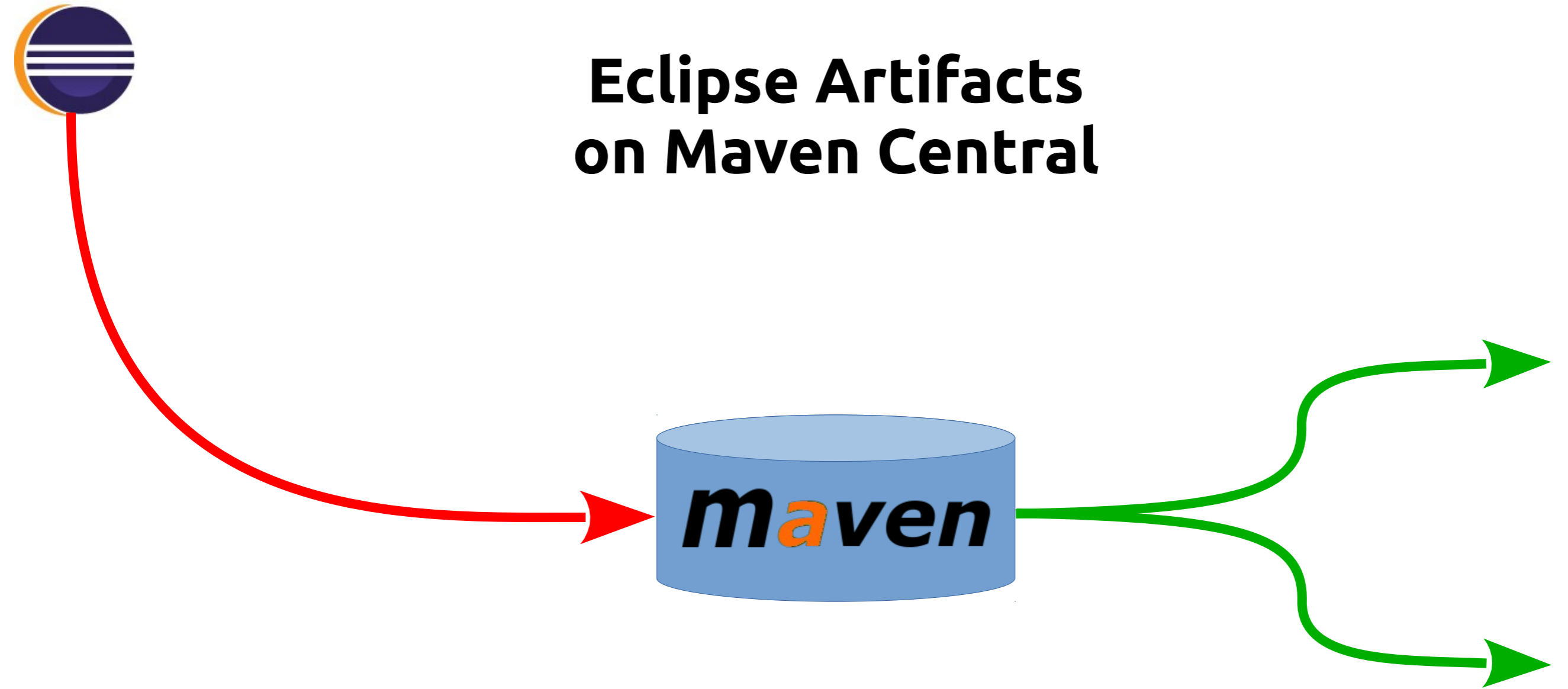


Eclipse Artifacts on Maven Central



Stephan Herrmann



GK SOFTWARE

Simply Retail.



