

Eine kurze Einführung in

Subversion

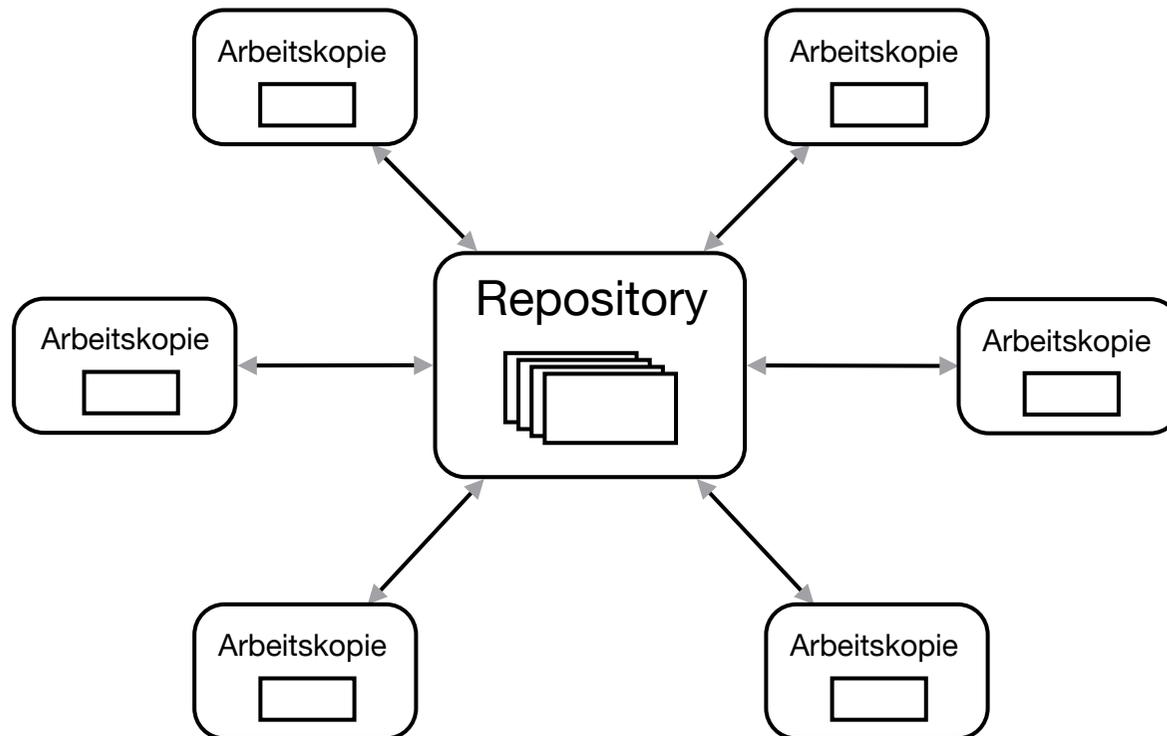
Michael Brenner

Warum Versionskontrolle?

- Ein Projekt - mehrere Autoren
- Alle Mitglieder sollten
 - Zugriff auf die Projektressourcen haben
 - gleichzeitig an der selben Ressource arbeiten können
 - Konflikte erkennen und beheben können
 - Veränderungen an Ressourcen verfolgen können

Subversion

- Zentrale Verwaltung aller Projektressourcen
- Lokale Arbeitskopien
- Zugriff auf aktuellen Stand sowie Historie

