Der Tagging Mechanismus im CVS

Aus der Diplomarbeit zur Automatisierung von Regressionstests

Jens Hanisch

1. August 2002

Für die Verwaltung getesteter Versionen des XCTL-Systems ist die Quelltext-Versionskontrolle CVS sehr hilfreich. Sie ermöglicht das Einfügen von sogenannten „Tags“, welche alle Quellen aus dem CVS-Modul in ihrem aktuellsten Zustand kennzeichnen, um im Laufe der Projektarbeit auf diese zugreifen zu können. Auf diese Weise kann eine bestimmte Version im CVS-Repository festgehalten werden. Lauffähige und vollständig getestete Versionen werden mit einer eindeutigen Marke (Tag), die das Datum des Testdurchlaufs beinhaltet, versehen.

Ein Regressionstest wird immer auf der aktuellsten Version des Testobjekts durchgeführt. Vom Tester muss sichergestellt werden, dass auch wirklich die überprüfte Version im CVS-Repository markiert wird, und nicht eine Version, die nach der Testdurchführung verändert wurde. Eine Möglichkeit dazu ist das „Einfrieren“ einer Version mittels einer Marke direkt vor dem Test. War der Test erfolgreich darf die Marke im CVS-Repository liegen bleiben, bei einem Misserfolg, muss sie dagegen wieder entfernt werden. Ein vollständiger Regressionstests wird mit den folgenden Schritten durchgeführt:

1. `% cvs co XC010701`
   Aktuelle Testobjektversion aus dem CVS-Repository holen.
   Dabei wird ein Verzeichnis mit dem Namen des Moduls im Homeverzeichnis des Testers erstellt. In diesem Kommando ist der Name des Moduls: `XC010701`.

2. `% cvs tag Test<Datum> XC010701`
   Markiert alle aktuellen Quelldateien im CVS-Repository, die sich im selben Zustand, wie die Dateien im Homeverzeichnis des Testers befinden. `<Datum>` ist durch das aktuelle Datum des Regressionstestes zu ersetzen.

3. Kopieren des Verzeichnisses aus dem Homeverzeichnis in ein temporäres Arbeitsverzeichnis, in dem der Test durchgeführt werden kann. Das Zielverzeichnis darf auf einem beliebigen PC mit Windows9x oder WindowsNT liegen.

4. Kompilieren der Quellen und Erstellen einer ausführbaren Programmversion im erstellten Arbeitsverzeichnis.

5. **Installation** des Testsystems *ATOS* (s. Benutzerhandbuch).

6. Durchführung des **Regressionstests** (s. Benutzerhandbuch).

7. **Auswertung** des Regressionstestlaufs.

8. Nach einem erfolgreichen Testdurchlauf:

   (a) Die Projektteilnehmer sind davon zu informieren, dass die Version mit der Marke `Test<Datum>` im CVS-Repository den Regressionstest bestanden hat.

   Nach einem gescheiterten Testdurchlauf:

   (a) Auf das Homeverzeichnis des Testers, in dem das Verzeichnis mit dem Namen des Moduls liegt, zurück wechseln.

   (b) `% cvs tag -d Test<Datum> XC010701`
   Die aktuelle Version im CVS darf nicht markiert bleiben, da sie sich in einem fehlerhaften Zustand befindet!

   (c) Die Projektteilnehmer sind davon zu informieren, dass die zuletzt aktualisierte Version im CVS-Repository den Regressionstest nicht bestanden hat. Die gescheiterten Testsequenzen sind mit Hilfe der Protokollausgaben zusammen zu stellen, um eine gezielte Fehlerverfolgung seitens der Entwickler starten zu können.

Vor der Durchführung eines Testdurchlaufs, sollte der Tester die anderen Projektteilnehmer darauf hinweisen, an den aktuellen Quellen keine weiteren Veränderungen vorzunehmen, bis die Arbeit des Regressionstestes beendet ist. Schlägt der Testdurchlauf fehl, müssen zunächst die dafür verantwortlichen Veränderungen ausfindig gemacht und repariert werden. Erst dann dürfen weitere Neuerungen in die Quellen aufgenommen werden. Somit kann verhindert werden, dass Entwickler ihre fehlerhaften Codezeilen, die zum Scheitern des Regressionstestes führen, in der zwischenzeit erweitern und erneut in das CVS-Repository einspielen.

Die Quelltext-Versionskontrolle CVS bringt noch einige nützliche Kommandos mit sich, die bei der Arbeit mit dem Tagging-Mechanismus sinnvoll sein können.

```
% cvs co -r Test200701 XC010701
```

Erstellt eine gültige und getestete Version aus dem CVS-Repository, die mit der Marke `Test200701` versehen wurde. Das erstellte Verzeichnis mit den Quelldateien steht immer noch unter der Kontrolle von CVS und kann somit als Ausgangspunkt für darauf aufbauende Versionen benutzt werden.

```
% cvs export -r Test200701 XC010701
```

Die Exportfunktion hingegen erstellt ein Verzeichnis mit den Quellen aus dem CVS-Repository, das nicht mehr unter der weiteren Kontrolle von CVS steht.

```
% cvs status -v XC010701
```

Um sich die Tags zu einem Modul anzeigen zu lassen, ist die Option `-v` sehr hilfreich. Dieses Kommando gibt zum Beispiel alle Tags des Moduls `XC010701` aus.

Da die Marke eines erfolgreichen Testdurchlaufs sehr wichtig für die - Regressionstest unterstützte - Entwicklung ist, wird sie in den Testpaketen zu den jeweiligen Anwendungsfällen im Web-Repository mitgeführt. Dabei kennzeichnet die Marke im Kopf des Dokumentes die letzte festgehaltene Version im CVS-Repository, welche einen erfolgreichen Durchlauf mit allen zugehörigen Testfällen dieses Testpaketes absolvieren konnte.