

Pa Hie

WAP

```
public class Titelfolie {  
  
    String ausgearbeitetVon = "";  
    String ausgearbeitetFuer = "";  
    String thema = "";  
  
    public Titelfolie() {  
  
        thema = "Eclipse";  
  
        ausgearbeitetFuer = "Wissenschaftliche Arbeitstechniken " +  
            "und Praesentation";  
  
        ausgearbeitetVon += "Christine Gfrerer ";  
        ausgearbeitetVon += "Alexander Baumgartner ";  
        ausgearbeitetVon += "Markus Kaufmann ";  
    }  
  
    public static void main(String[] args) {  
        Titelfolie tf = new Titelfolie();  
    }  
}
```

Outline

- Titelfolie
 - ausgearbeitetVon
 - ausgearbeitetFue
 - thema : String
 - Titelfolie()
 - main(String[])

Problems @ Javadoc Declaration

0 errors, 0 warnings, 0 infos

Description	Resource	Path	Location

Überblick

- Allgemeines, Geschichtliches
- Architektur
- Oberfläche
- Plugins und deren Einsatz



Was ist Eclipse?

- Open-Source-Framework zur Entwicklung von Software nahezu aller Art.
- Bekannteste Verwendung: Nutzung als Entwicklungsumgebung für Java.
- Auch für die Entwicklung von Rich-Client-Applikationen auf Basis der Eclipse Rich Client Platform (RCP).
- Eclipse wird aufgrund seiner offenen plugin-basierten Struktur mittlerweile für sehr unterschiedliche Entwicklungsaufgaben eingesetzt.

Vorteile

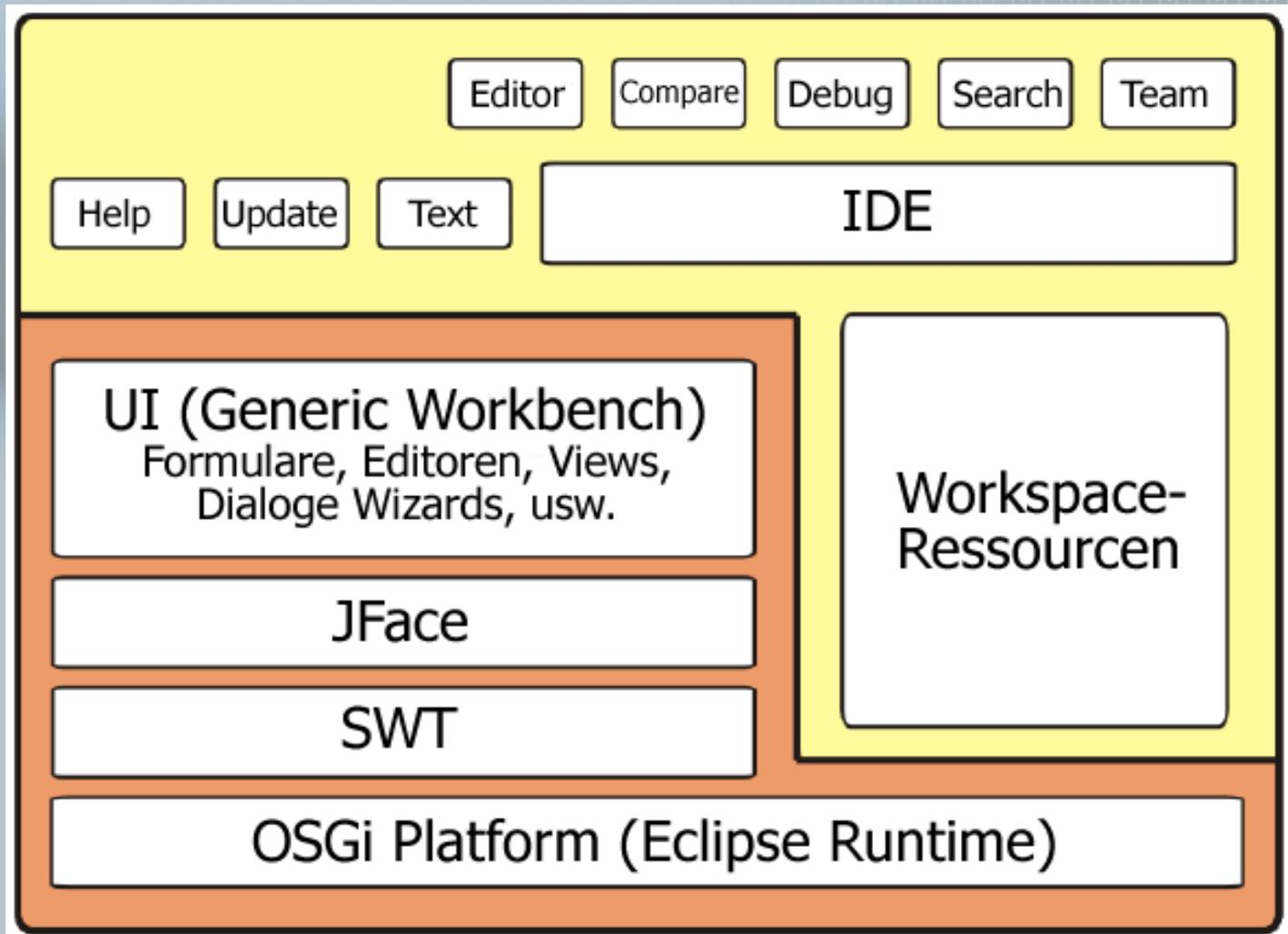
- Einsatz und Erweiterung mittels Plugins
- Basiert auf Java und damit auf vielen Betriebssystemen lauffähig
- Oberflächengestaltung mit SWT



