

# Versionskontrolle mit Git

## Erfahrungen eines Subversionnutzers

Hauro

2007-07-28

# Inhalt

## Warum Git?

- Mein Weg zu Git
- Verteilte VCS
- Vorteile von Git

## Git-Interna in 10 Minuten

- Interna? RTFM!
- Objektdatenbank
- Der Index
- SHA1-Summen

## Git in der Praxis

### Warum Git?

- Mein Weg zu Git
- Verteilte VCS
- Vorteile von Git

### Git-Interna in 10 Minuten

- Interna? RTFM!
- Objektdatenbank
- Der Index
- SHA1-Summen

### Git in der Praxis

### Zusammenfassung

# Die Anfänge

- ▶ Erstes Versionkontrollsystem: CVS
  - ▶ PITA: schnell Umstieg auf Subversion
- ▶ Subversion für mehrere Jahre
  - ▶ taugt nicht, wenn man “mobil” ist

# Der Wechsel: Verbesserung?

- ▶ Umstieg auf SVK
  - ▶ Erstes verteiltes VCS (DVCS)
  - ▶ Pro:
    - ▶ Subversion sehr ähnlich
    - ▶ Offline benutzbar
    - ▶ Merging deutlich angenehmer
  - ▶ Contra:
    - ▶ Großer Overhead (Daten/Laufzeit)
    - ▶ Branching umständlich
    - ▶ Kein Cherry Picking

# Nein, weiter suchen!

- ▶ Tests weiterer DVCS
  - ▶ Bazaar (bzd)
  - ▶ Monotone
  - ▶ Mercurial
  - ▶ Darcs
  - ▶ Git
- ▶ Fazit: Alle sehr gut!
  - ▶ Git passte zu meinem Workflow

# Vorteile verteilter VCS

- ▶ Zentrales Repository weiter möglich
- ▶ Commits gegen lokales Repository
- ▶ Leichtes Branching/Merging
- ▶ Cherry-Picking

# Vorteile von Git

- ▶ schnell!
- ▶ Bare Repository
- ▶ Feature Branches
- ▶ gute Dokumentation
- ▶ gute Tools (gitk, cogito, StGit, ...)
- ▶ Umschreiben der History möglich
- ▶ Branches im gleichen Repository

Versionskontrolle  
mit Git

Hauro

Warum Git?

Mein Weg zu Git

Verteilte VCS

Vorteile von Git

Git-Internia in 10  
Minuten

Internia? RTFM!

Objektdatenbank

Der Index

SHA1-Summen

Git in der Praxis

Zusammenfassung

# Inhalt

## Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

## Git-Interna in 10 Minuten

Interna? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

## Git in der Praxis

Versionskontrolle  
mit Git

**Hauro**

### Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

### Git-Interna in 10 Minuten

Interna? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

### Git in der Praxis

### Zusammenfassung

# Interna? RTFM!

- ▶ “Lies den Quellcode!”
- ▶ “Wieso willst du das wissen?”

→ Es hilft beim Verständnis!

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

Git-Interna in 10  
Minuten

**Interna? RTFM!**

Objektdatenbank  
Der Index  
SHA1-Summen

Git in der Praxis

Zusammenfassung

- ▶ Sammlung aller Objekte
- ▶ Objekt:
  - ▶ "Nach Inhalt benannt" über seine SHA1-Summe
    - ▶ Sinn: *nicht* kryptographische Sicherheit (Signatur), sondern Integrität wichtig
  - ▶ Vier mögliche Typen:
    - ▶ Blob: Dateiinhalte (Benutzerdaten)
    - ▶ Tree: Verbindet Blobs zu Verzeichnisstruktur
    - ▶ Commit: Beschreibt einen Tree zum Zeitpunkt des Commits
    - ▶ Tag: Symbolischer Name, enthält Identifier eines Objekts
  - ▶ Hierarchische Strukturen zwischen Objekten

