Formale Methoden: UML, OCL und vdm++

Prof. Dr. Hans-Jürgen Buhl

2004

Fachgruppe Mathematik / FB C Institut für Angewandte Informatik Bergische Universität Wuppertal

Praktische Informatik
 PIBUW-SS0401

April 2004 1. Auflage, 2004

Praktische Informatik 02

Link-Sammlung

• UML-Dokumentation und Beispiele

http://lci.cs.ubbcluj.ro/ocle/links.htm http://www.agilemodeling.com/style/classDiagram.htm http://www.agilemodeling.com/artifacts/classDiagram.htm

• UML-Tools

http://lci.cs.ubbcluj.ro/ocle/links.htm http://uml.sourceforge.net/index.php http://www.borland.com/together/ http://www.mid.de/de/ http://www-306.ibm.com/software/rational/offerings/design.html

• OCL-Standard-Drafts

http://neptune.irit.fr/Biblio/03-01-07.pdf http://neptune.irit.fr/Biblio/01-09-77.pdf

• OCL-Tools

http://lci.cs.ubbcluj.ro/ocle/links.htm http://www.klasse.nl/ocl/ http://www.eclipse.org/ http://www.eclipse.org/cdt/

• OCL-Beispiele

http://www.klasse.nl/ocl/ocl-reasons.html http://www.klasse.nl/ocl/ocl-introduction.html

Kleine Einleitung in die Benutzung von Octopus

Stand: Eclipse 3.0.0 mit Octopus 1.0.1

Starten Sie Eclipse durch den Aufruf von > Eclipse Sie erhalten:



Wählen Sie

File New Project

aus:

New Project	
Select a wizard	- Alian A
Bizardia Disardia Disardia Plag-lia Project Disa C Det Det Det Det Det Det Det Det	Bescriptions n elcard that helps creating an Detopos project.
E geer All Vizorde.	Bert + Earcel

Nach Anklicken der Option "Show all Wizards" und Auswahl von "Octopus project" erscheint nach Anwahl von Next:

topus I	roject. sreates a new actapus project.	
aject new	FirstBrispiel	
roject con	ants	
the defau	۹.	
reentrega	Assen/Subit. Ava Kapacov F Linst Beliapinet	Bosto

Vergeben Sie bitte einen Projektname (hier zum Beispiel FirstBeispiel)

und klicken Sie erneut auf Next:

Octopus Project Sub Judicate the folders serve	folders he uni model and oci expressions can be found.	=
<u>Folder</u> for coll expressions:	expressions	
Folder for sml models	sode1	
6		-
	1 Beck Sec. 1	Careel

Akzeptieren sie schließlich durch Klick auf Finish.

Im Navigator-Fenster erscheint das neue Projekt mit den (noch leeren) Unterordnern **expressions** sowie **model**:



Wählen Sie model an und erzeugen sie dar
in mit Hilfe der rechten Maustaste über

New Folder

einen weiteren (Unter-)
Unterodner, der einem UML/OCL-package entspricht:



P <u>r</u> oject	
Folder	
File	
<u>O</u> ther	Ctrl+N

New Folder		
Folder Breate a new folder resource.		
gnter or select the parent folders		
FirstBeispiel/wodel		
C Appkul.5.9 SiretBeigiel Sec B Sec Percinavervalturg		
Folder gant ours		
	Einim	Cancel

Im erschienenen Fenster "New Folder" ist nun ein Folder-Name (sprich: Package-Name) zu spezifizieren — im diesem Beispiel: core — und danach Finish anzuklicken.

