

Eclipse RCP

Inhalt

- Hintergründe zu Java
- Näheres zu Eclipse
- Weitere Informationen zu Rich Client Platform

Was ist Java und woher kommt es?

- Entstanden durch eine von Sun Microsystems ins Leben gerufene Projektgruppe Namens "Green Projekt" (1991)
- Erster Erfolg war OAK (Object Application Kernel)
- Nach einigen Modifikationen wurde OAK zu Java
- Erfolgreich wurde Java aber erst dadurch, dass es erlaubte kleine Java-Programme aus dem WWW zu laden und diese lokal auszuführen
- Führenden Web-Browser wie Netscape Navigator und MS Internet Explorer integrierten Java
- "Green Projekt" wurde zu JavaSoft
- Oracle übernahm Sun Microsystems und wurde für Java zuständig

Java Versions Übersicht

Version	Codename	Veröffentlichung
JDK 1.1.4	Sparkler	12. September 1997
JDK 1.1.5	Pumpkin	3. Dezember 1997
JDK 1.1.6	Abigail	24. April 1998
JDK 1.1.7	Brutus	28. September 1998
JDK 1.1.8	Chelsea	8. April 1999
J2SE 1.2	Playground	4. Dezember 1998
J2SE 1.2.1	(keiner)	30. März 1999
J2SE 1.2.2	Cricket	8. Juli 1999
J2SE 1.3	Kestrel	8. Mai 2000
J2SE 1.3.1	Ladybird	17. Mai 2001
J2SE 1.4.0	Merlin	13. Februar 2002
J2SE 1.4.1	Hopper	16. September 2002
J2SE 1.4.2	Mantis	26. Juni 2003
J2SE 5.0 (1.5.0)	Tiger	29. September 2004
JSE 6.0	(Mustang) ^[4]	11. Dezember 2006
JSE 7.0	(Dolphin) ^[4]	Mitte 2011
JSE 8.0	-	Ende 2012

Legende:

Ältere Version; nicht mehr unterstützt	Ältere Version; noch unterstützt	Aktuelle Version	Zukünftige Version
--	----------------------------------	-------------------------	--------------------

Quelle Wikipedia

Stärken und Schwächen von Java

Einfach

Objektorientiert

Verteilt

Dynamisch

Robust

Sicher

Plattform-unabhängig

Effizient

Multi-Threaded

Die Entstehung von Eclipse

- Nachfolger des von IBM entwickelten Visual Age
- Quellcode wurde November 2001 von IBM freigegeben
- IBM gründete damit das erste Eclipse-IT-Projekt und wurde von Softwareanbietern wie Borland, Rational Software, Red Hat, SuSE und TogetherSoft unterstützt
- Reorganisation zu einem echten Open-Source-Projekt erfolgte im Februar 2004 unter der Leitung der Eclipse Foundation
- Die von der Open-Source-Gemeinschaft entwickelte Technologie steht kostenfrei unter der Eclipse Public License weltweit zur Verfügung.
- Veröffentlichungstermine von neuen Java und Eclipse Versionen wurden angepasst, um Versionskonflikte zu vermeiden

Eclipse Versions Übersicht

Projektname	Version	Veröffentlichung
	3.0	28. Juni 2004
	3.1	28. Juni 2005
Callisto	3.2	30. Juni 2006 ^[2]
Europa	3.3	30. Juni 2007 ^[3]
Ganymede	3.4	25. Juni 2008 ^[4]
Galileo	3.5	24. Juni 2009 ^[5]
Helios	3.6	23. Juni 2010 ^[6]
Indigo	3.7	22. Juni 2011 ^[7]

