

# I/O Vivat

Jaargang 22, nummer 2,  
Februari 2007  
ISSN: 1389-0468  
Oplage: 2000

I/O Vivat is het periodiek orgaan van Inter-Actief (de studievereniging voor Informatica, BedrijfsInformatie Technologie en Telematica) en ENIAC, de alumnivereniging voor Informaticastudenten van de Universiteit Twente.

## Redactie

Alexander Spannenburg (voorzitter),  
Jan-Willem Dijkshoorn, Bram van de Laar, Tom Palsma, Mark Timmer

## Gastschrijvers

Sjoerd van der Spoel, Johan Noltes, Bert Perloy, Erwin Starke, Kimberly Lemmens, Gerrit van der Hoeven, Wiek Vervoort, Pascal Dürr, Wilke Havinga, Lodewijk Bergmans, Wim Fikkert, Suzanne Verlijdsdonk, Francis Henninger, Remko Nolten, Technolution

## Drukker

Van den Bosch & Fikkert



## Adressen

E-mail: [vivat@inter-actief.net](mailto:vivat@inter-actief.net)  
Telefoon: 053-4893756  
Fax: 053-4894571  
Postadres: Inter-Actief  
Postbus 217  
7500 AE Enschede  
Internet: [www.inter-actief.net](http://www.inter-actief.net)

Dank aan alle inzenders van kopij. De studievereniging wil de adverteerende bedrijven bedanken voor de goede samenwerking.

Copyright © 2007 Studievereniging Inter-Actief.

Foto voorkant: Jan Cees van Westering, 'Digital meeting' (foto door Bram van de Laar)

## Nut van een redactioneel

Met de studiereis als 'excuus' kon ik de afgelopen twee keer het schrijven van de redactioneel overlaten aan onze technisch redacteur, Jan-Willem. Maar nu ik op moment van schrijven al weer 2 maanden in ons koude kikkerland ben, is het onvermijdelijk weer mijn taak geworden. Tijd om na te denken over het nut van deze pagina, want alles kan beter en alles kan efficiënter, ook bij de I/O Vivat.

De blauwe kolom hiernaast is verplicht, maar het witte gedeelte van deze pagina zou compleet anders ingevuld kunnen worden dan met een redactioneel. Want wat is nou eigenlijk het nut van een redactioneel? Is het een inleiding, een opsomming van de inhoud, een column? Waarschijnlijk een mix van alle drie. Maar wie leest onze redactionelen nou eigenlijk? Want in deze 'gehaaste wereld' kijk je voor een overzicht van de inhoud logischerwijs naar de inhoudsopgave. En drie columns in de I/O Vivat is eigenlijk ook te veel van het goede.

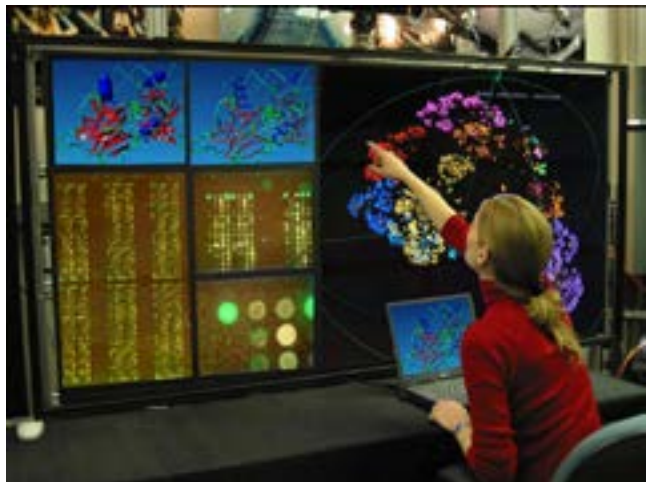
Terwijl ik deze constatering maak, bedenk ik me dat deze redactioneel al veel te veel op een column begint te lijken. Goed dus bovenstaande vragen door te schuiven naar een redactievergadering en te beginnen over iets wat deze I/O Vivat al wel is veranderd. We hebben namelijk de inhoud verdeeld onder een aantal categorieën en op alle pagina's zogenaamde 'tabs' toegevoegd zodat je altijd weet onder welke categorie het artikel valt.

Ook nieuw is het 28<sup>e</sup> bestuur wat inmiddels al een tijdje aan de bak is. Daarom onder de categorie Inter-Actief een 'middenwoord' van Sjoerd, de nieuwe voorzitter, en een aantal pagina's waarin het nieuwe bestuur zich voorstelt. De redactie wenst ze veel succes en vooral veel bestuur(s)plezier!

Om deze redactioneel verder niet al te lang te laten worden wil ik je als lezer graag doorverwijzen naar de inhoudsopgave op pagina 2. Want er is simpelweg niet genoeg ruimte om alle artikelen aan te stippen en ik wil geen schrijvers tegen de schenen schoppen na hun harde werk. ■

Alexander Spannenburg, voorzitter redactie I/O Vivat

# Inhoud



Human computing in the Life Sciences (pag. 25)



Levenskunst & e-life (pag. 34)

“Nou is Inter-Actief natuurlijk boven alles een vereniging, waarmee de leden het belangrijkste voor je zijn, zoals ook statutair vastgelegd is.”

3

“Uw schrijver en dit onderwerp (INF) onderhouden nu al vierentwintig-en-een-half jaar een relatie.”

9

“Door de toenemende drukte groeit de behoefte aan proactief en gebiedsomvattend verkeersmanagement.”

42

## Inter-Actief:

- 3 Middenwoord
- 4 Bestuur 28
- 7 Kryptos 2006

## Afdeling INF:

- 9 Wat zoekt Van der Hoeven...
- 13 Informatica aan de THT
- 17 Applying AOP in Industry
- 25 Human Computing in the Life Sciences

## ENIAC:

- 29 Van de voorzitter
- 30 Alumnidag 2006

## Extern:

- 32 Op bezoek bij... Océ
- 34 Levenskunst & e-life
- 38 Web 2.0 voor dummies
- 42 Nieuwe coalitie regelt ochtendspits



Op het moment dat dit tijdschrift van de persen rolt, ben ik alweer ongeveer twee maanden voorzitter van het 28e bestuur van *Inter-Actief*. Hoewel het schrijven van een beleidsplan daarmee ook alweer een hele tijd achter ons ligt, zit ik nu soms toch weer te denken over ons beleid. Dit is natuurlijk juist de tijd van het uitrollen van het beleid: het implementeren van nieuwe ideeën en het kijken in hoeverre die ideeën de toets met de werkelijkheid ook daadwerkelijk kunnen doorstaan.

Terwijl ik dit schrijf, is de verkiezingskoorts pas net weggeëbd en wordt in de achterkamertjes in Den Haag druk vergaderd over wie nou met wie gaat regeren. De afgelopen weken en dagen wordt, zo lijkt het, constant in alle media gedebatteerd over liberalen en socialen, progressief versus conservatief, links, rechts en elke andere richting. Het zet je toch aan het denken. Je bent nu een beleidsmaker, maar welke koers wordt de vereniging opgestuurd door de ideeën die je hebt. En als je daar toch over nadenkt, als de UT Nederland was, welke partij is *Inter-Actief* dan?

Nou is *Inter-Actief* natuurlijk boven alles een vereniging, waarmee de leden het belangrijkste voor je zijn, zoals ook statutair vastge-

legd is. Politiek vertaald: ons beleid is sociaal en er wordt geïnvesteerd in de leden. Met de nieuwe ideeën die we hebben zou je het beleid progressief kunnen noemen, hoewel we niet alle schepen achter ons willen verbranden. Lastiger is het om te benoemen of, en zo ja welke, liberale aspecten voorkomen in de ideeën die we hebben opgesteld, juist omdat *Inter-Actief* zich zo druk maakt voor haar leden. De koers die wordt ingezet is dus lastig te benoemen, ook als wordt gekeken naar oude beleidsplannen. Is dit beleid rechtser dan dat van het 27e bestuur, of juist linkser, en hoe zit dat met dat van het bestuur daar weer voor?

Alles bij elkaar is het bijzonder dat in een bestuur, waarin de politieke voorkeuren zo ongeveer het hele spectrum aan partijen beslaan, toch één beleid kan worden gevormd waar iedereen het over eens kan worden. Misschien dat de vergelijking tussen de campus en Den Haag daarmee ook gelijk de deur uit kan, tenslotte, een kabinet met D66, SP, PvdA, CDA en VVD klinkt op het moment van schrijven nog niet als een optie. Gelukkig maar dat *Inter-Actief* geen politieke organisatie is. Concluderend is het beleid dat wordt gevoerd niet zozeer links of rechts, maar hopelijk vooral vooruit. ■



## Sjoerd van der Spoel

Voorzitter I.C.T.S.V. Inter-Actief

Op 23 oktober 2006 heeft het 27e bestuur het stokje c.q. de voorzittershamer overgedragen aan het 28<sup>e</sup> bestuur der I.C.T.S.V. Inter-Actief. Bij dezen stellen zij zich voor.



## Bestuur 28

### Het bestuur stelt zich voor

#### Sjoerd van der Spoel voorzitter

We schrijven de nacht van zes op zeven november, het jaar is 1986. Het is een gure herfst dag, zoals mij wordt verteld, als ik diep in de nacht geboren wordt in het ziekenhuis te Almelo. Slechts vier jaar later begeef ik me voor het eerst in de wereld van het onderwijs, een wereld waar ik nou dus al zestien jaar inzit en hopelijk binnen een paar jaar uit ben. Ik begin mijn onderwijs carrière op een basisschool, niet ver van mijn ouderlijk huis, waar ik uiteindelijk maar zeven jaar op heb gezeten dankzij het overslaan van groep zes. Op sportief gebied houdt ik me op de basisschool bezig met atletiek, een sport waar de rest van mijn gezin zich ook actief mee bezighoudt.

Ik ben blijven atletieken bij de atletiekclub voor Almelo en omstreken tot ik richting middelbare school ga. Zes jaar heb ik daar doorgebracht, op de afdeling VWO van het Noordik te Almelo, een school waar ik veel leuke dingen heb meegemaakt, en hier en daar soms ook wat mindere, zoals dat gaat in het leven van

de middelbare scholier. Het is tevens de school waar mijn vader nog altijd werkt, maar grappig genoeg ook de school waar ik voor het eerst kennis mag maken met Johan Noltes, de altijd even productieve secretaris én onderwijsfunctionaris van mijn bestuur.



Aan het eind van mijn zesjarige loopbaan op de school in Almelo, richt ik mijn focus op een vervolgstudie. Ik heb daarbij vooral getwijfeld tussen Industrieel Ontwerpen en Elektrotechniek, maar ben uiteindelijk door puur toeval (ik ging met een vriend om de tijd te doden bij BIT kijken) toch bij BIT terechtgekomen, een studie die bevalt.

In het tweede jaar van mijn studie, nadat ik mijn propedeuse heb afgerond, begin ik te snuffelen aan activisme. Ik ben actief geworden bij Euros (de zeilvereniging), waar ik mijn zomers doorbreng met het zeilen op een platbodem van 25 meter, en mijn winters met het opknappen van diezelfde boot, de Ebenhaëzer. Niet lang nadat voor het eerst heb geproefd aan het verenigingsleven bij Euros ben ik gaan kijken bij In-

ter-Actief, en wat ik daar kan doen. In eerste instantie ben ik bij de aXi gaan kijken, en letterlijk een kwartier nadat ik besloten had die commissie in te gaan zat ik in de symposiumcommissie. Daarmee was vorig jaar een van de leukste en actiefste jaren uit m'n leven, hoewel de studie soms daar wel wat onder te lijden had.

Na het symposium, waar ik ondanks de lage opkomst erg van genoten heb, viel ik enigszins in het zwarte gat dat veel mensen meemaken nadat ze uit een grote commissie of het bestuur zijn. Ik wou erg graag meer doen met Inter-Actief, maar de gedachte aan een heel jaar bestuur, en daarmee bijna geen studiepunten, zag ik niet echt als een optie. Uiteindelijk heb ik voor mijzelf toch de knoop doorgehakt en gekozen voor een jaar vol uitdagingen, waar ik al dan niet (vooral niet) in studiepunten uitgedrukt veel ervaring uit zal halen, maar bovenal natuurlijk veel plezier. Het komende jaar zal ik me als voorzitter bezighouden met onder meer huisvesting, contact met andere verenigingen en instellingen op de campus, plus natuurlijk het bewaken van de grote lijn die is uitgestippeld in ons beleidsplan.

## Johan Noltes secretaris/functionaris derwijs

21 jaar geleden werd ik geboren in Wierden, de plaats waar ik nog steeds woon en dus ook de basisschool zou doorlopen. Op die basisschool ontstond mijn 'liefde' voor computers, programmeren en ontwerpen eigenlijk al.

Dat kreeg meer vorm



voor onze vereniging. Hoewel ik me had voorgenomen het pas volgend jaar serieus te overwegen, begon

het steeds aantrekkelijker te worden dit jaar al bestuur te gaan doen. Vlak voor de zomervakantie hakte ik de knoop door en besloot ervoor te gaan. Tijdens de intro leerde ik mijn mede-bestuursgenoten al snel kennen, voor zover dat nog niet het geval was. Het lag voor de

volgen, waarmee het gezin Perloy compleet was.

Op sportief gebied had ik natuurlijk ook ambities en al op vierjarige leeftijd begon ik aan een zwemcarrière die pas dertien jaar later zou stranden. Helaas staat de prijzenkast op enkele medailles van clubkampioenschappen na, stof te happen.

Op studiegebied was er onderweg ook een belangrijk kruispunt op mijn weg gekomen en ik besloot om de afslag richting het VWO te nemen. Tijdens deze leuke tijd waar ik nog redelijk besluiteloos was over wat ik in de toekomst zou gaan doen, volgde ik de vakken die behoorden bij het profiel Natuur en Gezondheid. Om alle opties open te houden moest daarnaast natuurlijk ook Economie gevolgd worden.

Na zes jaar werd het toch tijd om een keuze te maken en pakte ik in 2003 de bus richting Enschede waar ik een poging ging doen om Informatica te studeren. Omdat de overgang van het kleine Ulft naar de grote stad Enschede natuurlijk erg groot was, besloot ik om mezelf te vestigen in het net wat kleinere Hengelo, een plek waar ik nog steeds te vinden ben.