Preface

- › GK Software SE
 - › Adopted Modeling in 2007 based on Xtext 0.7
 - › Generators to be invoked using Maven
 - › Fornax maven plugin was voodoo
 - › Dependency management cargo-cult:
 - manually upload tons of jars to Nexus
 - manually add tons of artifacts to maven plugin dependencies
 - › Builds were frequently broken by changes beyond our control
 - › DSLs were used more and more widely
 - › Upgrading to newer versions was close to impossible



Preface

› GK Software SE

- › Adopted Modeling in 2007 based on Xtext 0.7
- › Generators to be invoked using Maven
- › Fornax maven plugin was voodoo
- › Dependency management
 - manually
 - manually
 - to maven dependencies
- › Builds were frequently broken by changes beyond our control
- › DSLs were used more and more widely
- › Upgrading to newer versions was close to impossible

Angst



Eclipse Artifacts on Maven Central

A Plea against Xenophobia



Eclipse Artifacts on Maven Central

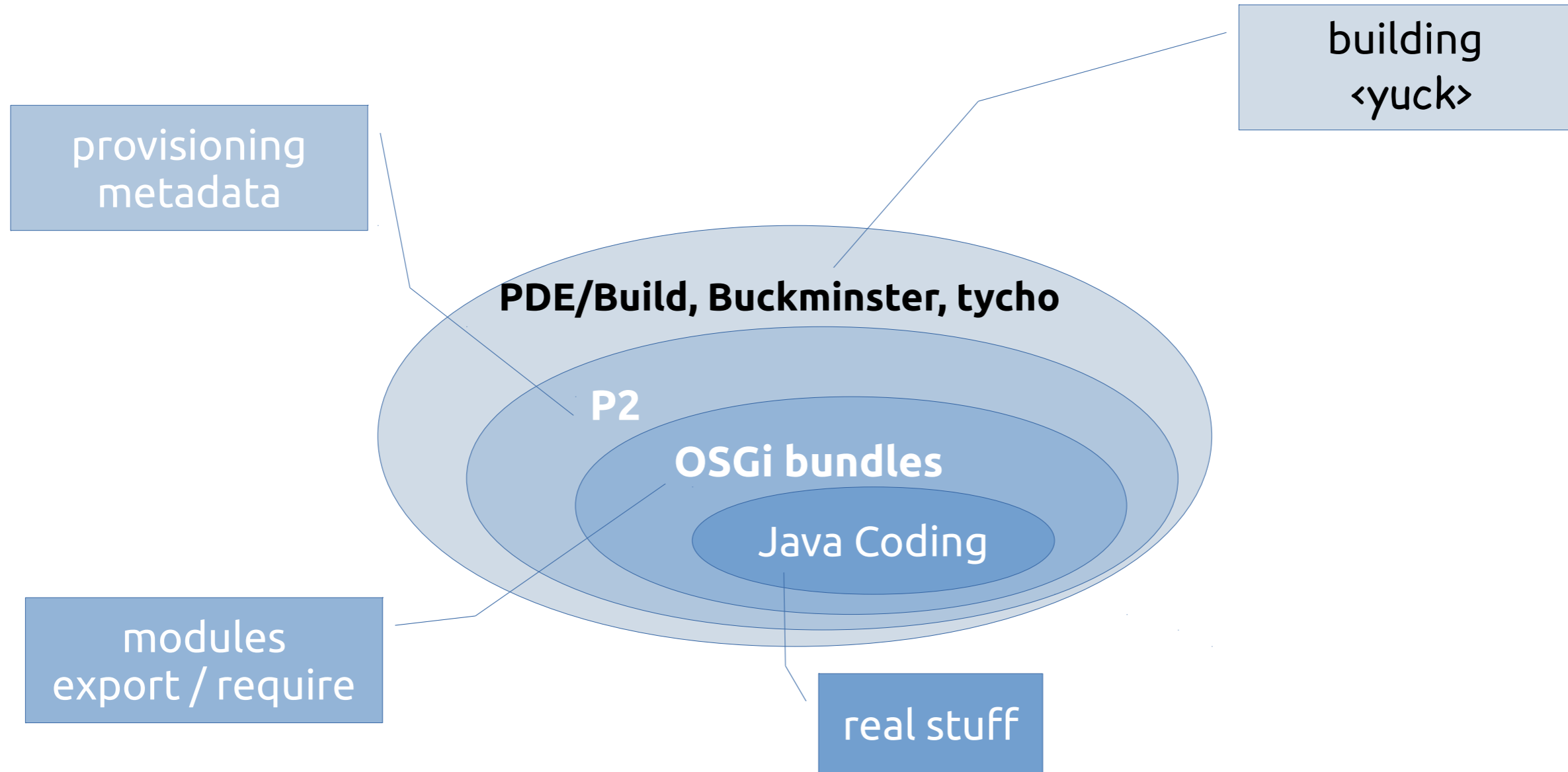
- › Artifact
- › Bundle
- › Plug-in
- › Module
- › Module
- › Feature

