

## **Overheids-IT-beleid, nostalgie, cynisme en hoop voor de verre toekomst**

Daan Rijsenbrij roert nogal wat zinvolle zaken aan maar is vaak erg algemeen en verwacht mogelijk teveel van bepaalde oplossingen.

Een Digitale Rijksbouwmeester lijkt best een aardig idee. Maar met bijvoorbeeld de (fysieke) Rijksbouwmeester zijn de resultaten lang niet altijd zo positief geweest. Architecten zijn nu eenmaal zelfbewuste (goud)haantjes die er vaak erg persoonlijke ideeën op na houden. Het in de boekjes komen of prijzen (van collega's) winnen is vaak belangrijker dan functionaliteit, kosten, gebruiksgemak, doelmatigheid en aanpasbaarheid.

Architecten in de echte wereld en in de IT behoren ook vaak tot bepaalde scholen die veelal andersdenkenden verketteren. Algemeen heb je in de IT wereld ook bepaalde scholen. De hautaine MAC-gebruikers, in het verleden de Linux figuren, de hoogleraren die wel 'even' een eigen Operating Systeem bouwen of allerlei 'slimme' software ontwikkelen zonder behoorlijke userinterface. Het staat natuurlijk goed te schimpen op Windows en de Microsoft producten. Hier is best (aardig) op af te dingen maar het is nu eenmaal een grote macht binnen de markt. Als een gemeente overgaat op OpenOffice en het loopt niet zoals gedacht dan is de reactie al gauw: "wie doet ook zo iets?" Gaat het met Windows fout dan is het excuus dat toch iedereen het gebruikt.

### **De oude doos**

In de jaren 70 heb ik meegewerkt aan een operating systeem voor een semiautomatische telefooncentrale. We hadden een Philips P9202 met zowaar 32 K geheugen. Ja octale K's, wel van 16 bits woorden. Binnen in de kast stond Honeywell DDP516. Ik kreeg zelfs 2K geheugen voor de totale diskhandler.

Ik denk nog altijd aan de toenmalige kreet: *no matter how clever the hardware, the software boys will fritter it away*. Je werkte nog veel in assembler. Kwam tot de conclusie dat de instructieset niet orthogonaal was. Dus dat je diverse tests waarop een sprong plaats vond kon combineren. Er waren al wel Winchesterdrives maar veel laden ging nog met papertape. Na een uurtje laden bleek een kleine routine niet aanwezig. Alles voor niets.

We hadden alleen source listings en objectcodes. Dus overlays maken via een sprong ergens. De linking loader bestuderen en ontdekken welke tabellen er allemaal waren. Dus die opslaan. Je kon zelfs relocatie, software op een ander adres laten beginnen. Tegenwoordig maken we ons daarover niet meer druk. Maar bibliotheken van verschillende versies of naam bevatten vaak dezelfde subroutines en kunnen veel ellende veroorzaken.

### **Configuration management**

Voor het testen van het telefoniesysteem was het mogelijk naast de 'officiële' versie vier testversies van elke 1K sector op te slaan. Wilde je er een verwijderen moest je eerst een ponsband uitprinten met vóór de objectcode in leesbare tekens datum en sector. Wisten wij veel. Tegenwoordig heet dat configuration management maar wordt er vaak met een petje naar gegooid. Natuurlijk was er een achterdeurtje om het systeem snel weer op te zetten na een crash. Maar die software was veilig opgeborgen en niet algemeen beschikbaar. Tegenwoordig lijkt het of iedereen met meer dan zeven jaar basisschool die achterdeurtjes weet te hanteren.

We hadden slechts 15 beschermde geheugenplaatsen. Dus echt goochelen om zonder een papertape bandje, al wel met plastic versterkt, het systeem op te kunnen starten. Het lukte mij maar om een geheugenplaats uit te sparen moest sector 0 van de Winchester drive op een speciale manier worden geformatteerd. Forma watte?