Geschichtliches

- Vorgänger „IBM Visual Age for Java“
- Gründung im November 2001
- Seit Februar 2004 eigenständige Foundation
- Version 3.1 – Java 5
- Version 3.2 – Java 6
- Projekte Eclipse Callisto, Eclipse Europa

Architektur - RCP/IDE



Quelle: <http://www.tilman.de/uni/Eclipse3.pdf>

OSGi (Open Services Gateway Initiative)

- Seit Eclipse 3
- Industriekonsortium das eine Softwareplattform für die Verwaltung von Diensten auf Komponentenbasis standardisiert.
- OSGi-Komponenten sind, wie Eclipse, in Java geschrieben und bieten Dienste an.
- Eclipse-Laufzeitumgebung erfüllt Aufgabe des OSGi-Servers.
(Genauer: Plugin org.eclipse.osgi = Teil der Laufzeitumgebung)
- Plugin Registry als Anlaufstelle für das Auffinden und Registrieren von Diensten.

SWT (Standard Widget Toolkit)

- Eigenes Framework für das UI.
- Abgrenzung zu Swing und AWT.
- Windows, Linux, Mac OS, ... unterstützt.
- GUI-Aufrufe über JNI direkt an das Betriebssystem oder emuliert.
- Wegen JNI sehr performant.
- Plattformspezifische Unterschiede