?

- › This is a Releng talk
 - › Do you want your stuff to be *consumed*?
- › No Java code!
 - › But this is the cool part!?



Comfort Zones





Comfort Zones

maven

metadata
provisioning
building

...



Welcome to the Real World

- › Stuff may / may not be *intended* for use via Maven
- › Stuff *is used* via Maven



Let's . Just . Do . It .

- › It's all about jar files
 - › Some produce them
 - › Others consume them
- › That's all?



Issues on the Road

- › 1. Extent
- › 2. Coordinates
 - › names
 - › versions
- › 3. Dependencies
 - › ranges
 - › optional, circular, transitive ...
- › 4. More metadata
- › 5. Javadoc & sources
- › 6. Sign, Seal & Deliver

Publishing to Maven Central:
Eclipse Platform
Eclipse JDT
Eclipse PDE



1. Extent

- › Grab any p2 repository
 - › Discard irrelevant stuff, e.g.:
 - features (no representation in Maven)
 - tests (?)
 - gc (stuff that was needed for removed stuff)





2. Coordinates

> Names

> artifactId: bundle symbolic name

> groupId?

- approx. 1st 3 segments:
 - org.eclipse.jdt
 - org.eclipse.pde
 - org.eclipse.platform (artificial name)

> Versions



2. Coordinates

> Names

> artifactId: bundle

> groupId?

- approx. 1st 3 segment
 - org.eclipse.jdt
 - org.eclipse.pde
 - org.eclipse.platform

> Versions

The screenshot shows a web browser window displaying the Sonatype Central Repository search results for the query 'a:org.eclipse.jdt.core'. The search results are presented in a table with two columns: 'Group ID' and 'Artifact ID'. The table lists several entries, including 'org.eclipse.jdt', 'org.eclipse.tycho', 'org.eclipse.scout.sdk.deps', 'com.github.adedayo.eclipse.sdk', 'com.google.code.maven-play-plugin.org.eclipse.jdt', 'com.reubenpeeris', 'org.eclipse.jetty.orbit', 'org.jibx.config.3rdparty.org.eclipse', and 'org.sonatype.tycho', all of which have 'org.eclipse.jdt.core' as the artifact ID.

Group ID	Artifact ID
org.eclipse.jdt	org.eclipse.jdt.core
org.eclipse.tycho	org.eclipse.jdt.core
org.eclipse.scout.sdk.deps	org.eclipse.jdt.core
com.github.adedayo.eclipse.sdk	org.eclipse.jdt.core
com.google.code.maven-play-plugin.org.eclipse.jdt	org.eclipse.jdt.core
com.reubenpeeris	org.eclipse.jdt.core
org.eclipse.jetty.orbit	org.eclipse.jdt.core
org.jibx.config.3rdparty.org.eclipse	org.eclipse.jdt.core
org.sonatype.tycho	org.eclipse.jdt.core



2. Coordinates

> Names

- > artifactId: bundle
- > groupId?