Na twee-en-een-halfjaar hard studeren besloot ik in mijn derde jaar ook iets actiefs naast mijn studie te gaan doen en nam ik samen met een hele leuke commissie de organisatie voor de opleidingsintro van dit jaar op me. Tijdens het

organiseren bleek het actief zijn toch wel erg te bevallen en uiteindelijk belandde ik in het kandidaatsbestuur van onze vereniging. Na een ietwat lange ALV mocht ik me dan eindelijk officieel penningmeester van *Inter-Actief* noemen. En waar het natuurlijk erg verleidelijk is om met zoveel geld snel richting een warm eiland te vertrekken, heb ik toch besloten om niet alleen aan mezelf te denken,

“Uiteindelijk koos ik toch niet voor het ‘exotische’ Telematica”

toen in naar de middelbare school ging, Het Noordik in Almelo. De profielkeuze Natuur en Techniek was voor mij dus ook snel gemaakt. Ook begon ik me langzamerhand met (de kwaliteit van) het onderwijs te bemoeien en alle zaken daaromheen; het cijferregistratiesysteem is daarvan een voorbeeld. Over mijn vervolgstudie heb ik toch even na moeten denken, uiteindelijk koos ik toch niet voor het ‘exotische’ Telematica, maar voor het ‘gewone’ Informatica.

Tijdens de intro van 2004 kwam ik echter toch in de Telematica-doe-groep TEGEL terecht, waarvan ik het jaar daarna samen met Ivo doegroep papa zou worden. Het eerste jaar heb ik nominaal afgerond, “dus had ik bewezen dat ik het kon, en werd het tijd om actief naast mijn studie te worden”, aldus een IAPC’er op zoek naar een nieuw bestuur. Toevallig werd ik later door Erik gevraagd om lid te worden van de Onderwijs Evaluatie Commissie en daar kon ik natuurlijk geen nee tegen zeggen. Toen ik dat tijdens een borrel aan Tin vertelde, kon hij het natuurlijk niet laten mij de Onderwijscommissie in te praten; mijn eerste stappen binnen *Inter-Actief* waren gezet.

In de loop van het jaar begon de zoektocht naar een nieuw bestuur

hand dat ik functionaris onderwijs zou worden, maar omdat we met z’n vijven zijn, werd ook de functie van secretaris aan mij toebedeeld. Het komende jaar ga ik me dus vooral bezighouden met de kwaliteit van het onderwijs op de faculteit en alle post binnen de vereniging.

## Bert Perloy penningmeester

Geduld is een schone zaak. Helaas wist ik dat 21 jaar geleden nog niet en een maand eerder dan gepland zag ik op 22 april in het jaar 1985 het kunstlicht van de couveuse. Geboren in het plaatsje Doetinchem in de Achterhoek werd ik al vroeg voor het eerst in mijn leven ontslagen en verruilde ik het ziekenhuis voor het onderkomen van mijn ouders en mijn één jaar oudere zus in Ulft.

Hierna vond ik het al snel tijd om mijn eerste stapjes op mijn weg naar een maatschappelijke carrière te zetten en maakte ik kennis met het fenomeen school, nog niet beseffend dat deze kennismaking best lang zou duren. Ondertussen was ik in 1988 een broertje rijker geworden en zou er in 1993 nog een tweede broertje



maar om er dit jaar een te gek jaar van te maken, met en vooral voor al onze leden.

**Erwin Starke**  
functionaris interne betrekkingen



Aan al het goede komt een eind, zo ook aan het bestuursjaar van het oud-bestuur. Tijdens de ALV hebben zij plaatsgemaakt

voor ons vijven. Nu zullen sommige mensen mij nog niet goed genoeg kennen, of soms helemaal niet kennen. Ik ben geboren in de mooie zomer van 1986, op 8 juli om precies te zijn. Dit gebeurde in het ziekenhuis van Winschoten in Oost-Groningen, tot twee-en-een-half jaar geleden nog mijn woonplaats. Daar heb ik nog het VWO gevolgd aan de lokale middelbare school met uiteindelijk het profiel Natuur en Techniek. Tijdens deze periode ben ik begonnen met badminton, waar ik momenteel in de regionale competitie meespeel en bij een groot aantal toernooien ben geweest en zelfs al een toernooi heb georganiseerd. Aan het einde van de middelbare school stond ik voor de keuze van mijn universiteit: Groningen of Twente. Uiteindelijk koos ik voor de gavere universiteit boven dichter bij ouders, en zo zitten we hier alweer in Enschede. Daar werd ik na een half jaartje studeren en activiteiten bezoeken actief bij een commissie door een overtuigende functionaris intern. Na een tijdje als "normaal" actief lid, vond ik dat het tijd werd om de dingen iets groter aan te pakken: ik besloot dat ik het bestuur van onze vereniging in wilde. Een beslissing waar ik tot nu toe nog geen spijt van heb gekregen en hopelijk ook geen spijt van hoeft te krijgen.

**Tom Palsma**  
Functionaris externe betrekkingen

(bestuurscontactpersoon I/O Vivat)

Op een mooie herfstdag in 1985 zag ik het levenslicht in Meppel. Vanaf mijn vierde jaar ging ik naar de basisschool waar ik acht jaar lang altijd de



jongste van de klas bleef. Na zes jaar braaf naar het VWO geweest te zijn, werd het tijd om naar de universiteit te gaan. Door een bezoek aan de voorlichtingsdagen van de Universiteit Twente werd het mij duidelijk dat ik Informatica wilde studeren in Enschede. Tijdens de zomer van 2003 verliet ik mijn ouders en zusje om te verhuizen naar de campus waar ik nog steeds woon. Inmiddels is dat ruim drie jaar geleden en op het moment dat je 20 jaar bent, nog steeds bij de jongste mensen van je jaar hoort en ook nog op je 22<sup>e</sup> klaar kan zijn met je studie, besef je dat je iets fout doet. Het organiseren van de

dig jaar van maken samen met de rest van het bestuur. ■

"Aan al het goede komt een eind"

opleidingsintroductie was erg leuk, maar het leverde geen noemenswaardige studievertraging op.

Om toch nog wat langer te kunnen genieten van het studentenleven en een jaar minder te kunnen werken tot de pensioengerechtigde leeftijd ging ik op zoek naar iets anders dan alleen maar studeren. Het bestuur van *Inter-Actief* is een nuttige invulling waar ik beide doelen mee kan bereiken. Naast deze persoonlijke doelen is het voor mij als functionaris externe betrekkingen natuurlijk belangrijk dat de contacten tussen *Inter-Actief* en het bedrijfsleven goed blijven. Verder wil ik er natuurlijk een gewel-

In de herfst van 2006 zijn we met dertig studenten en twee docenten van de afdeling Informatica naar de Verenigde Staten vertrokken voor een studiereis, een studiereis met als onderzoeksonderwerp IT Security en met de naam Kryptos.

## Kryptos 2006

Tijdens deze reis hebben we spraakmakende bedrijven en universiteiten bezocht in Boston, New York, Washington D.C., San Francisco en Silicon Valley. Gedurende deze bezoeken probeerden we antwoord te krijgen op de vraag: *Welke IT security trends en technologieën in de Verenigde Staten van Amerika kunnen worden onderscheiden en wat zijn de gevolgen voor organisaties in zowel de Verenigde Staten van Amerika als Nederland?* Het voorbereidend onderzoek en de resultaten van het onderzoek tijdens de studiereis zijn te vinden in respectievelijk het voorverslag en het naverslag. Deze zijn onder andere te vinden in de boekenkast van de Inter-Actief-kamer.

Niet alleen hebben we tijdens onze reis bezoeken gebracht aan verschillende organisaties als Sun Microsystems, Deloitte, Philips Medical Systems, Verisign, Symantec, National Institute of Standards and Technology, Boston University, Harvard University, Stanford University, Massachusetts Institute of Technology, maar ook hebben we veel gezien van de Amerikaanse cultuur en omgeving. Zo kregen we in Boston twee keer de stad te zien op zowel cultuur als historisch gebied: er werd ons, in de vorm van een ducktour en een wandeling door de stad, heel wat feitjes bijgebracht. In Washington deden de stadsrondleiding op de fiets en za-

gen we onder andere het Capitool en het Washington Monument.

Ook op sportief gebied hebben we ons vermaakt. Zo hebben we in het stadion van de Red Socks een baseballwedstrijd bijgewoond en bezochten we in San Francisco een volleybalwedstrijd op Stanford University, compleet met cheerleaders en volkslied vooraf.

Het natuurschoon heeft ook nog onze aandacht gekregen tijdens een weekend in Yosemite National Park. Hier sliepen we in tenten en hebben we door de prachtige bergen gewandeld.

Als aanvulling op de studiereis heeft de Kryptos studiereiscommissie een symposium georganiseerd met hetzelfde onderwerp als de studiereis, IT Security. Dit symposium was voor de studenten die deelnamen aan de studiereis een goede voorbereiding. Door de vele contacten die we op hadden gedaan tijdens de organisatie van de studiereis hadden we binnen een paar weken het indrukwekkende aantal van twaalf sprekers en workshops van zowel universiteiten als bedrijven. Mede door de hoeveelheid sprekers en door de grote hoeveelheid bezoekers was dit symposium een groot succes.



## IT Security in de USA

Met zijn allen kijken we terug op een geweldige ervaring die we door deze studiereis hebben opgedaan. ■

**De Kryptos studiereiscommissie:**  
Kimberly Lemmens – Voorzitter  
Stephan Roolvink – Penningmeester  
Eelco Eerenberg – Reiscoördinator  
Koen Voermans – Reiscoördinator  
Ivar Pruijn – Onderzoek en Contract Research  
Alexander Spannenburg – Onderzoek en Contract Research

*Voor een kleine foto-impressie zie ommezijde*



Van boven naar beneden: Chinatown; de deelnemers in Yosemite National Park; typische cab en bus; het Vrijheidsbeeld





# Wat zoekt Van der Hoeven ...



## INF!

In 2006 heeft de Universiteit Twente een historische mijlpaal bereikt. Zij had 25 jaar het recht diploma's uit te reiken in de informatica en aanverwante gebieden. Iets om bij stil te staan.

Uw schrijver en dit onderwerp (INF) onderhouden nu al vierentwintig-en-een-half jaar een relatie. Ik wil de roerige tijden die zij samen hebben meegemaakt, en meemaken, belichten, waarbij vele oude bekenden de revue kunnen passeren. Ik neem mij voor iets te zeggen over instroom, doorstroom en uitstroom, over OTV, OWI, OEC, WIP en HIC, over Inter-Actief en IAPC, over WUB en MUB, over geld en (re)organisatie, over EL en TW, over BICA en BIT en over TEL. De historische degelijkheid van mijn betoog zal mogelijk niet 100% zijn. Volledigheid streef ik niet na. Het vakgebied informatica zou er ook wat bekaaid van af kunnen komen. De zeden en gewoonten van de gemeenschap van mensen die zich in Twente om dat vakgebied hebben bekommerd en bekommeren, zowel docenten als studenten, staan centraal.

Is INF de serieuze titel voor deze zedenschets? Ja. Mijn verhaal gaat namelijk over de selecte groep mensen voor wie die drie letters iets betekenen. INF. Het zegt niemand in Nederland iets. Maar ons wel.

Alle onderwerpen die ik opsom, en de vele mogelijk cryptische afkortingen die ik daarvoor nodig heb, zouden kunnen suggereren

dat mijn betoog diepgravend wordt. Dat is niet helemaal de bedoeling. Maar samenhangend zou het wel moeten zijn. Ik heb gezocht naar een kapstok voor mijn zedenschets. Hoe vang ik de 25 jaren der Informatica in Twente in één woord? Ik heb gedacht dat dat woord wel "liefde" moest zijn. En ik ben zo vrij geweest die liefde te verbeelden in een variatie op een bekend beeldmerk. Maar vertel dat liever niet aan de huisstijlpolitie.

## INF loves studenten

We moeten bij het primaire proces beginnen, en dan staat de relatie tussen INF en studenten natuurlijk bovenaan de agenda. Maar hoezeer houdt INF van zijn studenten? Laten we eerst het thema omdraaien, eerst stellen



## Studenten love INF

Dat wisselt nogal. Als we kijken naar de instroom bedoel ik. (En INF staat hier niet voor één opleiding, maar voor alles waarvoor INF penvoerder is). Maar welke opleiding had al een studievereniging voordat de opleiding zelf er was? INF! En als studenten hier eenmaal zijn, willen zij hier ook blijven. Lang blijven. Soms heel lang blijven. Dat moet wel echte liefde zijn. En als zij eenmaal zijn vertrokken, wat velen tenslotte succesvol doen, dan denken zij met liefde aan hun tijd in Twente terug. Studenten love INF.

Dus nogmaals:

## INF loves studenten

Ja, maar het was vroeger heftiger dan nu. Meer borrels waar do-

# Gerrit van der Hoeven

## Opleidingsdirecteur

centen en studenten elkaar troffen. Meer docentmentoren. Aan de andere kant, INF is onverminderd trots op zijn *Inter-Actief*, op zijn IAPC, en op zijn OEC, die ook al heel wat jaren meegaan. En natuurlijk, ook op de prestaties van vele studenten. De verliefdheid van het eerste uur is misschien vervaagd, maar INF loves studenten.

### INF loves Microsoft

Studenten zijn core business, maar ICT is dat evenzeer. Een logische vraag dus, of INF van Microsoft houdt. Ik dacht het niet.

### INF loves apparatuurbegroting

Microsoft heeft het moeilijk in deze gemeenschap, maar superieure ICT-voorzieningen zijn wel ons aller (jongens)droom. Hoewel, ook daar lijkt de verliefdheid van de vroege jaren wat geblust. Hoe hebben we jaar in jaar uit met elkaar genoten van het gevecht om investeringen in onze infrastructuur. Soms leek het dat de waardering voor opvallende onderzoekprestaties op één plek terug te vinden was: in de apparatuurbegroting. Maar sinds ons onderzoek groeit als kool, is die apparatuurbegroting bijzaak. INF loved apparatuurbegroting.

### INF loves buiten spelen

We waren een gemeenschap die stamde uit de tradities van de technische hogeschool. Toepassingsgericht en met hechte banden met het bedrijfsleven. Contacten met de buitenwereld dus. Maar waren we dol op buiten spelen? Zulke kinderen waren we misschien niet. We zaten veel binnen en keken naar onszelf en naar elkaar. Misschien hebben we ons zelf en anderen daarmee tekort gedaan. Had de UT niet veel meer moeten profiteren van wat wij van ICT wisten (en weten), en hadden onze investeringsplannen in infrastructuur niet de UT-plannen moeten zijn?



