

Informationen zur Installation und Nutzung von Subversion (SVN)

Für detailliertere Problem ist das Rechenzentrum Ansprechpartner (rechenzentrum@ostfalia.de)

Auf den Seiten des Rechenzentrums finden Sie weitere Informationen zur Nutzung, siehe <https://code.ostfalia.de/>.

Der Link zum Online-SVN-Buch: <http://svnbook.red-bean.com/> (schon älter) könnte auch helfen.¹

Ein SVN-Repository ist von Herrn Helmke eingerichtet und mit ein paar Dateien² schon gefüllt. Der Zugang erfolgt je nach Ihrem Namen³ über

- <https://code.ostfalia.de/svn/i-wpf-cpp/Studenten/XXX>

XXX steht hier für Ihren Nachnamen (ohne Umlaute (ä = ae etc.)), Bei Doppelnamen mit Blank wurde nur der erste Name gewählt („Müller Meier“ wurde zu „Mueller“). Groß- und Kleinschreibung bitte beachten, d.h. das erste Zeichen ist immer groß, der Rest in kleinen Buchstaben.

Sowie Lesezugriff auf

- <https://code.ostfalia.de/svn/i-wpf-cpp/AlleGruppen>

Sie sollten zum Zugriff auf Ihrem privaten Rechner die Tortoise-SVN Software (public-domain) installieren oder alternativ das Eclipse-Plugin oder ... Einen Client inklusive sehr ausführlicher Dokumentation finden Sie auf der Tortoise-Homepage <http://tortoisesvn.net/downloads.de.html>. Die Basisfunktionen werden im Folgenden auch hier erklärt.

¹ Mir ist bekannt, dass GIT die bessere Option wäre. Hierfür bekomme ich aktuell aber keine Unterstützung vom Rechenzentrum z.B. für das Rechtemanagement. Um für mich ein akzeptables Kosten/Nutzen-Verhältnis zu haben, habe ich mich entschieden, nochmals SVN zu verwenden.

² Aktuell (21. September 2024) habe ich nur bei AlleGruppen schon ein paar Dateien abgelegt.

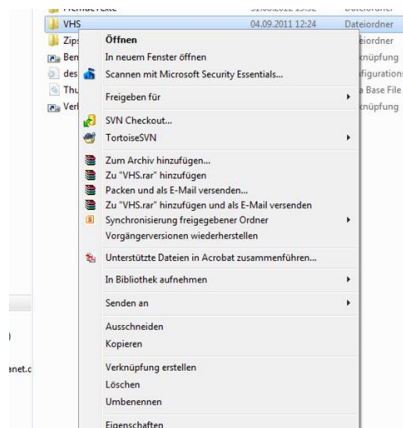
³ Derzeit (21. September 2024) sind für 6 StudentInnen die Zugriffsrechte beim Rechenzentrum freigeschaltet und 0 weitere beantragt. Von den restlichen Vorlesungsteilnehmern liegen mir keine Informationen über die Vorlesungsteilnahme vor. Sie loggen sich mit Ihrem ganz normalen Benutzernamen und Passwort der Ostfalia ein. Sie haben hier auf alle Unterverzeichnisse lesenden und schreibenden Zugriff. Bitte prüfen.

Auf das Verzeichnis „AlleGruppen“ haben Sie nur lesenden Zugriff. Hier möchte ich Ihnen als Betreuer Informationen zu den Aufgaben zur Verfügung stellen. Außerdem befindet sich hier einer Kopie der pdf-Dateien von meiner Homepage.

Nutzung von Tortoise-SVN unter Microsoft Windows⁴

Im Folgenden ist die Version 1.9.1 beschrieben (Stand der Beschreibung 15. September 2015, angepasst 21.09.2023); mittlerweile (21. September 2024) haben wir die Version 1.14.7. Nach erfolgreicher Installation von Tortoise-SVN unter Microsoft Windows stehen im Datei-Explorer nun neue Funktionen zum Auschecken, Einchecken etc. zur Verfügung.

Bei der Erstbenutzung, geht man im Datei-Explorer in das Verzeichnis, wohin ausgecheckt werden soll, d.h. dort wo Sie Ihre Codedateien ablegen wollen. Hier führt man einen Rechts-Klick durch und wählt mit der rechten Maustaste "SVN-Checkout..."⁵.



