Maven/Tycho für Eclipse RCP-Applikationen
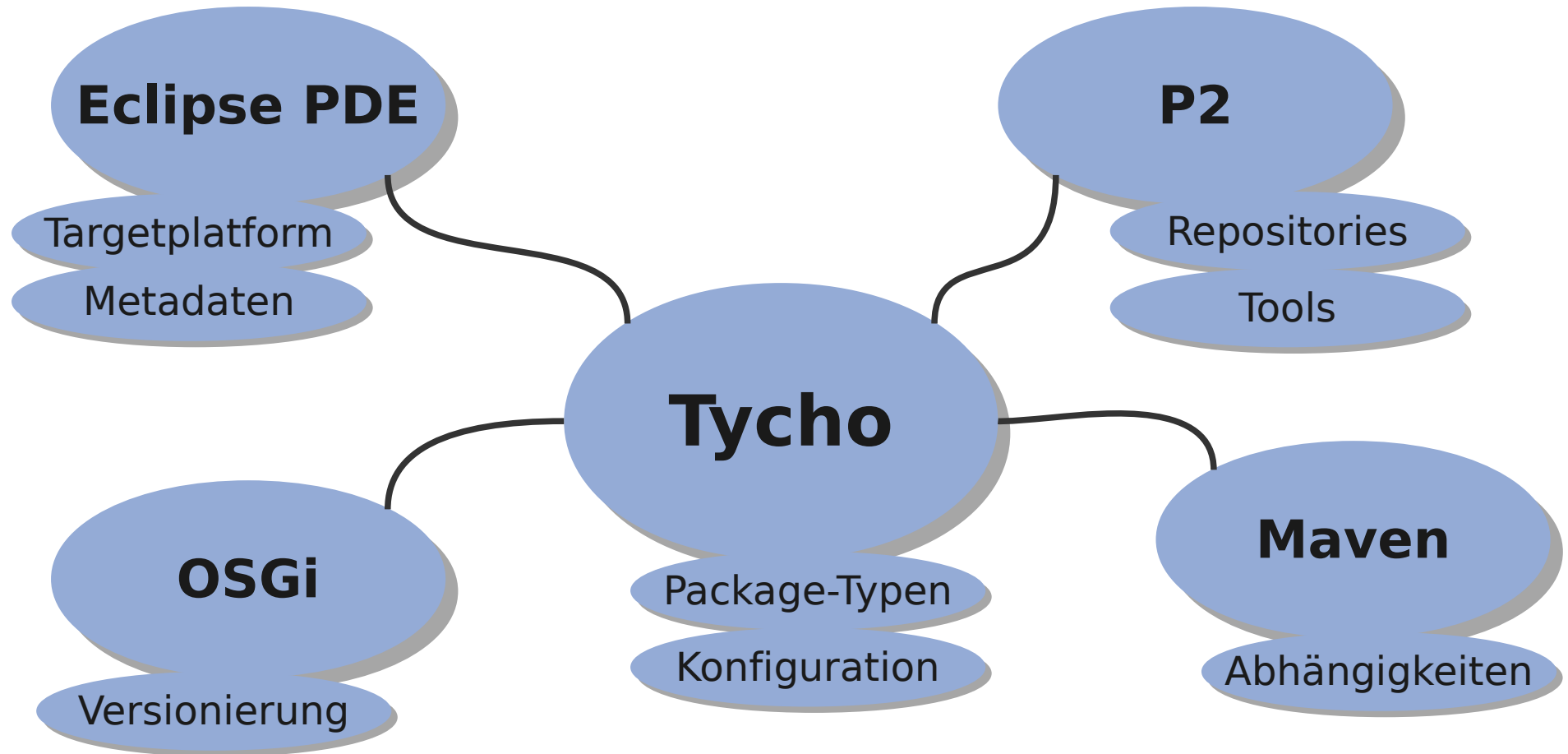
Das neue Build-Werkzeug im Eclipse-Universum