### INF loves de burens

Zoals het was met buiten spelen, zo was het met de burens. Burens, dan bedoel ik natuurlijk EL en TW. We hebben afstand van hen genomen. Niet zo gek, als de burens ook je ouders zijn. Zo gaat het als je opgroeit. Een radicale breuk is het niet geworden. We blijven de verwantschap met wiskunde herkennen. Op het oog is de afstand tot Elektrotechniek groter, er was soms zelfs echte wrijving. Maar met deze buurman delen we twee leerstoelen, dat is een inniger band dan met TW, dat duidt toch op enige liefde.

### De burens love INF

Zijn wij geliefd bij onze burens? Dat moeten we de burens vragen. Onze ouders lijken nog altijd met enige zorg te kijken naar wat wij doen en

is misschien niet het goede woord. Het was een zinvol ritueel, zolang het de helderheid en openbaarheid van de taakverdeling en de taakbelasting diende. Zo bedoelde hij het ook. Maar het werd een beloningsinstrument. Je uren in de OTV deden recht aan je positie, het was de moeite waard te strijden voor uren in dat overzicht. Daarmee sleet de waarde van het ritueel. We doen er niet meer aan. INF loved OTV. Wat niet wegneemt dat die OTV één waarheid zichtbaar maakte, die onveranderd waar is gebleven: wij doen heel veel dingen in heel beperkte tijd.

### INF loves FVM

We hebben sinds een paar jaar de OTV vervangen door het FVM. Het financieel verdeelmodel. Een verdeelsleutel voor middelen over leerstoelen. Niet meer dan dat. Maakt één ding duidelijk (en dat is jaren zo), en dat is dat wij heel veel dingen doen voor weinig geld. We zijn intussen druk bezig in het FVM allerlei kwaliteitsoordelen te lezen, die er niet in staan. Dat moesten we eigenlijk niet doen. Daar werd de OTV ook alleen maar slechter van.

“Of mijn punt stond zo los van zijn werkelijkheid dat hij echt niet kon geloven wat hij hoorde”

willen. Maar tegelijk is de indruk dat zij vrede hebben met wat er van ons geworden is. En trouwens, wie praat er over burens? We wonen weer in één huis.

### INF loves OTV

OTV betekent onderwijstaakverdeling. Een gedetailleerd overzicht van alle taken in het onderwijs. Met daarbij de uren die nodig waren om elk van die taken (althans volgens ons model) uit te voeren. En natuurlijk en vooral ook de inbreng die elk van de docenten had bij het uitvoeren van die taken. Het vaststellen van dat overzicht was een jaarlijks ritueel. Ik sprak een ex-collega, die dat een “gezellig” ritueel noemde. Gezellig

### INF loves programmeren

Jaren geleden, toen ik nog niet zo lang de taken van Karel de Jonge had overgenomen, was ik gastheer van een IFIP bijeenkomst over informatica-onderwijs. Ik stelde daar de vraag aan de orde of programmeer-onderwijs voor het vakgebied nu werkelijk zo belangrijk was. Een Japanse bezoeker op de eerste rij vroeg mij beleefd dat nog eens te herhalen. En nog eens. Ik sprak slecht Engels. Of ik sprak goed Engels maar hij kon het slecht verstaan. Of mijn punt stond zo los van zijn werkelijkheid dat hij echt niet kon geloven wat hij hoorde. Dat laatste dus. De hele dag heeft hij mij achtervolgd om mij bij te staan in mijn droevig lot. Dat ik



kon werken op een plek waar informatici zich daadwerkelijk afvroegen of programmeren voor een studie een essentiële vaardigheid was.

Gek genoeg stellen wij onszelf die vraag. En zijn wij de eersten om het belang van programmeervaardigheden voor het ware informatica-inzicht te relativieren. Maar tegelijkertijd is programmeren echt onze gemeenschappelijke taal. Wij hebben er altijd veel in geïnvesteerd om onze studenten die taal te laten spreken. Tegendraads en eigenzinnig. Algol-

de universiteit is de dag waarop we de stichting van de universiteit herdenken, de verjaardag dus) waarin hij betoogde dat de grote kracht van Twente was, dat de universiteit bedrijfskunde en informatica kon combineren tot een vakgebied met een grote toekomst. Het heeft even geduurd, maar het is ervan gekomen. BIT! Houdt Informatica van BIT? Informatica mag graag de wetenschappelijke diepgang van BIT wat relativieren. Informatica wil zich sowieso wel afzetten tegen mensen die

(tegenwoordig Zilverling). Het hield ons bezig. Twee kleine versjes in mijn archief getuigen van die tijden.

(Eerst een fragment uit een smeekbede aan Sinterklaas, toen de helft van INF in het TO-gebouw woonde, en de andere helft nog in het TW/RC-gebouw)

Ongetwijfeld hebt u de verhalen over dat onherbergzame oord, gebouwd in een vijver op palen, ook wel eens gehoord.

Je kan er van hier uit gemakkelijk komen;  
over de vijver loopt een brug.  
Maar wie er geweest is blijft er van dromen  
en wil er nooit ofte nimmer naar terug

(Dan een Ollekebolleke, dichtvorm in het verleden met verve beoefend door Drs P. en consorten, ter gelegenheid van het betrekken van de Zilverling, waarvan het uiterlijk niet iedereen aansprak)

#### NIEUWBOUW INFORMATICA

Broeinest van wetenschap!  
J. van de Lagemaat  
Kijkt uit zijn venster  
bedrukt naar benee.

Geeft deze Graubau het  
CTIT-gevoel?  
Als je maar binnen blijft,  
valt het wel mee!

(Jeroen van de Lagemaat was destijds zakelijk directeur van het CTIT)

#### INF loves organisatieverandering

In de 25 jaren INF is veel veranderd in de organisatie. Externe omstandigheden zijn anders dan ze toen waren. De wet bijvoorbeeld zegt nu iets anders over de universitaire organisatie dan toen (Toen hadden we WUB, maar we kregen MUB, modernisering universitair bestuur). Maar ook los van externe invloeden grepen wij in in de organisatie. Tenslotte zelfs

### “BIT is een succes waar INF collectief trots op moet zijn”

68 als eerste programmeertaal, functioneel programmeren als eerste kennismaking, Java als voertuig voor het object-georiënteerde denken. Wij liepen voorop en boden kwaliteit. INF loves programmeren!

#### INF loves curriculumherziening

De essentie van dat eerste curriculum, met zijn keten van programmeren 1 tot en met programmeren 6 heeft lang stand gehouden. Pas na meer dan 10 jaar ging het op de schop, toen uit de BICA-variant al BIT aan het ontstaan was, en TEL aan de poort rammelde. Het was een moeizaam proces, die eerste herziening. Maar de veranderingen zijn elkaar in steeds sneller tempo op gaan volgen, naarmate wij ouder werden. Major-minor, bachelor master, semesters en kwartielen, wij kunnen het allemaal aan. En let wel: met behoud van kwaliteit. Met verbetering van kwaliteit, waar we de kans hadden. Maar INF loves curriculumherziening? Hm. Te masochistisch.

#### INF loves BIT

Bij de relatie met de burens heb ik één buur niet genoemd. Bedrijfskunde. Al in de zeventiger jaren hield de grondlegger van de zelfstandige informatica in Twente, professor Duijvestijn een Diesrede (De Dies van

met hun voeten in de modder staan van de alledaagse werkelijkheid van personen en organisaties, waarin wel eens wat geneuzeld wordt dat zich slecht vanuit wiskundig en natuurwetenschappelijk perspectief laat begrijpen. Niet verstandig. BIT is een succes waar INF collectief trots op moet zijn.

#### BIT loves INF

Nou vooruit, misschien houdt BIT een beetje van INF. Maar de groeistuipe waren hevig. De eerste BITOC-vergaderingen die ik mocht bijwonen (alweer 10 jaar geleden) waren heftige aanklachten tegen het gebrek aan aandacht dat INF voor BIT had. Het was stilzitten en afwachten hoe je geschoren werd. Maar zo gaat dat, we zullen het een generatiekloof noemen.

#### INF loves verhuizen

Een heel ander thema. In een samenvatting van wat tekenend is voor INF zou ik het haast vergeten, omdat het niet zo belangrijk meer lijkt. Maar als ik kijk in de mappen met dingetjes die ik bewaard heb, speelt het een hoofdrol. Verhuizen. In de tijd die ik heb meegemaakt gingen we van TW/RC (tegenwoordig Cubicus) naar het TO-gebouw (tegenwoordig Ravelijn) naar het KCT-gebouw (tegenwoordig Citadel) naar het informaticagebouw

zover dat de faculteit INF opging in de faculteit EWI. We hebben ons erg druk gemaakt over dat veranderen. Of het veel zin had, er zijn weinigen die daar van overtuigd zijn geraakt. Of het ons pijn heeft gedaan? We herinneren ons veel tumult, maar we zijn niet beschadigd. Misschien ook wel omdat BOZ en laboratorium steeds fier overeind zijn gebleven. En dat zijn natuurlijk de steunpilaren waarop wij allen rusten, hoe onze organisatie ook in elkaar steekt.

#### INF loves UT

In de UT-gemeenschap is INF altijd een buitenbeentje geweest. Het braafste jongetje van de klas, dat zijn eigen gang ging. Houdt INF van de UT? Ja, maar ze moeten het niet te gek maken.

#### UT loves INF

De gevoelens zijn geheel wederkerig, zo lijkt het.

#### INF loves TEL

Nu wordt het lastig. De relatie tussen INF en TEL is nog veel ingewikkelder dan die tussen INF en BIT. Eén ding moet worden gezegd. Toen INF er niet aan toe was om buiten te spelen, was TEL daar al volop mee bezig. Met plezier en succes. En dat verdient alle lof.

#### TEL loves INF

O ja?

#### INF loves vrouwen

Ja.

#### Vrouwen love INF

Ach. De gevoelens zijn niet wederzijds. Wat moet ik daar nu nog van zeggen?

#### INF loves INF

INF heeft alle redenen om van INF te houden. Maar als je altijd binnen zit, raak je met jezelf in de knoop. Zoals dat gebeurt met pubers. Maar we zijn de puberteit voorbij. We zijn 25 jaar, tijd voor volwassen trots. INF loves INF!

(En begrijp mij goed, die drie letters staan voor een gemeenschap van mensen, ook HMI en BIT en TEL en ondersteuning) ■

*Deze rede is gehouden door dr. G.F. van der Hoeven tijdens de lustrumviering van de oude faculteit Informatica en de nieuwe faculteit EWI, afdeling INF.*

“In de UT-gemeenschap is INF altijd een buitenbeentje geweest”



Ter gelegenheid van het vijftienvijftigjarig jubileum van de opleiding Informatica is het boek Informatica aan de THT verschenen. Ik ben er vier jaar geleden aan begonnen en het was klaar voor de drukker toen ik op 1 december 2005, na veertig 'dienstjaren' de UT verliet. Het boek beschrijft de geschiedenis vanaf ongeveer 1964 - de oprichting van de Technische Hogeschool Twente (THT) - tot 2002 toen de inmiddels in 1981 opge-

richte faculteit Informatica ging fuseren met Elektrotechniek en Toegepaste Wiskunde tot EWI.



## Informatica aan de THT

### geschiedenis van een faculteit

Voor de eerste periode, van 1964 tot 1981 komt uitgebreid aan bod. Dat komt omdat ik die tijd zelf intens beleefd heb. Ik ging op 1 oktober 1965 als 21-jarige ambtenaar-student werken en studeren bij de afdeling Elektrotechniek, in de leerstoel Numerieke wiskunde en Programmeratuur van Arie Duijvestijn.

Op dat moment was er op de hele THT nog geen enkele computer. Er was er al wel een besteld bij IBM, de IBM-360/30, maar die kwam pas begin 1966. Elders waren er al wel enkele computers in gebruik, onder andere bij Philips Computer Industrie (PCI) in Apeldoorn, waar Duijvestijn werkte tot zijn benoeming aan de THT. Duijvestijn regelde een vaste telefoonverbinding tussen de THT en Philips en zo konden we al beginnen met oefenen in het maken en draaien van programma's, in afwachting van de grote IBM-computer. Bovendien moesten we de studenten die straks programmeerpractica moesten doen een stapje voorblijven!

De programmeertaal Algol-60 kon op de computer bij Philips verwerkt worden. We typten ons Algol-programma op een schrijfmachine die tevens ponsband kon ponsen (je moest dan wel foutloos typen, want geponst is geponst). De ponsband ging daarna in een lezer die de gecodeerde bits over de telefoonlijn naar Apeldoorn stuurde. Daar werd de code weer geponst, de band in de computer gestopt, de uitvoer uit de computer weer geponst en terug naar de THT gestuurd. Op de THT werd het dan weer geponst en vervolgens door de schrijfmachine gelezen en uitgeprint. Dan pas zag je wat je fout gedaan had en kon je de hele cyclus opnieuw beleven.

#### Nulpunten van een derdegraads polynoom

Voor iedereen was programmeren een volkomen nieuwe bezigheid. We hadden geen idee hoe dat netjes moest. De strategie was een flowchart tekenen en die met al zijn sprongen en vertakkingen letterlijk

in Algol-code omzetten, net zolang tot het leek te werken.