Bei der URL des Projektarchivs gibt man nun an:

„<https://code.ostfalia.de/svn/i-wpf-cpp/Studenten/XXX>“ bzw. <https://code.ostfalia.de/svn/i-wpf-cpp/AlleGruppen>; XXX siehe oben

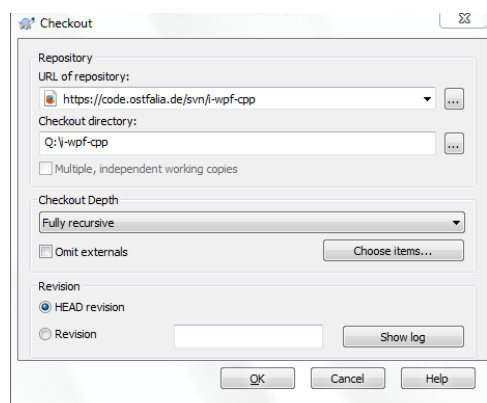


Abbildung 1 Initialer Bildschirm nach Wahl von SVN-Checkout

Es wird dann evtl. nach dem Zertifikat gefragt und hier gibt man bei der Fehlermeldung "permanent akzeptieren" oder "temporär akzeptieren" ein.

⁴ Die Beschreibung bezieht sich hier auf Windows. Sie wird unter Linux aber ganz ähnlich aussehen. Die Bilder beschreiben das Vorgehen, wenn ein deutscher SVN-Client installiert ist. Sofern die englische Sprache eingestellt ist, sind die Menüs entsprechend zu übersetzen.

⁵ Oder „SVN-Checkout“, wenn die englische Sprachversion aktiv ist. Die entsprechende englische Wortwahl gilt dann entsprechend auch für das Restliche hier vorliegende Dokument.

Login mit dem username und zugehörigem password durchführen, welches Sie sonst auch für Ihren Ostfalia-Zugang verwenden.

Nun erhalten Sie alle Dateien, die schon für Sie im Repository (teilweise von mir) eingecheckt sind.⁶

Den Vorgang wiederholen Sie nochmals für „AlleGruppen“ (also SVN-Auschecken wählen und bei „URL of Repository“ nun [„https://code.ostfalia.de/svn/i-wpf-cpp/AlleGruppen“](https://code.ostfalia.de/svn/i-wpf-cpp/AlleGruppen) eingeben, siehe Fußnote 6 auf Seite 2.

Wenn Sie selbst eine Datei geändert haben, wird Ihnen diese Datei als geändert (commitable) im Datei-Explorer angezeigt.⁷

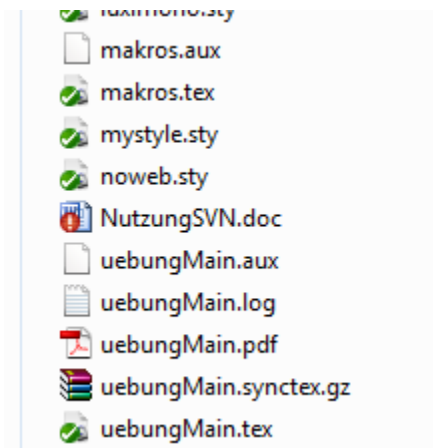
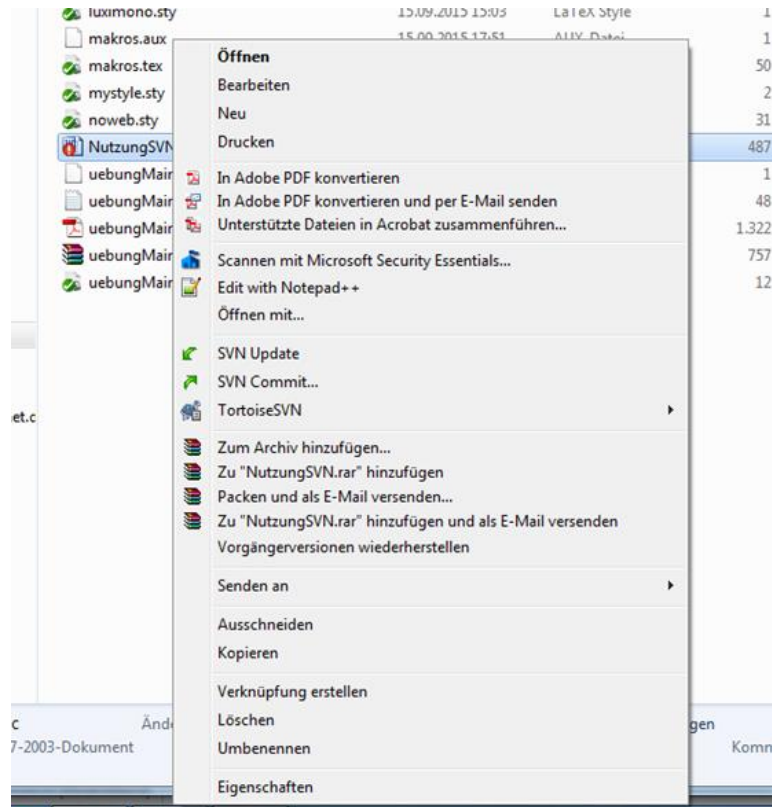