Projects must own their names incl. groupId

- org.eclipse.platform

> Versions

The screenshot shows a web browser window displaying the search results for 'a:org.eclipse.jdt.core' on the Sonatype Maven Central Repository. The search results are displayed in a table with two columns: 'Group ID' and 'Artifact ID'. The search results are as follows:

Group ID	Artifact ID
org.eclipse.jdt.core	org.eclipse.jdt.core
org.eclipse.jdt.core	org.eclipse.jdt.core
org.eclipse.jdt.core	org.eclipse.jdt.core
org.eclipse.jdt.core	org.eclipse.jdt.core
org.eclipse.jdt.core	org.eclipse.jdt.core
com.google.code.maven-play-plugin.org.eclipse.jdt	org.eclipse.jdt.core
com.reubenpeeris	org.eclipse.jdt.core
org.eclipse.jetty.orbit	org.eclipse.jdt.core
org.jibx.config.3rdparty.org.eclipse	org.eclipse.jdt.core
org.sonatype.tycho	org.eclipse.jdt.core



2. Coordinates

- › Names
- › Versions
 - › syntax
 - OSGi: `<major>.<minor>.<service>.<qualifier>`
 - Maven: anything is accepted
 - some formats have specific meaning
 - others are just strings
 - › matching



2. Coordinates

- › Names
- › Versions
 - › syntax
 - › matching
 - real world-ish example:

needed:

g:org.eclipse.core
a:org.eclipse.core.runtime
v:1.2.3

resolved to



g:org.eclipse.core
a:org.eclipse.core.runtime
v:1.2.3.v20140815



2. Coordinates

- › Names
- › Versions
 - › syntax
 - › matching
 - real world-ish example:

needed:

g:org.eclipse.core
a:org.eclipse.core.runtime
v:1.2.3

not resolvable

g:org.eclipse.core
a:org.eclipse.core.runtime
v:1.2.3~~v~~20140815

g:org.eclipse.core
a:org.eclipse.core.runtime
v:1.2.3~~v~~20140815



2. Coordinates

- › Names
- › Versions
 - › syntax – be nice to maven:
 - 3 – part versions for releases
 - some 3rd party artifacts omit service .0
 - *future: snapshots?*
 - › matching

```
junit:junit  
- 4.8  
- 4.8.1  
- 4.8.2  
- 4.9  
- 4.10  
- 4.11  
- 4.12
```



2. Coordinates

- › Names
- › Versions
 - › syntax
 - › matching

Never publish versions with unclear semantics.



3. Dependencies

> Ranges?

- > re-usable library uses ranges (as we do in Eclipse)
- > application freezes all versions
 - Eclipse: target platform
 - Maven: dependencies pom, required with scope "import"

flexible

reproducible

```
<dependency>
  <groupId>com.mycomp</groupId>
  <artifactId>myplatform</artifactId>
  <version>${platform.version}</version>
  <scope>import</scope>
</dependency>
```



3. Dependencies

- › Ranges?
- › Capabilities & Filters
 - › Supported by OSGi & P2
 - › Cannot be modeled in Maven
 - › Example: platform specific fragments



3. Dependencies

- › Ranges?
- › Capabilities & Filters
 - › Supported by OSGi & P2
 - › Cannot be modeled in
 - › Example: platform specific **fragments**





3. Dependencies – fragments

> Bundle

```
Bundle-SymbolicName: org.eclipse.swt
Export-Package:
  org.eclipse.swt,
  org.eclipse.swt.accessibility,
  org.eclipse.swt.awt,
  ...
```

