#### In dieser Starthilfe

- Team-Entwicklung mit CVS
  - Anbinden von Eclipse an den CVS Server
  - Einchecken eines vorhandenen Projektes
  - Erzeugen einer Revision
  - Verwalten der Revisionen einer Ressource
  - Ersetzen der aktuellen Revision durch eine andere
  - Projektstand als Version definieren
  - Laden einer Version in den Workspace

*Team-Entwicklung unter Eclipse* 

### 1.1. Um was geht's?

Sie wollen sich mit **Eclipse**, einer gratis Java-IDE aus dem IBM *Eclipse Projekt* (Eclipse.org), vertraut machen. Ich wollte das auch und finde Eclipse sehr brauchbar. Einiges finde ich besser als im JBuilder (Refactoring, Plugins, Imports organisieren u.v.m.). JBuilder hat andere Vorteile (Help, UML, grosser Funktionsumfang [Enterprise Edition]). Aber es gibt noch viele andere IDE's, wie beispielsweise intelliJ (*www.intellij.com*).

### 1.2. Team-Entwicklung mit CVS

CVS steht für Concurrent Version System. Es ist das bekannteste Open-Source Versionsverwaltungssystem. Sie können beispielsweise Quelldateien verwalten und dabei unterschiedliche Versionen definieren und kontrollieren, ohne dass Sie jeweils den gesamten Code abspeichern müssen. Der Zugriff auf die einzelnen Versionen und Archive sind Passwort-geschützt. Beim Einchecken neuer Dateien kann der Benutzer einen Kurzkommentar angeben und damit die Suche nach bestimmten Versionen erleichtern.

Die Dateien werden in einem Repository (=Verzeichnis mit Unterverzeichnissen) abgespeichert. Jede Veränderung erhält eine Revisionsnummer, wobei mit 1.1 gestartet wird:

Ele Edit Source Refactor Navigate Search Project Run Window Help
」 武 ・ □ 風 合   弥 ・ ★ ・ 矢 ・   メ   留 丞   ひ ク ・ ⇔ ×   目 書 単 部 QVS Repositories 金 ゆ ゆ 砂   ズ ▼ X JExpose J HelloWo J Status.j J Summen J Expose 』Cle >
🗄 🔯 CVS Repositories 🎽 🗇 🚸 💢 🔻 🗙 刘 Expose J HelloWo J Status.j J Summen J Expose 🔬 Cle >
<pre>package kursbeispiele; package kursbeispiele; package kursbeispiele; package kursbeispiele; package kursbeispiele; package kursbeispiele; package kursbeispiele; package kursbeispiele; /** * Title: XML-RPC Standalone Server * Description: Einfacher XML-RPC Server * Description: Einfacher XML-RPC Server * Copyright (c) 2001 * Company: http://www.joller-voss.ch * @author Josef M. Joller * @version 1.0 */ import java.util.Vector; * CVS Console * CVS Console * CVS Console</pre>

(Unsere Dateien befinden sich im Hauptzweig (HEAD) und besitzen die Revisionsnummer 1.1.)

Unter einer Version versteht man den Stand eines Projekts bestehend aus Dateien mit unterschiedlichen Revisionsnummern.

Das CVS hat einen Nachteil: gleichzeitige Änderungen an einer Datei sind möglich. Beim Hineinstellen (*merge*) einer Datei in ein Repository muss der Entwickler entscheiden, wie die verschiedenen Stände kombiniert werden sollen. In der Praxis kann man das Problem so lösen, dass jeder Entwickler seinen eigenen Subpfad verwendet und nur zu bestimmten Zeitpunkten "synchronisiert" wird und eine neue Version im Hauptpfad angelegt wird.

Die Unterstützung für CVS ist direkt in Eclipse integriert, man kann also sofort loslegen. Natürlich muss irgendwo ein CVS-Server zur Verfügung stehen. Zum Testen oder als lokales Repository kann man mit CVSNT (<u>http://www.cvsnt.org/</u> für Windows) oder CVS (<u>http://www.cvs.org/</u>) schnell einen CVS-Server einrichten. Neben der Möglichkeit im Team an einem Projekt zu arbeiten, sind die wichtigsten Vorteile:

- Änderungshistorie zu jeder Datei
- Änderungskommentare (Begründung jeder Änderung)
- Übersichtliche Änderungsanzeige (was hat sich getan?)
- Verwalten von Tags (Versionsständen oder Releases)
- Verwalten von unterschiedlichen Entwicklungszweigen (Branches)

Unabhängig, ob im Team oder nicht gearbeitet wird, sollte man allein aus diesen Gründen immer mit CVS arbeiten.