Beat Strasser – Inventage

:inventage

Themen

:inventage

www.eclipse.org/tycho

Tycho: Set von Maven3-Plugins

> `org.eclipse.tycho:tycho-*-plugin:0.13.0`

> In Maven Central verfügbar
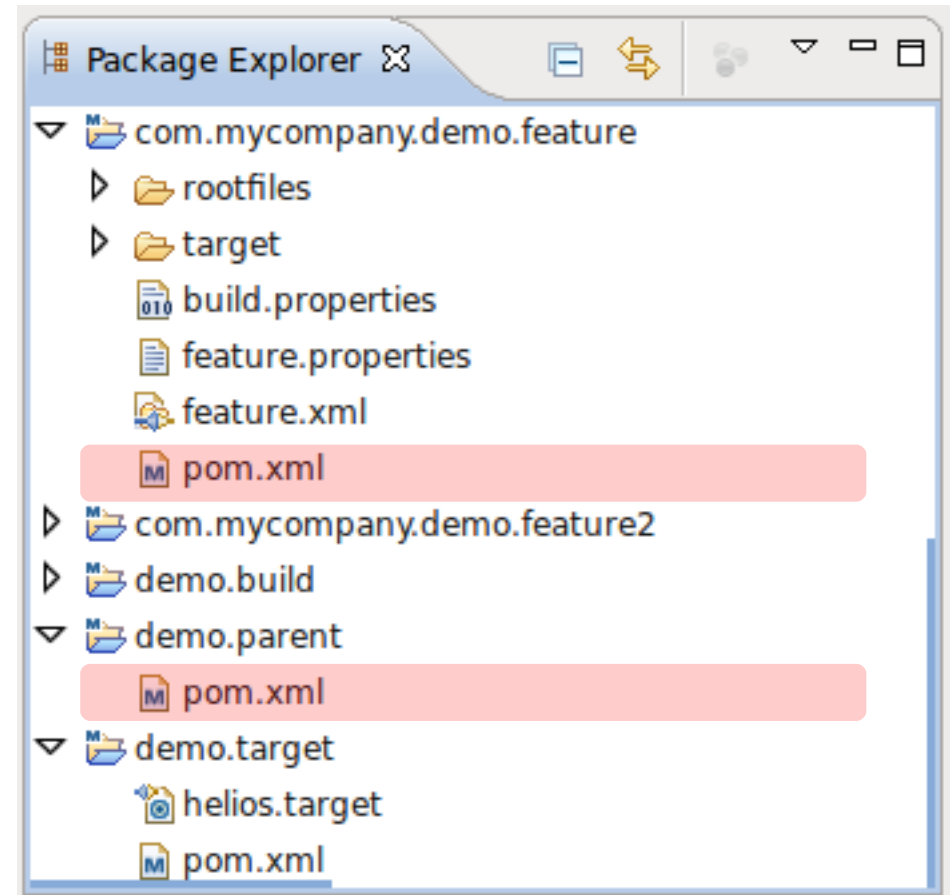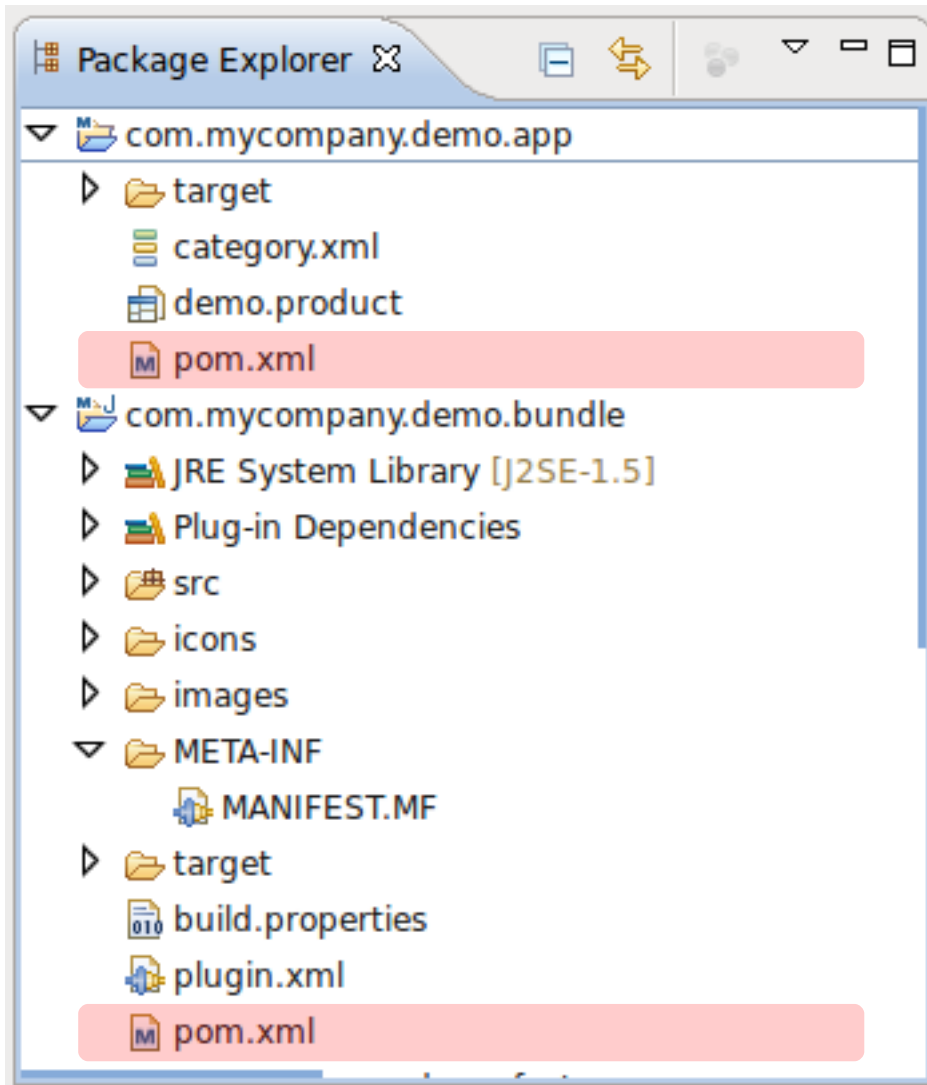
Konkurrent zu PDE-Headless-Build