Eclipse-ExtensibleAPI: true

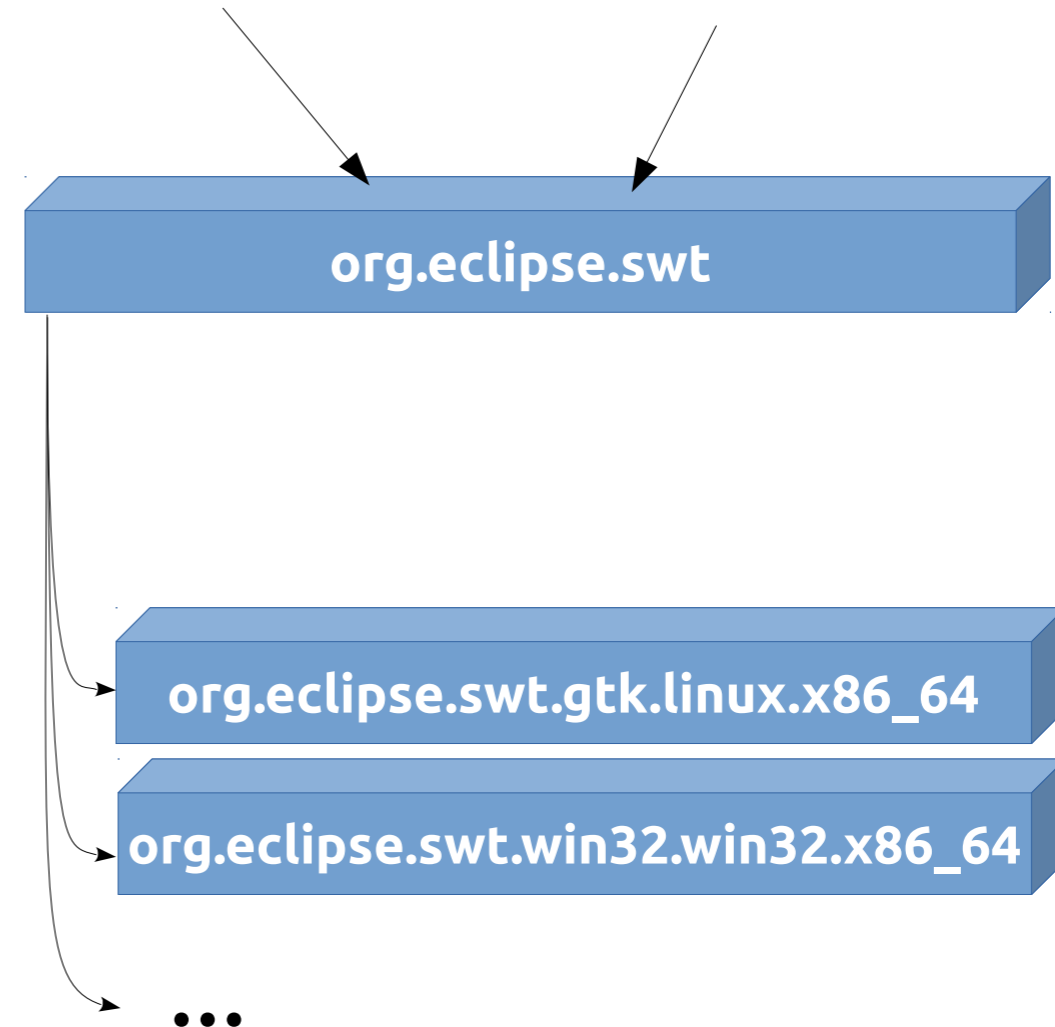
> Fragment(s)

```
Bundle-SymbolicName: org.eclipse.swt
  .gtk.linux.x86_64
```

Fragment-Host: org.eclipse.swt

Eclipse-PlatformFilter: (& (osgi.ws=gtk) (osgi.os=linux) (osgi.arch=x86_64))

```
Export-Package:
  org.eclipse.swt,
  ...
```





3. Dependencies – fragments

› Bundle (p2 metadata)

```
<unit id='org.eclipse.swt' version='3.108.0...'>
<provides size='20'>
  <provided namespace='java.package' name='org.eclipse.swt' /> ...
<requires size='8'>
  <required ... name='org.eclipse.swt.gtk.linux.x86_64' ...>
    <filter>
      (& (osgi.arch=x86_64) (osgi.os=linux) (osgi.ws=gtk)...)
    ...
  ...
```