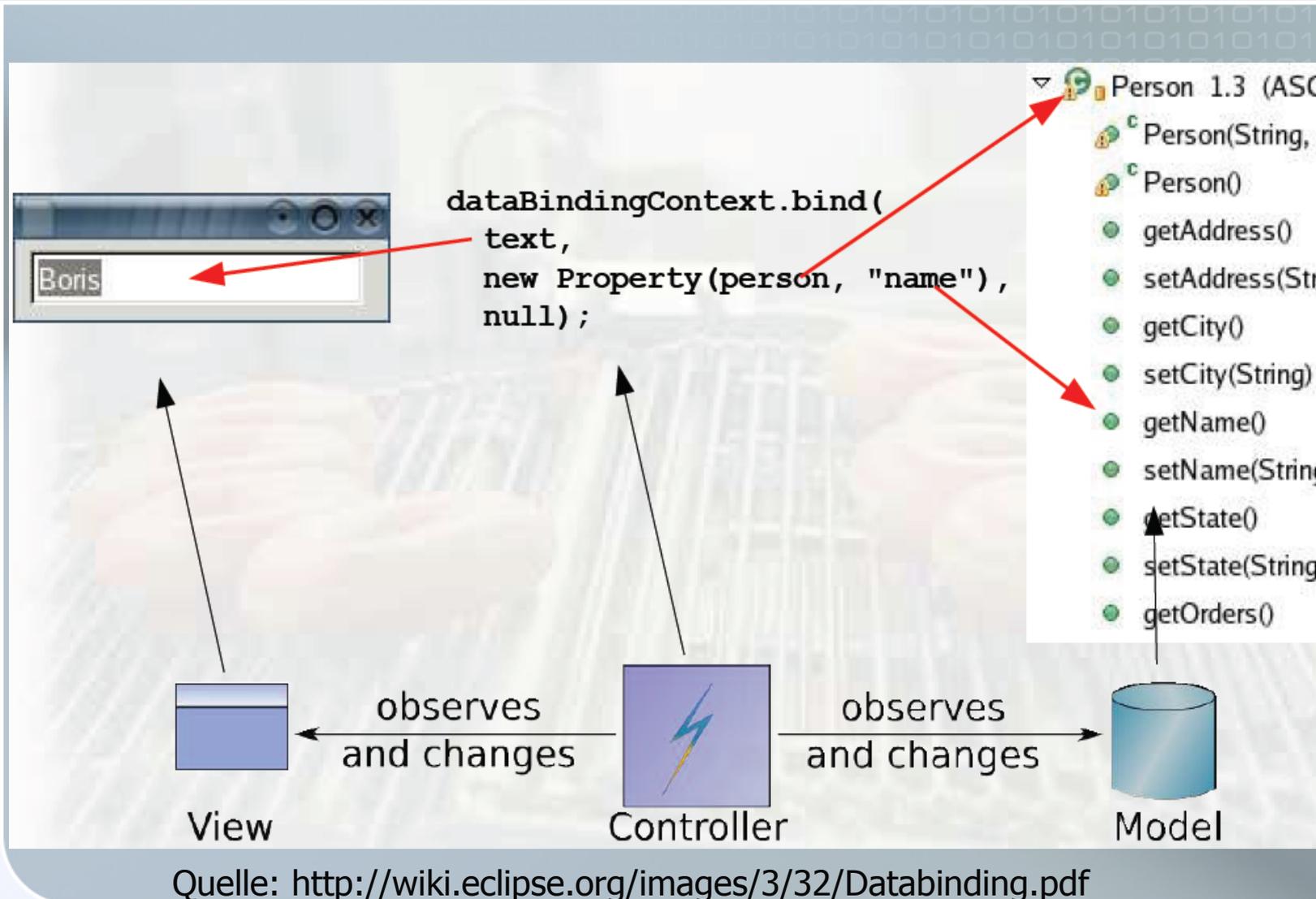
JFace (UI toolkit)

- Abstraktion von komplexen SWT-Komponenten wie Dialoge und Wizards
- Trennung von Modell und Darstellung
- JFace Data Binding (Eclipse 3.2)
 - ◆ Modell soll keine oder möglichst wenige Abhängigkeiten zur UI-Technologie (SWT) haben und sogar auch ohne GUI verwendet werden können.



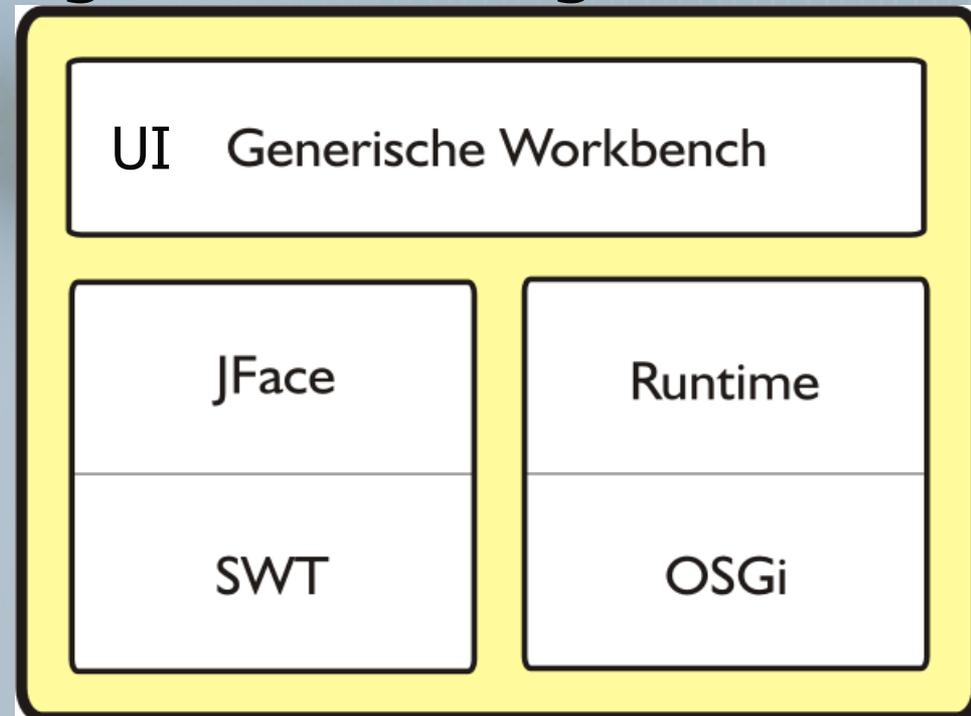
- ◆ Controller spielt beim Eclipse DataBinding eine zentralere Rolle und agiert als Vermittler zw. Modell (JavaBeans) und View.
- ◆ Datenvalidierung und Konvertierung

