

= Git für Philosophen == Kurzer geschichtlicher Abriß

über [Versionsverwaltung](#) allgemein. Die (unvollständige) Entwicklung ohne kommerzielle Lösungen:

* 1972 [SCCS](#) (Source Code Control System) im Unix Umfeld (Bell Labs) * 1982 [RCS](#) (Revision Control System), [diff](#)-Sammlung parallel zu jeder Datei, * 1989 [CVS](#) (Concurrent Versions System), Netzwerk-Erweiterung von RCS (diff Sammlungen auf einem zentralen Server), * 2000 [SVN](#) (Subversion) http://de.wikipedia.org/wiki/Apache_Subversion - „CVS ohne dessen Fehler“, * 2005 [Git](#) - Neuentwicklung zur Verwaltung der Linux-Kernel Quellen, kein zentraler Server nötig * 2005 [Mercurial](#) - „git“ für CVS/SVN Umsteiger, auch kein zentraler Server nötig

== Konzepte * [Lock-Modify-Write](#) pfui, * [Copy-Modify-Merge](#)

== Topologien

* lokal, im Dateisystem eingestreut (SCCS, RCS), * zentrales Repository (CVS, SVN), * verteilte, gleichberechtigte Repositories (git, mercurial)

== Terminologie http://de.wikipedia.org/wiki/Verteiltes_Versionskontrollsystem

* Repository (Repo): Archiv mit kompletter Änderungsgeschichte (was, wann, wer) * checkout: erstmalig Dateien aus dem Repository holen bzw. wiederholt auffrischen, * update: Dateien mit aktuellen Stand aus dem Repository auffrischen, * commit: Änderung(en) in's Repository schreiben, * changeset: Zusammengehörige Änderungen an mehreren Dateien in einem Commit * branch: „Zweig“ - von der Hauptentwicklung abzweigende aufeinanderfolgende Änderungen

- trunk: Hauptzweig (git: 'master')

== Git Benutzung * [Installation für Mac](#) * [git GUI für commit/log](#) * [Bash Completion](#)

git... * [...init](#) - lokal neues Repo anlegen * [...add status diff commit reset](#) (dafür ist GitX sehr gut) * [...checkout branch merge](#) (Branch anlegen, wechseln, löschen) * [...clone](#) - Repo kopieren, * [...fetch pull push](#) - Sync mit anderen Repos == Hosting * [extrem flexibel](#) * Open Source gerne bei <http://github.com> - ich auch: <http://github.com/mro>

== Links * <http://git-scm.com/> - Homepage * <http://gitref.org/> - sehr gute Doku * <http://www.kernel.org/pub/software/scm/git/docs/> - Manpages * <http://www.kernel.org/pub/software/scm/git/docs/user-manual.html> - Original Manual

[git](#)

From:

<https://wiki.mro.name/> - □

Permanent link:

https://wiki.mro.name/playground/git_fuer_philosophen?rev=1288451854

Last update: **2010/10/30 17:17**

