
Eclipse-Plug-in Tutorial

Software-Architektur 1, WS 2011/12 (Holger Gast, Universität Tübingen)

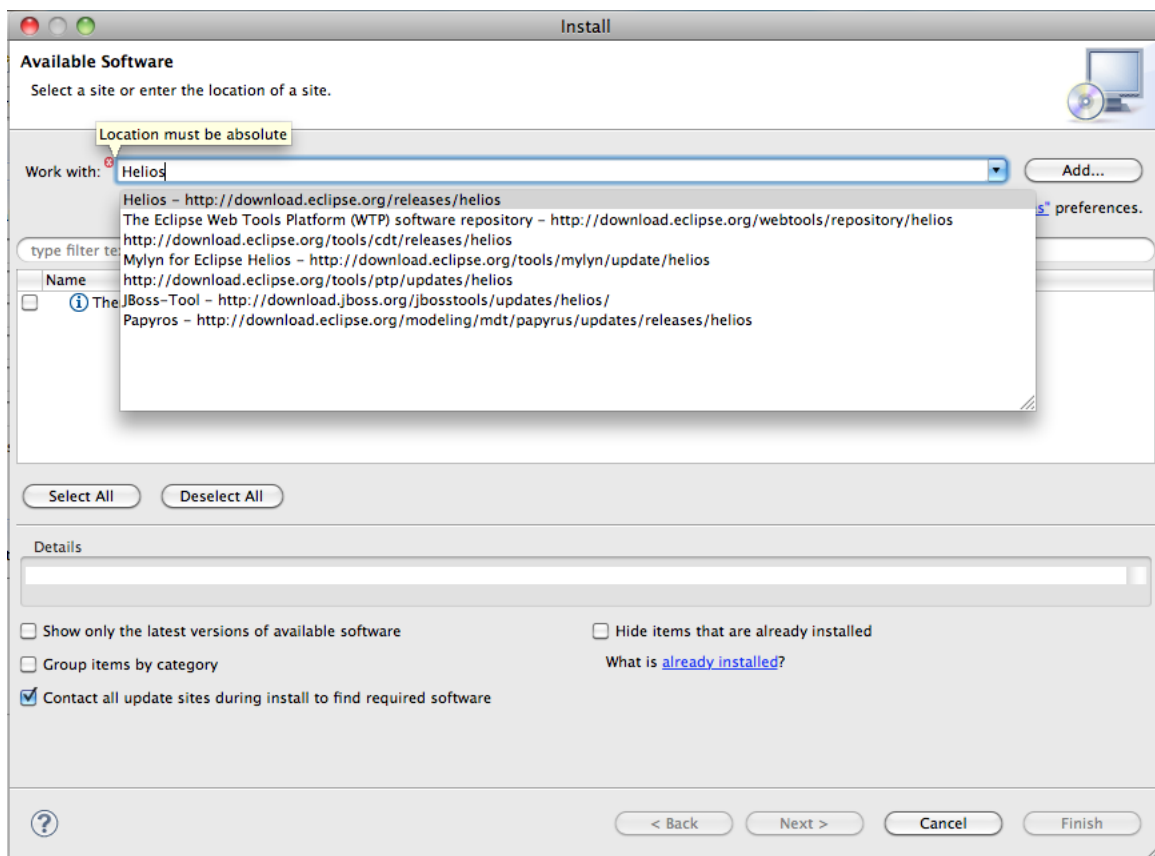
Matthias Hirzel

Erforderliche Eclipse-Software



Für die Entwicklung von Plug-ins müssen in Eclipse zusätzliche Software-Komponenten installiert werden:

- Eclipse Plug-in Development Environment
- Eclipse RCP SDK
- Eclipse RCP Plug-in Developer Resources

1. Wählen Sie hierfür im Eclipse-Menü 'Help' → 'Software' aus.



2. Geben Sie in der Zeile "Work with:" den Namen Ihrer Eclipse-Installation ein (z.B. Helios oder Indigo).

3. Deaktivieren Sie die Option "Show only the latest version of available software", sodass Sie an dem Plug-in-Symbol ( ↔ ) in jeder Zeile erkennen, ob die erforderliche Software schon installiert ist.