MVC Beispiel



RCP (Rich Client Platform)

- Framework für Entwicklung clientzentrierter Anwendungen.
- Baut auf die Plugin-Technologie auf.
- Seit Eclipse 3 saubere Trennung von UI und IDE



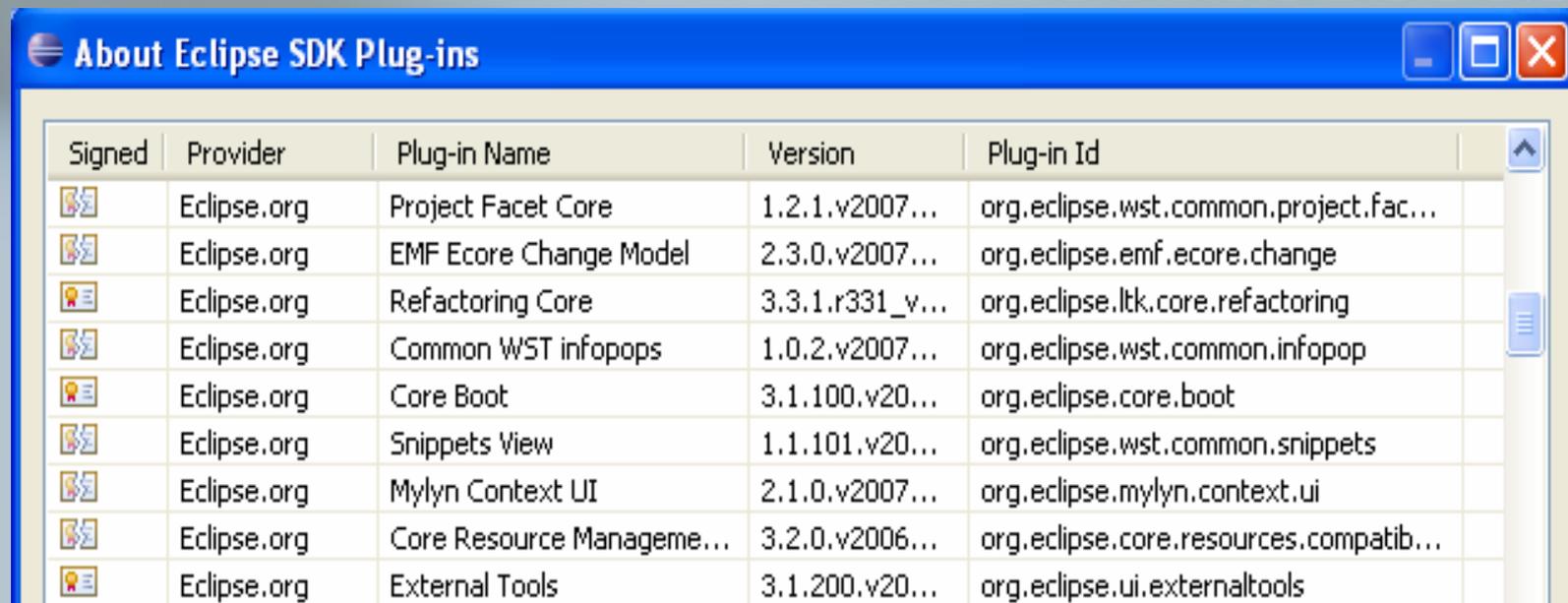
Quelle: <http://www.tilman.de/uni/Eclipse3.pdf>

