Important for releases
- More important for source releases



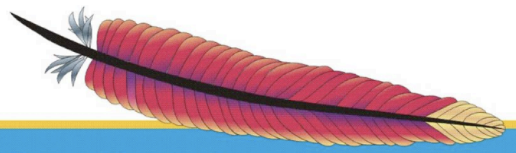
Reproducibility

- Important for releases
- More important for source releases
- Build must be isolated from change



Reproducibility

- Important for releases
- More important for source releases
- Build must be isolated from change
- Portability is a pre-requisite



The Enforcer

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-enforcer-plugin</artifactId>
  <version>1.0-alpha-4</version>
  <executions>
    <execution>
      <goals>
        <goal>enforce</goal>
      </goals>
      <configuration>
        <rules>
          <requirePluginVersions>
            <banLatest>true</banLatest>
            <banRelease>true</banRelease>
          </requirePluginVersions>
        </rules>
      </configuration>
      ...
```

The Enforcer

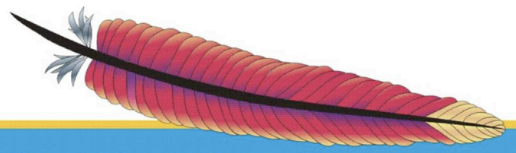
```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-enforcer-plugin</artifactId>
  <version>1.0-alpha-4</version>
  <executions>
    <execution>
      <goals>
        <goal>enforce</goal>
      </goals>
      <configuration>
        <rules>
          <requirePluginVersions>
            <banLatest>true</banLatest>
            <banRelease>true</banRelease>
          </requirePluginVersions>
        </rules>
      </configuration>
      ...
```

The Enforcer

```
<plugin>
  <groupId>org.apache.maven.plugins</groupId>
  <artifactId>maven-enforcer-plugin</artifactId>
  <version>1.0-alpha-4</version>
  <executions>
    <execution>
      <rules>
        <requirePluginVersions>
          <banLatest>true</banLatest>
          <banRelease>true</banRelease>
        </requirePluginVersions>
      </rules>
    </configuration>
    ...
```

The Enforcer

- Help ensure build will be reproducible
- Based on rules
  - force specific plugin versions
  - ban snapshots
  - global exclusions
  - force Maven/Java/OS version
  - can write your own

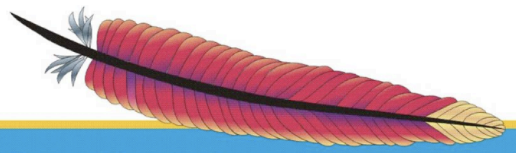



ApacheCon
Releasing


ApacheCon
Releasing
Release Early, Often and Right

Releases

- Set the project version to a `-SNAPSHOT`
- Make them early and often
- Use Maven tools to automate



Questions?

Wendy Smoak - wsmoak@apache.org

http://wsmoak.net/

http://people.apache.org/~wsmoak/maven