## Normalisatie

Deze eeuw heb ik aan de TU/e als tweede studie 39 jaar na mijn eerste doctoraal experimentele natuurkunde een Master Computer Science & Engineering behaald. Natuurlijk is er sinds het begin heel wat veranderd in de IT-wereld. Dat is als je op detailniveau kijkt. Maar als we eerlijk zijn gaat er nog veel op dezelfde weinig gestructureerde manier als toen. Natuurlijk kennen we de reclamekreet *structured programming* waarbij ingewikkelde constructies nodig zijn voor een eenvoudig probleem.

Normalisatie was een onbekend woord aan de TU/e. Ja, één docent had ooit in een IEEE werkgroep gezeten. De meesten denken dat ISO staat voor *International Standards Organization*, daarmee aantoonend dat ze nog nooit een echte ISO norm hebben gezien.

De Nederlandse universiteiten liepen indertijd weg met Algol. Fortran, waarin iedereen toen programmeerde was taboe. Dus behoorlijke opleidingen ontbraken hier. Je had hoogstens een NOVI opleiding T4. De exameneis was standaard Fortran. Maar de examencommissie had die norm nog nooit gezien en dacht dat hun IBM-machines de norm waren. Als docent heb ik hele gevechten moeten voeren. Aan het eind van elke les ruimde ik een kwartiertje in om alle fouten tegen de norm in het dikke NOVI-boek te corrigeren.

Is het nu zoveel beter? Elke paar maanden of jaren komt er wel een nieuwe taal om in te ontwikkelen. Is die dan beter te testen? Moet dat eigenlijk niet al bij de beroemde architectuur worden meegenomen? Kunnen de cowboys minder allerlei trucjes uithalen die ze natuurlijk niet documenteren?

Een taal als Fortran is in de academische wereld hier in Nederland al vele malen dood verklaard. Maar voor parallelle highperformance toepassingen is het wereldwijd, wel in een daarvoor aangepaste vorm, nog steeds een veel gebruikte. Ook ik was in de jaren 70 niet echt tevreden over Fortran. Maar Algol was zo mooi omdat je er recursief in kon werken. In die tijd waren de Torentjes van Hanoi het paradepaardje. Ik heb toen heel wat lieden uitgedaagd om me tenminste één praktische en ook werkende toepassing te tonen. Toen in ieder geval niet gekregen.

Schelden op een taal is gemakkelijker dan er zelf wat aan zien te doen. Ik ben toen lid geworden, later voorzitter van de Nederlandse Fortran Commissie onder het NNI, zoals het toen nog heette. Je kon indertijd alleen suggesties per brief naar ANSI in New York sturen. Op het gebied van character handling, aanvankelijk een zwak punt in Fortran, hebben we nog wel een en ander gedaan gekregen in wat Fortran 77 zou gaan heten.

Er was heel wat kritiek op de draft norm. Gebruikers van software blijken erg conservatief te zijn. Elke fout uit het verleden blijft via het bekende backward compatible eeuwige leven. Eindelijk kwam er in 1977 na vele jaren weer een bijeenkomst van (toen) ISO/TC97/SC5 Programming Languages. Meteen de Amerikaanse delegatie opgezocht en het Fortran contingent daarin. Verteld dat ook wij niet helemaal gelukkig waren met het voorstel maar het stukken beter vonden dan wat we als Fortran 66 hadden. Als Nederland zouden we zelfs voorstellen om het ANSI document als ISO norm te aanvaarden. Echter op één voorwaarde, dit zou de allerlaatste keer zijn dat ANSI X3J3 de norm alleen zou ontwikkelen. Daaruit zijn internationale werkgroepen ontstaan voor diverse talen. De Nederlandse academische wereld schittert daarin door afwezigheid, behalve misschien bij de ontwikkeling van de propaganda taal Ada, waar soms geld mee was te verdienen, maar die voor een flink deel wat het synchronisatie mechanisme betreft stoelde op de Duitse taal Pearl (en niet op Dijkstra). Het goed formele beschrijvingsmodel van Pearl werd weer verprutst door een beschrijving in het Engels.