Oberfläche

- Sichten (kleine Fenster)
Durch Drag and Drop beliebig anordbar
(Tabs, Fast views, ...)
- Editoren
Ändern von Dateien (Quelltexte, XML, ...)
- Perspektiven
Speichert Anordnungen von Menüleisten,
Symbolleisten, Views und Editoren

Plugins

- HELP>About Eclipse>Platform>Plug-in Details
- Window>Show View>Plug-ins

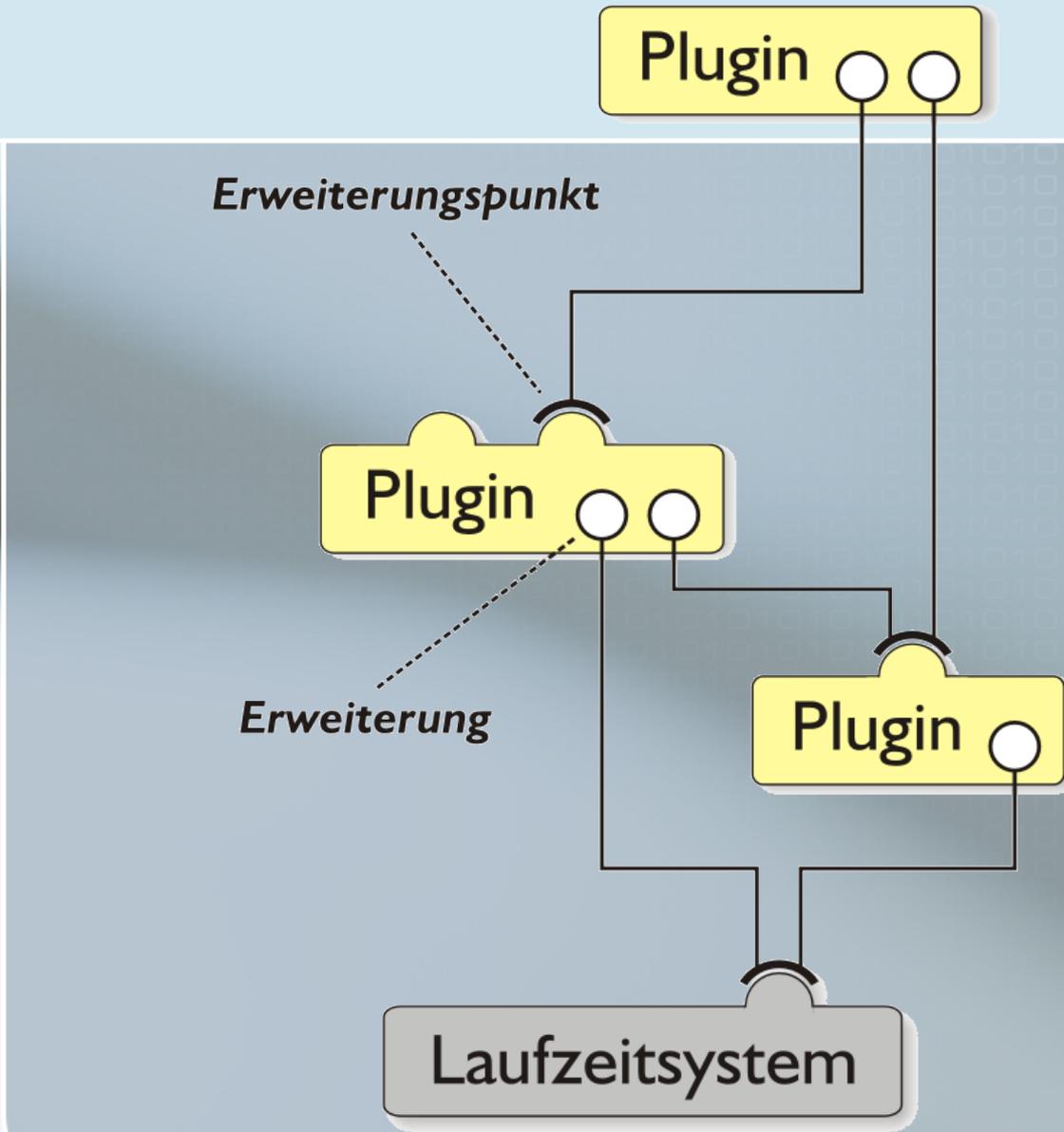


The screenshot shows the 'About Eclipse SDK Plug-ins' dialog box. It contains a table with the following columns: Signed, Provider, Plug-in Name, Version, and Plug-in Id. The table lists several plugins, including Project Facet Core, EMF Ecore Change Model, Refactoring Core, Common WST infopops, Core Boot, Snippets View, Mylyn Context UI, Core Resource Managem..., and External Tools.