Weitere Informationen zum CVS (Installation, Tests usw.) finden Sie im *Skript zum CVS* (Starthilfe). Hier beschränken wir uns auf die Nutzung eines CVS Systems aus Eclipse.

## 1.2.1. Anbinden von Eclipse an den CVS Server

Nachdem Sie einen CVS Server aufgesetzt haben (separates Skript, s.o.) möchten Sie sicher Eclipse mit diesem Server verbinden.

1. öffnen Sie die Perspektive "CVS Repository Exploring"



2. Die entsprechende Perspektive wir geöffnet. Selektieren Sie die View "cvs Repositories"

Wenn Sie jetzt auf die rechte Maus-Taste klicken, erscheint ein Auswahlmenü

) 🖈	EVS Repository Exploring - Eclip:	se Platform					
File	Edit Source Refactor Navigat	e Search Project To	mcat Run Wi	ndow Help			
Ď	1 II II A K K K K K K K K K K K K K K K K	1 🖬 🖌 🖎	- 🛛 🖓	<b>a</b>			
É	🕅 CVS Repositories	<b>≜</b> ≑ ⇒ ∲ <del>▼</del> ×	🚺 Test.jav	a X			
⊕J			packa	ge test;			<u>^</u>
愁			/**				
Cus			* @au *	ithor Ada	inistrati	or	
=	New 🕨	Describer Leasting	* To	change th	nis gener bes>J	ated comme ava>Templa	ent edit the templa ates.
		Repository Location	a antina ta dave a	-lin	isabl	e the crea	ation of type comme
	Refresh View	Anonymous/Repusicory L	ocation to devie	clipse.org			Seneration.
			public	c class	lest (		
			pu	blic sta System	out.prin	l main(Str tln("Hallo	ring[] args) { p. Welt!");
			}				
			,				
			CVS Res	ource History			
			Revision	Tags	Date	Author	Comment
							A
							×
	<u> </u>					<u>اخار</u>	
						1	<u> </u>

3. Im neuen Popup Fenster können Sie Ihren CVS Server und Benutzerinformationen sowie Protokoll angeben:

🚝 Add CVS Repo	sitory	×
Add a new CVS Add a new CVS Rep	Repository pository to the CVS Repositories view	cvs
Location Host:	localhost	<b>•</b>
Authentication – User: jjole Password:	1	
Connection Connection type:	pserver	
C Use Port: ✓ Validate Conne	tion on Finish	
		Finish Cancel

Diese Angaben müssen Sie unter Umständen erfragen. Die obigen Angaben entsprechen meinen Einstellungen und dem lokalen CVS.

- 4. Wählen Sie die Option "Validate Connection on Finish".
- 5. falls alles korrekt eingegeben wird, wird die Verbindung aufgebaut und Sie können in Zukunft Ihre Programme einchecken.

### 1.2.2. Einchecken eines vorhandenen Projektes

Nun wollen wir aus Eclipse mit CVS Systemen arbeiten. Alle Aktivitäten sind direkt aus Eclipse möglich. Hier das praktische Vorgehen:

- 1. öffnen Sie eine Perspektive, welche die View "Package Explorer" enthält (oder öffnen Sie in der aktiven Perspektive zusätzlich diese View)
- 2. Klicken Sie auf das einzucheckende Projekt (selektieren Sie das gewünschte Projekt).
- Klicken Sie auf die rechte Maustaste. Ein Kontextmenü öffnet sich und u.a. folgende Optionen werden wählbar: "New", "Go Into", ... "Team", ... "Properties"
- 4. Wählen Sie "Team".

Ein weiteres Kontextmenü öffnet sich und u.a. folgende Optionen werden wählbar: falls Sie das Projekt noch nicht ins CVS gestellt haben: "Apply Patch" (kann auch fehlen), "Share Project"; falls Sie das Projekt bereits eingecheckt hatten: "Synchronize with ...", "Commit", "Tag as Version",... "Share Project". Sie können den Status auch unter den Projekteigenschaften überprüfen: nach dem Einchecken sehen Sie unter den Projekt Properties ein Feld "cvs" mit den CVS Server-Informationen.

5. Wählen Sie "Share Project".



6. Sie können nun das Repository auswählen, in das Sie Ihr Projekt stellen möchten:



Wie Sie sehen, könnten Sie auch gleich ein neues Repository einrichten.