> Verarbeitet OSGi- und PDE-Metadaten

> Erweitert Maven-Abhängigkeits-Modell um OSGi-/Eclipse-RCP-Abhängigkeiten

> P2-Repository-Support für Maven

Hauptsächlich Manifest-first

:inventage

Projekt-Struktur

Package Explorer ⊠

- ▽ com.mycompany.demo.app
  - ▷ target
  - category.xml
  - demo.product
  - pom.xml
- ▽ com.mycompany.demo.bundle
  - ▷ JRE System Library [J2SE-1.5]
  - ▷ Plug-in Dependencies
  - ▷ src
  - ▷ icons
  - ▷ images
  - ▽ META-INF
    - MANIFEST.MF
  - ▷ target
  - build.properties
  - plugin.xml
  - pom.xml

Package Explorer ⊠

- ▽ com.mycompany.demo.feature
  - ▷ rootfiles
  - ▷ target
  - build.properties
  - feature.properties
  - feature.xml
  - pom.xml
- ▷ com.mycompany.demo.feature2
- ▷ demo.build
- ▽ demo.parent
  - pom.xml
- ▽ demo.target
  - helios.target
  - pom.xml

JUGS: Maven/Tycho für Eclipse RCP-Applikationen

:inventage

Tycho-Konfiguration

```xml
<repositories><repository>
  <id>eclipse-indigo</id>
  <layout>p2</layout>
  <url>http://download.releases.org/releases/indigo/</url>
</repository></repositories>

<build><plugins>
 <plugin>
    <groupId>org.eclipse.tycho</groupId>
    <artifactId>tycho-maven-plugin</artifactId>
    <version>0.13.0</version>
    <extensions>true</extensions>
 </plugin>

 <plugin>
  <groupId>org.eclipse.tycho</groupId>
  <artifactId>target-platform-configuration</artifactId>
  <version>0.13.0</version>
  <configuration>
```

```xml
<environments>
 <environment>
    <os>linux</os>
    <ws>gtk</ws>
    <arch>x86_64</arch>
 </environment>
 <environment>
    <os>win32</os>
    <ws>win32</ws>
    <arch>x86_64</arch>
 </environment>
 <environment>
    <os>macosx</os>
    <ws>cocoa</ws>
    <arch>x86_64</arch>
 </environment>
</environments>
  </configuration>
 </plugin>
</plugins></build>
```