- ▶ aka: Current Directory Cache
- ▶ Datei mit virtuellem Verzeichnisinhalt
- ▶ Wichtige Eigenschaften:
  - ▶ Kann den gecachten Zustand wiederherstellen
  - ▶ Kann Inkonsistenzen zwischen gecachtem und gegenwärtigem Zustand feststellen
  - ▶ Kann Informationen über Merge-Konflikte darstellen
- ▶ (fast) alle Tools arbeiten auf dem Index
- ▶ Normalerweise keine direkte Manipulation

- ▶ Alle Objekte nach ihren SHA1-Summen benannt
- ▶ Tools arbeiten mit SHA1-Summen (nur Anfang nötig)
- ▶ Stellt Integrität sicher (git fsck)

# Inhalt

## Warum Git?

- Mein Weg zu Git
- Verteilte VCS
- Vorteile von Git

## Git-Interna in 10 Minuten

- Interna? RTFM!
- Objektdatenbank
- Der Index
- SHA1-Summen

## Git in der Praxis

Versionskontrolle  
mit Git

**Hauro**

### Warum Git?

- Mein Weg zu Git
- Verteilte VCS
- Vorteile von Git

### Git-Interna in 10 Minuten

- Interna? RTFM!
- Objektdatenbank
- Der Index
- SHA1-Summen

### Git in der Praxis

Zusammenfassung

# Git in der Praxis

Der nachfolgende Teil wurde live gezeigt.  
Die Slides wurden zum besseren Verständnis der  
Audioaufzeichnung nachträglich hinzugefügt.

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

Git-Internia in 10  
Minuten

Internia? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

**Git in der Praxis**

Zusammenfassung

# Anlegen und Konfigurieren

```
git init
git config user.name Hauro
git config user.email hauro@cccmz.de
cat .git/config
```

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

Git-Internia in 10  
Minuten

Internia? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

Git in der Praxis

Zusammenfassung

# Hinzufügen von Dateien

```
echo 'Hello , World!' >hello.txt  
git status  
git add hello.txt  
git status  
git commit
```

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

Git-Interna in 10  
Minuten

Interna? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

Git in der Praxis

Zusammenfassung

# Änderungen ansehen

```
echo 'Hello , Git!' >>hello.txt
git status
git diff
git add hello.txt
git diff
git diff --cached
git commit
```

# Weitere Änderungen

```
echo 'Goodbye, Git!' >goodbye.txt  
git add goodbye.txt  
git diff  
git diff --cached  
git commit
```

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

Git-Internia in 10  
Minuten

Internia? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

Git in der Praxis

Zusammenfassung

# Neue Arbeitszweige (Branches)

```
git branch
git branch feature -x
git branch
git checkout feature -x
echo feature >feature.txt
git add feature.txt
git commit
```

# Wechseln von Arbeitszweigen

```
git branch
git checkout master
ls
git checkout feature-x
ls
```

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

Git-Internia in 10  
Minuten

Internia? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

Git in der Praxis

Zusammenfassung

gitk ist ein graphisches Programm, das das Repository sehr übersichtlich anzeigt.

gitk

# Zusammenfügen von Zweigen (Merge)

```
git checkout master  
git merge feature-x
```

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git

Verteilte VCS

Vorteile von Git

Git-Internia in 10  
Minuten

Internia? RTFM!

Objektdatenbank

Der Index

SHA1-Summen

Git in der Praxis

Zusammenfassung

# Bare Repository anlegen

```
cd ..  
git clone --bare repos repos.git  
ls repos.git/  
ls repos/.git/
```

# Klonen eines Repositories

```
git clone repos.git my_new_repos
cd my_new_repos/
ls
```

Versionskontrolle  
mit Git

**Hauro**

Warum Git?

Mein Weg zu Git  
Verteilte VCS  
Vorteile von Git

Git-Internia in 10  
Minuten

Internia? RTFM!  
Objektdatenbank  
Der Index  
SHA1-Summen

Git in der Praxis

Zusammenfassung

- ▶ Git ist besser als sein Ruf
  - ▶ andere DVCS sind auch gut
- ▶ DVCS sind die Zukunft
  - ▶ große Auswahl, leicht erlernbar
- ▶ Besser jetzt umsteigen als später
  - ▶ schont Nerven, spart Arbeitszeit