

ALLGEMEIN

In dieser Übung soll aus einem Task Flow Diagramm der entsprechende Java Quelltext mittels Xpand¹ generiert werden.

VORBEREITUNG

Zur Vorbereitung müssen folgende beiden Features in Eclipse installiert sein:

- MWE SDK
- Xpand SDK

Damit ihr generierter Quelltext vollständig kompiliert, benötigen Sie SDMLib. Die Bibliothek kann vom SE Blog heruntergeladen werden: <https://www.se.eecs.uni-kassel.de/fileadmin/se/courses/SE2/SDMLib.zip>.

Importieren Sie das Projekt in ihren Runtime Workspace. Konvertieren Sie ihr Beispielprojekt in ein Plugin Projekt (Rechtsklick auf Projekt -> Configure -> Convert to Plug-In Projects...) und fügen sie SDMLib als Abhängigkeit hinzu.

Laden Sie das Fujaba Modell CodeGenClasses.ctr vom SE Blog herunter: <http://www.se.eecs.uni-kassel.de/fileadmin/se/courses/SE2/CodeGenClasses/CodeGenClasses.ctr>. Fügen Sie das Fujaba Modell anschließend als Abhängigkeit zur ihrem Task Flow Diagramm Modell hinzu.

Abgabe: Bis spätestens zum **Sonntag den 16.02.2013** über das Hausaufgabenabgabesystem zur Veranstaltung. Die Abgabe muss eine .zip Datei mit folgendem Inhalt sein:

- Alle 4 Eclipse Plugin Projekte (Modell, Edit, Editor, Diagramm)
- Im Modell Plugin:
 - Workflow Datei (z.B. generatorWorkflow.mwe)
 - Xpand Template (z.B. TFDDefault.xpt)
- Beispielprojekt aus ihrem Runtime Workspace. Dieses enthält mindestens:
 - Modell und Diagramm (siehe Aufgabe 4 – Quelltext erzeugen)
 - Generierte .java Datei für Beispieldiagramm (siehe Aufgabe 4 – Quelltext erzeugen)

¹ <http://www.eclipse.org/modeling/m2t/?project=xpand>

AUFGABE 1 – GENERATECODE ACTION ERZEUGEN (3P)

Legen Sie eine neue Klasse mit dem Namen „GenerateCodeAction“ in ihrem Fujaba Modell an. Sorgen Sie dafür, dass die Action auf einem Diagrammobjekt aufrufbar ist und (wie in der Vorlesung und der Übung gezeigt) den Xpand Codegenerator programmatisch aufruft.

AUFGABE 2 – WORKFLOW DEFINITION ERZEUGEN (2P)

Erzeugen Sie in Ihrem Modell Plugin im src Verzeichnis eine Workflow Definitionsdatei. Orientieren Sie sich am in der Vorlesung und der Übung gezeigten Vorgehen.

AUFGABE 3 – TEMPLATE ERZEUGEN (8P)

Erzeugen Sie in Ihrem Modell Plugin im src Verzeichnis ein Xpand Template. Orientieren Sie sich am in der Vorlesung und der Übung gezeigten Vorgehen. Das Template soll für das Task Flow Diagramm auf dem die GenerateCodeAction aufgerufen wird, Java konformen Quelltext generieren, der folgenden Ansprüchen genügt:

- Es wird eine .java Datei erzeugt, die den Namen des LaneContainers trägt. Diese muss von der Klasse `org.sdmlib.model.taskflows.TaskFlow` erben und das Interface `org.sdmlib.utils.PropertyChangeInterface` implementieren.
- Innerhalb der generierten .java Datei
 - Ein Java enum, welches für jeden Tasknamen im Diagramm eine Konstante enthält
 - Für jede im Diagramm definierte globale Variable, ein entsprechendes Feld
 - Eine Methode mit der Signatur

```
public Object[] getTaskNames()
```

welche ein Array mit den Tasknamen zurückgibt

- Eine Methode mit der Signatur

```
public void run()
```

die den Code für den Taskflow enthält. Innerhalb dieser Methode soll es für jeden im Diagramm dargestellten Task eine Switch-Case Anweisung geben, die folgendes enthält:

- Die Lane-lokalen Variablen
- Den Quelltext des Tasks

AUFGABE 4 – QUELLTEXT ERZEUGEN (1P)

Erzeugen Sie für das folgende Beispieldiagramm den zugehörigen Quelltext (es handelt sich hierbei um Ihr Diagramm aus Hausaufgabe 6):