Om wat te oefenen heb ik eind 1965 geprobeerd een Algolprogramma te schrijven voor het berekenen van de nulpunten (ook wel wortels genoemd) van een derdegraads polynoom. Het programma is gebaseerd op het steeds halveren van een interval waarop de functie een tegengesteld teken heeft. Het fraaie staaltje van mijn eerste programmeerkunst is hieronder te bewonderen. Let in het bijzonder op de zinvolle naamgeving en de vele goto-sprongen.

```
begin comment oplossen derdegraads vergelijking d.m.v. halveren,vervoort;
integer n,p,i;
real e,a,b,c,d,x,y,r,s,t,u,k,l;
array T[1:16];
format read1=<I2>;
format read2=<F5.5,B3F5.5, B3F5.5, B3F5.5//>;
format write2:<T13//>;
format write3:<E0.9//>;
real procedure f(x,a,b,c); f:=x*x*x+a*x*x+b*x+c;
in(4,read1,n);
vier: in(4,read2,e,a,b,c);
uit(read2,e,a,b,c);
a:=a/e;b:=b/e;c:=c/e;e:=-a/3;d:=a*a-3*b;
for i:=9 step 1 until 16 do T[i]:=10^(i-9);
for i:=1 step 1 until 8 do T[i]:=10^(8-i);
if d=0 then begin r:=e;t:=f(r,a,b,c);
if t<0 then goto een
else begin s:=r;goto twee end
```

```

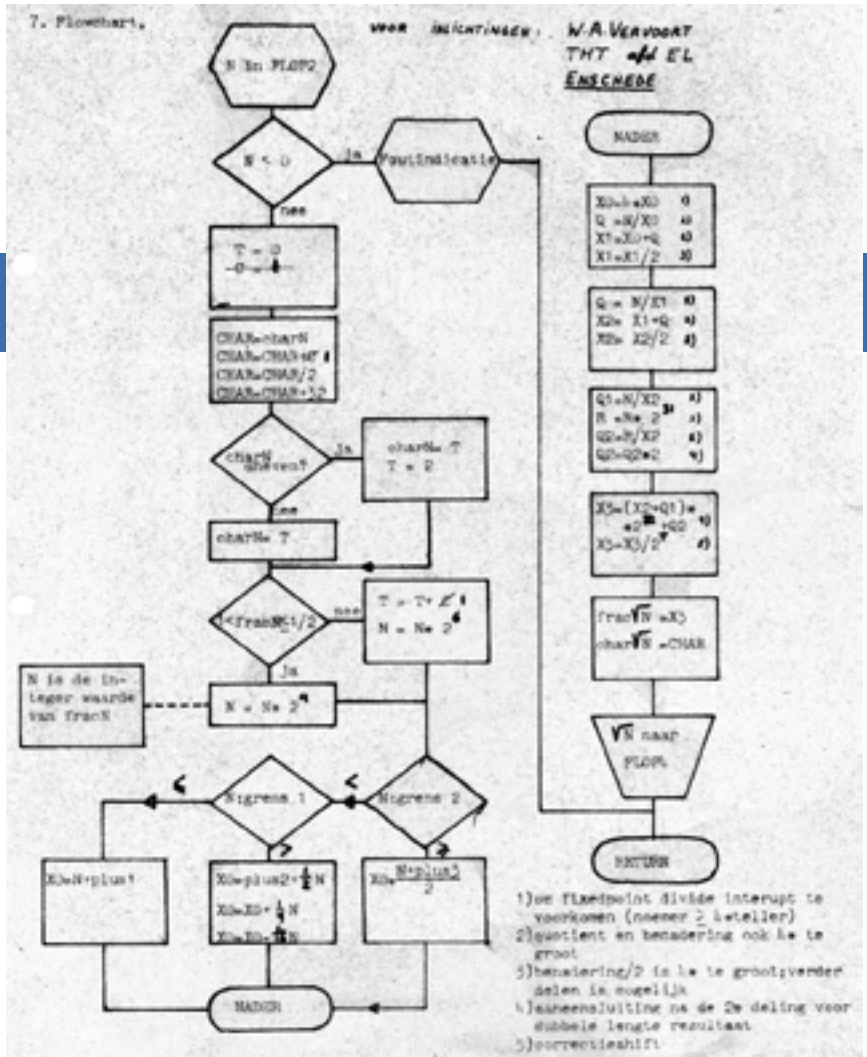
end
else begin d:=sqrt(d)/3;k:=r:=e+d;l:=s:=e-d;t:=f(r,a,b,c);d:=f(s,a,b,c);
if sign(t)?sign(d) then begin p:=1;uit(write2,<drie wortels>);goto een end
else if t>0 then goto twee
else goto een
end;
end;
een: i:=0;
terugl:i:=i+1;
if i>16 then begin uit(write2,<plus oneindig>);goto vijf
end
else begin x:=T[i];if x=r then begin y:=r;goto terugl end
else if i=1 then begin uit (write2,<min oneindig>);goto vijf
end
else if f(x,a,b,c)=0 then begin y:=x;goto terugl end
else goto verder
end;
drie: p:=2;
twee: i:=17;
terug2:i:=i-1;
if i<1 then begin uit(write2,<min oneindig>);goto vijf
end
else begin x:=T[i];if x=s then begin y:=s;goto terug2 end
else if i=16 then begin uit (write2,<plus oneindig>);goto vijf
end
else if f(x,a,b,c)=0 then begin y:=x; goto terug2 end
else begin e:=x;x:=y;y:=e; goto verder end
end;
verder: if abs(x) < 1 then t:= 1 else t:=x;
if abs((x-y)/t)=10-9then begin uit (write3,(x+y)*0.5);
if p=2 then begin y:=r;x:=s;p:=0;goto verder end else goto vijf
end
else begin u:=(y+x)*0.5;
if f(u,a,b,c)>0 then begin x:=u;goto verder end else begin y:=u;goto verder end
end;
vijf: if p=1 then goto drie
else begin n:=n-1;if n=0 then goto einde
else goto vier
end;
einde:
end;

```

### Assembler voor de IBM-360/30

Toen in februari 1966 onze eerste mainframecomputer in Hal-B werd geïnstalleerd, ongeveer op de plek van de Inter-Actief-kamer, stapten we niet alleen over van ponsband naar ponskaarten, maar ook van Algol naar de assembleertaal voor de IBM-360/30. Om daar netjes in te programmeren was helemaal een verschrikking. Ik kreeg van Duijvestijn een klusje om een assembleroutine te schrijven voor het berekenen van de wortel uit een getal. Die instructie zat niet op de computer die voor 250.000 gulden per jaar werd gehuurd.

Ik heb hieronder een beschrijving en een plaatje van de flowchart voor dat programma toegevoegd en de listing (op pagina 16) van een voorlaatste versie van de assemblercode, voor de puzzelaars onder de lezers. IBM had inmiddels ook een routine voor de vierkantswortel geschreven, maar die van mij deed het in 4,5 msec terwijl IMB er 8,6 msec voor nodig had. Beating Big Blue!



## SQRT - routine

## 1. Werkgebied

Het getal wordt zo getransformeerd, dat voor zijn fractie  $F$  geldt:  $1 < F = 1/4$ , omdat voor  $1/4 F = 1/16$  de fout in de Newtonbenaderingsmethode minder dan kwadratisch blijkt af te nemen.

## 2. Benaderingsintervallen

Het gebied is in drie intervallen verdeeld, waarin de wortel door een rechte, met een „binair - eenvoudige” richtingscoëfficiënt wordt benaderd tot een fout =  $2^{-7}$ . De vermenigvuldiging met deze richtingscoëfficiënt geschiedt d.m.v. een shift - instructie. De intervallen met bijbehorende rechten zijn:

interval	rechte
$0,25 = x = 0,390625$	$x + 0,25 - 2^{-7}$
$0,390625 < x < 0,677700$	$(11/16)x + 0,3600409$
$0,677700 = x < 1$	$(1/2)x + 0,5 - 2^{-7}$

## 3. Aantal iteraties

Na drie iteraties is de fout =  $2^{-56}$ , dus voldoende klein voor een long - format floatingpointgetal. Daar na twee iteratie de fout nog  $2^{-28}$  is, kunnen, mits ten minste 28 bits van de fractie in een register liggen, deze twee iteraties in enkele lengte worden uitgevoerd. Met een linkse shift over 4 plaatsen

wordt de denkbeeldige komma geplaatst tussen bit 3 en 4, en komen er 28 bits in het register.

## 4. Gebruikte instructies en totaal tijd

De bewerking geschiedt geheel in de FIXED - point registers 2 t/m 8. Opgave en resultaat staan in resp FLOP 2 en 4. De tijd bedraagt 4,5 msec op een IBM 360/30. De huidige IBM - SRQT routine doet hierover 8,6 msec

## 5. Plaats van de denkbeeldige komma tijdens deling

In de Newton wortelbenadering  $X_{i+1} = 1/2 (X_i + N/X_i)$  worden  $N$  en  $X_i$  als integers behandeld. Bij de deling  $N/x_i$  moeten we ervoor zorgen dat  $X_i = 4 * N$  om te voorkomen dat het quotient doorloopt in zijn tekenbit. Door  $X_i$  twee plaatsen naar links te shiften, en daarna de deling uit te voeren krijgt het quotient zijn denkbeeldige komma tussen bit 1 en 2. We kunnen er nu zondermeer  $X_i$  bijtellen en het geheel met een shift naar rechts, door twee delen. De denkbeeldige komma van  $X_{i+1}$  staat dan meteen weer goed voor de deling in de volgende iteratie.

## 6. De correctie shift

Bij de correctieshift, aan het eind wordt de komma weer geplaatst tussen bit 7 en 8 en na toevoeging van de gehalveerde karakteristiek gaat de wortel naar FLOP 4. Als  $1/4 < F = 1/16$ , dan  $F := F * 4$  en bij correctieshift : WORTEL /2. Als de karakteristiek oneven is, dan  $kar := kar + 1$  en bij correctieshift: WORTEL / 4.

## Het boek

Mocht je na het bekijken, lezen of bestuderen van bovenstaand geworstel met de vroege informatica willen weten hoe daaruit in veertig jaar zo'n bloeiende faculteit heeft kunnen groeien dan nodig ik je uit om eens een kijkje te nemen in het geschiedenisboek. Je kunt het krijgen bij BOZ en anders verwijzen zij je wel weer verder.

Veel leesplezier! ■

Wiek Vervoort  
w.a.vervoort@utwente.nl



	STM	2,8,SAVE
	STD	2,ADRX FLOP NAAR ADRX
	TM	ADRX,X'80' NEGATIEF
	BO	NEG NAAR FOUTINDICATIE
	SR	8,8 TOESTANDSTELLER = 0
	LA	7,1 CONSTATE = 1
	SR	4,4 KLAAR VOOR KARAKTERHALVERING
LA 4,1(4,0)	IC	4,ADRX KARAKTER NAAR 4
	AR	4,7 PLUS 1
	SRL	4,1 HALVEREN
LA 4,32(4,0)	A	4,HALFK PLUS 32
	TM	ADRX,X'01' KARAKTER ONEVEN
	STC	8,ADRX KARAKTER UITWISSEN
	BZ	EVEN
	LA	8,2 TELLER = 2 INDIEN ONEVEN LATER DELEN DOOR 4
	LM	2,3,ADRX FRACTIE NAAR WERKREGISTERS 2 EN 3
EVEN	TM	ADRX+1,X'C0' TUSSEN 1 EN 1/4
	BZ	KLEIN SPRONG INDIEN KLEINER 1/4
	SLDL	2,4 OM 2 ITERATIES ENKEL TE KUNNEN DOEN
LA	B	COMP ZOEK GEBIED
8,1(8,0)	KLEIN	AR 8,7 TELLER = TELLER + 1 INDIEN KLEINER 1/4 (/2)
	SLDL	2,6 MAAL 4 EN EXTRA SHIFT VOOR 2 ENKELE ITERATIES
	COMP	STM 2,3,ADRX INTERNE WAARDE TERUG NAAR ADRES X
*	*	APPROXIMATIE GEBIED ZOEKEN
	C	2,GRENS2 GRENS2 = 0,6777000
	BL	COMP1
*		GEBIED 3 $X0=1/2*(N + 2* 0,4921875)$
DRIE	L	5,PLUS3 PLUS3 HEEFT 2*OORSPRONKELIJKE WAARDE
	AR	5,2
	SRL	5,1 EERSTE BENADERING IN GEBIED 3
	B	NADER
*		VERGELIJK MET GRENS1
COMP1	C	2,GRENS1 GRENS1 = 0,3906250
	BH	TWEE
	L	5,PLUS1 GEBIED 1 $X0=N + 0,2421875$
	AR	5,2 EERSTE BENADERING IN GEBIED 1
	B	NADER
TWEE	L	5,PLUS2 GEBIED 2 $X0=11/16* N + 0,3600409$
	LR	3,2
	SRL	3,1
	AR	5,3 $0,3600409 + 8/16 * N$
	SRL	3,1
	AR	5,3 $+ 4/16 * N$
	SRL	3,2
	SR	5,3 $- 1/16 * N$
	L	3,ADRX+4 EERSTE BENADERING IN GEBIED 2
NADER	SLL	5,2 TWEE ENKELE LENGTE ITERATIES $1/2 *(X0+N/X0)$
	DR	2,5 $N/X0$
	AR	5,3 $X0+N/X0$
	SRL	5,1 $X1 = 1/2*(X0+N/X0)$
	LM	2,3,ADRX
	DR	2,5 $N/X1$
	AR	5,3 $X1+N/X1$
	SRL	5,1 $X2 = 1/2*(X1+N/X1)$
	LM	2,3,ADRX DERDE ITERATIE DUBBEL
	DR	2,5 $N/X2$
	LR	6,3 EERSTE DEEL QUOTIENT BEWAREN
	SR	3,3 NULLEN TOEVOEGEN
	SRL	2,1 SHIFT OM OVERFLOW TE VOORKOMEN
	DR	2,5 REST DOORDELEN
	SLL	3,1 TEKENBIT 2E QUOTIENT WEG
	LR	2,6 KOPPELEN AAN 1E QUOTIENT
	AR	2,5 $X2+N/X2$
	SRDL	2,7(8) SHIFT AFHANKELIJK VAN TOESTANDSTELLER
	STM	2,3,ADRX WORTEL NAAR ADRX
	STC	4,ADRX KARAKTER NAAR ADRX
	LD	4,ADRX WORTER NAAR FLOP4 N NOG IN FLOP2
	B	EINDE
NEG	WTO	'NEGATIEF ARGUMENT IN WORTELROUTINE'
EINDE	EQU	*
	LM	2,8,SAVE
SAVE	DS	8F
HALFK	DG	F'32'
ADRX	DS	D
GRENS1	DC	X'06400000'
GRENS2	DC	X'0AD7DBF4'
PLUS1	DC	X'03E00000'
PLUS2	DC	X'05C2BA40'
PLUS3	DC	X'0FC00000'





This article presents the preliminary results of a case study we carried out at ASML. According to their website, "ASML is the world's leading provider of lithography systems for the semiconductor industry, manufacturing complex machines that are critical to the production of integrated circuits or chips".

Lithography machines are large embedded systems. The software components of these machines consist of over 15 million lines of

code (LOC), written in the C programming language. The system is divided into several architectural layers and consists of more than 200 components. Within this context the Ideals[8] project aims at identifying crosscutting concerns at both the implementation and architecture level, and modularizing these crosscutting concerns using (new) Aspect Oriented Programming techniques.

# Applying AOP in Industry

## Benefits and Challenges

### Problem: Cross-Cutting Concerns

Figure 1 shows a visualization of a representative ASML component. In this figure, each vertical bar represents a single file in this component. We have colored the lines in the source code that deal with strongly related pieces of functionality, also called concerns. In this case the red lines deal with the concern of Parameter Tracing. This concern has two key responsibilities. Firstly, for each function that is called it must trace (that is, write to a log file) the values of input parameters at the start of that function. Secondly, it must trace any output parameters and the return value at the end of the function. The green lines are concerned with Profiling. This concern ensures that the duration of each function is measured and stored in a profiling database. Finally, the blue lines deal with providing Reflective Information to the program, viz. the defining (ASML) module. We also identified several other recurring concerns in this code, namely: Error Handling, Parameter Checking, Memory Handling and Startup/Shutdown routines. However, these will not be discussed in detail in this article.

