GIT – Verteilte Versionskontrolle

Robin Tibor Schirrmeister
22.04.2013

Vorteile GIT

Gegenüber SVN:

- offline lokale Versionskontrolle
- einfacher neue Funktionalitäten entwickeln / reintegrieren
- Repository wiederherstellbar, wenn zentrales Repository kaputt geht
- GitHub! :)

Demo

- Erstellen wir ein kleines Progamm mit GIT! :)
- https://rawgithub.com/robintibor/dbpedia-spotlight-finaltest/master/dbpedia-annotate-ui.html

SVN-Befehle → GIT-Befehle

svn checkout	git clone
svn add	git add
svn commit	git commit -a git push
svn update	git pull

GIT Branching/Merging

- git branch *branchname* – Erstellen
- git checkout *branchname* – Hinwechseln
- git push *remotename branchname* - Teilen
- git merge *branchname* – Vereinigen
- git branch -d *branchname* – Lokal Löschen
- git push origin - -delete *branchname* – Entfernt Löschen

GIT verteilte Workflows

- Weitere dezentralere Workflows möglich, z.B.:
  - Fork Repository, Entwickle → Pull Request

Nachteile

- Lokale Branches und Commits können verloren gehen
- Kompliziertere Benutzung
- Nicht gedacht für Rechteverwaltung über einzelnen Ordnern

Tutorials

- http://rogerdudler.github.io/git-guide/
- https://git.wiki.kernel.org/index.php/GitSvnCrashC
- http://learn.github.com/

Use Cases ;)

- Elmar im Zug (lokale Commits)
- Florian erstellt neues UI in branch
- Ich mache Broccoli kaputt, aber nur lokal :)