Abbildung 2 NutzungSVN.doc wurde geändert (rotes Ausrufezeichen); noweb.sty und makros.tex etc. sind im Repository bekannt, aber nicht geändert (grünes Aufrufezeichen). Einige Dateien sind noch nicht im Repository eingecheckt (UebungMain.aux z.B.).

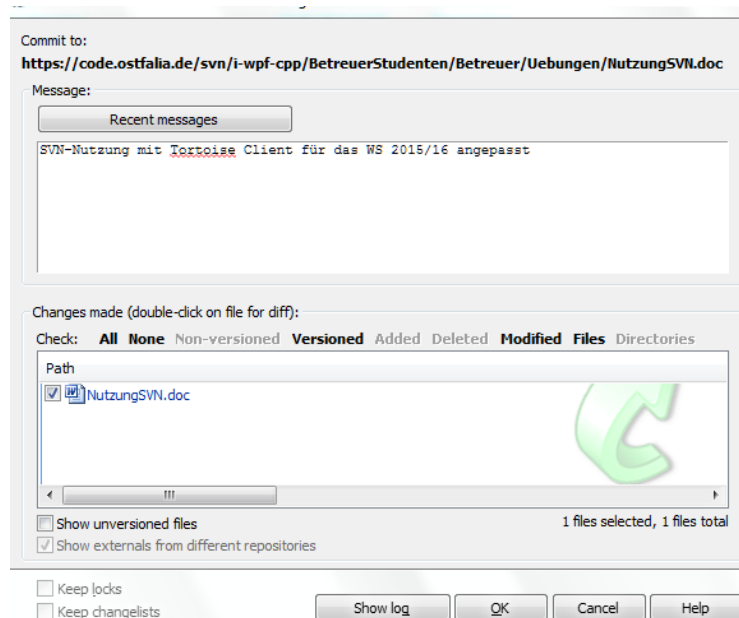
Eine geänderte Datei klicken Sie mit der rechten Maustaste im Datei-Explorer an und wählen „SVN-Commit“ aus (Sie können auch mehrere Dateien gleichzeitig auswählen; Wenn Sie ein oder mehrere Verzeichnisse auswählen, werden Ihnen in einem Menü alle geänderten Dateien angezeigt und Sie können selbst auswählen, was Sie davon übertragen wollen):

⁶ Ich muss zunächst den Zugang für Sie beantragen und dann auch noch ein paar Beispieldateien einchecken. Dazu benötige ich eine E-Mail von Ihnen mit Namen und Ostfalia-E-Mail-Adresse.

⁷ Je nach gewählten Einstellungen, werden Ihnen für geänderte Dateien etc. andere nette Symbole angezeigt. Es kommt auch vor (zumindest bei mir), dass gar keine Symbole angezeigt werden und beim nächsten „Booten ist wieder alles gut“.



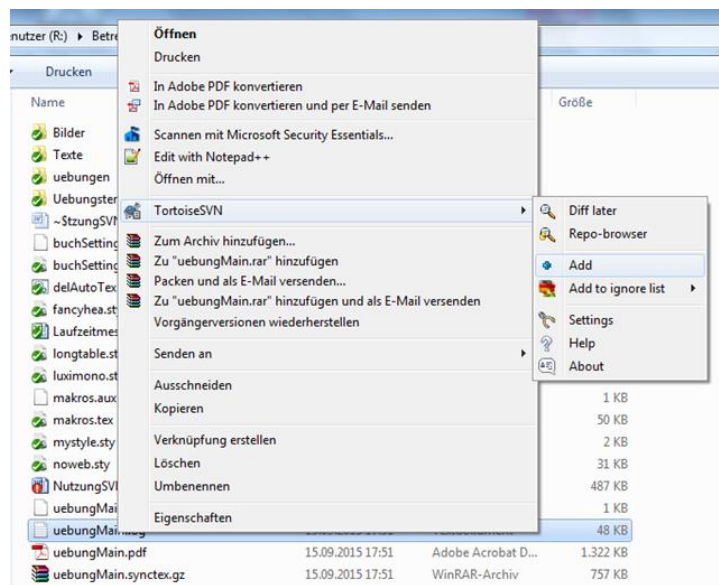
Nun erscheint ein Fenster, in dem Sie spezifizieren **sollten**⁸, was Sie an der Datei geändert haben, damit Ihre Team-Mitglieder Ihre Änderung nachvollziehen können:



