Grundlagen
der Versionierungssoftware
Git

Hochschule Darmstadt
SS 13

Norbert Bastius

Was ist Git ?

Git ist eine Versionierungssoftware, die im Jahre 2005 von der Linux Entwickler Community entwickelt wurde, weil die kostenlose Nutzung des damals populären Tools „Bitkeeper“ durch dessen Betreiber nicht mehr erlaubt wurde.

Ursprünglich wurde Git zur Quellcode-Verwaltung des Linux-Kernels entwickelt.

Wozu braucht man VCS und wie arbeitet Git ?

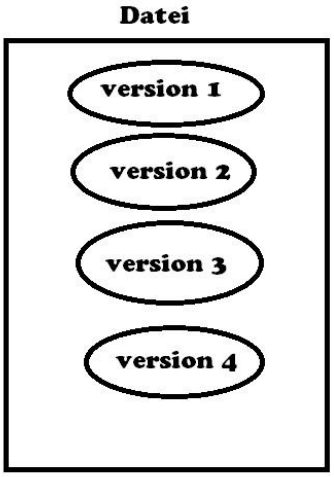
Versionskontrollsysteme erlauben es dem Benutzer zum Beispiel, eine Datei in einen früheren Zustand zurückzusetzen, also in eine frühere Version.

Dies erlaubt es, mögliche Probleme zu erkennen und Fehler ganz einfach rückgängig zu machen.

Ein weiteres Feature von VCS ist, dass man genau zurückverfolgen kann, wer, wann welche Änderungen an dieser Datei vorgenommen hat.

Durch jede Änderung an einer Datei, wird von der Software eine aktuelle Version erstellt. Falls man z.B. Version 1-4 hat, kann man jederzeit ganz leicht von Version 4 zur Version 2 oder 3 springen und Änderungen vornehmen.

Hat man eine Datei ausversehen zerstört und, kann man sie ohne grossen Aufwand in eine funktionierende Version zurücksetzen.



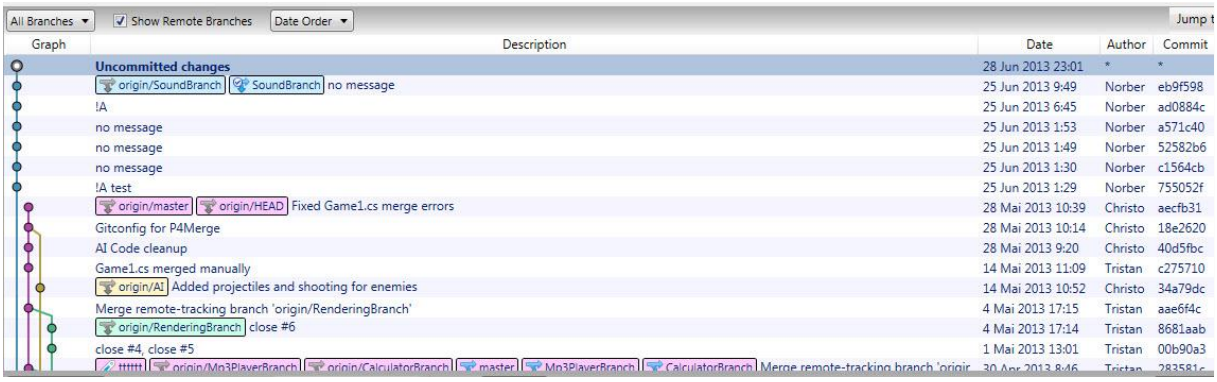
Beispiel: Eine Datei in Git enthält hier 4 Versionen, also 4 verschiedene Zeitzustände der gleichen Datei (Snaphots)

Git sieht eine Datei als eine Reihe von Snapshots, also Momentaufnahmen (= Versionen).

Wenn man also eine Datei verändert (beispielsweise ein Foto) und dann committet, dann wird diese Datei nicht als neue Datei abgespeichert oder kopiert, sondern es wird neuer Snapshot generiert, also eine neue Version, die als Verknüpfung zur vorherigen Version eingefügt wird.

In der oben gezeigte Skizze wäre dieser neu generierte Snapshot also ein neues Oval mit der Bezeichnung „Version 5“

- Git kann den Verlauf einer Datei oder eines Projektes aus der lokalen Datenbank der festplatte entnehmen, ohne auf einen externen Server zugreifen zu müssen.
  Man kann z.B. lokal von der Festplatte auf die aktuelle Version der Datei zugreifen und sie mit der Version vergleichen, die schon einen Monat alt ist, um Unterschiede festzustellen.
- Änderungen werden in Git in Checksummen umgerechnet bevor sie gespeichert werden. Git sucht nach Informationen über Dateien aus der Datenbank nicht nach dem Dateinamen sondern nach dem Wert der Checksumme. (Beispiel im Screenshot unter Commit)



- So sieht ein Arbeitsprozess in Git aus

1. Dateien im lokalen Arbeitsordner bearbeiten.
2. Dateien für den nächsten Commit markieren
3. Committen, damit Git die Snapshots dauerhaft im lokalen Ordner abspeichert.

Befehle

- **Clone -** Ein existierendes Repository zum Arbeiten verwenden bzw auf die Festplatte abspeichern.
- **Push –** Änderungen an ein entferntes Repository senden
- **Checkout –** Änderungen rückgängig machen
- **Fetch –** Herunterladen von Änderungen in das lokale Repository
- **Merge –** Verschmelzt lokales Repository mit dem vom entfernten Server, damit beide auf dem gleichen Stand sind.
- **Pull –** Lokales Repository mit den neusten Änderungen updaten (Pull ist Fetch und dann Merge)

Drei Hauptzustände in Git

Committed : Daten sind in der lokalen Datenbank gesichert

Modified : Die Datei wurde geändert, aber noch nicht committed

Staged : Eine Datei ist in ihrem aktuellen Zustand für einen commit vorgemerkt