Waarom vertel ik dit allemaal? Ik ben er nog steeds, ook uit eigen ervaring, van overtuigd dat de Nederlandse IT-opleidingen sterk onder de maat zijn voor bijna alle **praktische** toepassingen.

Natuurlijk zijn er juweeltjes vanuit opleidingsdoeleinden maar verder dan wat proefprojecten komt het niet. Er zijn vaak hele gevechten wat er allemaal voor syntax in de taal moet komen. DO .. UNTILL, DO ... WHILE, wat voor FOR statements. Eigenlijk zijn slecht twee statements nodig, een ASSIGN-statement [waarbij zowel bron als doel kunnen worden opgegeven, dus inclusief I/O] en een conditionele sprong. Velen kennen natuurlijk de kreet van Dijkstra: *GOTO considered harmful*. Logisch als je in een luie stoel hardop ligt te programmeren en een ander het op laat schrijven. Dan is een GOTO die terug springt een ramp.

### **Overheidsorganisatie**

Een heel ander terrein. Hoe is de Overheid opgebouwd? Nagenoeg bijna altijd puur hiërarchisch. Dus de baas heeft de hoogste rang met dito salaris en echte specialisten worden niet werkelijk gewaardeerd. Om verderop te komen moet je naar management functies. De Balkenende norm is het ultieme bewijs. Natuurlijk is het goed de exorbitante salarissen van bepaalde directeuren aan banden te leggen. Tenslotte zijn die veel gemakkelijker te vervangen, ondanks dat ze zichzelf onmisbaar achten. Denk aan lieden als Rijkman Groenink van de ABN-AMRO.

In het verleden had je die schoorsteenklimmers. De boekhouder die als eenoog hoofd IT werd. Ze zijn nog niet geheel uitgestorven. Daarnaast had je de projectswingers. Zorg altijd weg te zijn voordat een project af is. Alles wat goed ging heb jij gedaan en wat fout gaat die klungels na jou. Ik weet er zelfs die op die manier president-directeur van een groot bedrijf zijn geworden.

Kijken we naar de Kamer. Een handjevol lieden met enige technische kennis. In Provinciale Staten barst het van de adviseurs die klusjes voor het eigen bedrijf of dat van een vriendje zien binnen te halen. Elkaar de bal toespelen loont blijkbaar. Rijkswaterstaat, klassiek een technisch bolwerk heeft alweer jaren geleden bijna alle technici de laan uitgestuurd. Pas nu worden er weer mondjesmaat binnengehaald. Ze willen digitale projectplannen met risicoanalyse maar die juristen snappen er de ballen van.

We moeten ineens overal CRM systemen en dat soort ongein. Ooit gehoord van bijvoorbeeld een SAP implementatie die binnen 14 dagen perfect werkte? Bij de concurrenten is het echt niet beter. Ik heb mijn afstudeerproject bij een bekend software house gedaan. Ze gebruikten een tekenprogramma voor stroomschema's. Bepaalde pijltjes stonden verkeerd. "Nee dat gaat helaas niet anders met dit programma." Ik heb er een kwartiertje naar gekeken en vond de oplossing. Maar het was voor hen in de uurtje-factuurtje filosofie niet mogelijk er tijd aan te besteden.

### **Toekomst**

Nee, er moet op ontzettend veel terreinen iets gebeuren om het grote geld verspillen bij de Overheid op IT-gebied te verminderen. Helaas zijn veel voorgestelde oplossingen voor een deel frutselen in de marge. Hoe dan wel? Ons parlementaire stelsel zorgt er helaas voor dat het zeker nog enkele generaties zal duren. De super CIO en Digitale Rijksarchitect zullen vooral nieuwe stokpaardjes teweeg brengen als er geen goed tegenwicht is. Dat zal nog wel even duren. Daarnaast zijn de banden tussen politiek en bedrijfsleven zo hecht dat vriendjespolitiek helaas niet is uit te sluiten. Kijk maar naar uitgerangeerde politici die topbanen in het bedrijfsleven krijgen. Echter, hoop doet leven.

*drs.ir. Ampt*