

# Übungen zur Vorlesung Softwaretechnologie

-Wintersemester 2011/2012-

Dr. Günter Kniesel

## Übungsblatt 1

Zu bearbeiten bis: 21.10.2012

Bitte fangen Sie **frühzeitig** mit der Bearbeitung an, damit wir Ihnen bei Bedarf helfen können. Checken Sie die Lösungen zu den Aufgaben bitte in Ihr SVN-Repository ein, „Erklärungen“ bitte als Textdatei. Fragen zu Übungsaufgaben respektive zur Vorlesung können Sie auf der Mailingliste [swt-tutoren@lists.iai.uni-bonn.de](mailto:swt-tutoren@lists.iai.uni-bonn.de), bzw. [swt-vorlesung@lists.iai.uni-bonn.de](mailto:swt-vorlesung@lists.iai.uni-bonn.de) stellen.

### **Aufgabe 1.** *Erste Schritte mit Subversion* (5 Punkte)

Bei Bearbeitung der Aufgabe 1 des vorausgehenden Aufgabenblattes haben Sie sich für eine Übungsgruppe angemeldet. Sobald diese Anmeldung abgeschlossen war, haben Sie per E-Mail Ihre persönlichen Zugangsdaten für das SVN-System erhalten.

a) Jede Übungsgruppe hat ein eigenes SVN-Repository, in dem sie gemeinsam die Aufgaben der Übung bearbeitet und ihre Ergebnisse ablegen kann. Der SVN-Pfad dazu lautet:

[https://svn.iai.uni-bonn.de/repos/IAI\\_Software/se/swt2012/gruppe/XX/](https://svn.iai.uni-bonn.de/repos/IAI_Software/se/swt2012/gruppe/XX/)

Fügen Sie dieses Repository Ihrer Eclipse-Installation hinzu (vgl. Aufgabe 6 auf Blatt 0) und ersetzen Sie dabei bitte das XX durch die zweistellige Nummer Ihrer Übungsgruppe (z.B. 05).

In Ihrem Repository befindet sich bereits ein Projekt, *HelloTutor*. Checken Sie das Verzeichnis aus, ergänzen Sie das Projekt so, dass Ihr Name mit ausgegeben wird und committen Sie anschließend Ihre Änderungen.

b) Legen Sie nun ein neues Java-Projekt an und darin ein Programm, das alle durch 3 oder 5 teilbare Zahlen zwischen 11 und 37 (jeweils inklusive) ausgibt. Sichern Sie dieses in Ihrem SVN-Repository. Dazu können Sie den Befehl *Team->Share Project...* aus dem Kontextmenü des Projektes verwenden.

### **Aufgabe 2.** *Dateioperationen mit SVN* (6 Punkte)

a) Erzeugen sie in Eclipse ein neues Projekt „Dateien“ und legen Sie darin einen Ordner „Nummer 1“ mit einer Datei „Datei 1.txt“ an. Importieren Sie nun das Projekt ins SVN (*Team->Share project...*).

b) Fügen Sie in dem Ordner eine neue Datei „Datei 2.txt“ hinzu. Vergleichen Sie die Symbole der beiden Textdateien im *Package Explorer*. Was fällt auf? Welche Bedeutung haben die jeweiligen Symbole? Welche Schritte müssen Sie unternehmen, damit sich „Datei 2.txt“ im gleichen Zustand wie „Datei 1.txt“ befindet?

c) Benennen Sie nun „Datei 2.txt“ in „Datei 2a.txt“ um (Kontextmenü: *Refactor->Rename*). Was bemerken Sie? Was bedeutet dies für Dateioperationen bei SVN?

d) Öffnen Sie nun den Projektordner mit einem Dateiverwaltungsprogramm (z.B. Finder oder Windows Explorer)<sup>1</sup>. Benennen Sie nun hier, außerhalb von Eclipse, „Datei 1.txt“ in „XX.txt“ um. Wechseln Sie zu Eclipse und wählen Sie aus dem Kontextmenü des Projekts den Befehl „Refresh“ aus. Was bemerken Sie im Unterschied zu Aufgabenteil c) ? Welche Änderungen zeigt Ihnen die Synchronisations-Ansicht des Projekts (*Team->Synchronize*)?

e) Committed Sie das Projekt. Benennen Sie nun außerhalb Eclipses den Ordner „Nummer 1“ in „Nummer 2“ um und aktualisieren Sie die Dateiansicht in Eclipse. Was bemerken Sie nun?

f) Wechseln Sie in die Synchronisations-Ansicht des Projekts. Was bemerken Sie nun? Was passiert bei einem Update? Was können Sie nun tun, um das Projekt zu „reparieren“?