JUGS: Maven/Tycho für Eclipse RCP-Applikationen

:inventage

Plugins, Fragmente, Features

Plugins, Fragmente: **eclipse-plugin**

Test-Fragmente: **eclipse-test-plugin**

Eclipse-Features: **eclipse-feature**

Artifakt-Id und -Version analog zu PDE/OSGi

POM-Dateien generieren:

```
mvn org.eclipse.tycho:tycho-pomgenerator-
plugin:generate-poms -DgroupId=...
```

:inventage

Eclipse RCP-Applikation

```
<packaging>eclipse-repository</packaging>
```

```
<build><plugins><plugin>
  <groupId>org.eclipse.tycho</groupId>
  <artifactId>tycho-p2-director-plugin</artifactId>
  <version>0.13.0</version>
  <executions>
    <execution>
      <id>materialize-products</id>
      <goals><goal>materialize-products</goal></goals>
    </execution>
    <execution>
      <id>archive-products</id>
      <goals><goal>archive-products</goal></goals>
    </execution>
  </executions>
</plugin></plugins></build>
```

:inventage

Spezial-Konfiguration

Support für Zusatzdateien analog zu PDE

build.properties in Feature:

```
root = file:README.txt
root.win32.win32.x86 = rootfiles/win
root.linux.gtk.x86 = rootfiles/linux
root.linux.gtk.x86.permissions.755 = scripts/hello
root.linux.gtk.x86.link = scripts/hello,hello_alias
```

Compiler-Optionen: **tycho-compiler-plugin**
Packaging-Optionen: **tycho-packaging-plugin**
Source-Optionen: **tycho-source-plugin**
Test-Optionen: **tycho-surefire-plugin**

: **inven**tage

Abhängigkeiten bei Eclipse PDE – Targetplatform
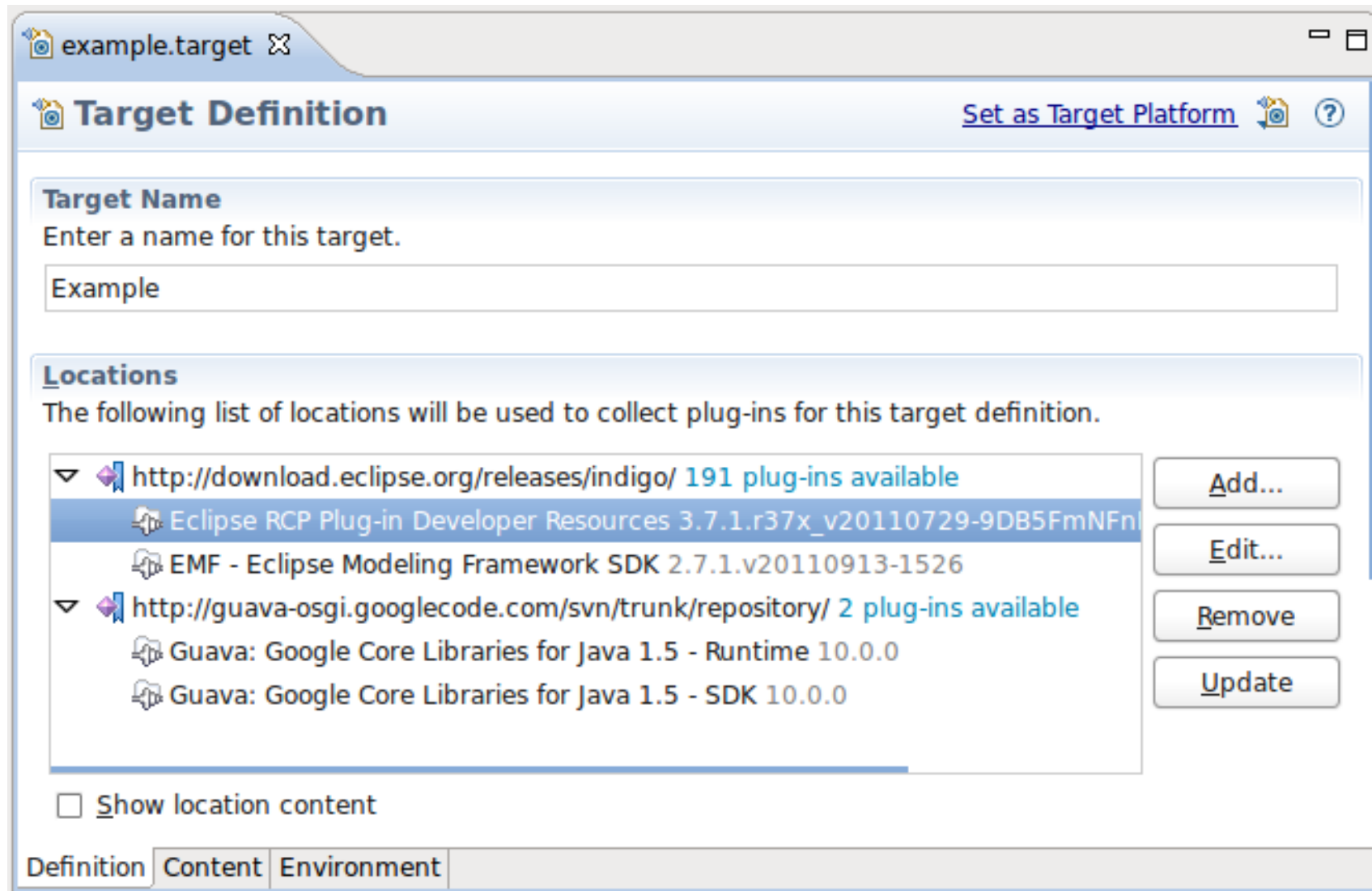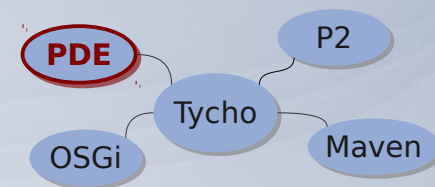
Abhängigkeiten bei Tycho