Klicken sie nun im Navigator-Fenster auf das OCT-Icon (Octopus), so wird der Octopus-UML-Model-View aktualisiert (statt dummy erscheint core):



Wählen Sie im Model-View-Fenster **core** an und klicken Sie danach auf das UML-Icon des Model-View-Fensters: Es erscheint ein schon vorausgefüllter Dateiinhalt für das Package **core** im UML-Editor oben rechts:



Für die weitere Arbeit ist es wichtig, dass Sie die Window-Preferences richtig eingestellt haben. Rufen Sie zur Überprüfung über

Window Preferences

das zugehörige Fenster auf

Workbench	Workbench					
<pre>> Novexelation → Ant → Build Order → C/C++ → Help → Install/Update → Java → Java → Run/Debug → Team</pre>	 Perform <u>build</u> automatically on resource modification Save all <u>modified</u> resources automatically prior to manual build <u>P</u>rompt for Workspace on startup <u>R</u>efresh Workspace on startup <u>C</u>onfirm exit when closing last window Show Problems <u>v</u>iew when build has errors or warnings 					
	Workspace save interval (in minutes): 5 Open mode ◆ Double click ◇ Single click					
	<pre> telect on hower Open when using errow keys Note: This preference may not take effect on all views</pre>	aly				

und vergewissern Sie sich, dass sowohl "Perform build automatically on resource modification" als auch "Show Problems view when build has errors or warnings" eingeschaltet sind.

Nun sollten sie textuell eine UML-Klasse eingeben. Nach Eintippen von <cla wird mittels der Tastenkombination

<Ctrl Leertaste>

die automatische Schlüsselwortergänzungsfunktionalität angefordert:

Resource - core_additions.v	ml - Eclipse	Platform	9		_ O X
Eile Edit Herigete Gegrof Broject	Ben Binden Bel	lø .			
10.00 14.14 2.	100.0	*			E Broose
Cincipator II	Dates of	CoPerson-sol	4 core_addition.ml	Carl and antition and Carl	5
Bighal.3.7 Graduitation Gr	<pre>/* This file // This file // file</pre>	file contains generated by (TE elemente created Intapas en Toe Hay 18	in fattpus, and added to packag	
1 1	itable	Insert	10 + 8	1	

Ergänzen Sie die Klasse zur Klasse ${\tt Person}$ nach folgendem Muster

Resource - core_additions.u	umi - Eclipse Platform	_ D X
file Edit Berigets Segroh Broject	Ben Binden Brip	
10.00 0.00 0.00	1 0 0 • 0 •	/ El Crease
The sign for 22 and 24	Control Contro Control Control Control Control Control Control Control Control Co	
	Canada Viene IS Problems	19 1 A & & + PB
	itable Smart 12 + 7	

und klicken Sie auf das Disketten-Icon, um mit der manuellen Abspeicherung aller Projektänderungen auch (implizit) ein build mit einem Syntaxcheck auszulösen: Es erscheint der neue OCT-Model-View



oder eine Fehlermeldung im Problems-Fenster.

Resource - core_additions.un	nl - Eclipse Plat	form			_ C X
file Edit Beigets Gegrob Brouest B	en Bindev Bello				
12·00 s· * 2·	1 0 0 · 0 ·				E Cresserce
Charigeter 11	Coloran col Cor	Person-sel ¥eore_addition	and there,	additions, and 22	0
Provide the second trains	Pile group */ *package* 00/6 *elans* Person *attributes* * nome: 25 *endtilans* *endpackage* Radet Time (* (* (* (*)))) Personalized Of Hence) Description	teing	y 38 16 19 48	16 Folder	H D T PD

Ein leeres Problems-Fenster (wie hier) zeigt an, dass keine Syntaxfehler fest-gestellt wurden.

Klicken Sie nun im Model-View-Fenster auf die Klasse **Person** und danach auf das OCL-Icon in der Model-View/Problems- Kopfzeile, um OCL-Constraints für die Klasse **Person** einzugeben: Im OCL-Editor erscheint eine vorausgefüllte OCL-Datei



die Sie nach Aktualisierung des Model-View-Fensters



jetzt nur noch entsprechend zu modifizieren brauchen



um dann nach Anklicken des Save-Icons von der syntaktischen Korrektheit der OCL-Klassen-Invarianten

name <> ''

überzeugt zu sein



beziehungsweise einen Fehler sowohl im Model-View



als auch im Problems-Fenster feststellen zu können (rotes Problem-Kreuz):



Und nun viel Vergnügen bei der Planung von formal spezifizierten Software-Modellen...