› Fragment(s) (p2 metadata)

```
<unit id='org.eclipse.swt.gtk.linux.x86_64' version='3.108.0...'>
  <provides size='25'>
    <provided namespace='java.package' name='org.eclipse.swt' /> ...
  <requires size='1'>
    <required namespace='osgi.bundle' name='org.eclipse.swt' /> ...
  <filter>
    (& (osgi.arch=x86_64) (osgi.os=linux) (osgi.ws=gtk)) ...
```




3. Dependencies – fragments

> Bundle (p2 metadata)

```
<unit id='org.eclipse.swt' version='3.108.0... '>  
<provides size='20'>  
  <provided namespace='java.package' name='org.eclipse.swt' /> ...  
<requires size='8'>  
  <required name='org.eclipse.swt.gtk.linux.x86_64' ...>  
    <filter>  
      (&amp; ...  
  ...
```

This is not supported in Maven

```
<unit id='org.eclipse.swt.gtk.linux.x86_64' version='3.108.0... '>  
<provides size='25'>  
  <provided namespace='java.package' name='org.eclipse.swt' /> ...  
<requires size='1'>  
  <required namespace='osgi.bundle' name='org.eclipse.swt' /> ...  
<filter>  
  (& ; (osgi.arch=x86_64) (osgi.os=linux) (osgi.ws=gtk)) ...
```



3. Dependencies – fragments

› Bundle (pom)

```
<artifactId>org.eclipse.swt</artifactId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <artifactId>org.eclipse.swt.${osgi.platform}</artifactId>
```

› Fragment(s) (pom)

```
<artifactId>org.eclipse.swt.gtk.linux.x86_64</artifactId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <artifactId>org.eclipse.swt</artifactId>
```



3. Dependencies – fragments

> Bundle (pom)

```
<aId>org.eclipse.swt</aId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <aId>org.eclipse.swt.${osgi.platform}</aId>
```

> Usage single platform

> Fragment(s) (pom)

```
<aId>org.eclipse.swt.gtk.linux.x86_64</aId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <aId>org.eclipse.swt</aId>
```



dependency



3. Dependencies – fragments

> Bundle (pom)

```
<aId>org.eclipse.swt</aId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <aId>org.eclipse.swt.${osgi.platform}</aId>
```

> Usage single platform

```
<exclusion>  
  <groupId>org.eclipse.swt</groupId>  
  <artifactId>org.eclipse.swt</artifactId>  
</exclusion>
```

> Fragment(s) (pom)

```
<aId>org.eclipse.swt.gtk.linux.x86_64</aId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <aId>org.eclipse.swt</aId>
```



dependency



3. Dependencies – fragments

> Bundle (pom)

```
<aId>org.eclipse.swt</aId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <aId>org.eclipse.swt.${osgi.platform}</aId>
```

> Usage multi platform (recommended)



dependency

> Fragment(s) (pom)

```
<aId>org.eclipse.swt.gtk.linux.x86_64</aId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <aId>org.eclipse.swt</aId>
```

```
$ mvn -Dosgi.platform=gtk.linux.x86_64 ...
```



3. Dependencies – fragments

> Bundle (pom)

```
<aId>org.eclipse.swt</aId>  
<version>3.108.0...</version>  
<dependencies><dependency>  
  <aId>org.eclipse.swt.${osgi.platform}</aId>
```