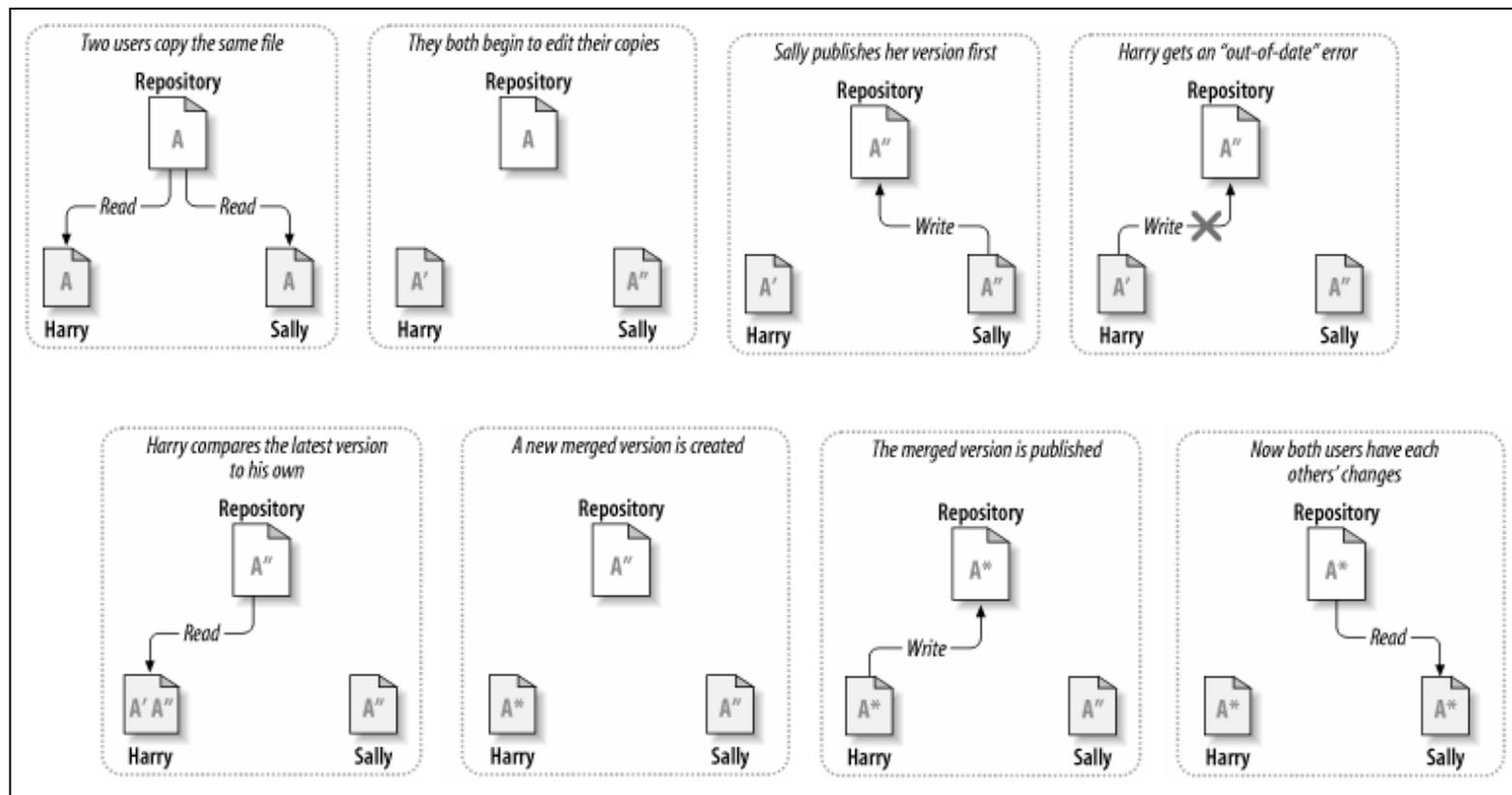


Subversion

- Mehrere Personen können gleichzeitig die selbe Ressource modifizieren
- Änderungen automatisch zusammenfügen (merge) bzw. auf unauflösbare Konflikte hinweisen
- Änderungen nachvollziehen (logs, diff) und wieder rückgängig zu machen
- Verschiedene Versionen und Varianten (tags, branches)