The three concerns mentioned above have a high impact on the component shown in figure 1: together, they account for over 25% of the

statements in this component. We can also see that the implementation of these concerns is not neatly modularized, as the lines of each color are scattered over multiple files, and often occur in multiple locations within each file. We call these kinds of concerns Crosscutting Concerns. Such concerns cause several kinds of problems, which we will discuss based on three characteristics of such crosscutting concerns: Scattering, Replication and Tangling.

### Scattering

The first characteristic of crosscutting concerns is that the code which

implements the desired functionality is distributed over the base system. Figure 2 illustrates this for the Profiling concern.

A scattered concern occurs when one logical piece of functionality is distributed over multiple locations. Practically, this results in an increase of maintenance effort (and hence, cost). For example, if the base program is extended or altered (e.g. a new function is added), the programmer must take the Profiling concern into account (e.g. remember to add the 'boilerplate' lines of code that implement profiling to this new function).

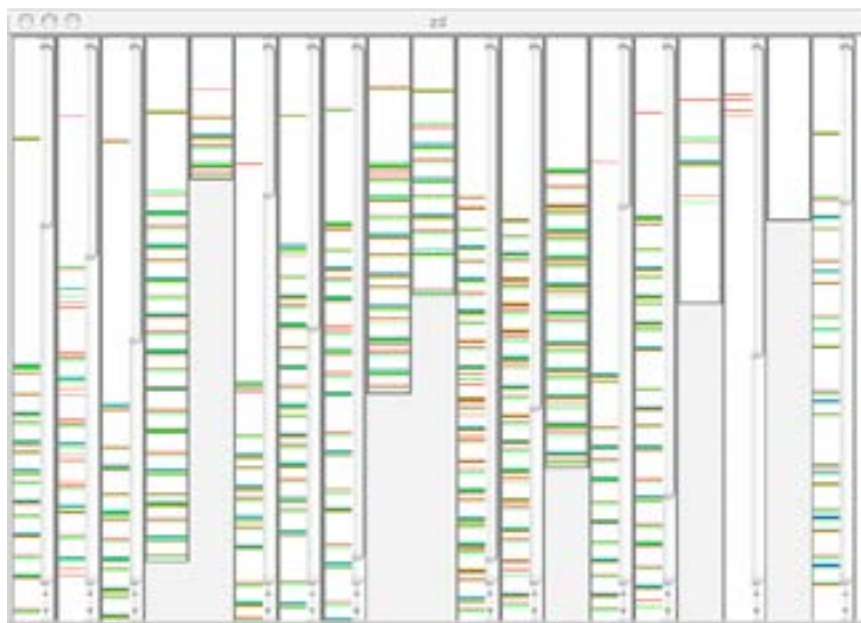


Figure 1: Example ASML Component

In addition, if we want to update the Profiling concern itself (e.g. to include more information), this may require updating the code related to this concern in all the places shown in figure 2. Furthermore, it is hard to see where exactly the Profiling concern is implemented. This is a problem when, for example, we want to create a version of the software without the profiling code at all.

### Replication

Due to the scattered implementation of the concern, we often see that (almost) the same ‘boilerplate’ code is implemented at many different locations. There may be some context specific information present in the concern code, e.g. the name of the function, but otherwise the code is replicated many times. Such code is often written using a ‘copy-paste-edit’ process, from templates or already existing code. There are several problems with this kind of code replication. The first one is that there is still a lot of effort involved to write and maintain this “simple” code. Secondly, we know from experience that even in such “simple” code, a lot of errors occur. Thirdly, if we want to alter the functionality we may have to update the replicated code in all the locations where it occurs. Finally, we see that often the same (or very similar) functionality is designed and implemented inconsistently (that is, in different ways) in different subsystems.

### Tangling

We say that a concern is tangled (with other concerns) when a single module implements not only its main functionality (i.e. the functionality usually suggested by the name of the module or function), but also deals with other concerns, i.e. if a program unit contains a mixture/interleaving of concerns. In Listing 1, a typical ASML function is presented.

Here we have colored all the lines that deal with one of the distinct concerns identified before (using the

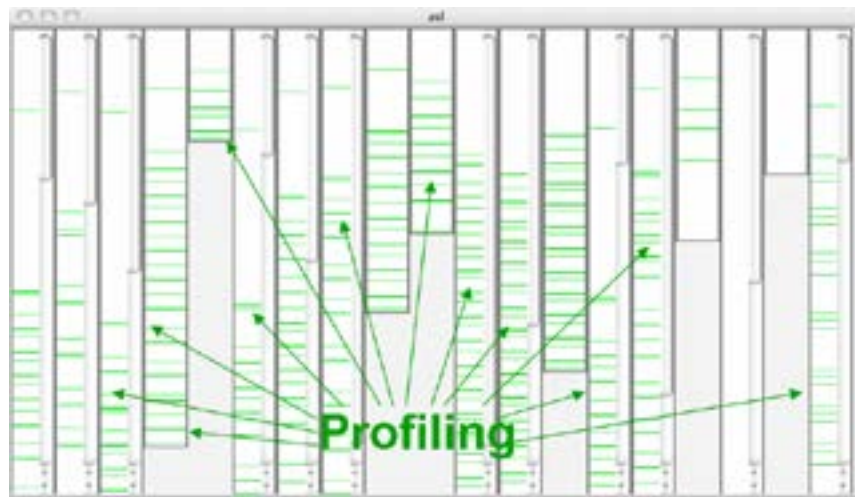


Figure 2: Profiling Concern

same colors). In addition, the grey lines deal with error handling and the black lines present the core functionality of this function. As one can see there are many concerns interleaved with each other. This reduces the comprehensibility and the maintainability, e.g. updating one concern may break surrounding code.

To summarize, crosscutting concerns can be identified as being: tangled with the base code, scattered throughout the base code and/or presenting replicated functionality. Crosscutting concerns cause additional effort and more (lines of) source code, increased lead time and increased risk of

### CASE STUDY SETUP

For our case study, we selected a subsystem driver component, which is composed of 30458 lines of source code, containing 6389 control and executable statements. We selected this component, because it is representative in terms of crosscutting concerns. In addition, it was recently maintained for better modularity, which would increase our chances to identify the “real” crosscutting concerns, i.e. not the ones due to the “aging” of the component. To identify crosscutting concerns, we “manually” investigated source code, and used AspectBrowser [4] to visualize the crosscutting concerns as shown

“We often see that (...) the same ‘boilerplate’ code is implemented at many (...) locations”

errors in the code. Furthermore, they reduce maintainability and may cause inconsistencies in the application of such concerns, when the same concern is implemented differently in several modules. It is also important to realize that conventional languages like Object-Oriented and procedural languages lack the ability to address these concerns effectively.

The Aspect Oriented Programming paradigm is meant to address the problem of modularizing crosscutting concerns. Figure 3 (page 21) depicts schematically how aspects (Profiling and Tracing, in this figure) are separated from the base code.

in figure 1. We identified six crosscutting concerns in the component: reflective information (e.g. listing 1, line 2), exception handling (e.g. lines 1, 3, 7, 10, 11, and 14), profiling (e.g. lines 4, 5, and 13), parameter tracing (e.g. lines 6 and 12), parameter checking (not shown), and memory handling (not shown). Note that most of the functions contain more core functionality than this one – however, keep in mind that in such a large code base, it is necessary to enforce strict coding guidelines. Therefore, even simple functions have to adhere to such guidelines on the implementation of recurring concerns such as error handling – even if no actual



errors could possibly occur in the current implementation of this particular function. Admittedly, the overhead created by such crosscutting concerns is not usually as extreme as in this example.

Out of the identified crosscutting concerns, we addressed only parameter tracing, profiling, and reflective information, because these crosscutting concerns made up for 28% of the code size, and realizing them using aspects was feasible within our limited time frame. To apply AOP, we

re we consider this set as a collection of common or fundamental issues that must be addressed by any AOP mechanism, if it is to be used in an industrial context.

### Migration

The concerns we handled in this case were mostly implemented on separate lines in the code and were easy to identify with “simple” tooling. However, migrating the tracing concern was more difficult, as determining whether to trace a variable at

### Availability and maturity of tooling and process

The current implementation and process has been developed as a proof of concept. These tools have to become industry strength, and this requires substantial investments. Using AOP for the selected concerns has clear benefits, but a large impact, too. The use of AOP requires a seamless integration into the development and build-process. The tools used in these processes should be robust and mature. Furthermore, in an industrial development process there are usually specific roles, with different responsibilities. These roles should remain clearly separated even in the presence of aspects. If one uses AOP, there will be additional roles in the development life-cycle, most notably an aspect developer, a weave tool maintainer, and an integrator. The latter is responsible for the sanity of the entire system (i.e. the base code combined with aspects).

### Ability to switch off aspects

It should be possible to switch aspects on or off, for two reasons. Firstly, for tracking down errors, if the aspect code contains a bug the system would still be able to compile without a specific aspect. This can only be done if the impact of the aspects is not too severe. For example, turning off an exception handling aspect is not feasible as it would compromise the entire system. The second reason is for more control over the aspect application. In development versions of the software some aspects are turned on, while in the releases these aspects are turned off. Or, if some component is called often, Tracing might need to be turned off for that components to improve runtime efficiency. As most identified aspects are cleanly separated from the base code, these can be turned off, if required. However, if there exist base-aspect or aspect-aspect dependencies this may not be trivial. (As noted however, if the implementation of a particular concern is scattered over

“The current implementation and process has been developed as a proof of concept”

first removed the code related to the selected crosscutting concerns from the component using `sed`, a regular expression engine. Next, we specified the aspects for the three crosscutting concerns. Finally, we used our weaver, namely `WeaveC` [6], to weave the aspects with the component.

### Issues and Worries

This section describes some of the issues which should be addressed for applying an AOP approach in an industrial context. Although these might seem ASML specific, we do feel that these issues are shared by most, if not all, companies. Therefo-

the start or end of a function relies on the parameter’s input and output specification. We implemented this by annotating which parameters are input, output or both. Automatically generating these annotations requires control flow analysis and read-write analysis of the parameters.

The CWI [7] is currently developing tooling to automate the migration process, i.e. to remove the crosscutting concern code, to insert the tracing annotations and to indicate the deviations which must be handled manually.

```

1.  int get_data(Data_struct* data_ptr)
2.  {
3.      const char* func_name = "get_data";
4.
5.      int result = OK;
6.
7.      Timing_handle timing_hdl = NULL;
8.
9.      TIMING_IN;
10.
11.     Trace_in(tr_handle, func_name);
12.
13.     if (result == OK)
14.     {
15.         /* Retrieve data */
16.         *data_ptr = data;
17.     }
18.
19.     CHECK_ERROR(result, CCCXSF_GET_DATA_FAILED_obj);
20.
21.     Trace_out(tr_handle, func_name, data_ptr, result);
22.
23.     TIMING_OUT;
24.
25.     return result;
26. }
```

Listing 1: An example ASML C function

the base code, turning it off/removing it completely is even harder.)

### Understandability (mental model)

ASML uses the C programming language, hence switching to AOP is considered to be a real issue. When applying AOP, there is a clear separation between aspect developers and core developers. Core developers still want to understand what goes on “behind the scenes”, i.e. what gets added to their functions by aspects, as they are understandably wary of techniques that influences the execution of “their” code. Hence, they may want to verify the woven code.

### Compile-time & Run-time performance

In large systems with many developers and many dependencies between components, the build performance can be quite crucial. As the throughput at runtime must be guaranteed, a runtime performance hit is usually simply unacceptable. A static weaving approach is thus more attractive. A compile-time performance hit is usually acceptable as long as the performance hit is not (potentially) several orders of magnitude; i.e. the performance hit at compile time should not be exponential to the number of join points or files to be processed.

### Ability to Debug

Debugging a program with aspects is also considered to be an issue. Especially, if an error occurs who is responsible for fixing this error? First of all this requires determining whether the error is in the aspect code, in the base code or in the combination. As the core developers may not be aware of aspects, they may not want to see the code with the aspects applied. However, when debugging the total system, one needs to be aware of the aspects. We came up with the following options: debug with the inserted code, debug with inserted code hidden, and debug normally and while executing the inserted code

jump to the aspect definition. In our case study we implemented the second option.

### EVALUATION AND BENEFITS

This section will briefly explain the results we achieved with our case study. Most of these are considered to be the key motivations for using AOP. We provide these in the context of the case study.

#### Statement reduction

As stated previously, our experiment was carried out on a component of about 6K LLOC. We use the term LLOC to indicate all logical lines, thus excluding white space and comments. In our experiment, we achieved a reduction of 26% of LLOC as compared to the original code. In our case study, we only implemented three aspects. If we would also address parameter checking and error handling, the reduction would be up to 50%. The migration path, especially for the error handling concern, is more difficult as the error handling code is extremely tangled with the base code.

constrained. However, there are situations where local deviations are necessary from a business perspective.

#### Better software structure

Achieving better software structure has been stated as one of the cornerstone benefits of AOP and separation of concerns in general. The pointcut and advice mechanisms provide an easy way to deal with changing coding guidelines. The benefit of AOP was clearly visible when we applied the profiling aspect on a different component (which did not previously implement profiling) within 10 minutes. Although this flexibility can partly be achieved by using C macros, these do not offer a pointcut-like mechanism. This means the macros always have to be included in each file, and the macro “calls” have to be manually added and maintained in all the correct locations.