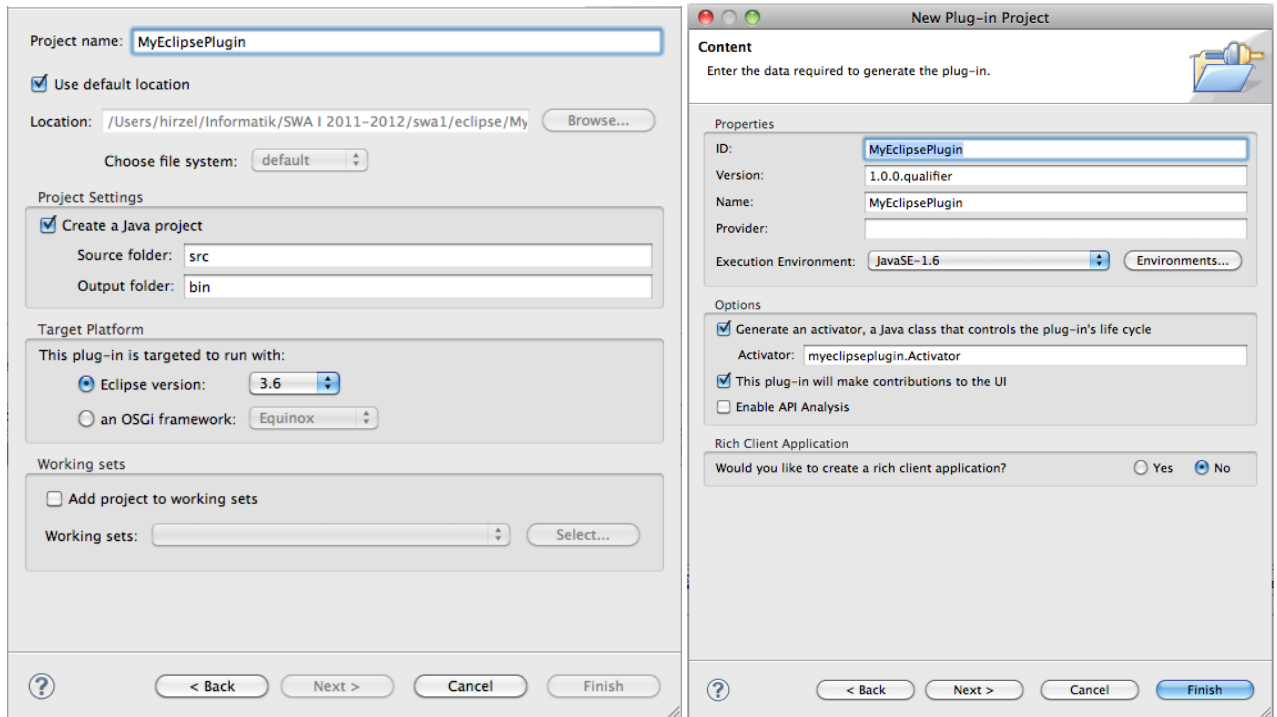
Name	Version
<input type="checkbox"/> Eclipse Plug-in Development Environment	3.6.2.r362_v20110203-7b7mFL2FET3dhHal
<input type="checkbox"/> Eclipse Plug-in Development Environment	3.6.1.r361_v20100727-7b7mFL-FET3dhESC
<input type="checkbox"/> Eclipse Plug-in Development Environment	3.6.0.v20100525-7b7mFKtFEx2XmfZ4_B7NI
<input type="checkbox"/> Eclipse Project SDK	3.6.2.r362_v20100929-7Q7m7HD5cWfGdC
<input type="checkbox"/> Eclipse Project SDK	3.6.1.r361_v20100714-0800-7Q7m6DDaKf
<input type="checkbox"/> Eclipse Project SDK	3.6.0.v20100427-7Q7m-DPY2dP0s7TnPlvx
<input type="checkbox"/> Eclipse Pulsar	1.1.0.M20100811-0600
<input type="checkbox"/> Eclipse Pulsar	1.1.0.I20100517-0318

