

# Git-Schnellreferenz

Stefan Gast

27. März 2009

## 1 Repository anlegen

Aufgabe	Befehl
Lokales Repository erzeugen	<pre>cd myproject git init git add . git commit</pre>
Öffentliches Repository anlegen	<pre>newgrp projektgruppe mkdir meinprojekt.git cd meinprojekt.git git init --bare</pre>
Öffentliches Repository mit Schreibzugriff für Gruppe	<pre>newgrp projektgruppe mkdir meinprojekt.git cd meinprojekt.git git init --bare --shared=group</pre>
Kopie eines Projektes erzeugen	<pre>git clone ssh://username@server.de/pfad</pre>

## 2 Informationen abfragen

Aufgabe	Git-Befehl	SVN-Befehl
Log anzeigen	<code>git log</code>	<code>svn log</code>
Änderungen seit letzter Übertragung	<code>git diff</code>	<code>svn diff</code>
Versionen vergleichen	<code>git diff &lt;key1&gt;..&lt;key2&gt;</code>	<code>svn diff &lt;re_a&gt;:&lt;re_b&gt;</code>
Status ausgeben	<code>git status</code>	<code>svn status</code>

## 3 Git für SVN-Benutzer

Schritt	Git-Befehl	SVN-Befehl
1. Neueste Version aus zentralem Repository holen	<code>git pull</code>	<code>svn update</code>
2. Bearbeiten der Dateien	-	-
3. ggf. Dateien hinzufügen	<code>git add &lt;file1&gt; &lt;file2&gt;</code>	<code>svn add &lt;file1&gt; &lt;file2&gt;</code>
4. Änderungen kontrollieren	<code>git diff</code>	<code>svn diff</code>
5. Neue Version an zentrales Repository schicken	<code>git commit -a</code> <code>git push</code>	<code>svn commit</code>

## 4 Änderungen rückgängig machen

Aufgabe	Befehl
Arbeitskopie zurücksetzen	<code>git reset --hard HEAD</code>
Commit zurücknehmen	<code>git revert &lt;key&gt;</code>

## 5 Tags

Aufgabe	Befehl
Tag erzeugen	<code>git tag &lt;tagname&gt; [&lt;key&gt;]</code>
Tags anzeigen	<code>git tag</code>
Tag löschen	<code>git tag -d &lt;tagname&gt;</code>

## 6 Arbeiten mit mehreren Entwicklungszweigen

Aufgabe	Befehl
Zweig anlegen	<code>git branch &lt;name&gt; [&lt;start&gt;]</code>
Zweige anzeigen	<code>git branch</code>
Zweig wechseln	<code>git checkout &lt;zweig&gt;</code>
Zweige zusammenführen	<code>git checkout &lt;ziel&gt;</code> <code>git merge &lt;quelle&gt;</code>
Zweig löschen	<code>git branch -d &lt;zweig&gt;</code>
Neuen lokalen Zweig an öffentliches Repository schicken	<code>git push &lt;remote&gt; &lt;neuer_zweig&gt;</code>

## 7 Verteilte Entwicklung

Aufgabe	Befehl
Mit anderem Repository verbinden	<code>git remote add &lt;name&gt; &lt;adresse&gt;</code>
Remote Branches auf neuesten Stand bringen	<code>git fetch &lt;remote_name&gt;</code>
Änderungen ansehen	<code>git diff HEAD..&lt;remote&gt;/&lt;zweig&gt;</code>
Änderungen in aktuellen lokalen Zweig übernehmen	<code>git merge &lt;remote&gt;/&lt;zweig&gt;</code>