

Versionskontrolle mit Subversion

Was ist Versionskontrolle?

- Ermöglicht gemeinsames Editieren von Quelltext und anderen Dateien
- Meist zentrale Ablage (Repository), Zugriff über Netzwerk
- Bearbeitung immer über lokale Kopie, nie auf Server
- System speichert Kopien aller alten Versionen:
 - Zugriff auf alte Dateizustände
 - Zugriff auf alte Zustände des Gesamtprojekts
 - Bei Textdateien Ausgeben von „Diffs“ zwischen beliebigen Versionsständen
- Beispiele: RCS, CVS, BitKeeper, Subversion, arch...

Warum besonders wichtig/nützlich bei XP?



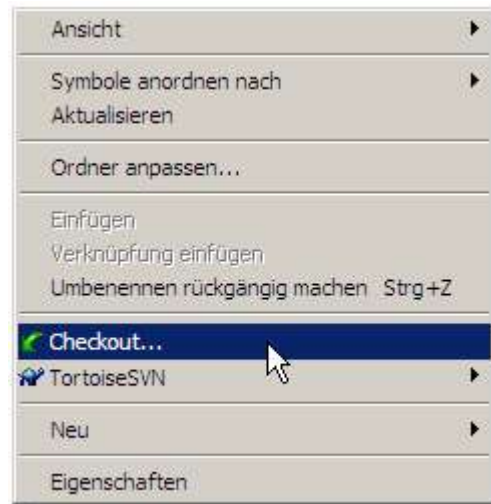
Ersatz für CVS (concurrent version system): Ähnliche Benutzung, behebt viele Designfehler, bessere Nutzung über das Netzwerk, Open Source, für viele Betriebssysteme erhältlich

- Vorbereitung: „Checkout“, d.h. Kopie der aktuellsten Dateiversionen vom Server auf den eigenen Rechner
- Wiederholtes Kopieren von geänderten Versionen auf den Server:
 - „Update“ - Etwaige Änderungen der anderen in eigene lokale Kopie übernehmen
 - „Commit“ - Eigene Änderungen auf den Server kopieren (auch „checkin“/„einchecken“ genannt)

*Warum nicht einfach Locks auf dem Server?
Was passiert bei gleichzeitigen Änderungen?*

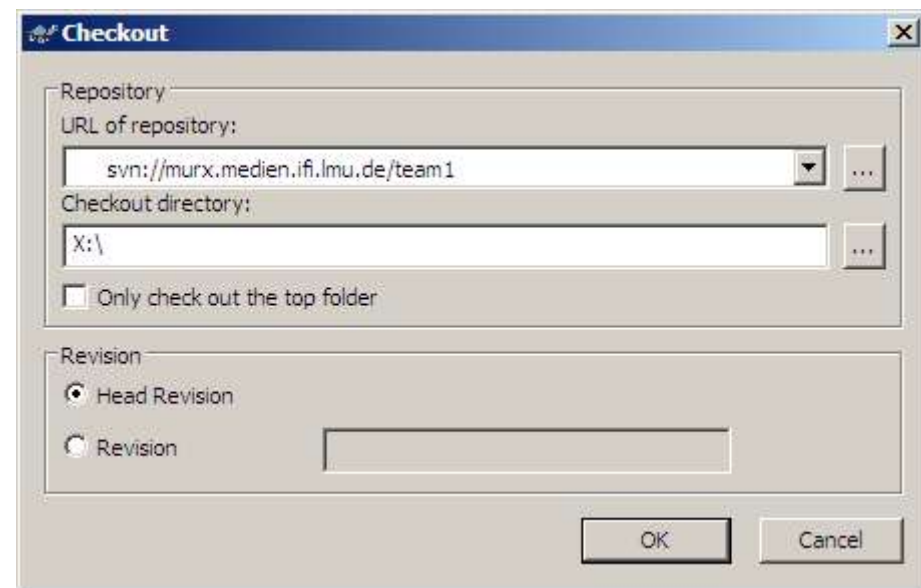
Neuer Checkout der letzten Version

TortoiseSVN ist ein Windows-Client für SVN, in den Explorer eingebettet.