4. Wählen Sie die zu installierende Software aus und folgen Sie dem Installations-Wizard.

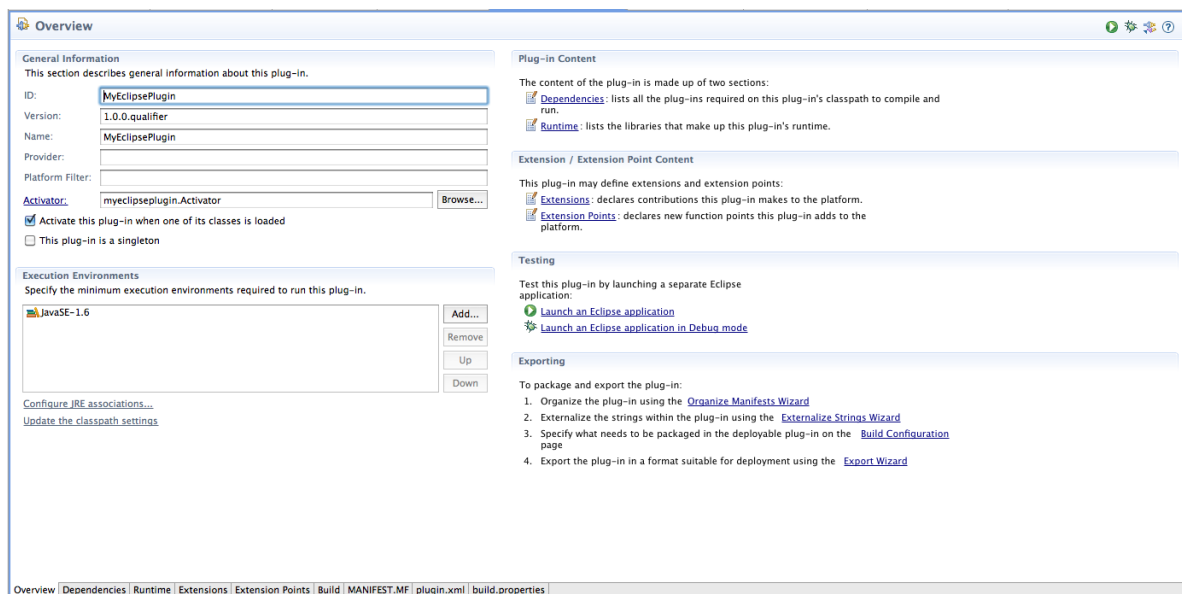
Project Setup

Eclipse bietet für die Programmierung von Plug-ins einen eigenen Wizard an, mit dem das grundlegende Setup festgelegt und ein neues Eclipse-Projekt erstellt wird. Mit Hilfe des Manifest-Editors lassen sich die Einstellungen anpassen und erweitern.

1. Im Eclipse-Menü 'File' auswählen → 'New' → 'Other' → 'Plug-in Development' → 'Plug-in-Project' (bzw. 'General' → 'Plug-in-Project') → 'Next'.
2. Projekt-Namen angeben (hier: MyEclipsePlugin) → 'Next' → 'Finish'.



Angezeigt wird die Datei `MANIFEST.MF` aus dem Verzeichnis `META-INF`. Daneben wurde noch eine Klasse `Activator.java` im Package `<PluginName>` (hier: `myeclipseplugin`) generiert.



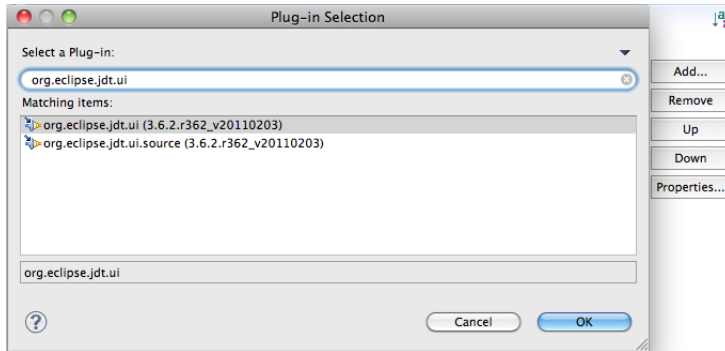
3. Tab 'Dependencies' auswählen. Folgende Dependencies sind schon vorhanden:

- org.eclipse.ui
- org.eclipse.core.runtime

Für die Beispiele in der Übung werden weitere Dependencies benötigt:

- org.eclipse.core.resources
- org.eclipse.jdt.core
- org.eclipse.jdt.ui
- org.eclipse.ui.ide

Wählen Sie hierfür 'Add' → geben Sie nacheinander die Dependencies ein (z.B. org.eclipse.core.resources) → 'OK'.