7. Nach Anklicken von "Next" erscheint die folgende Auswahl:



Wenn Sie möchten, könnten Sie Ihrem CVS Modul einen spezifischen Namen geben. Wir lassen die obige Einstellung stehen und gehen zu "Next" und "Finish" 8. Eclipse checkt Ihr Projekt ins CVS ein und Sie können in der CVS View

("Synchronize - Incoming/Outgoing Mode") das Ergebnisprotokoll ansehen:

9. In der "Synchronize" View : klicken Sie auf die rechte Maustaste (öffnen Sie das Kontextmenü). Die Optionen"Expand all", "Add To Version Control", ... erscheinen. Wählen Sie "Add To Version Control":



Eclipse fügt Ihr Projekt in die Versionskontrolle ein.

10. Nach einer Weile, je nach Projektgrösse: wählen Sie nun aus dem Kontextmenü (linke Maustaste) "Commit".

🚰 Synchronize - Outgoing Mode		÷	<b>*</b>	⇔ ₽	<u>۱</u>	×
Structure Compare						
HeinProjekt [localhos	Expand All Remove From View Show in Navigator Show in Resource History					
	Add to Version Control					
	Commit					
	Override and Commit Add to .cvsignore	<u>م</u> د				
	Mark as Merged Select New Resources					
Tasks Console Synchronize						
nge(s), no new resources.	W	/ritable	Insert		14:42	

11. Sie können noch einen erklärenden Text eingeben (dieser sollte in irgend einem Zusammenhang mit abgespeicherten Version sein).



- 12. Klicken Sie auf "OK". Jetzt werden die Dateien ins CVS übertragen.
- In der View "CVS Repositories" können Sie die eingecheckten Projekte finden:



wobei die Startversion mit 1.1 bezeichnet wird (die Nummer ist willkürlich – einfach ein Index).

## 1.2.3. Erzeugen einer Revision

Sobald Sie Ihre Dateien / Programme verändern und neu synchronisieren, ergibt sich ein neuer Revisionstand auf dem CVS Server.

Schritt für Schritt:

- 1. Ändern Sie eines Ihrer Programme aus einem eingecheckten Projekt (fügen Sie beispielsweise einen Kommentar ein).
- 2. Speichern Sie die geänderte Datei (Ctrl+S).
- 3. Jetzt müssen wir die geänderte Datei mit der im CVS synchronisieren. Kontextmenü: "Team->Synchronize with Repository"



Ihre Datei wird verarbeitet und es erscheint die View "Synchronize"



 Jetzt müssen Sie die Änderung(en) bestätigen. Kontextmenü: "Team->commit"

EclipseTeamTutorial.doc

🗄 📳 Package Explorer 🛛 🔻 🗙	J Aufgabe_1.java 🗙
	/*
🕞 🗈 Apache XML-RPC	* Created on 24.08.2003
JavaGrundkurs	*
src	* To change the template fo
	* Window> Preferences>
	•
Den Open	<pre>ckage kapitel_1;</pre>
Open With	•
Open Type Hierarchy	<b>port</b> java.io.BufferedReade
the at a cut	<b>port</b> java.io.IOException;
	port java.io.InputStreamRe
ten	+
🕀 🕪 🚺 JRE Sys 🗰 Delete	Rauther iicller
Source	• eaution jjorren
Refactor	▶ /
Import	blic class Aufgabe 1 {
Export	/**
	* Addition zweier float
Reterences	* @param pl
Decial actoris	* @param p2
🔅 Re <u>f</u> resh	+ 0+ 0
Team	Synchronize with Repository
Comp <u>a</u> re With	Commit
Replace With	Update
Restore from Local Histor	2 Create Patch
Properties	
	Tag as Version
	Branch
	Struc Show in Resource History
	End Chapter ASCIT/Pinary Property
	Add to Version Control
	Add to .cvsignore
	5CII -kkv) [ou Show Editors
	Unedit
	Edit

5. Sie werden aufgefordert einen Kommentar einzugeben (mit dem Sie die aktuelle Revision kennzeichnen können):

🚰 Commit	×
Edit the commit comment:	
Aufgabe 1 mit Kommentaren	
	_
Choose a previously entered comment:	
	•
_	
	OK Cancel

Klicken Sie auf "OK".

### 1.2.4. Verwalten der Revisionen einer Ressource

Wenn Sie jetzt sehen wollen, welchen Revisionslevel die gerade mutierte Datei im CVS hat, können Sie wiefolgt vorgehen:

1. In der View "Package Explorer" selektieren Sie die neu eingecheckte Datei, nicht das Package!

Kontextmenü auf dieser Datei: "Team->Show in Resource History"



2. Die View "CVS Resource History" erscheint und zeigt die oben ausgewählte Ressource:



Die aktuelle Revision ist mit einem "\*" gekennzeichnet.

