



## De beste drielettertools

**G**oede software-ingenieurs hebben vaak sterke voorkeuren en communiceren daar fanatiek over. Ook over tools voor versiebeheer, ook wel revisieadministratie of configuratiemanagement genoemd. Ik heb dan altijd drie vragen: wat is de beste tool? Waarom is dat gereedschap het beste? En vooral: waarvoor gebruik je het eigenlijk?

Immers, configuratiemanagement is, volgens de Itil-definitie, zoiets als het bijhouden van wie wat waarom en waar doet. Lees je verder, dan moet die inventarisatie jaarlijks worden bijgewerkt, waarbij je controleert of iedereen die configuratie-items nog steeds gebruikt, je oude items afschrijft en nieuwe toevoegt. In mijn beleving staat dit ver af van efficiënt goede code ontwikkelen.

Voor ontwikkelaars is versiebeheer zoiets als regelmatig alle broncode opslaan, tezamen met meta-informatie: wie heeft wat wanneer en waarom veranderd.

En al lijkt dat op bovenstaande Itil-definitie, het is echt anders. Zo is de frequentie veel hoger en bovendien verandert het item: *de code*. Dat laatste is essentieel. Het gaat niet om het beheren van de code, maar het beheersen van de veranderingen.

Dat bewaren kan heel primitief met Zip of Tar, of heel geavanceerd en met een goed doordacht proces. Helaas gaat de discussie vooral over tools, zowel in de media als bij ontwikkelaars. Gelukkig zijn er heel veel gereedschappen, zodat we daar veel over kunnen communiceren. Ook ik heb mijn lijstje favorieten. Ik begon ooit met RCS, later werd dat CVS en nu heeft Bzr mijn voorkeur. Waarom? Waarschijnlijk selecteer ik op naamlengte: drielettertools lijken mij het beste. Al het goede komt immers in drieën. Blijft de derde vraag over: waarvoor?

Alle tools, zowel de open, de *free* als de commerciële, ondersteunen dezelfde drie stappen: *check-out*, *edit* en *commit*. De naamgeving kan anders zijn, maar dat ritme is altijd de basis. Logisch, een ontwikkelaar werkt in die volgorde. Maar het kan snel complexer worden als er meerdere ontwikkelaars zijn. Een voorbeeld in *plain old C*: ontwikkelaar 1 is bezig met file *a.c*, ontwikkelaar 2 met file *b.h* en ontwikkelaar 3 met *c.c*. Die laatste wil zich concentreren op zijn eigen veranderingen, niet op de integratie met *a* en *b*. Elke tool belooft hierbij te helpen, maar zodra een koppelvlak zoals *b.h* verandert, is toolsupport in de praktijk zeer beperkt. Als één prototype in *b.h* een extra parameter krijgt, moet dat direct in alle *C*-files worden doorgevoerd, zowel de implementatie (*a.c*) als bij alle aanroepen in *c.c*. Dat maakt een file dus ongeschikt als configuratie-item.

Helaas geldt dit ook voor alle andere denkbare items zoals classes, modules of directory's. Er is altijd een eenvoudig tegenvoorbeeld te vinden: zodra je het koppelvlak verandert, gaat het fout, of dat nu prototype, Api of interface heet.

Dus nogmaals de vraag: waarvoor gebruiken we die tools? Want ervaren ingenieurs weten dat je niet zomaar koppelvlakken kunt veranderen. Dit geldt in alle engineeringdisciplines, maar software heeft nu eenmaal het imago dat je alles gratis kunt aanpassen, dus veranderen interfaces en Api's vaak. Dat dat in de praktijk niet tot problemen leidt, komt niet zozeer door de versiebeheertools maar door de bekwaamheid van de ontwikkelaars. Die zullen het koppelvlak pas na onderlinge afstemming aanpassen, of desnoeds pas na goedkeuring van de architect. Het gaat om die afstemming. Daarmee beheers je veranderingen. De revisiebeheertool zorgt *hopelijk* voor consistente code, een zipfile vlak voor de veranderingen, een tarfile vlak erna of twee labels in het codearchief.

Om code beheerst te veranderen, hebben we een andere tool nodig: iets als Twitter lijkt me heel bruikbaar. Of Yammer, of ... Kies maar uit; er zijn eindeloos veel communicatie-tools. Elk van die gereedschappen is geschikt om te overleggen, om codeveranderingen te beheersen. Ook hier gebruik ik een drielettertool. Want al is mijn favoriet het equivalent van de tarball binnen de moderne communicatie, het werkt: gewoon even bellen per gsm.



**Albert Mietus**  
albert@mietus.nl