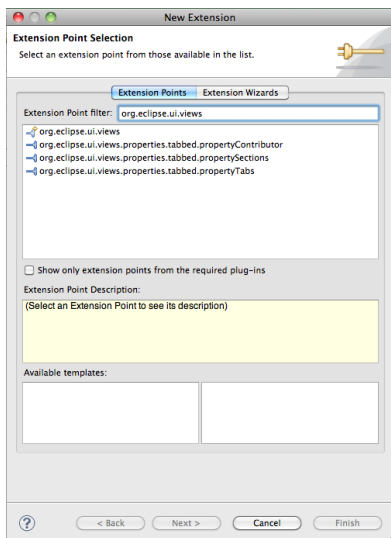
Die Datei MANIFEST.MF (Tab 'MANIFEST.MF') sollte jetzt folgendes Aussehen haben:

```
1 Manifest-Version: 1.0
2 Bundle-ManifestVersion: 2
3 Bundle-Name: MyEclipsePlugin
4 Bundle-SymbolicName: MyEclipsePlugin
5 Bundle-Version: 1.0.0.qualifier
6 Bundle-Activator: myeclipseplugin.Activator
7 Require-Bundle: org.eclipse.ui,
8   org.eclipse.core.runtime,
9   org.eclipse.core.resources;bundle-version="3.6.1",
10  org.eclipse.jdt.core;bundle-version="3.6.2",
11  org.eclipse.jdt.ui;bundle-version="3.6.2",
12  org.eclipse.ui.ide;bundle-version="3.6.2"
13 Bundle-ActivationPolicy: lazy
14 Bundle-RequiredExecutionEnvironment: JavaSE-1.6
15
```

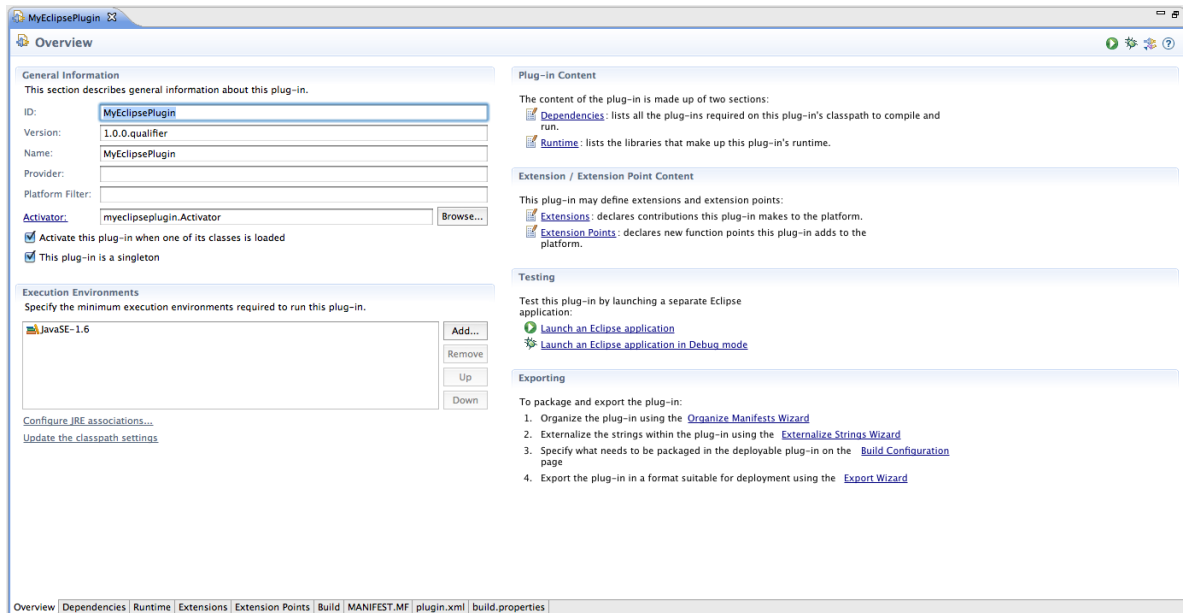
View deklarieren

In unserem Plug-in stellen wir eine zusätzliche Anzeige für Eclipse bereit. Dieses View stellt eine Erweiterung dar und muss somit als Extension zum Rest des Systems angelegt werden.

4. Tab 'Extensions' auswählen → 'Add...' → 'org.eclipse.ui.views' selektieren bzw. 'org.eclipse.ui.views' in die Eingabezeile "Extension Point filter:" eingeben → Finish.

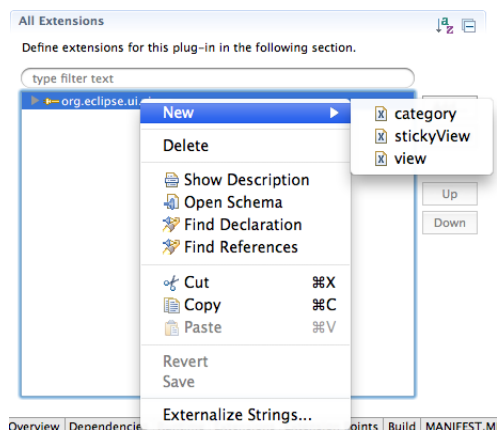


Damit wird eine neue `plugins.xml`-Datei erstellt und mit sinnvollen Standardwerten gefüllt.



