

CVS-Kurzreferenz

Wichtige CVS-Befehle und -Optionen

Invokation

```
cvs [ global-opt ] command [ command-opt ] [ command-args ]
```

Globale Optionen

-d Spezifiziere das CVSROOT

-q Unterdrücke unrelevante Ausgaben.

-n Verändere keine Dateien (weder lokale noch solche im CVSROOT). Kann mit **-q** und **update** benutzt werden, um gut lesbare Ausgabe der Stati der Dateien zu erhalten. Beispiel:

```
$ cvs -qn update | grep -v ^?
```

-z *gzip-level* Setze den Kompressionslevel. Gültige Level reichen von 1 (schnell, schwache Kompression) bis 9 (langsam, gute Kompression), oder 0, um Kompression zu deaktivieren (Standard).

--help Gibt Hilfe aus. Wenn *command* angegeben ist, wird eine Liste mit Befehlsoptionen und kurzen Erklärungen ausgegeben.

Befehle

```
add [ opt ] entry0 [ ... ]
```

Markiere die Dateien oder Verzeichnisse *entry0*, ... zum Hinzufügen. Die angegebenen Einträge müssen lokal existieren.

-k* Lege die Art der *Schlüsselwortsubstitution* fest. Am häufigsten wird **-kb** benutzt. Damit wird eine Datei als binär markiert: Schlüsselwortsubstitutionen sind deaktiviert und Zeilenenden werden nicht konvertiert.

```
checkout [ opt ] module0 [ ... ]
```

Lege eine Arbeitskopien der Module *module0*, ... an.

-D *date* Benutze die neueste(n) Revision(en) nicht älter als *date*. Diese Option ist dauerhaft (ein *sticky date*) und impliziert **-P**. Ein Datum kann auf sehr verschiedene Weisen angegeben werden. Falls es ein Leerzeichen enthält, muss ggf. gequotet werden. Beispiele:

```
1972-09-24
"1972-09-24 20:05"
"24 Sep 1972 20:05"
yesterday
"1 hour ago"
```

-r *tag* Benutze die Revision *tag*. Diese Option ist dauerhaft (ein *sticky tag*) und impliziert **-P**. *tag* ist eine Revisionsnummer, eine symbolische Revision (s. *tag*) oder der Name eines Entwicklungsstranges (*branch*, s. *tag*).

-P Lösche leere Verzeichnisse.

-j tag1 Verschmelze die Änderungen zwischen dem größten gemeinsamen Vorfahren (von *tag1* und der Arbeitskopie) und *tag1* in die Arbeitskopie. *tag1* darf ein Datum enthalten: z.B: **-j Branch:Date**.

-j tag2 Verschmelze die Änderungen zwischen dem größten gemeinsamen Vorfahren (von *tag1*, *tag2* und der Arbeitskopie) und *tag2*, ohne die Änderungen zwischen dem Vorfahren und *tag1*, in die Arbeitskopie. *tag2* darf ein Datum enthalten.

commit [*opt*] [*entry0 ...*]

Reiche lokale Änderungen von *entry0*, ... ein. Die angegebenen Einträge müssen mit dem Repository synchronisiert sein (s. *update*).

-r tag Ist *tag* eine Revisionsnummer, größer als die höchste vorhandene: Benutze Revisionsnummer *tag*. Ist *tag* Name einer *branch*, reiche als neueste Revision der *branch* ein.

-m text Benutze *text* als Log-Eintrag und rufe nicht einen Editor dazu auf.

diff [*opt*] [*entry0 ...*]

Betrachte Unterschiede der Einträge *entry0*, ... zwischen Revisionen.

-D date1 Betrachte die Unterschiede der Revisionen an *date1* zur Arbeitsversion.

-D date2 Betrachte die Unterschiede der Revisionen an *date1* zu den Revisionen an *date2*.

Keine, eine, oder beide **-D date** Optionen können durch **-r tag** ersetzt werden.

release [*opt*] *dir*

Gib ein Arbeitsverzeichnis frei (beseitige CVS/ Unterverzeichnisse).

-d Lösche das angegebene Verzeichnis nach der Freigabe lokal.

remove [*opt*] *entry0* [...]

Markiere die Dateien oder Verzeichnisse *entry0*, ... zum Löschen. Die angegebenen Einträge dürfen lokal nicht mehr existieren.

-f Lösche die angegebenen Einträge vor dem Entfernen lokal.

status [*opt*] [*entry0 ...*]

Gib Statusinformationen über Dateien oder Verzeichnisse *entry0*, ... aus.

-v Beziehe Informationen über tags mit ein.

tag [*opt*] *tag* [*entry0 ...*]

Markiere die Einträge *entry0*, ... mit der symbolischen Revision *tag*. Tags können, sollten aber möglichst nicht, verschoben oder gelöscht werden.

-D, -r siehe *checkout*.

-b Es wird *tag* nicht als symbolische Revision eingetragen. Stattdessen wird ein Entwicklungszweig (*branch*) mit dem Namen *tag* erstellt, zu dem die Einträge *entry0*, ... gehören.

update [*opt*] [*entry0 ...*]

Synchronisiere die Einträge *entry0*, ... mit dem Repository.

-D, -r, -P, -j siehe *checkout*.

-A Setze alle *sticky tags / dates / options* vor der Synchronisierung zurück.

-d Hole auch Verzeichnisse, die im Repository existieren, im Arbeitsverzeichnis aber fehlen.