#### Conclusion

We presented the case study described above to circa 100 developers, and the feedback was mostly positive. The results of the presentation and case study were convincing enough to initiate a transfer project. The

“Aspect-Oriented Programming is still a relatively young field”

#### Local deviations

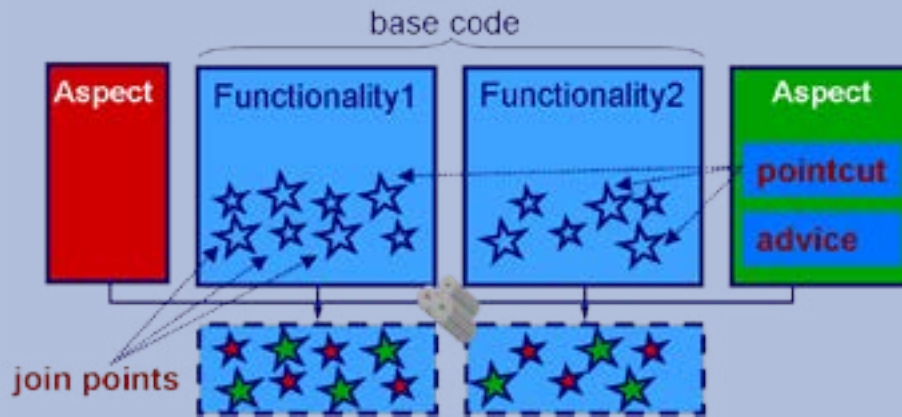
The solution we proposed still allows for local deviations. For example, there are functions which are called often and for these functions tracing is turned off. There are also low level components, which should not be traced, as these are active before the logging and tracing component is initialized. Although we do not endorse these deviations, one can implement these deviations with the use of annotations and a more powerful pointcut description language. Most aspects discussed here implement coding guidelines. These should therefore be enforced and applied uniformly. For this reason, the usage of local deviations has to be

results showed that there were no fundamental issues with this aspect-oriented approach. The purpose of the transfer project is to mature the aspect language, process and tooling. It is planned that from early 2007, the first ASML components will use aspects. This also implies training the developers currently working on those components. There will also be a separate training for the people who are going to be responsible for writing and maintaining the aspects.

Aspect-Oriented Programming is still a relatively young field; not many industrial case studies have been published until now. Industrial application is also hampered a bit by the lack



Figure 3: Ingredients used in Aspect Oriented Programming:



- Aspect: module encapsulating crosscutting behavior. An aspect usually consists of:
  - Pointcuts: describe some selection criteria over the base code to select those places where behavior has to be adapted. The actual points in the base code that are selected by a particular pointcut are called 'join points'.
  - Advices: describes the actions that have to be executed at selected join points
- Weaver:
  - Composes the aspects with base code, resulting in a complete system (as shown at the bottom of figure 3).
  - For the Ideals project, a C weaver was developed
  - Provides source-to-source weaving (i.e. it transforms aspect source + base C source -> woven C source code, which can be compiled using a regular C compiler)
  - Only works on preprocessed source (so it does not have to deal with macros)
  - Supports the following weaving capabilities:
    - function execution: insert behavior before/after
    - function: insert variable declaration
    - global: insert variable declaration
    - global: insert field declaration into a structure
    - function call: insert behavior before/after
  - Supports several pseudo variables to access context information in advices:
    - E.g. %MODULE\_NAME% in advice is replaced with the base filename where the advice is woven
  - Available as open source software: <http://weavec.sourceforge.net>

of suitable tools; in this case, ASML decided to invest in developing tooling. Because a number of crosscutting concerns will apply to most of the ASML code base of 15 MLOC, the potential benefits are substantial, and warrant an initial investment in this technology.

There are also a number of other issues with AOP that have not been addressed satisfactorily. One of the most important issues is that modularizing a crosscutting concern may make it difficult to understand what exactly happens in the base code of a particular module (function), since there is no explicit reference to aspect code. As a result, the application of aspects may yield unexpected results, perhaps even mutually inconsistent. This is an area of active research, which tries to detect and/or prevent such unexpected and incorrect behavior to occur.

In the Software Engineering group at the UT, the composition filters

approach [9] aims at offering aspect-oriented programming that is more robust. Some of its characteristics are: (a) language-independent interface-level specification, (b) respects strong encapsulation, (c) extensible library of well-defined advice as filter types, and (d) offers a simple, restricted specification language for matching and manipulating messages (e) which enables static reasoning about the (safe) behavior of an aspect-oriented program. The concept of composition filters is recently applied in the Compose\* language [9], with an implementation for the .NET platform, and proof-of-concept implementations for Java and C. ■

Pascal Dürr  
 Wilke Havinga  
 Lodewijk Bergmans  
 Software Engineering Group  
 EWI faculty  
 University of Twente  
 the Netherlands  
 {durr, havinga, bergmans}@ewi.utwente.nl

## References

- [1] ASML. <http://www.asml.com/>
- [2] Technical University. Eindhoven. TUE. <http://www.tue.nl/>
- [3] Embedded Systems Institute. ESI. <http://www.esi.nl/>
- [4] University of California. Aspect-Browser. <http://www-cse.ucsd.edu/users/wgg/Software/AB/>
- [5] University of Twente. <http://www.utwente.nl/>
- [6] University of Twente. WeaveC. <http://weavec.sourceforge.net>
- [7] Centrum voor Wiskunde en Informatica. CWI. <http://www.cwi.nl/>
- [8] Ideals project page. <http://www.esi.nl/ideals/>
- [9] University of Twente. Compose\*. <http://composestar.sourceforge.net>

**ADVERTENTIE**

**BD/CICT**

# ADVERTENTIE

## Deloitte 1



In future computing environments you will be surrounded and supported by all kinds of technologies. You will be able to interact with them in a natural way: you can speak to, point at, or even frown at some piece of presented information: the environment will understand your intent. Natural interaction approaches will improve the way we work. However, this is still in the future. True automated understanding can only come from



### Human Media Interaction

context. The BioRange project at the Human Media Interaction (HMI) group addresses such natural interfaces from the viewpoint of scientific experiments in the molecular biology domain.

## Human Computing in the Life Sciences

### What does the future hold?

#### Where Molecular Biology and HCI meet

Bio-informatics is a new research domain. Like biochemistry and biophysics, bio-informatics has its roots in the biology domain. Increasing amounts of data are gathered by biologists, mainly as a result of new technological developments in molecular biology. Such vast data sets need to be interpreted and managed, generating even more data in the process. New tools are needed that allow us to perform data analysis, management and annotation. Such tools are rooted in computer science (CS) [1]; hence the name bio-informatics.

A key role in molecular biology is played by bio-informatics. Molecular biology teams are highly multidisciplinary, and include biologists, statisticians, bio-informaticians, computer scientists and microarray experts. Diverse types of expertise contribute to tackling the complex biological challenges of both the present and

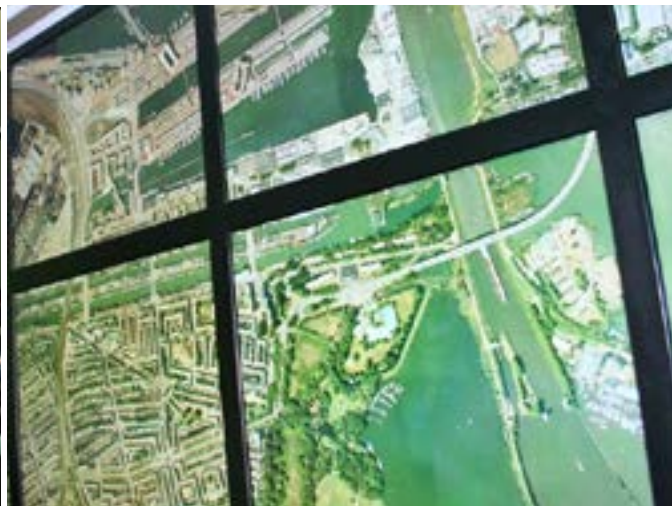
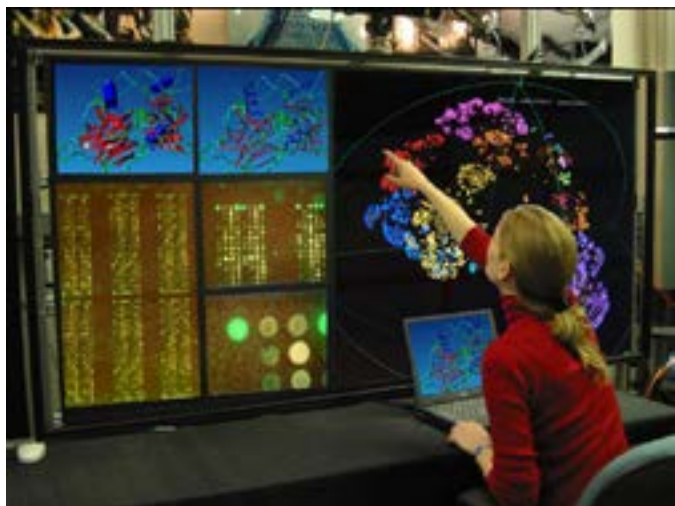
the future, such as curing Multiple Sclerosis and Huntington's Disease. These teams need some way to work with these vast data sets in a single location. Understanding one another in the team is as important as a joint understanding and interpretation of the acquired data. More clarity in the relationships between data and the current working progress can help. A picture can say more than a thousand words.

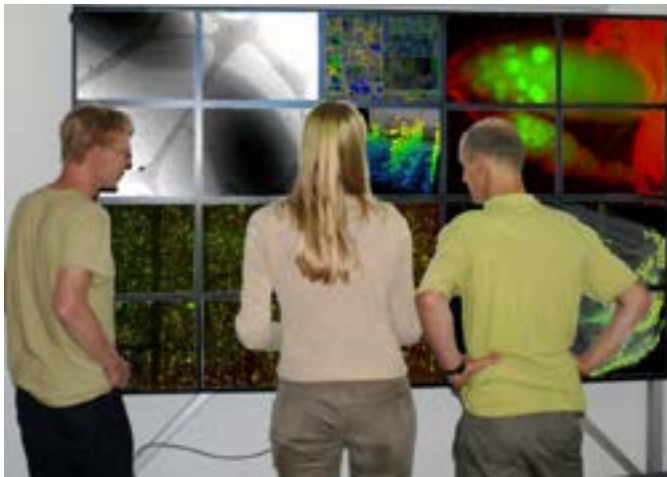
Bio-informaticians use their knowledge of both biology and computer science to develop tools that allow biologists access to data acquired in their complex experiments. Opposite to the current standard in CS, user-centred design is something to be scared of in bio-informatics. Tools are not based on their users' needs and so biologists are lost in a forest of functionality that they do not understand. They simply let it be and pass it by [3]. Biologists should not be struggling to operate sophisti-

cated tools and devices. They should work, in teams, with their data in intuitive ways. It is here that HCI, human-computer interaction, and molecular biology meet.

#### Smart Surroundings

With the BioRange project, the Netherlands has positioned itself at the forefront of the rapid developments in bio-informatics. HMI's BioRange group develops interaction tools for molecular biology teams [5]. We develop these tools for the *e-BioLab* [4]. The *e-BioLab* is a future computing environment which is under continuous development. Its facilities include large, high-resolution displays and portable PCs. However, state-of-the-art technologies add nothing of themselves. Our aim is to support the molecular biology team with the possibilities that the *e-BioLab's* facilities can offer. In the *e-BioLab*, team members will better understand each other, the team will be more creative due to both new insights and new





functionality, and the surroundings will understand and anticipate users' behavior. It will be a truly smart environment.

The following sections describe how we are addressing this vision. More specifically, they describe our main research questions.

### Mutual Understanding

An initial study of current working practices in molecular biology indicates that work is done in a semi-structured way, research is explorative. It is difficult to explicitly define team members' responsibilities for parts of the experiment at hand and to validate and reproduce results, therefore experiments need to be modelled.

Currently, we are investigating the uses of workflows to describe dynamic molecular biology experiments. These are already in wide use today in, for example, business and commerce. Workflows describe which resources are required for each process that is started, in progress and completed. For example, what happens after you have taken an exam at the UT is defined by a workflow system: the relevant teachers and staff will be notified in turn so you can, after some weeks, find your grade in TOST. We ask ourselves how a workflow system can be adopted to describe the dynamic nature of an experiment in molecular biology. A fine line needs to be walked concerning our design's rigidity: too rigid will prevent our team thinking "out of the box" and too generalized will result in meaningless models.

After modelling an experiment, it is vital to assess how such a model can guide an experiment and how a model can provide a focus for discussion. During a meeting, the structure and current progress of an experiment can be displayed. Our research addresses the question of when, where and how to depict an experiment's progress, datasets, user roles and processes. We aim to guide the team members in their work by increasing their mutual understanding of these topics.

### Group Creativity

The multidisciplinary nature of molecular biology teams itself stimulates complex thinking and reasoning. In the *e*-BioLab, scientists will work together to interpret very large

We are currently observing teams in the *e*-BioLab, in a series of meetings. By describing the processes that take place and the way in which team members interact with each other and their environment, we work towards an evaluation framework for collaborative, scientific environments. This framework includes requirements and design concepts for such future smart rooms.

### Understanding Behavior

A smart room understands its inhabitants. Users can act naturally: the environment will understand their intentions, and react to them. In essence, the traditional computer interface disappears. The ideal natural interface can recognize speech, facial gestures, hand gestures, body

## "A smart room understands its inhabitants"

and complex datasets. To visualize a large quantity of data you simply need a lot of physical space to depict them. Large displays immerse their users more than widespread and small displays. But how do the large displays and other technologies in the *e*-BioLab affect the way the team works together? In order to answer this question, we describe the influence of mutual situational awareness: the environment's awareness of its users and their work, and the team's awareness of the opportunities that their new environment offers. To assess this influence, we need to describe the tasks and users we find in molecular biology experiments.

postures and so on: it can deal with all modalities that humans use to communicate. Humans express significant parts of their messages using non-verbal signals. We are exploring intuitive two-handed gesture-based interaction with large displays. Users will be able to navigate, select and manipulate complex experimental data using their hands. They will do so both individually and in groups. However, the current state-of-the-art in gesture-based interaction [6] does not yet allow us to do this out-of-the-box, especially in multi-party unconstrained environments.

So what are the challenges we face? First, we must detect two-handed



gestures from multiple users simultaneously. Specialized gloves, touch screens or computer vision can be used here. Then, we must be able to identify a hand gesture and the person who is making it. Was it Billy or Bob? And did he want to see graph A or did he simply want another coffee? The same pointing gesture can mean both things. So, to resolve this ambiguity, we need a language of hand gestures. User intent needs to be understood: was it the coffee or the graph? Such understanding can

### More information

Are you interested in any of the topics we have addressed here? Please contact us to participate in or carry out projects and Capita Selecta assignments! Look for more information at: <http://hmi.ewi.utwente.nl/project/BioRange> or mail us: [biorange@ewi.utwente.nl](mailto: biorange@ewi.utwente.nl).

### Acknowledgement

This work is part of the BioRange program carried out by the Netherlands Bio-informatics Centre

### References

[1] Jacques Cohen. Bio-informatics - an introduction for computer scientists. *ACM Computing Surveys*, 36(2):122–158, 2004.

[2] Wim Fikkert, Marco D'Ambros, Torsten Bierz, and TJ Jankun-Kelly. Interacting with visualizations. In Andreas Kerren, Achim Ebert, and Joerg Meyer, editors, *Human-Centered Visualization Environments*, LNCS Tutorial. Springer, August 2006. GI-Dagstuhl Seminar.