5. Das Plug-in soll später in Eclipse als eigenes View angezeigt werden wie es z.B. vom Package Explorer oder der Console bekannt ist. Da in Eclipse alle Plug-ins in Kategorien unterteilt sind, müssen nun Informationen wie der Name des Views und die Kategorie, in der das Plug-in später zu finden ist, festgelegt werden. Erledigen Sie dies *nicht* in der `plugins.xml` von Hand, sondern nutzen Sie den in Eclipse eingebauten Plug-in Editor:

Tab 'Extensions' auswählen → Rechtsklick auf 'org.eclipse.ui.views' → 'new' → 'category'.



6. In der Rubrik 'Extension Element Details' eine ID und einen Namen festlegen (hier: id: myEclipsePlugin.cat, name: MyEclipsePlugin).



7. Rechtsklick auf 'org.eclipse.ui.views' → 'new' → 'view'.

8. In der Rubrik 'Extension Element Details' id, name, class und category festlegen (hier: id: MyEclipsePlugin.view, name: MyEclipsePlugin, class: myeclipseplugin.ExplorerView, category: myEclipsePlugin.cat). Ein Klick auf "class*:" erzeugt automatisch die entsprechende Klasse.
Die Datei `plugins.xml` (Tab 'plugins.xml') sollte nun folgende Elemente beinhalten:

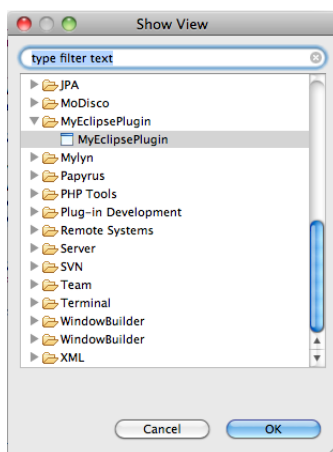
```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <?eclipse version="3.4"?>
3 <plugin>
4   <extension
5     point="org.eclipse.ui.views">
6     <view
7       category="myEclipsePlugin.cat"
8       class="myeclipseplugin.ExplorerView"
9       id="MyEclipsePlugin.view"
10      name="MyEclipsePlugin"
11      restorable="true">
12     </view>
13   <category
14     id="myEclipsePlugin.cat"
15     name="MyEclipsePlugin">
16   </category>
17 </extension>
18
19 </plugin>
20
```


9. Mit dem *WindowBuilder* setzen Sie nun das View zusammen.
Achtung: Die mit dem WindowBuilder zu bearbeitende Klasse müssen Sie mit dem WindowBuilder Editor öffnen (Rechtsklick auf die gewünschte Klasse → 'Open With' → 'WindowBuilder Editor').

Starten des Plug-ins

Das Plug-in kann bereits lokal gestartet und angewandt werden. Im *Eclipse-Update-Site-Tutorial* sehen Sie, wie Sie das neue Plug-in für anderer Nutzer bereitstellen, sodass diese das Plug-in als neue Software in Eclipse installieren können.

10. Starten der Applikation:
Tab 'Overview' → In Rubrik 'Exporting' auf 'Launch an Eclipse Application' klicken → es startet eine weitere Eclipse-Instanz.
Hinweis: 'Launch an Eclipse Application' ist ein Shortcut für die "Run Configurations". Es wird empfohlen, das Plug-in zunächst über 'Launch an Eclipse Application in Debug Mode' zu starten.
11. Im Eclipse-Menü 'Window' → 'Show View' → 'Other' → ⟨ Neu angelegte Kategorie ⟩ (hier: MyEclipsePlugin) → ⟨ Plug-in ⟩ (hier: MyEclipsePlugin) → OK.



12. Evtl. müssen Sie das Plug-in beim ersten Mal über die Schaltfläche "FastView"  am linken unteren Bildschirmrand starten.

Herzlichen Glückwunsch! Sie haben Ihr erstes Eclipse-Plug-in programmiert!

Literatur

- Jeff AcAffer, Jean-Michel Lemieux, and Chris Aniszczyk. *Eclipse Rich Client Platform*. Addison-Wesley, 2010.
- Eclipse. New Project Creation Wizards. http://help.eclipse.org/helios/topic/org.eclipse.pde.doc.user/guide/tools/project_wizards/new_project_wizards.htm. [Stand 24. November 2011].
- Eclipse. Plug-in Overview. http://help.eclipse.org/helios/topic/org.eclipse.pde.doc.user/guide/tools/editors/manifest_editor/overview.htm. [Stand 24. November 2011].