### 1.2.5. Ersetzen der aktuellen Revision mit einer anderen:

Falls Sie eine Ressource 'versaut' haben und auf eine alte Version zurück greifen möchten, können Sie folgendermassen vorgehen.

- 1. Selektieren Sie die Ressource (anklicken)
- 2. Lassen Sie die View "CVS Resource History" anzeigen (jene mit den verschiedenen Revisions).

Kontextmenü: "Team->Show in Resource History".

3. Jetzt können Sie in der View "CVS Resource History" Ihre gewünschte Revision selektieren.

Kontextmenü:"CVS Resource History->Get Sticky Revision"

vision	Tags	Date	Author	Comment	
2	Tugs	08 10 03 11:04	iioller	Aufgabe 1 mit Kommentaren	
		04.10.03 20:14	ijoller	Grundkurs Java	
Get Contents					
Get Sticky Re	vision				
Tag with Exis	ting				
Compare					
Open					
A Defined House					
A REAL PROPERTY AND A REAL AND A					

4. Bevor die ältere Revision die neuere *im CVS* als aktuelle ersetzt, erhalten Sie eine Warnung:



Klicken Sie auf "OK".

5. Jetzt ist die ältere Revision mit einem Stern versehen:

Tasks CV5 Editors						
Revision	Tags	Date	Author	Comment		
1.2		08.10.03 11:04	jjoller	Aufgabe 1 mit Kommentaren		
*1.1		04.10.03 20:14	jjoller	Grundkurs Java		

Beachten Sie: bisher haben wir nur CVS-seitig Änderungen gemacht. Im Workspace selber haben Sie immer noch die selbe Revision.

Sie können beide Versionen mit einem Doppelklick im Editor anzeigen lassen, aber mit der Datei-Extension 1.1 bzw. 1.2

cclipse Platform
jate Se <u>a</u> rch <u>P</u> roject <u>R</u> un <u>Wi</u> ndow <u>H</u> elp
・●・   ◎ ● ◎ ◎・   ● タ   * ♀ + ♀ -   圖 ☆ ?
▼ 🗙 🔄 *Aufgabe_1.java 🔄 Aufgabe_1.java 1.1 🚺 Aufgabe_1.java 1.2 🗙
/ * localhost:/CVSServer/Prod/JavaGrundkurs/src/kapitel_1/Aufgabe_1.java 1.2
* Created on 24.08.2003
*

Zudem könnten Sie jetzt Probleme bei einer erneuten Synchronisation erhalten: EclipseTeamTutorial.doc



Mit der Revision 1.2 tritt dieses Problem nicht auf.

## 1.2.6. Projektstand als Version definieren

Die Definition einer Version geschieht im CVS. Hier das Vorgehen Schritt für Schritt:

- 1. Selektieren Sie die View "CVS Repositories"
- 2. Selektieren Sie das Projekt



3. Kennzeichnen Sie diese Revision als Version. Kontextmenü:



4. Geben Sie ein Versions-Kennzeichen ein, beispielsweise "Kurs2003"

🚰 Tag Resources		×
Please enter a version tag:		
Kurs2003		
	OK Cancel Cancel	5
Existing Versions:		
	Refresh from Repository	gs

Klicken Sie auf "OK".

5. Jetzt finden wir unter CVS Versionen unsere eben definierte Version:



## 1.2.7. Laden einer Version in den Workspace

Falls Sie nun Ihre aktuelle, im Workspace befindliche Version durch eine im CVS vorhandene ersetzen möchten, brauchen Sie nur einen Schritt auszuführen.

- 1. Wählen Sie die View "CVS Repositories"
- 2. Selektieren Sie die gewünschte Version.
- 3. Jetzt können Sie die aktuell im Workspace befindliche Version überschreiben. Kontextmenü:"Check Out As Projekt"



4. Falls das Projekt bereits im Workspace vorhanden ist erscheint eine Warnung:



5. Die im CVS ausgewählte Version überschreibt Ihre Workspace Version.

EINLE	EITUNG	1
1.1	Um was geht's?	1
1.2.	TEAM-ENTWICKLUNG MIT CVS	1
1.2	2.1. Anbinden von Eclipse an den CVS Server	2
1.2	2.2. Einchecken eines vorhandenen Projektes	4
1.2	2.3. Erzeugen einer Revision	8
1.2	2.4. Verwalten der Revisionen einer Ressource1	0
1.2	2.5. Ersetzen der aktuellen Revision mit einer anderen:1	1
1.2	2.6. Projektstand als Version definieren1	2
1.2	2.7. Laden einer Version in den Workspace1	3