

ftw.
Forschungszentrum Telekommunikation Wien
[Telecommunications Research Center Vienna]

CAMPARI-Trainings-Workshop Versionskontrolle mit Subversion

Martina Umlauf
umlauft@ftw.at

Kplus
Erasmus-Programme

Zielgruppe und Vorkenntnisse ftw.

- Praktikanten CAMPARI und Interessierte
- Computer-Kenntnisse (Commandline)
- keine weiteren Vorkenntnisse erforderlich (CVS o.ä. keine Voraussetzung)
- Benutzung, NICHT Administration von SVN

© ftw. 2005 2 Kplus
Erasmus-Programme

Overview ftw.

- Zielgruppe & Vorkenntnisse
- Einführung
- Filesharing im Team
- Arbeitsablauf & Kommandos

© ftw. 2005 3 Kplus
Erasmus-Programme

Einführung - Motivation ftw.

Wer braucht das?

- jeder, der schon mal 27 verschiedene Versionen eines Files gespeichert hat
- speziell, wenn im **Team** gearbeitet wird
- nicht nur für Sourcecode, sondern auch:
 - Schriftstücke (LaTeX, Word,...)
 - Config-Files
 - Content für Websites
 - ...
- Sogar beliebige Binärfiles!

© ftw. 2005 4 Kplus
Erasmus-Programme

Geschichtlicher Abriss ftw.

- Subversion (SVN)
- freies, Open-Source Version Control System
- Entwicklungsbeginn 2000
- Verbesserung von CVS (weit verbreitetes OSS Version Control System)
- Komplette Neuimplementierung - Befehlssatz weitgehend kompatibel zu CVS
- gesponsert von CollabNet und RedHat
- URL: <http://subversion.tigris.org>
- freies Buch *Version Control with Subversion* von Ben-Collins-Sussmann, Brian W. Fitzpatrick, C. Michael Pilato zum Downloaden

© ftw. 2005 5 Kplus
Erasmus-Programme

Repository + Sandboxes ftw.

© ftw. 2005 6 Kplus
Erasmus-Programme

Overview



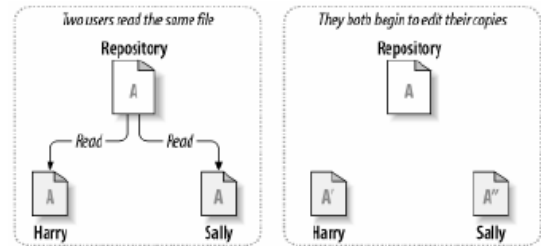
- Zielgruppe & Vorkenntnisse
- Einführung
- Filesharing im Team
- Arbeitsablauf & Kommandos

© ftw. 2005

7



Filesharing im Team - Problem / 1

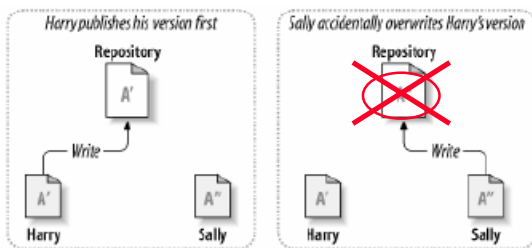


© ftw. 2005

8



Filesharing im Team - Problem / 2

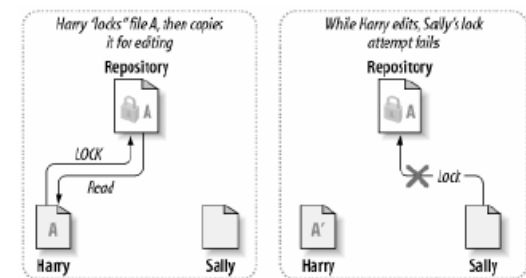


© ftw. 2005

9



Filesharing im Team - Locking / 1

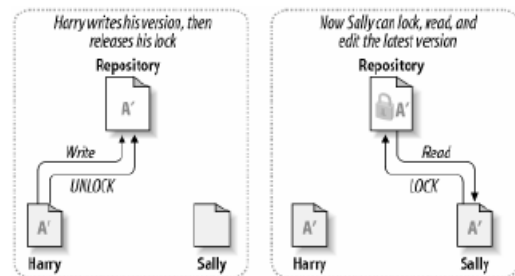


© ftw. 2005

10



Filesharing im Team - Locking / 2



© ftw. 2005

11



Filesharing im Team - Locking / 3



Probleme

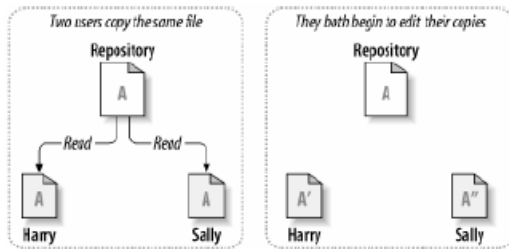
- Lock vergessen; Urlaub => niemand anders kann zugreifen
- Oft garnicht nötig - zb. Harry editiert Anfang, Sally Ende vom File ohne Überschneidungen
- Falsches Gefühl der Sicherheit: Änderungen in 2 verschiedenen Files können sich trotz Locking semantisch "schlagen"