Unzureichende Metadaten in Maven-Repositories

- Manifest-first (default): Dependencies in POM werden nicht beachtet

Targetplatform-Definition durch

- Angabe aller benötigten P2-Repositories (Layout p2)
- Angabe einer PDE-Target-Definitionsdatei

Verwaltung pro Maven-Modul

`tycho.targetplatform`-Property nicht mehr benutzen
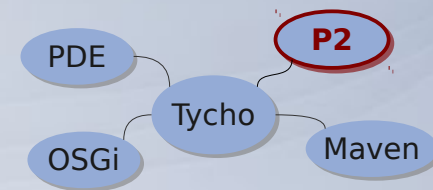
Lokales Maven-Repository dient auch als P2-Cache

Targetdefinition verwenden in Tycho

Eigenes Artifact für Target-Definition erstellen (z.B. mit org.codehaus.mojo:build-helper-maven-plugin:attach-artifact)

Tycho-Konfiguration:

```xml
<build><plugins><plugin>
  <groupId>org.eclipse.tycho</groupId>
  <artifactId>target-platform-configuration</artifactId>
  <version>0.13.0</version>
  <configuration>
    <target>
      <artifact>
        <groupId>com.example</groupId>
        <artifactId>example.target</artifactId>
        <version>1</version>
        <classifier>example</classifier>
      </artifact>
    </target>
  </configuration>
</plugin></plugins></build>
```
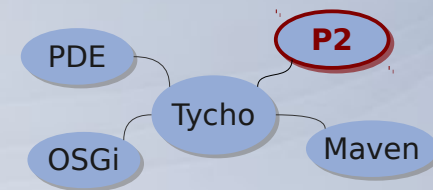
:inventage

Eclipse Software-Verteilungs-Plattform mit Beschreibungen für Installable Units

> **P2 Core**
>
> **P2 UI**
>
> **P2 Repository** (statisch)
>
> > content.xml
> >
> > artifacts.xml