[3] Olga Kulyk and Ingo Wassink. Getting to know bio-informaticians: Results of an exploratory user study. In *Workshop on Combining Visualisation and Interaction to Facilitate Scientific Exploration and Discovery* in conjunction with *British HCI 2006*, pages 30–38. ACM Press, 2006.

[4] Han Rauwerda, Marco Roos, Bob Hertzberger, and Timo Breit. The promise of a virtual lab in drug discovery. *Drug Discovery Today*, 11:228–236, March 2006.

[5] Paul van der Vet, Olga Kulyk, Ingo Wassink, Wim Fikkert, Han Rauwerda, Betsy van Dijk, Gerrit van der Veer, Timo Breit, and Anton Nijholt. Smart environments for collaborative design, implementation, and interpretation of scientific experiments. In *Workshop on AI for Human Computing* in conjunction with the *International Joint Conference on Artificial Intelligence*, volume 20. AAAI Press, January 2007.

“User intent needs to be understood: was it the coffee or the graph?”

only flow from context and it is context that we model using our workflows. We thus explore how context can help us understand behavior. Also, context helps us to define appropriate feedback.

### Looking at the Future

Clearly, our work does not provide solutions for all open topics in creating future “smart” environments. We focus on a narrow use case: supporting life science experiments. However, our findings will be applicable in a broad domain. They contribute to the future vision of human computing: see, hear and feel you in there! ■

Wim Fikkert

On behalf of HMI's BioRange group  
[fikkert@ewi.utwente.nl](mailto:fikkert@ewi.utwente.nl)



**ADVERTENTIE**

**ASML**

Op het moment van schrijven staan de feestdagen nog voor de deur, Sinterklaas moet zijn verjaardag nog vieren, de geboorte van kindje Jezus nog herdacht en het oude jaar moet nog in stijl worden afgesloten en de start van het nieuwe jaar uitbundig gevierd. Maar op het moment van verschijnen heeft 2007 zich al aangediend en is een nieuw jaar vol nieuwe mogelijkheden begonnen.

## Van de voorzitter

Voor het bestuur van ENIAC betekent dit dat we ons hebben moeten beraden op wat we in 2007 gaan doen. Aan het eind van 2006 hebben we een enquête uitgeschreven. Deze enquête kent een twintigtal veelal meerkeuzevragen over het reilen en zeilen van ENIAC. Bijvoorbeeld over het aankondigen van activiteiten, de soort activiteiten, de periodieken en wat je verwacht van ENIAC. De antwoorden op deze vragen willen we graag gaan gebruiken om onze plannen voor 2007 vorm te geven, dus heb je de enquête nog niet ingevuld dan verzoek ik je om dit alsnog te doen. Het invullen hiervan duurt maximaal vijf minuten en geeft ons als bestuur een idee of we het roer volledig moeten omgooien of dat we op de goede weg bezig zijn. Daarnaast geeft het je ruimte om wat wensen betreffende ENIAC door te geven. De enquête is anoniem.

Voorlopig hebben we al wel een planning van activiteiten gemaakt voor het komende jaar. Die vind je

onderaan dit stuk. Wat ik daarvan zeker onder de aandacht wil brengen is de ALV eind maart. Op deze bijeenkomst presenteert het bestuur namelijk de plannen voor 2007. Een mooie kans om te zien wat de resultaten van de enquête hebben opgeleverd, maar ook om je zegje te doen over deze plannen!

2007 is sowieso een speciaal jaar voor ENIAC, onze vereniging bestaat dan namelijk 25 jaar. Dit willen we natuurlijk gaan vieren. Houd dus de agenda de komende maanden in de gaten voor meer informatie over de lustrumactiviteit.

Dan rest me nu nog het aanspreken van die mensen die voor 2007 goede voornemens hebben gemaakt wat betreft vrijwilligerswerk: ENIAC zoekt ook voor 2007 mensen die op projectbasis een activiteit willen organiseren. Daarnaast zijn we nog op zoek naar een bestuurslid! ■

Met vriendelijke groet,  
Suzanne Verlijdsdonk

### Agenda

Datum	Locatie	Activiteit
Februari 2007	Amsterdam	BOL Tandberg
24 maart 2007	Nog onbekend	Algemene ledenvergadering
Nog onbekend	Enschede	Elysium/ENIAC dag over de ICT ontwikkelingen op het gebied van gezondheid
Oktober	Nog onbekend	Lustrumfestiviteit

Op 18 november 2006 kwam Sinterklaas aan in zijn pakjesboot te Middelburg. Aan de andere kant van het land vond echter nog een ander evenement plaats, dat heel weinig te maken had met Sinterklaas, namelijk de bachelorvoorlichting aan de UT. Echter daarover zal dit stuk ook niet gaan, maar over de UT Alumnidag 2006. Voor de alumni was de dag in grote lijnen een parallel programma verzorgd door de verschillende alumniver-

enigingen, gevolgd door een algemeen en centraal programma van de universiteit.

## Alumnidag 2006

Voor de INF-, BIT- TEL-alumni begon de dag al vroeg om half elf met een ontvangst van koffie, thee en cake. Dit keer geen Twentse krentenwegge, onder de aanname dat de universiteit een krentenwegge zou serveren bij het middagprogramma. Uiteraard bleek dit een speculaas-koekje, omdat het de intocht van Sinterklaas was.

Na de ontvangst en een kleine opening van Suzanne Verlijdsdonk, voorzitter ENIAC, was het aan Arend Rensink om de eerste presentatie te geven. Deze luchtige en opgewekte presentatie ging over getallen, grote getallen en nog grotere getallen. Maar ook over absurd grote getallen dat die niet meer opgeschreven kunnen worden, omdat er te weinig atomen ( $\pm 10^{80}$ ) in het zichtbare universum zijn. Uiteindelijk kwam de presentatie

weer terug bij de informatica over de testbaarheid van software en de groei van mogelijke 'features' of fouten in een softwaresysteem.

De geschiedenis van Informatica was het onderwerp van de tweede presentatie en gaf in een vogelvlucht weer hoe de studie en later ook de faculteit Informatica ontstond uit Wiskunde en Electrotechniek. Het interessante hieraan is dat het idee voor een zelf-



standige studie en faculteit al bestond net na of rond de oprichting van de toenmalige Technische Hogeschool Twente. Het volgende deel van de presentatie was het rondzwerven van Informatica over de verschillende locaties op de campus, voordat ze uiteindelijk terecht kwam in het huidige gebouw Zilverling, ook wel gebouw 11, Informatica en voor sommigen 'de legbatterij'. De presentatie werd afgesloten met het samengaan van de faculteiten Wiskunde, Electrotechniek en Informatica tot de faculteit

derdelen humor bestaat en hoe je ze het beste kan gebruiken.

Na de cursus volgde de introductie van het speciaal gebrouwen Alumnusbier door, uiteraard, Grolsch, en werd onder begeleiding van de beveiliging naar binnengebracht. Dit speciale bier kon vervolgens gedronken worden tijdens de borrel en het opvolgende diner. De afsluiting van de dag vond plaats door middel van een tweetal concerten van oude campusbands en een concert van Ilse de Lange. ■

"Deze (...) presentatie ging over getallen, grote getallen en nog grotere getallen"

EWI. De afsluitende sheet was dan ook 'format c' uit Informatica.

Francis Henninger  
Functionaris activiteiten

Na de staande lunch, gezien het aantal mensen was het restaurant in de Drienerburght te klein maar wel gezelliger, was de laatste presentatie aan Daan Velthausz over de huidige activiteiten en organisatiestructuur van het Telematica Instituut. Na de korte introductie werd er overgegaan naar het Freeband-programma en het bijbehorende 'Living Labs'-concept. In dit concept wordt de proefpersoon of potentiële gebruiker niet naar een testomgeving gebracht, zoals het Home-lab van Philips, maar observeert de onderzoeker de persoon in zijn of haar eigen omgeving.

Het laatste onderdeel van het ENIAC-programma was een borrel, zodat er nog voldoende tijd was om herinneringen op te halen en eventueel te kunnen netwerken, voordat er overgegaan werd naar het algemene programma van de universiteit.

Het groots opgezette centrale programma bestond uit een ontvangst met koffie of thee en de eerder genoemde speculaas. Het eerste programmaonderdeel, na de opening door het college van bestuur, bestond uit een spoedcursus humor. In deze cursus werd op een 'wetenschappelijke' manier uitgelegd uit welke on-

Hoewel een printer er wellicht van buiten niet al te ingewikkeld` uitziet, blijkt er toch zeer veel te komen kijken bij het ontwikkelproces van de huidige apparatuur. Vroeger betrof dit nog voornamelijk elektrotechniek en mechanica, maar nu de trend richting multifunctionele documentoplossingen gaat komt er ook steeds meer informatica aan te pas.

Wij namen een kijkje in de keuken van de afdeling Research & Development van Océ; één van de marktleiders in de branche.

## Op bezoek bij...



### De bedrijfsstructuur

De afdeling Research en Development is nog verder opgesplitst in drie subafdelingen: research, development en engineering.

Bij research wordt voornamelijk gekeken naar de mogelijkheden die wellicht interessant zijn voor toekomstige ontwikkelingen. Hierbij is het doel niet direct het maken van een product, maar meer het verbreden van de kennis binnen Océ.

Bij development wordt er gewerkt aan ideeën die al wel interessant en nuttig zijn gebleken, maar waarvan het nog onduidelijk is hoe deze het best in een concreet product omgezet kan worden.

Op het moment dat de hoofdlijnen duidelijk zijn, zal engineering het product afmaken. Hierbij wordt onder andere nog gekeken naar hoe er het meest efficiënt geproduceerd kan worden, repareert men de laatste bugs en worden bijvoorbeeld allerlei functionaliteiten die een goede ondersteuning mogelijk maken ingebouwd.

Projecten worden bemand met medewerkers uit verschillende afdelingen. Deze afdelingen hebben elk een bepaald expertisegebied. Zo zijn er onder andere de gebieden mechanica, elektrotechniek en software. Een

medewerker werkt meestal aan één project, waarbij hij zowel een projectleider (die verantwoordelijk is voor de voortgang van het project) als een afdelingshoofd (die verantwoordelijk is voor de ontwikkeling van het expertisegebied & de medewerker) heeft. Daarnaast is er nog een coach, die wat dichterbij de medewerkers staat en hen van allerlei adviezen kan voorzien.

### Een loopbaan bij Océ

Om een beeld te krijgen van hoe het is om als informaticus bij Océ te werken spraken wij met Arjan van der Hoogt. Hij heeft Informatica aan de Universiteit Twente gestudeerd en is ruim acht jaar geleden bij Océ terechtgekomen, na eerst bij Philips stage te hebben gelopen en een afstudeeropdracht over applicatieprotocollen te hebben gedaan.

Bij Océ is Arjan begonnen met een werkopdracht om het bedrijf te leren kennen. Hij onderzocht hierbij welk besturingssysteem het best kon worden gebruikt voor een product. Na een half jaar was het tijd voor een nieuwe uitdaging: het schrijven van requirements voor een bedieningsapplicatie. Deze opdracht - waarbij het voornamelijk om interfaces ging - was een stuk minder technisch dan de overige werkzaamheden die Arjan bij Océ heeft verricht.



Arjan van der Hoogt

Vervolgens heeft Arjan nog een jaar als software-engineer gewerkt en werd hij coach binnen deze afdeling.

Via de functie integrator (het samenvoegen van deeloplossing tot een totaaloplossing) is hij op dit moment weer coach, deze keer binnen de breedformaatmarkt. Daarbij gaat het om printers die op A0 en dergelijke formaten kunnen printen. Bovendien zit Arjan samen met andere disciplines in de technische commissie van het project. Deze commissie zorgt voor de goede voortgang van het project en neemt belangrijke technische beslissingen.

Wie van de universiteit komt weet veelal nog niet wat hij kan verwachten van het bedrijfsleven. Arjan koos



dan ook voor Océ, aangezien de mogelijkheden hier erg uitgebreid zijn en hij dus verwachtte hier in ieder geval wel wat interessants te kunnen doen. Het viel op dat de technische achtergrondkennis wel voldeed, maar dat voornamelijk het samenwerken in een groep met meerdere disciplines even wennen was. Ook veel praktische zaken, zoals kennis van ontwikkelmethodes, miste Arjan nog.

Gelukkig was de begeleiding bij Océ prima; een ervaren coach legde telkens duidelijk uit wat wel en wat niet

m'n eentje lukt het ook wel' kom je er bij Océ niet.

Aangezien het een platte organisatie betreft met slechts drie lagen (researchmedewerkers, afdelingshoofden en directie) is het geen probleem om even bij de baas binnen te lopen.

Erg leuk aan werken bij Océ vindt Arjan bovendien de vrijheid en met name het multidisciplinaire karakter. Dit karakter komt voornamelijk doordat er aan software voor apparaatbesturing gewerkt wordt. Der-

Mocht je interesse hebben in een kennismaking met Océ, dan is het bijna altijd mogelijk om er een stage- of afstudeeropdracht uit te voeren. De nieuwe, steeds geavanceerdere apparatuur zal alleen maar meer uitdagingen gaan bieden voor de informaticastudent. ■

Mark Timmer  
[m.timmer@student.utwente.nl](mailto:m.timmer@student.utwente.nl)

“Met ‘in m'n eentje lukt het ook wel' kom je er bij Océ niet”

handig was. Daarnaast heeft Arjan in de loop van de tijd zo'n vijftien cursussen gevolgd, waarbij zowel technische aspecten als persoonlijke ontwikkeling aan de orde kwamen.

Op een gemiddelde werkdag zit men bij Research & Development redelijk veel achter de computer. Er wordt immers veel kennis vergaard, die uiteraard gedocumenteerd dient te worden. Het leukst is volgens Arjan echter de afstemming van projectgerelateerde zaken met collega's. Met soms wel 100 man in een project kan dit tot hevige, maar daardoor zeer interessante discussies leiden. Met ‘in

gelijke software heeft vaak strikte tijdseisen, bijvoorbeeld om bij een printer die op hoge snelheid vele afdrucken maakt de beelden perfect op de vellen papier te leggen. Dit wordt mede mogelijk gemaakt door ‘state of the art’ softwareontwikkeling. Dergelijke problemen zullen niet alleen maar door middel van software opgelost kunnen worden, maar moeten ook met de mechanische en de elektrotechnische afdelingen overlegd worden. Dit zorgt voor een brede technische ontwikkeling, die het werk erg uitdagend maakt.