© ftw. 2005

12



Filesharing - Copy-Modify-Merge / 1

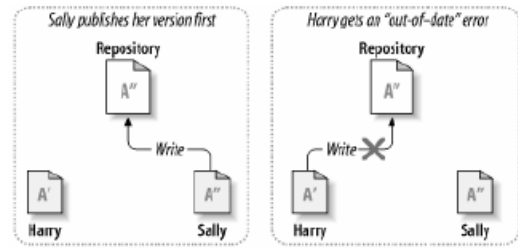


© ftw. 2005

13



Filesharing - Copy-Modify-Merge / 2

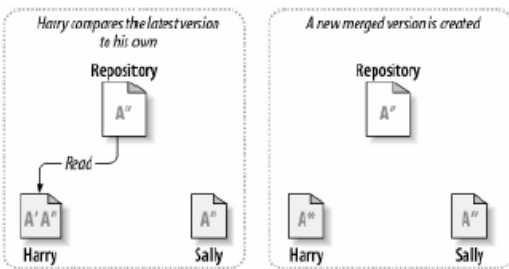


© ftw. 2005

14



Filesharing - Copy-Modify-Merge / 3

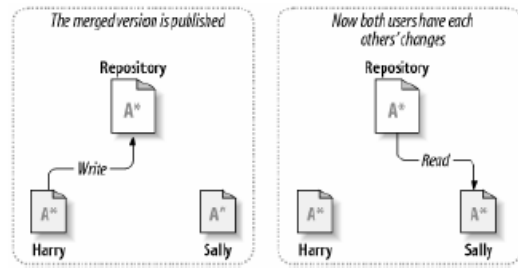


© ftw. 2005

15



Filesharing - Copy-Modify-Merge / 4



© ftw. 2005

16



Filesharing - Copy-Modify-Merge / 5

- Default-Mechanismus von SVN; neueste Version kann zusätzlich Locking
- Schaut nur auf ersten Blick chaotisch aus
- In der Praxis wenige Überlappungen
- "Conflicts" müssen immer von Menschen behoben werden
- Dazu ist **Kommunikation** nötig!
- Bei schlechter Kommunikation hilft auch Locking nichts

© ftw. 2005

17



Overview

- Zielgruppe & Vorkenntnisse
- Einführung
- Filesharing im Team
- **Arbeitsablauf & Kommandos**

© ftw. 2005

18



Arbeitsablauf

- **Working Copy auschecken**
 - svn checkout <url> <local dir>
- **Editieren**
 - svn add / delete / copy / move
- **Vergleich mit Repository**
 - svn status / diff / revert
- **Änderungen der Anderen Einbinden**
 - svn update / resolved
- **Eigene Änderungen committen**
 - svn commit

© ftw. 2005 19

Checkout

```

$ svn checkout http://server/repos/calc
A calc
A calc/Makefile
A calc/integer.c
A calc/button.c
Checked out revision 2499.
$ ls -A calc
Makefile integer.c button.c .svn/

```

file:///	lokale Disk
http://	via WebDAV zu Apache Server
https://	mit SSL Verschlüsselung
svn://	Proprietäres Protokoll zu svnsrve server
svn+ssh://	durch SSH Tunnel

© ftw. 2005 20

Editieren

- Files editieren mit beliebigem Editor
- Strukturänderungen SVN mitteilen

```

$ svn add newfile.c
$ svn delete somefile.c
$ svn copy foo.c bar.c
$ svn move foo.c bar.c

```

© ftw. 2005 21

Vergleich mit Repository / 1

- Oft weiß man garnicht mehr, was man alles editiert hat

```

$ svn status
M bar.c           wurde modifiziert
C stuff/loot/lump.c   Conflict!
A stuff/loot/bloo.h  zum hinzuf. Vorgemerkt
D stuff/fish.c       zum Löschen vorgemerkt
? foo.o             nicht im SVN

```

```

$ svn status --show-updates --verbose
M * 44 23 sally README
M 44 20 harry bar.c
* 44 35 harry stuff/trout.c
D 44 19 ira stuff/fish.c
Status against revision: 46

```

© ftw. 2005 22

Vergleich mit Repository / 2

```

$ svn diff
Index: bar.c
-----
--- bar.c (revision 3)
+++ bar.c (working copy)
@@ -1,7 +1,12 @@
+#include <sys/types.h>
+#include <sys/stat.h>
int main(void) {
- printf("Sixty-four slices of American
  Cheese...\n");
+ printf("Sixty-five slices of American
  Cheese...\n");
  ...
$svn diff > patchfile

```

© ftw. 2005 23

Vergleich mit Repository / 3

```

$ svn revert README
Reverted 'README'

```

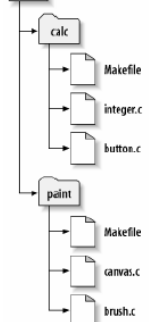
- Kann auch versehentliches Hinzufügen oder Löschen von Files rückgängig machen

© ftw. 2005 24

Fremde Änderungen einbinden / 1



repos



```
$ svn update
U INSTALL
G README
C bar.c
Updated to revision 46.

U ... Updated
G ... merGed
C ... Conflict!
```