Signed	Provider	Plug-in Name	Version	Plug-in Id
	Eclipse.org	Project Facet Core	1.2.1.v2007...	org.eclipse.wst.common.project.fac...
	Eclipse.org	EMF Ecore Change Model	2.3.0.v2007...	org.eclipse.emf.ecore.change
	Eclipse.org	Refactoring Core	3.3.1.r331_v...	org.eclipse.ltk.core.refactoring
	Eclipse.org	Common WST infopops	1.0.2.v2007...	org.eclipse.wst.common.infopop
	Eclipse.org	Core Boot	3.1.100.v20...	org.eclipse.core.boot
	Eclipse.org	Snippets View	1.1.101.v20...	org.eclipse.wst.common.snippets
	Eclipse.org	Mylyn Context UI	2.1.0.v2007...	org.eclipse.mylyn.context.ui
	Eclipse.org	Core Resource Managem...	3.2.0.v2006...	org.eclipse.core.resources.compatib...
	Eclipse.org	External Tools	3.1.200.v20...	org.eclipse.ui.externaltools

Plugin-System

- Konzept einer reinen Plugin-Architektur.
- Jeder Teil von Eclipse ist ein Plugin.
- Plugins werden über Extension Points verschachtelt (MANIFEST.MF).
- Wurzel bildet das von der Laufzeitumgebung gelieferte „Halbplugin“ `org.eclipse.core.runtime`.
- Zwingend erforderlich: „`plugin.xml`“
- Erst in Speicher geladen, wenn benötigt.
> „Unbegrenzte“ Anzahl Plugins möglich.



Quelle: <http://www.tilman.de/uni/Eclipse3.pdf>

Plugin-Typen

- Modellierung mit UML und AOM
- Plugins für spezielle Java-Werkzeuge
- Entwicklungsumgebungen für andere Programmiersprachen
z.B. für C++ oder AspectJ

Plugin-Entwicklung

- Funktionalität existierender Eclipse-Plugins
 - Ressourcenverwaltung des Eclipse-Workspace
 - Projekte, Verzeichnisse und Dateien
 - GUI-Komponenten
 - Hilfesystem
- Plugin-Erstellung basiert auf klar definierten Schnittstellen
 - > Große Teile durch Wizards automatisiert
- Installation ins Verzeichnis eclipse/plugins

Quellen für Plugins

- <http://www.eclipse.org>
- <http://eclipse-plugins.2y.net>
- <http://www.eclipseplugincentral.com>



Wichtige Plugins I (Callisto)

- Web Tools Platform
- C/C++ Development Tools
- Data Tools Platform (DTP)
- Visual Editor



Wichtige Plugins II (Sonstige)

- Grafik und Benutzeroberfläche
 - WindowBuilder
 - Jigloo SWT /SwingGui Builder
- Modellierung
 - MagicDraw, Slime UML, Kleen
 - Poseidon for UML, OMONDO
- Programmiersprache und Compiler
 - AspectJ, JavaCC, TruStudio

Wichtige Plugins III (Sonstige)

- Web
 - MyEclipse, JSEclipse
- Datenbankbereich
 - QuantumDB, DB Copy Eclipse, Clay
- Qualitätsmanagement / Tests und Codeanalyse
 - Eclipse Test & Performance Tools Platform TPTP
 - PMD / Eclipse PMD, Findbugs

ENDE

Vielen Dank für
Eure Aufmerksamkeit



Bibliographie/Internetquellen

- Daum, Berthold: Java Entwicklung mit Eclipse 3.2, dpunkt.verlag, 2006, 4. Auflage
- Cuber, Ulrich: Eclipse 3, Herdt Verlag, 2005
- C't 19 / 2007, Seite 130 - 140
- <http://wiki.eclipse.org/images/3/32/Databinding.pdf>
- <http://www.tilman.de/uni/Eclipse3.pdf>
- [http://de.wikipedia.org/wiki/Eclipse_\(IDE\)](http://de.wikipedia.org/wiki/Eclipse_(IDE))