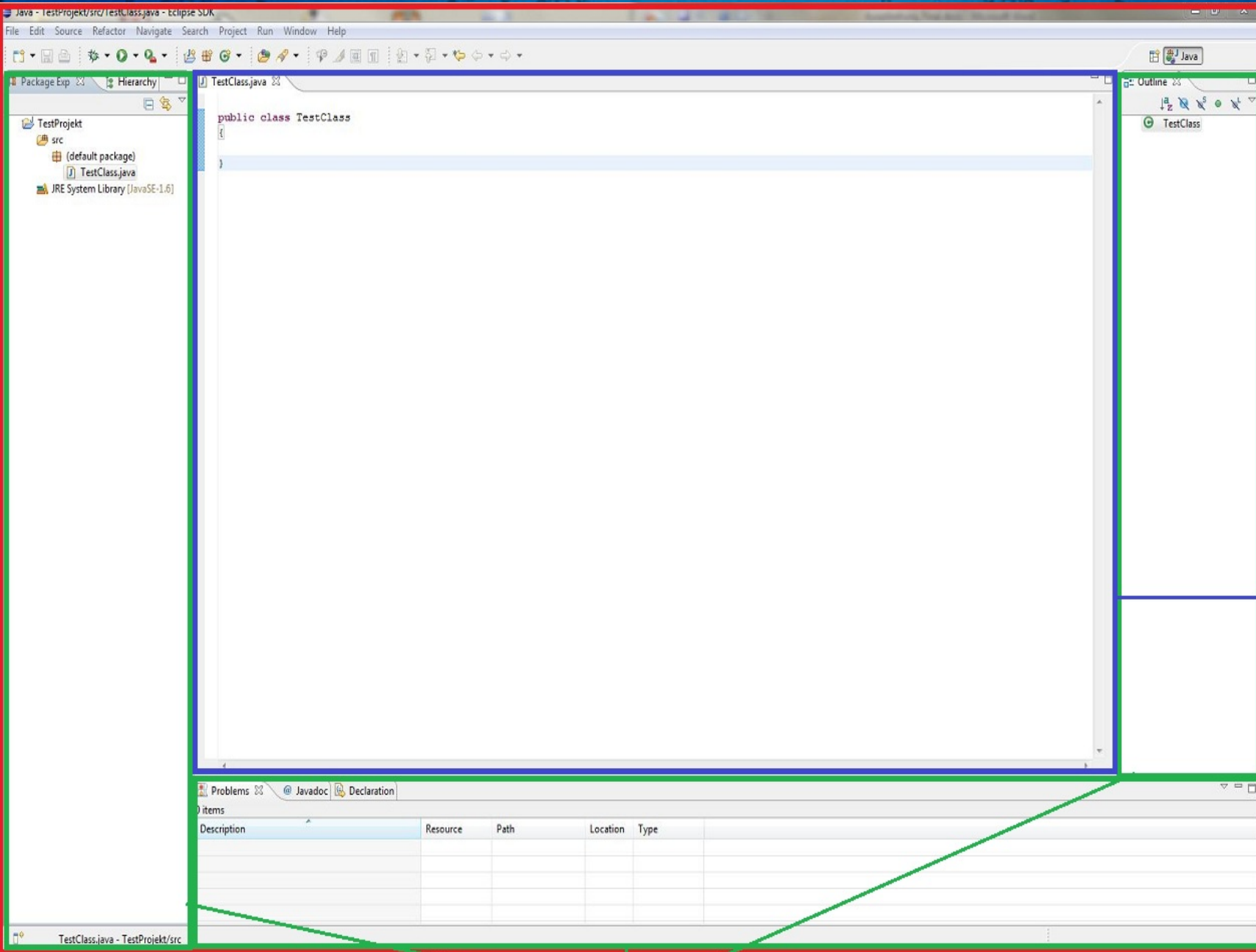
Legende:	Ältere Version; nicht mehr unterstützt	Ältere Version; noch unterstützt	Aktuelle Version	Zukünftige Version
-----------------	--	----------------------------------	-------------------------	--------------------

Quelle Wikipedia

Vorteile von Eclipse

- Team Support
 - Unterstützt Cooperative Work durch Team Repositories
- Plug-Ins
 - Es gibt Plug-Ins nahezu jede Aufgabe
- Eclipse UI
 - Erweiterbare UI die man nach belieben ändern und erweiter kann mittel Plug-Ins

Eclipse UI



Editor

Workbench Window

Views

Was ist ein Rich Client

Fat Client

- Vollständiges System
- Anwendungen Lokal
 - Lange Aktualisierungen
- Wird Lokal auf dem Betriebssystem des Users ausgeführt

Thin Client

- Hat alle Wichtigen Daten auf einem Server
- Ohne Verbindung zum Server ist der Thin Client "nutzlos"
- Meistens über ein Web-Interface zugänglich

Der Rich Client

- Der Begriff "Rich Client" beschreibt eine Anwendung bei der die Datenverarbeitung Lokal ausgeführt wird
- Meistens besitzt die Anwendung auch eine GUI
- Ist Plattform-unabhängig
- Unterstützt Online und Offline Arbeiten
- Bietet Client Aktualisierungsmöglichkeiten

Eclipse RCP

- Eine Rich Client Platform ist ein Framework zur Entwicklung von Rich Client Anwendungen
- In diesem Fall werden mittels Eclipse-RCP Plug-Ins entwickelt
- Mit Eclipse-RCP entwickelte Plug-Ins können über RCP selbst ausgeführt werden, doch das ist nicht zwangsläufig notwendig
- So entwickelte Anwendungen sind selbstständige Rich Clients, doch bleiben sie immer Eclipse-Plug-Ins

Eclipse RCP

- RCP nutzt die schon in Eclipse vorhandenen GUI-Elemente wie Windows, Perspektiven, Views und Editoren
 - Dadurch nutzt RCP auch SWT und JFace
- Eclipse RCP unterstützt Extension-Points und Extensions
 - Diese Extension-Points kann man sich als Steckdosen vorstellen, wo sich Plug-Ins einstecken können um z.B. Funktionen zu erweitern
- Mit Hilfe des "Hello RCP-Template" kann man sich recht einfach eine kleine RCP Applikation zusammen stellen

Vielen Dank!