Nach Eingabe des Kommentars wählen Sie „O.K.“

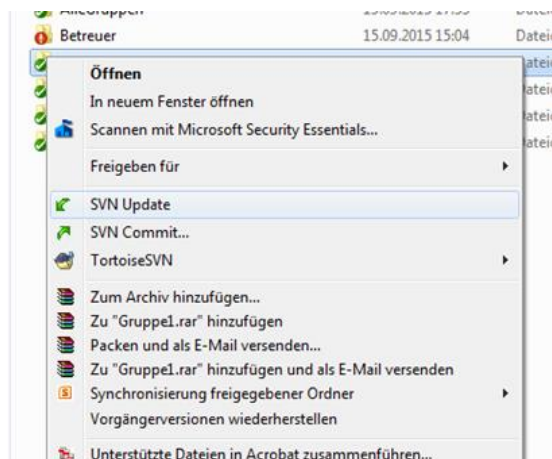
⁸ Der gute Software-Entwickler **muss** einen sinnvollen Kommentar eingeben. Für ihn ist es selbstverständlich, seinen anderen Teamkollegen und den Betreuern mitzuteilen, welche Änderung er/sie gerade durchgeführt hat. Es geht natürlich auch ohne.

Wenn Sie eine neue Datei erstellt haben, die noch nicht im Repository vorhanden ist, müssen Sie diese zunächst hinzufügen. Wählen Sie wiederum mit der rechten Maustaste die Datei aus und wählen im Menü vom „TortoiseSVN“ den Unterpunkt „Add“ aus.



Damit ist die Datei allerdings nur bereit für das Hinzufügen zum Repository. Sie ist noch nicht dorthin übertragen worden. Dazu müssen Sie die Datei nun abermals mit der rechten Maustaste markieren und „SVN Commit“ auswählen. Nun befindet sich die Datei im Repository und steht allen anderen Teammitgliedern (also auch mir) zur Verfügung.

Wollen Sie die Änderungen der anderen Teammitglieder (oder von mir) auf Ihren Rechner übertragen, wählen Sie das Verzeichnis, das aktualisiert werden soll, mit der rechten Maustaste aus und wählen hier nun „SVN-Update“.⁹



⁹ Gut wäre es, wenn Sie, bevor Sie mit der Arbeit „morgens um 4:30 Uhr“ beginnen, zunächst IhrName und AlleGruppen auschecken, damit Sie nicht Änderungen an einem alten Stand vornehmen. Dann sollten Sie regelmäßig (nicht einen ganzen Tag warten) Ihre Änderungen einchecken. Vor dem Einchecken sollten Sie allerdings zuvor nochmals „SVN Aktualisieren“ wählen, damit Sie sicher sind, dass Ihre Teamkollegen nicht in der Zwischenzeit schon Änderungen durchgeführt haben.

Selbstverständlich laden Sie nichts „hoch“, wenn Sie nicht mindestens einmal geprüft haben, dass es auch funktioniert. Nach dem von uns im Rahmen der Veranstaltung praktizierten Vorgehensmodell „Test First“ bedeutet das, dass Sie zuvor die TestSuite durchlaufen lassen und nur bei „grün“ einchecken.

Normalerweise wählen Sie also nach dem Einschalten des Rechners etc. zunächst „SVN-Aktualisieren“ auf Ihrem gesamten Verzeichnis aus (IhrName und AlleGruppen). Sie übersetzen das Programm zunächst und führen dann alle Tests aus. Es sollte nun die Meldung „alle Tests erfolgreich“ oder etwas Entsprechendes ausgegeben werden. Andernfalls haben Sie keine lauffähige Version ausgetestet. Klären Sie diesen Zustand mit Ihren Teammitgliedern.

Nun führen Sie Ihre Änderungen durch und fügen ggf. neue Dateien zum Repository hinzu. Es reicht im Normalfall aus, wenn im Repository die C++-Dateien (Endungen „.h“ und „.cxx“ bzw. „.cpp“) und ggf. noch Datendateien (Partygäste mit Aufstehzeiten, Flugzeuge mit frühesten Zeiten etc.) eingecheckt werden.