Das Copy-Modify-Merge-Modell

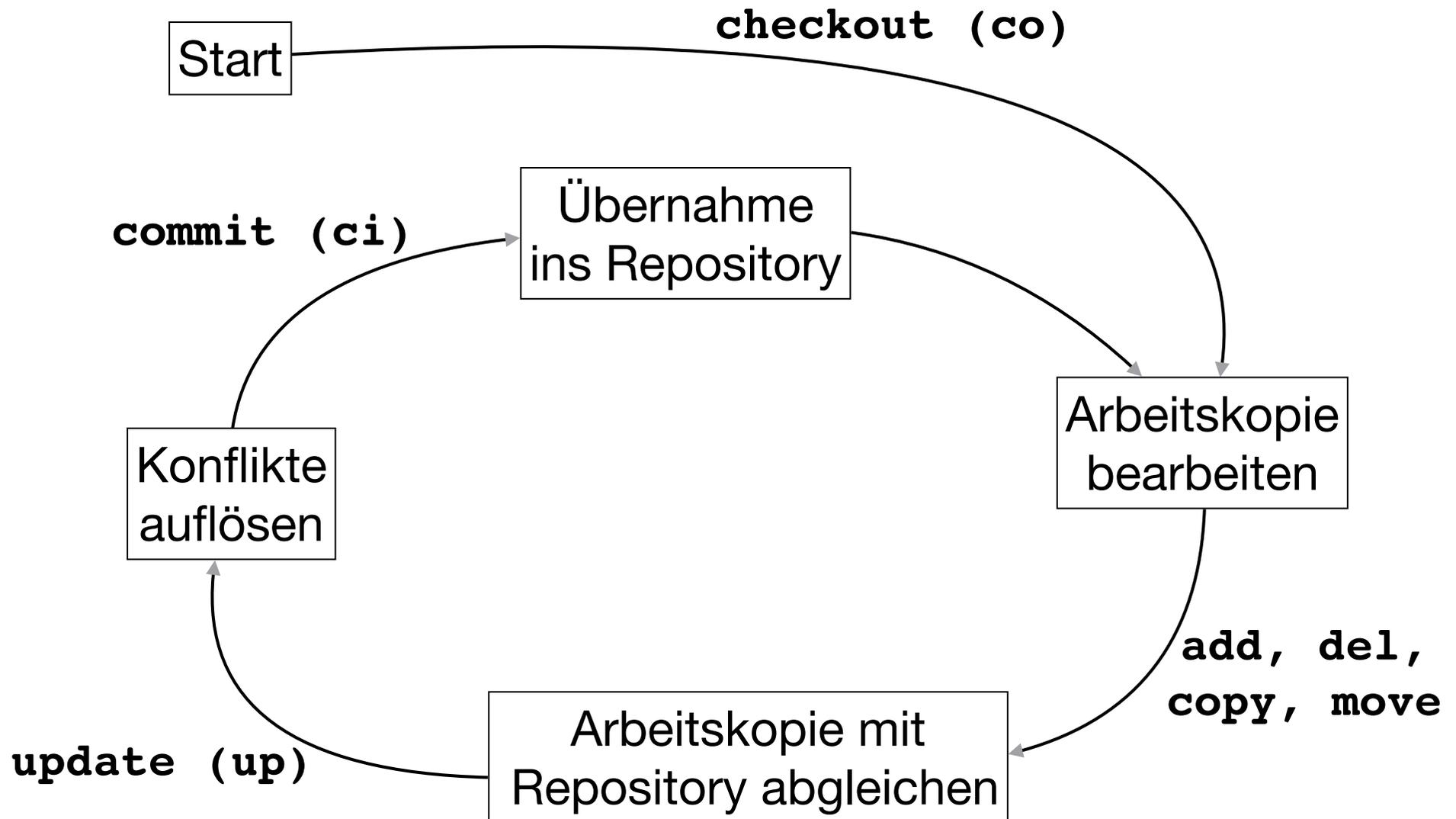
- Probleme bei gleichzeitiger Modifikation vermeiden



Arbeiten mit Subversion

- Benutzer holt Daten erstmalig aus dem Repository und legt eine Arbeitskopie an
 - `$ svn checkout URL`
- Dateien hinzufügen, löschen, etc
 - `$ svn add foo` (ähnlich: `del`, `move`, `copy`, `mkdir`)
- Aktualisierung der Arbeitskopie (Änderungen von anderen einarbeiten)
 - `$ svn update`
- Änderungen ins Repository bringen
 - `$ svn commit [file] -m "informativer Logeintrag"`
- Überprüfen von Änderungen
 - `$ svn status` (zeigt, welche Dateien verändert wurden)
 - `$ svn diff` (Änderungen im Detail)
 - `$ svn log` (Historie der Logeinträge)

Typischer Arbeitszyklus



Revisionen

- Repository enthält *alle* Versionen der verwalteten Dateien (kompakt repräsentiert)
- Revisionen: Versionsnummer für jede Version einer Datei
- Zugriff auf bestimmte Version
 - meist mit: `-r revision`
 - Z.B.: `svn up -r 42`

Ist das alles? Von wegen!

- Branches
- Tags
- Einrichten und Verwalten von Repositories
- ...

- Das Subversion-Buch
 - <http://svnbook.red-bean.com/nightly/de/>
 - sehr lesbar, sehr hilfreich, umsonst!
- `svn help` is your friend
 - `svn help` listet alle Befehle
 - `svn help commit` (o.ä) gibt Hilfe für spezifische Befehle

Das Praktikums-Repository

Dear Michael Brenner,

you have been granted access to the aipractical09 repository.

To check out a working copy, save the attached file into your `~/.ssh` directory (create it if it does not yet exist), and append the following three lines to your `~/.ssh/config` file (again, create it if it does not yet exist):

```
Host aipractical09
HostName alfons.informatik.uni-freiburg.de
IdentityFile ~/.ssh/aipractical09.key
User helmert
```

Then type:

```
chmod 600 ~/.ssh/aipractical09.key
svn checkout svn+ssh://aipractical09 DIRNAME
```

where DIRNAME is the directory name for your working copy. If you choose a non-existing directory, svn will create it automatically.

Best regards,

your aipractical09 repository bot

Das Praktikums-Repository

- Struktur:

```
projects/  
  project1/  
  project2/  
  project3/  
  project4/  
  project5/  
groups/  
  group1/  
    project1/  
    project2/  
    project3/  
    ...  
  group2/  
    ...  
  ...
```

- Mehr Informationen per Mail, wenn ihr euch in Gruppen zusammengetan habt