# Zusammengesetztes P2-Repository

```xml
<?xml version='1.0' encoding='UTF-8'?>
<?compositeMetadataRepository version='1.0.0'?>
<repository name='Demo Repository'
    type='org.eclipse.equinox.internal.p2.metadata.repository.
            CompositeMetadataRepository'
    version='1.0.0'>
  <properties size='1'>
    <property name='p2.timestamp' value='1'/>
  </properties>
  <children size='3'>
    <child location='http://download.eclipse.org/releases/indigo/'/>
    <child location='http://guava-osgi.googlecode.com/svn/trunk/repository/'/>
    <child location='other'/>
  </children>
</repository>
```

```
p2repo/
├── compositeArtifacts.xml
├── compositeContent.xml
└── other/
        ├── artifacts.jar
        ├── content.jar
        ├── features/
        └── plugins/
```

Publizieren von IUs
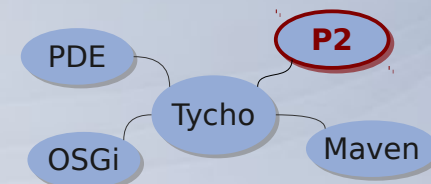
```
java -jar ~/.m2/repository/org/eclipse/tycho/tycho-p2-
runtime/0.13.0/eclipse/plugins/org.eclipse.equinox.launch
er_1.2.0.v201110502.jar
   -application org.eclipse.equinox.p2.publisher.
             FeaturesAndBundlesPublisher
   -metadataRepository file:/tmp/myNewRepo
   -artifactRepository file:/tmp/myNewRepo
   -source file:/tmp/bundles
   -metadataRepositoryName "My new repo"
   -artifactRepositoryName "My new repo"
   -publishArtifacts -compress -append


<p2.publish.featuresAndBundles
   metadatarepository="file:/tmp/libs"
   metadatarepositoryname="My new repo"
   artifactrepository="file:/tmp/libs"
   artifactrepositoryname="My new repo"
   source="/tmp/bundles"
   compress="true" append="true"
   publishartifacts="true" />
```

:inventage

Nexus Repository Manager
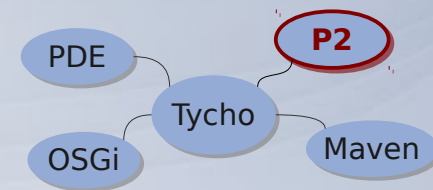
P2-Proxies und P2-Gruppen

  seit September 2011 auch in Nexus OSS!

Experimentelle Features (OSS):

  nexus-p2-repository-plugin:
  P2-Ansicht aller OSGi-Bundles eines Maven-
  Repos

  nexus-bundle-maker-plugin:
  Automatisches OSGi-fizieren von Artifakten

:inventage

Versionierung

Maven

Snapshot- und Release-Versionen

OSGi

Nur Release-Versionen

Versions-Semantik – major.minor.micro.qualifier

Package-Versionierung

Tycho

Zweifache Versionsangabe → konsistent halten

```
1.2.3-SNAPSHOT ≙ 1.2.3.qualifier
```

```
tycho-versions-plugin:set-version
    -DnewVersion=1.2.3-SNAPSHOT -Dartifacts=...
```

:inventage

Versionsnummer-(In)Konsistenz

Maven: `1.2.3-SNAPSHOT < 1.2.3`

OSGi: `1.2.3.20111013 > 1.2.3`

Lösungsansätze:

Qualifier auch für Releases (Timestamp, FINAL, RELEASE, STABLE, …)

Gerade/ungerade Nummern für Release/Snapshot

OSGi Draft: negative Qualifier
`1.2.3-20111013 < 1.2.3 < 1.2.3.20111013`

:inventage

Eclipse-API-Tooling

Fazit

Maven-Build

- Einheitlicher Build
- Einfacher Setup, wenig Konfiguration
- Guter PDE-Support
- Transitive Abhängigkeiten nicht ersichtlich über Tycho-/Maven-Grenze
- Maven-Repositories ungebraucht
- Versionierung mühsam

P2

- Repository-Management wenig effizient
- P2-Core-Tools ohne Offline-Support

http://www.eclipse.org/tycho/

http://wiki.eclipse.org/Category:Tycho

Mailing-Listen:

tycho-user@eclipse.org

tycho-dev@eclipse.org

git://git.eclipse.org/gitroot/tycho/org.eclipse.tycho.git

EclipseCon Europe (2.-4.11.2011)