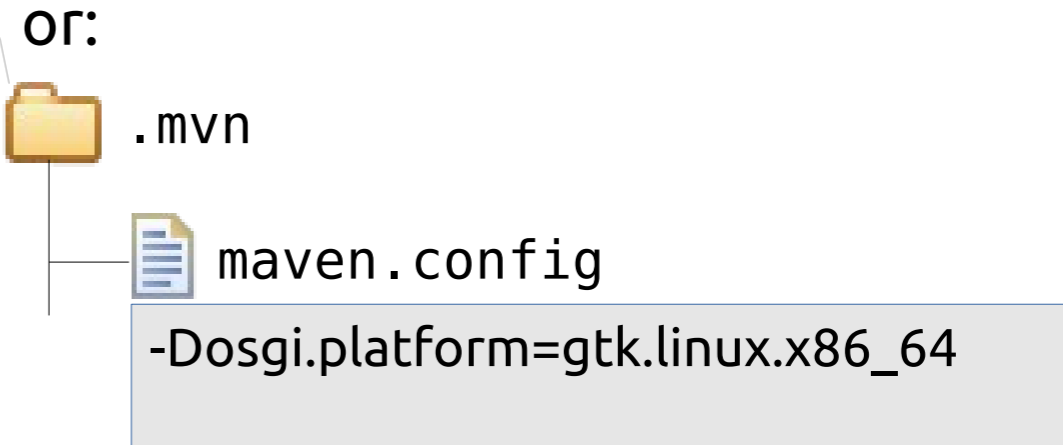
.mvn folder:
Located with in the projects top level folder, the files maven.config and extensions.xml contain project specific configuration for running Maven.

```
<dependencies><dependency>  
  <aId>org.eclipse.swt</aId>
```

> Usage multi platform (recommended)



```
$ mvn ...
```





4. More Metadata

› Required (by OSSRH et al.):

› **constant content**

- copyright
- license
- organization
- issue management

Sonatype OSS Repository Hosting

› **per-artifact content** (retrieved mostly from MANIFEST.MF):

- name (*typically matches what the aggregator filled in as description*)
- url
- scm connection, tag and url

› **semi constant content**

- developers (*"Who's Involved" page of corresponding project in PMI*)



```
<?xml version="1.0" encoding="UTF-8"?>
<!--
  Copyright (c) 2016, 2018 GK Software SE and others. ... -->
<project xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" ...>
  <modelVersion>4.0.0</modelVersion>
  <groupId>org.eclipse.jdt</groupId>
  <artifactId>org.eclipse.jdt.core</artifactId>
  <version>3.15.0</version>
  <description>Java Development Tools Core</description>
  <name>Java Development Tools Core</name>
  <url>http://www.eclipse.org/jdt</url>
  <licenses><license>
    <name>Eclipse Public License</name>
    <url>http://www.eclipse.org/legal/epl-v10.html</url>
    <distribution>repo</distribution></license></licenses>
  <organization>
    <name>Eclipse Foundation</name>
    <url>http://www.eclipse.org/</url></organization>
  <issueManagement>
    <system>Bugzilla</system>
    <url>https://bugs.eclipse.org/</url></issueManagement>
  <scm>
    <connection>scm:git:git://git.eclipse.org/gitroot/jdt/eclipse.jdt.core.gi...
    <tag>I20180905-0800</tag>
    <url>https://git.eclipse.org/c/jdt/eclipse.jdt.core.git</url></scm>
  <developers><developer>
    <url>https://projects.eclipse.org/projects/eclipse.jdt/who</url></devel...
```




5. More Artifacts

› Required (by OSSRH et al.):

› **javadoc**

- we have .doc bundles
 - no 1:1 correspondence
- create .javadoc jar on the fly:
 - create README.txt
 - refer to the corresponding doc bundle

› **sources**

- most projects correctly build .sources
 - last remaining offender: `org.eclipse.core.net.*`
 - source bundle created on the fly (ad-hoc)



6. Sign, Seal & Deliver

- › Sign & Upload
 - › **maven-gpg-plugin**
 - keys howto: ask webmaster :)
 - goal: `gpg:sign-and-deploy-file`
- › Release
 - › **Via Web interface of OSSRH Nexus**
 - “close” => runs validations
 - “release” & “drop”



Tools Used (status quo)

- › 1. Extent (rm & gc) — bash script
- › 2. Coordinates
 - › names (groupid)
 - › versions (format)
- › 3. Dependencies
 - › ranges (fragments)
 - › optional, circular, transitive ...
- › 4. More metadata — Java (< 500 LOC)
- › 5. Javadoc & sources — bash script
- › 6. Sign, Seal & Deliver — mvn



Tools Used (perspective)