### **Aufgabe 3. Etiketten und Zweige (15 Punkte)**

Neben dem Hauptstamm eines mit SVN verwalteten Projektes (genannt „trunk“) gibt es noch die Konzepte der Etiketten („tags“) und Zweige („branches“).

a) Erklären Sie, welchen Zweck *tags* und *branches* erfüllen und wie diese in SVN angelegt werden. Gibt es auf technischer Ebene im SVN Repository einen Unterschied zwischen *trunk*, *tags* und *branches*?

b) Legen Sie von Ihrem Projekt in Aufgabe 1b einen Tag an und benennen Sie diesen sinnvoll.

c) Erzeugen Sie einen Branch, checken Sie diesen aus und verändern Sie das verzweigte Projekt so, dass zusätzlich auch durch 7 teilbare Zahlen ausgegeben werden. Checken Sie das modifizierte Branch-Projekt wieder in das SVN Repository ein.

d) Ergänzen Sie das Programm im *trunk* so, dass durch 6 teilbare Zahlen übersprungen werden und committed Sie das Projekt. Fügen Sie nun den *branch* aus Aufgabenteil c) wieder mit dem *trunk* zusammen („mergen“) und committed sie das Projekt erneut.

e) Warum steigt beim Kopieren von Zweigen innerhalb des SVN Repositories der Speicherbedarf desselben – solange keine Modifikationen committed werden – nur minimal an?

### **Aufgabe 4. SVN Properties (4 Punkte)**

In Subversion lassen sich Dateien und Verzeichnisse um Meta-Informationen, die so genannten Eigenschaften („Properties“) ergänzen. Geben Sie für die folgenden Eigenschaften wieder jeweils deren Bedeutung und einen beispielhaften Wert an:

a) `svn:ignore` (für Verzeichnisse)

b) `svn:mime-type` (für Dateien)

c) `svn:keywords` (für Dateien)

d) `svn:externals` (für Verzeichnisse)

*Hinweis: Verwenden Sie zur Bearbeitung der Aufgaben 3 und 4 auch weiterführende Literatur bzw. geschickte Web-Suchanfragen und deren Ergebnisse.*

---

<sup>1</sup> Den Pfad des Projekts finden Sie in den Eigenschaften des Projektes bei „Resource“ unter „Location“.

### **Aufgabe 5.** *Erste Schritte mit GIT* (12 Punkte)

Nach Kennenlernen von SVN, wollen wir uns jetzt kurz mit Git beschäftigen. Installieren Sie dazu einen Git-Client auf ihrem System (wir empfehlen SmartGit), und bearbeiten Sie dann die folgenden Aufgaben. Informationen zur Installation von SmartGit erhalten Sie hier:

<https://sewiki.iai.uni-bonn.de/teaching/lectures/se/2012/git>

- a) Holen Sie sich zuerst unser Repository auf ihren lokalen Rechner, indem sie ein Clone durchführen. Das Repository finden Sie unter:  
[git://github.com/swt201112/uebungsblatt\\_1.git](git://github.com/swt201112/uebungsblatt_1.git)
- b) Erstellen Sie nun für sich einen eigenen Branch, und führen Sie folgende Änderungen durch:
  1. Korrigieren Sie die enthaltenen Rechtschreibfehler in der Datei *briand-kellogg.txt*. Der Text ist in drei Abschnitte unterteilt: Jeder von Ihnen soll genau einen Abschnitt bearbeiten. Ordnen Sie sich selbst die Abschnitte zu, indem Sie ihre Vornamen alphabetisch sortieren. Der 1. Vorname bearbeitet den 1. Abschnitt, der 2. Vorname den zweiten Abschnitt, usw....
  2. Benennen Sie die Datei *briand-kellogg.txt* in *kellogg-briand.txt* um.
  3. Fügen Sie in der Datei *gedicht.txt* eine neue Strophe hinzu
  4. Comitten Sie die Änderungen in ihr lokales Repository.
- c) Übergeben Sie ihren lokalen Git-Ordner (gepackt) an ihre Teamkollegen (E-Mail, USB-Stick, ...)
- d) Mergen Sie die beiden Repositories ihrer Kollegen auf ihren eigenen Branch (dazu müssen Sie zuerst die beiden Repositories unter Remotes/Manage Remotes als Remote Repository hinzufügen, und einen Pull durchführen). Eventuell auftretende Konflikte lösen Sie auf, indem sie auf die entsprechende Datei rechtsklicken und „Show Changes“ auswählen. Jetzt in dem sich öffnenden Fenster manuell die Änderungen durchführen und speichern. Nun ein „Stage“ durchführen und anschließend wieder in Ihr lokales Repository committen.
- e) Vereinbaren Sie, wer in ihrem Team als "Teamleiter" für die finale Version verantwortlich ist. Schicken Sie ihren Git-Ordner nun an ihren Teamleiter. Dieser hat die Aufgabe alle ihre drei Branches zu einem zusammenzuführen. Sein Git-Ordner bildet ihre Gruppenabgabe.