Nach erfolgreicher Durchführung Ihrer Änderungen laufen natürlich immer noch alle Tests¹⁰ (erst prüfen, dann erst einchecken). Nun führen Sie „Hinzufügen“ und „SVN Übertragen“ durch und das Repository enthält auch Ihre Änderungen.

Dies ist der Normalfall. Dann gibt es noch die Problemfälle. Hierzu wird auf die oben genannten Links verwiesen, z.B. ein Konflikt, da auch Ihr Team-Mitglied gerade die gleiche Datei geändert hat.

Zum „Herumspielen“ mit Subversion erstellen Sie am besten ein Verzeichnis „PlayGround“ oder ein ähnlicher Name. Dies sollten Sie während der ersten Laborübungen zum Kennenlernen der Subversion-Funktionalitäten verwenden.

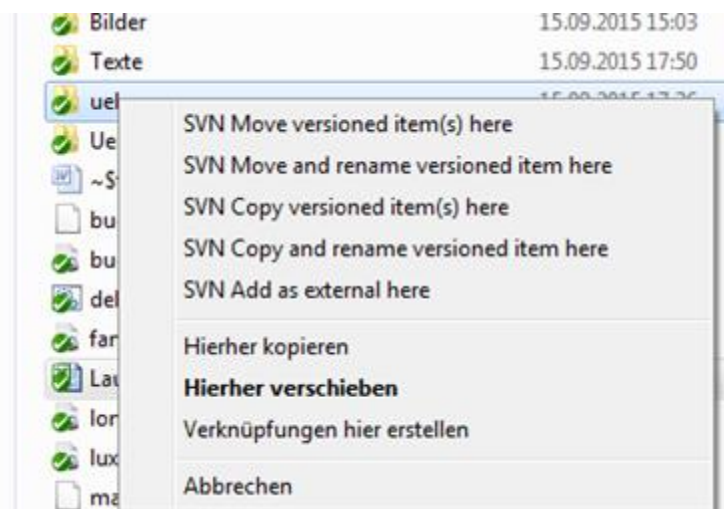
Verschieben von Dateien in andere Verzeichnisse.

Grundsätzlich ist es keine gute Idee, zum Umbenennen einer Datei oder zum Verschieben einer Datei in einem anderen Ordner, die Datei in dem einen Ordner zu löschen (TortoiseSVN→Delete) und in dem anderen Ordner wieder hinzuzufügen (TortoiseSVN→Add). Man verliert in diesem Fall die Log-Historie.

Es gibt einen besseren Weg.

- Zum Umbenennen einer Datei im gleichen Ordner sollte deshalb der Rename-Befehl verwendet werden (TortoiseSVN→Rename).
- Zum Verschieben in ein anderes Verzeichnis müssen Quell- und Zielverzeichnis bereits beide unter SVN-Kontrolle stehen (Hinzufügen für beide durchgeführt sein). Dann kann man die Quelldatei im Quellordner mit der rechten Maus-Taste markieren (festhalten der Maustaste) und dann in den Zielordner ziehen. Es öffnet sich dann der folgende Dialog .

¹⁰ Das müssen Sie als gewissenhafter Programmierer sicherstellen. Es wird nicht (nie!) eingecheckt, wenn die Tests nicht laufen und schon gar nicht (niemals nie!!!), wenn noch Compilerfehler drin sind.



Es wird nun „SVN Move versioned files here“ ausgewählt. Ein anschließendes Commit auf beide Verzeichnisse führt dazu, dass die Datei in dem einen Verzeichnis aus dem Repository entfernt wird und in dem anderen Verzeichnis hinzugefügt wird. Dieses ist zwar auch ein Löschen- und ein Hinzufügen-Schritt. Nun hat SVN allerdings erkannt, dass die beiden Dateien die gleichen sind und die Historie im SVN-Repository bleibt erhalten.

Welche Dateien sollen eingchecked werden?

Checken Sie nur die Dateien ein, die nicht automatisch erzeugt werden. Das ganze bin-Verzeichnis ist schon mal **nicht** einzuchecken. Entsprechendes gilt für das von z.B. doxygen erzeugte doc-Verzeichnis. Sehen Sie sich die von mir erstellte Kommandoprozedur delAuto.bat an. Dieses Batch-Skript löscht vieles Überflüssige, aber Achtung die Datei geht rekursiv in alle Unterverzeichnisse und löscht alle exe-Dateien.

Ganz schlechte Idee, diese Datei auf dem Rootverzeichnis von C:\ auszuführen.