© ftw. 2005

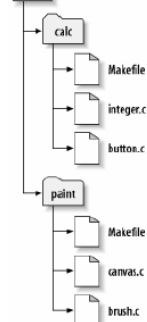
25



Fremde Änderungen einbinden / 2



repos



Bei Konflikt:

filename
- mit Conflict Markers

filename.mine
- letzte eigene Änderungen

filename.OLDREV
- eigenes File vor meinen Änderungen

filename.NEWREV
- neue Version vom Server

© ftw. 2005

26



Konfliktresolution Beispiel / 1



- Hannes und ich arbeiten am gleichen File
- Hannes checkt als erster ein
- ich bekomme Conflict:

```
$ svn update
C sandwich.txt
Updated to revision 2.

$ ls -l
sandwich.txt
sandwich.txt.mine
sandwich.txt.r1
sandwich.txt.r2
```

© ftw. 2005

27



Konfliktresolution Beispiel / 2



```
$ cat sandwich.txt
Top piece of bread
Mayonnaise
Lettuce
Tomato
<<<<<< .mine
Salami
Mortadella
Prosciutto
=====
Sauerkraut
Grilled Chicken
>>>>>> .r2
Creole Mustard
Bottom piece of bread
```

Fileinhalt

Meine Änderungen

Trennzeichen

Änderungen von Hannes

© ftw. 2005

28



Konfliktresolution Beispiel / 3



- Also was tun?
- **Miteinander reden!!**
- Und gemeinsamen Kompromiß finden
- am besten gemeinsam hinter den Rechner setzen und File editieren

© ftw. 2005

29



Konfliktresolution Beispiel / 4



```
$ vi sandwich.txt
Top piece of bread
Mayonnaise
Lettuce
Tomato
Salami
Prosciutto
Grilled Chicken
Creole Mustard
Bottom piece of bread
```

Fileinhalt

Kompromiß

- Nicht vergessen: alle Trennzeichen löschen!

```
$svn resolved sandwich.txt
```

© ftw. 2005

30



Eigene Änderungen Comitten



```
$ svn commit
$ svn commit -m "log message"
$ svn commit --file logmsg.txt
Sending sandwich.txt
Transmitting file data.
Committed revision 3.
```

- Bei "out-of-date":
 - svn update
 - svn commit
- Bei Conflict:
 - svn update
 - svn resolve
 - svn commit
- Vor commit prüfen:
 - svn status -u

© ftw. 2005

31



Revisions



- Revision Numbers: 0...n
- Revision Keywords:
 - HEAD ... Neueste im Repository am Server
 - BASE ... Vor eigenen Änderungen ("pristine")
 - COMMITTED ... Letzter Commit
 - PREV ... COMMITTED-1
 - \$ `svn diff --revision PREV:COMMITTED foo.cc`
 - Änderungen zu letztem Commit
 - \$ `svn log --revision BASE:HEAD`
 - alle logs seit letztem update
- Revision Dates:
 - \$ `svn log --revision {2005-07-13}:{2005-08-05}`

© ftw. 2005

32



Diverses



- `svn log`: zeigt Log Messages
- `svn cat`: zeigt File Inhalt an
 - \$ `svn cat --revision 2 foo.cc`
- `svn list`: Repository Inhalt
- `svn cleanup`: bei Inkonsistenzen durch Programmabbruch
- `svn help`

© ftw. 2005

33



Wie krieg ich mein File zurück? / 1



- `svn revert` - verwirft lokale Änderungen vor commit
- Wenn schon committed:

1. Suchen

```
$ svn log --verbose
```

```
...
```

```
-----
r808 | joe | 2003-12-26 14:29:40 -0600 (Fri, 26 Dec 2003) |
      3 lines
Changed paths:
  D /calc/real.c
  M /calc/integer.c
Added fast fourier transform functions to integer.c.
Removed real.c because code now in double.c.
...
```

© ftw. 2005

34



Wie krieg ich mein File zurück? / 2



2. Altes File wiederherstellen

```
$ svn copy --revision 807 http://server/repos/calc/real.c
./real.c
```

```
$ svn status
```

```
A + real.c
```

3. Commit mit entsprechender Log-Message

```
$ svn commit -m "Resurrected real.c from revision 807,
/calc/trunk/real.c."
```

```
Adding real.c
Transmitting file data .
Committed revision 1390.
```

© ftw. 2005

35



References



- **Version Control with Subversion**, Ben Collins-Sussmann, Brian W. Fitzpatrick, C. Michael Pilato, O'Reilly, sowie frei downloadbar unter <http://svnbook.red-bean.com>
- **Subversion Homepage**
<http://subversion.tigris.org>

© ftw. 2005

36



Folgetermine



- **Di, 9. Aug, 9:00-12:00**
Tools: make, doxygen, Kdevelop, vi,...
- **Di, 23. Aug, 9:00-12:00**
Extreme Programming light and good development practices
- **Fr, 26. Aug, 9:00-12:00**
offen, wahrscheinlich Kurzvortrag Python

The End



**Danke für die
Aufmerksamkeit!**