In leerem Verzeichnis „Checkout“
im Explorer-Kontextmenü wählen

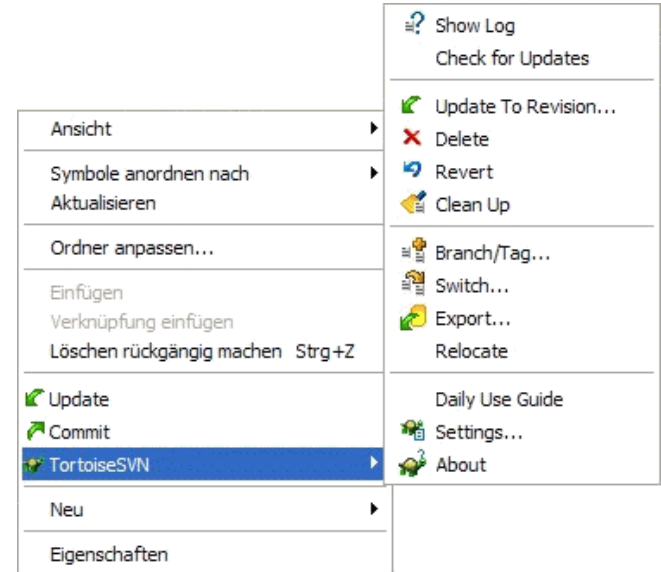
„svn:"-URL und
lokales Verzeichnis
angeben



Kommandozeile (Checkout ins aktuelle Verzeichnis):
`svn checkout svn://murx.medien.ifi.lmu.de/team1`

Weitere Aktionen

- Update, Commit: Vom Kontext-Menü aus, möglich sowohl für einzelne ausgewählte Dateien als auch ganze Verzeichnisse, rekursiv (Kommandozeile: `svn update`, `svn commit`)
- Umbenennen, Löschen (`svn move`, `svn delete`)
- Revert: Verwerfen lokaler Änderungen, Überschreiben der Datei(en) mit aktuellster Repository-Version (`svn revert`)
- Alte Versionen ansehen: Repository-Browser (`svn cat`, `svn diff`)



Konflikte

Was passiert, wenn zwei Leute dieselbe Datei editieren?

- A und B holen Datei vom Server, editieren sie lokal
- A führt Commit durch
- B führt Commit durch, bekommt Fehler: Lokale Version der Datei ist älter als die aktuellste im Repository (*„Commit failed: Out of date“*)
- B führt Update durch
 - 1. Fall: Änderungen an verschiedenen Stellen, oder dieselbe Änderung an derselben Stelle – Subversion führt automatisch „Merge“ durch (während des Update: *„merged: Dateiname“*)
 - 2. Fall: Änderungen überschneiden sich – manuelle Konfliktauflösung nötig

Konfliktauflösung

- Beim Update erscheint die Meldung „*Conflicted: Dateiname*“
- Datei öffnen – sie enthält an der Stelle des Konflikts beide Versionen:

Änderung des anderen Teammitglieds A (aktuelle Repository-Version 14):

a
bx
c
d
e
f

```
a  
<<<<<<< .mine  
by  
=====  
bx  
>>>>>>> .r14  
c  
d  
e  
f
```

Original (zuletzt von mir, B, aus dem Repository geholte Version 13):

a
b
c
d
e
f

Meine (B's) versuchte Änderung:

a
by
c
d
e
f

- Nach Behebung des Konflikts im Explorer-Kontextmenü für Datei „Resolved“ auswählen (`svn resolved dateiname`)
- SVN speichert Dateien mit Endungen `.mine`, `.r13` und `.r14`

Tipps

- Nur Code einchecken, der getestet ist und funktioniert
- Möglichst kleine Änderungen, möglichst oft einchecken
- Vor dem Einchecken immer Update durchführen und etwaige Konflikte auflösen
- Wenn während des Updates Änderungen der anderen vom Server geholt werden, den Code nochmals vor dem eigenen Checkin testen
- Umbenennen/Löschen von Dateien immer über SVN, nicht direkt im Dateisystem - *warum?*
- Änderungen über den Repository-Browser vermeiden

Tipps für das Flash-Projekt

- Ins Repository sollten *alle* Dateien des Projekts eingecheckt werden: Code, `.fla`, `.flp`, Bilder & Sound, Dokumentation... (Binärdateien werden von Subversion erkannt)
- Es sollten keine automatisch generierten Dateien eingecheckt werden, d.h. kein `.swf`
- Umbenennen von Dateien zu Backup-Zwecken ist nicht nötig – alle alten Versionen sind über den Repo-Browser verfügbar!
- Zugriff auf den SVN-Server nur vom Rechnerpool aus möglich

Hiiiiiiifäääää!!!

- *Update, Commit usw. funktioniert nicht mehr* – neuen Checkout in anderes Verzeichnis machen, zuletzt geänderte Datei(en) hinüberkopieren
- *Wie sah der Code gestern aus, als noch alles funktionierte?* -
neuer Checkout mit Revision „{2002-02-17 15:30}“
(`svn diff --revision "{2002-02-17 15:30}" dateiname`)
- *Nach einem Konflikt lässt sich die Datei nicht mehr einchecken (remains in conflict)* – Konflikt beheben, „Resolved“ im Menü auswählen
- Sonst: richard.atterer@ifi.lmu.de oder 2180-4654 ;-)

Links

<http://subversion.tigris.org/>

<http://svnbook.red-bean.com/svnbook/>

<http://tortoisesvn.tigris.org/docs.html>