

# 'Scripts in waterbeheer bieden onvermoede mogelijkheden'

Tien jaar geleden stelde Harry Boukes in het blad *Stromingen* al dat 'een hydroloog moet kunnen programmeren'. Ondergetekenden halen zijn stelling graag weer uit de kast. Anno 2010 is het voor een goede hydroloog nog steeds een vereiste enige vorm van programmeren te beheersen om alle gegevensstromen en modellen onder controle te kunnen houden. We willen echter de stelling veralgemeniseren tot 'iedere (water)adviseur die met veel data werkt, moet kunnen (laten) programmeren'. De kwaliteit van het werk zal daardoor hoger worden.

In het waterbeheer zijn steeds meer beslissingen gebaseerd op berekeningen met modellen of *spreadsheets*.

Die nemen in omvang toe en moeten worden gevoed met grote hoeveelheden gegevens uit diverse bronnen. Het is vaak monnikenwerk, met veel digitaal knippen en plakken, om deze data in het juiste formaat te krijgen.

Ook in andere toepassingen wordt vaak handwerk toegepast bij het bewerken van gegevens, bijvoorbeeld bij het aanvullen of combineren van reeksen, omzetten van gegevens naar het juiste formaat, bepalen van statistieken of andere afgeleide gegevens. Behalve tijdrovend is het risico op moeilijk traceerbare fouten groot. Als later toch verbeteringen noodzakelijk blijken, is er vaak geen tijd of geld meer om het werk opnieuw te doen en blijft deze verbetering uit. Dit kan ook anders.

Iedereen die exact kan uitleggen hoe men een databewerking uit moet voeren, kan in theorie een eenvoudig script (laten) programmeren. Zo'n script leest invoergegevens, knipt, plakt en schrijft de resultaten weg. Met een beetje enthousiasme is het script gebruikersvriendelijk te maken. We hebben het hier niet over het programmeren van volledige applicaties; dat is een vak apart. We doelen op scripts voor relatief eenvoudige, maar tijdrovende databewerkingen. Behalve de bekende voordelen van tijdswinst en een kleinere kans op fouten, zijn er nog andere, onvermoede voordelen. We geven twee praktijkvoorbeelden voor Waterschap Rivierenland: een eenvoudig en een vrij complex script.

## Interpolatiescript

Rivierenland maakte tijdens de uitvoering van een project gebruik van neerslaggegevens. Hierin bleken veel korte hiaten voor te komen, waarvoor aanvulling via interpolatie nodig was. Tot voor kort werd dit in Excel met formules en knip- en plakwerk opgelost. Een vervelend klusje. Hiervoor is in Excel een VBA-script geschreven, zodat geselecteerde cellen via lineaire interpolatie zijn op te vullen. Nu het script er is, blijkt dat het ook is te gebruiken voor andere gegevens. Zo wordt het nu bijvoorbeeld gebruikt bij het interpoleren van concentraties van stoffen die slechts eenmaal per maand zijn gemeten. Opeens blijkt het prettig in plaats van frustrerend om met deze gegevens te werken en is de energie te gebruiken voor belangrijker zaken.

## Stofbalansscript

Het waterschap is ook bezig met het opstellen van stofbalansen voor de deelstroomgebieden. Per gebied en hydrologisch jaar kostte dit bijna twee dagen knip- en plakwerk. Een adviseur is ingehuurd om voor vijf deelstroomgebieden stofbalansen te maken. Deze adviseur kon echter programmeren en ontwikkelde ruim binnen de beschikbare tijd een script, compleet met foutafhandeling en tests. Hiermee was uiteindelijk binnen een half uur per deelstroomgebied en hydrologisch jaar de klus te klaren. Uiteraard zijn de eisen voor de formaten van de invoerbestanden nu iets strenger dan voorheen, maar dat bleek een kleine prijs. Door de grote tijdswinst zijn scenario's heel snel door te rekenen om de effecten van de invoerparameters te analyseren, iets dat voorheen onhaalbaar was. Er is nu meer tijd beschikbaar voor analyse van de resultaten. En door de automatische controle op de invoergegevens komen fouten daarin nu sneller boven water.

## Conclusie

Met het tweede voorbeeld willen we de reikwijdte van de mogelijkheden van scripts aangeven, maar eigenlijk gaat het ons vooral om scripts van het eenvoudige type. We willen stimuleren dat databewerkende vakgenoten de voordelen van programmeren gaan inzien. Zelf programmeren is geen vereiste, vaak loont het ook iemand anders zo'n script te laten schrijven. De voordelen van tijdswinst en verkleinde kans op fouten zijn evident. Onverwachte voordelen zijn mogelijkheden voor scenario- en gevoeligheidsanalyse of om nieuwe informatie uit bestaande gegevens te verkrijgen, bijvoorbeeld via statistieken. Maar het grootste onverwachte voordeel is dat weer tijd resteert voor datgene waarmee je werkelijk bezig wilt zijn. Bewerkelijke analyses worden weer leuk. Er blijft meer tijd over voor plezier in het werk!

**Koen van der Hauw (Grontmij)**

**Hella Pomarius (Waterschap Rivierenland)**