Océ VarioPrint 6250 met revolutionaire Gemini Instant Duplex technologie

Het internet ontsluit een wereld aan nieuwe mogelijkheden: communiceren op grote afstand is nog nooit zo makkelijk en goedkoop geweest, informatie vinden en alles wat je maar bedenken kan bestellen is maar 'a mouse click away'. Toch, of beter: juist daarom, dien je een relatie te ontwikkelen met het internet. Dit artikel is een blik op het internet in al zijn diversiteit die gekleurd wordt door de levenskunst.



## Levenskunst & e-life

### e-government ook voor u/jou

#### Levenskunst

Levenskunst is de filosofische stroming die zich bezighoudt met de kunst van het leven, waarbij je kunst dient te lezen als vaardigheid: hoe verhoud je je tot de zaken die in jouw leven op jou afkomen, hoe ga je met die dingen om? Een voorbeeld hiervan is iets heel banaals als gewoontes: de mens is een gewoontedier, en dat is maar goed ook omdat dat je ontslaat om over alles wat je doet na te denken. Als je echt overal over na zou moeten denken zou dat niet alleen heel vermoeiend voor je zijn maar ook heel tijdrovend. Tegelijkertijd dien je wel zo flexibel te zijn dat als er een gewoonte doorbroken wordt je niet volledig in de stress schiet maar de gewoonte aan te passen aan de veranderde omstandigheden. [SCH01]

#### Snelle communicatie

Een van de vele voordelen van het internet is de e-mail: deze vorm van communicatie maakt "de levenswijze mogelijk van een stoïsche inkeer in jezelf, terwijl de communicatie met de hele wereld in stand blijft", zo zegt Schmid in [SCH04]. In gewone mensentaal: je kunt in een hutje op de Mookerhei gaan zitten en toch communiceren met de wereld. De post gaat met lichtsnelheid door de ruimte, maar laat je toch vrij in je tijd: je opent de post als jou dat uitkomt.

Zoals bekend heeft elk voordeel zijn nadeel: e-mail mist het persoonlijke van een brief, mist die exotische postzegel, mist het gevoel van spanning bij het openen van een onwillige envelop. Het gemak van e-mail verleidt je ook tot lichtzinnigheid en ondoordachte uitdrukkingen, terwijl je bij het schrijven van een brief veel meer de inhoud overdenkt. De getypte tekst houdt geen rekening met de ontvanger, het wordt een snel neergetypte tekst die zo vluchtig is dat de ontvanger niet voelt dat er waardering uit de mail blijkt. Als uitvloeisel van het gebruik van e-mail ontstaan allerlei vormen van ontbinding, Schmid constateert er drie: technische, omdat de wereld via virussen en spam met behulp van de e-mail je leven binnenkomen; menselijke, vanwege de zekerheid dat alle privé-communicatie in feite openbaar is, en historische, want in de toekomst zal men zich afvragen waarom de mensen in het elektronische tijdperk zo weinig met elkaar communiceerden omdat er niets bewaard is gebleven.

Internet is een nieuwe wereld waarin de mens van de 21e eeuw zijn e-life leeft: de zo ontstane wereldgemeenschap is de grootste polis die er ooit is geweest, en het forum is het internet. Discussiëren gebeurt in chatrooms en dergelijke, als een moderne agora: snelle communicatie, die een heel eigen etiquette heeft. Omdat je

niet weet hoe zinnen overkomen bij mensen, je mist de stembuigingen en lichaamstaal, dien je een virtuele fijngevoeligheid te ontwikkelen zodat je niemand onbedoeld tegen de schenen schopt. Een etiquette opschrijven is een lastige zaak en de pogingen die gewaagd zijn, zijn een moderne variant van een eeuwenoud gezegde: Mattheüs 7:12a+b formuleert het als "Alles nu wat gij wilt, dat u de mensen doen, doet gij hun ook aldus" [NBG51], terwijl Kant flink wat eeuwen later in zijn categorische imperatief stelde "Handle nur nach derjenigen Maxime, durch die du zugleich wollen kannst, dass sie ein allgemeines Gesetz werde" [KAN88]; de moderne chat-variant wordt dan zoiets als "Bejegen andere chatters altijd met respect en beleefdheid. Dan zullen ze jou ook met respect en beleefdheid behandelen".

#### Het world wide web (www)

Ook in het www schuilt een gevaar voor jezelf: je kunt verslaafd aan internet raken zoals je kunt verdwalen in de ruimte van de fantasie; we zijn aan de transformatie daarvan in de ruimte van de technologie niet gewend omdat het ruimtelijke aspect dat vaak nog verbonden is met bijvoorbeeld informatieinwinning (naar een bibliotheek gaan, een boek lezen, dat kost tijd) getransponeerd is naar een muisklik. Je kunt dan het internet in-

gezogen worden zonder dat je er erg in hebt en verdwalen in die enorme virtuele ruimte, met als gevolg dat je de echte ruimte verwaarloost.<sup>1</sup> Door de informatieoverload van het www en de communicatie via mail en chat is er een, zoals Schmid dat noemt, ‘virtuele relatie met jezelf’ mogelijk geworden, volledig gemedieerd door techniek, waardoor je een soort van cyborg wordt.

Een soortgelijk gevaar is dat je jezelf verliest in een avatar. Door je voor te doen als iemand anders op het net is de kans aanwezig dat je je niet-virtuele identiteit kwijtraakt en een soort van identiteitscrisis ontwikkelt. Ook het omgekeerde komt voor, en dat nog veel vaker: doordat er veel websites zijn als CU2.nl, orkut.com, hyves.nl en noem allemaal maar op word je verleid je hele hebben en houden op het internet te parkeren, tot op de

Het woord avatar komt van oorsprong uit het Sanskrit en betekent afdaling (vanuit de hemel); het is een incarnatie van een Hindu-god, vooral Vishnu, in menselijke of dierlijke vorm. Een citaat van [WIK05]: “Volgens bepaalde hindoe stromingen (met name binnen het Vaishnavisme) zou Vishnoe tien keer op aarde zijn geweest, in de vorm van tien zogenaamde avatars (speciale incarnaties van God), om de wereld en de mensheid in verschillende perioden van de geschiedenis van de ondergang te redden. De tien avatars van Vishnoe zijn volgens deze stromingen:

1. Matsya, de vis
2. Kurma, de schildpad
3. Narasimha, de man-leeuw
4. Varaha, het wilde zwijn
5. Vamana, de dwerg
6. Parusha-Rama, de krijger
7. Rama, de ideale man
8. Krishna, de god van de bhakti yoga
9. Boeddha
10. Kalki, de ruiter op het witte paard, die nog moet komen”



Tegenwoordig wordt het woord avatar gebruikt om een klein plaatje aan te duiden dat iemand gebruikt om zich af te beelden: dit kan zowel een foto van de persoon zelf zijn als iets totaal anders.

als Supermario levens kan verzame-

gen is verwijs ik u naar de film “The Lawnmower Man”, die zich voor een groot deel afspeelt in cyberspace en meteen toont dat je je zeker niet zonder gevaar kan digitaliseren.

### “Hieraan ligt een gevoel van onsterfelijk willen zijn aan ten grondslag”

meest intieme details toe, waardoor iedereen alles over je te weten kan komen. Hieraan ligt een gevoel van onsterfelijk willen zijn aan ten grondslag: in eerste instantie niet lichamelijk, maar door wel iets blijvends na willen laten. Tegelijkertijd zie je een technisch streven naar onsterfelijkheid. René Munnik ziet het als volgt op [DIJ02\08]: “in de Middeleeuwen was de onsterfelijkheid van de ziel een uitgangspunt in wetenschap, in religie, in alles; dit lijkt met de boeken van Julien Offray de la Mettrie, “L’homme machine” en “Histoire naturelle de l’âme”, afgeschafte te zijn, maar stiekem probeert het verwetenschappelijkt denken een soort van onsterfelijkheid terug te halen: Nietzsche signaleert dat vooral in de medische vooruitgang van die tijd, nu is dat vooral via informatietechnologie (IT) en virtual reality (VR), dat enkel een indruk is, maar waarin je niet kunt stervensterven (...). Zo-

len, is er een soort van metafysisch verlangen van opnieuw beginnen.” Dit door techniek gemedieerde onsterfelijkheidsverlangen zie je terug in cyberpunk: een literaire stroming die zich afspeelt op een wereldwijde cyberspace waar mensen zich letterlijk aan hangen en zich in kunnen voortbewegen. In de roman “Neuromancer”, geschreven door William Gibson, de vader van de cyberpunk, beschrijft hij het begrip “Cyberspace”: “a consensual hallucination experienced daily by billions of legitimate operators, in every nation, by children being taught mathematical concepts... A graphical representation of data abstracted from the banks of every computer in the human system. Unthinkable complexity. Lines of light ranged in the non-space of the mind, clusters and constellations of data. Like city lights, receding...” (citaat afkomstig van [IRV97]). Omdat cyberspace niet goed uit te leg-

Het zichzelf blootgeven op internet gebeurt niet alleen bewust, maar nog veel meer en vaker onbewust, door middel van cookies, spyware en meer van dergelijke zaken.

Uiteraard, en daar ben ik vrijwel aan voorbij gegaan, heeft het internet ons heel veel voordelen opgeleverd. Het punt dat de levenskunst wil maken is dat je vaardigheid in de omgang met jezelf in de virtuele ruimte dient op te bouwen: weet nog steeds in de informatiestroom virtuoos te zijn door richting te kiezen in de mogelijkheden van een informatiestroom en niet te lang te treuren over de verloren mogelijkheden. Structureer je eigen tijd in de ruimte van het alledaagse leven, zodat de virtuele tijd daar een eigen plek in krijgt. Doe je dat niet, dan lijkt je overgeleverd aan een nieuwe vorm van cartesianisme – alle informatietechnologie is een verschijningsvorm van de res cogi-

<sup>1</sup> Bovendien is informatieinwinning via internet een precare zaak: iedere Jan Doedel kan van alles op het www zetten en controle is er amper. Dit vraagt een heel kritische blik van de zoeker van informatie, meer dan in een bibliotheek.

tans, waar tegenover de res extensa alle betekenis verliest, zelfs in de relaties met de medemens.

Door het alledaagse leven in te richten kan men ook met verschillende identiteiten op internet overweg, zonder dat zij meteen de overhand krijgen in dat alledaagse leven; kan men in een gesprek via het internet tegelijk een gesprek met zichzelf voeren door je gedachten te overdenken en daarmee een goed gesprek te voeren waarbij de ander begrijpt wat je nu precies wilt zeggen. Ook de virtuele communicatie is immers een mogelijkheid tot inspiratie van het zelf, zolang de werkelijke wereld met al zijn aandachtspunten als mimiek, gebaren, beweging, kleding en dergelijke niet verwaarloosd wordt.

Als dit allemaal zijn vorm krijgt, wordt de virtuele relatie met jezelf weer een reële. Zo is e-government niet alleen iets voor regeringen en bedrijven, maar ook voor uzelf, om zo niet alleen een consumer van het internet te zijn, maar een user. ■

Jan-Willem Dijkshoorn  
Filosoof

### Referenties

[DIJ02\08] J.W. Dijkshoorn, Meedenken met moderne filosofen, 161662, achtste college, <http://home.student.utwente.nl/j.w.dijkshoorn/mmf/college8.html>, 2002, geraadpleegd op 7 augustus 2006

[IRV97] Martin Irvine, Postmodern Science Fiction and Cyberpunk, Po-Mo SF: William Gibson's Neuromancer and Post-Modern Science Fiction, <http://www.georgetown.edu/faculty/irvinem/technoculture/pomosf.html>, 1997, geciteerd op [DIJ02\08]

[KAN88] Immanuel Kant, Kritik der praktischen Vernunft, 1788, geciteerd op [WIK06]

[NBG51] Nederlands BijbelGenootschap: Bijbel, 16e druk, Nederlands BijbelGenootschap, Haarlem, 1991

[SCH01] Wilhelm Schmid, Filosofie van de levenskunst, inleiding in het mooie leven, eerste druk, Ambo, Amsterdam, 2001

[SCH04] Wilhelm Schmid, Handboek voor de levenskunst, eerste druk, Ambo/Anthos Uitgevers, Amsterdam, 2004

[WIK01] GianVella, Immanuel Kant, [http://de.wikipedia.org/wiki/Immanuel\\_Kant](http://de.wikipedia.org/wiki/Immanuel_Kant), 2001, geraadpleegd op 7 augustus 2006

[WIK05] Andries Krugers Dagneaux, Vishnoe, <http://nl.wikipedia.org/wiki/Vishnu>, 2005, geraadpleegd op 7 augustus 2006

“Als dit allemaal zijn vorm krijgt, wordt de virtuele relatie met jezelf weer een reële”



**ADVERTENTIE**

**Deloitte 2**

Overall zijn er tegenwoordig referenties naar het zogenaamde Web 2.0-principe. Steeds vaker komen er websites die zijn voorzien van fancy grafische effecten en handige principes als Ajax, waar in de vorige I/O Vivat al het één en ander over te lezen viel. Bovendien springen de frameworks die het ontwikkelen van zulke webapplicaties makkelijk moeten maken als paddestoelen uit de grond. De vraag rijst of het allemaal wel een revolutie is

en of het niet meer is dan de zoveelste web-hype die we over een paar jaar weer helemaal vergeten zijn.

## Web 2.0 voor dummies

### Historie

De term Web 2.0 werd voor het eerst gebruikt door O'Reilly media terug in 2004. In die tijd bedoelde O'Reilly vooral de nieuwe generatie websites die waren gericht op samenwerking en het delen van informatie. Voorbeelden hiervan zijn de wiki's (denk aan Wikipedia) en de sociale netwerksites (denk aan del.icio.us).

Een belangrijk onderdeel van Web 2.0 is dus het delen van informatie. Niet zo gek dus dat vooral in het begin de term vaak door de war werd gehaald met het 'Semantic Web'. Dit principe, wat momenteel in ontwikkeling is bij 'de bedenker van het web' – Tim Berners-Lee van het World Wide Web-Consortium (W3C), betekent vooral dat alle informatie op het web begrijpelijk moet zijn voor computers zodat die gemakkelijk alle data kunnen analyseren en doorzoeken. Het gewone web is op het moment vooral geschreven voor mensen, en een computer heeft vaak hulp nodig van zijn gebruiker om data te onderscheiden en daar bepaalde acties op te ondernemen. Informatie die goed door systemen te begrijpen is begint mondjesmaat steeds meer op het web tevoorschijn te komen. Op dit moment zijn vooral de zogenaamde RSS-feeds daar een mooi voorbeeld van. Voor meer informatie over het 'Semantic Web' kan je kij-

ken op Wikipedia waar veel informatie en voorbeelden te vinden zijn.

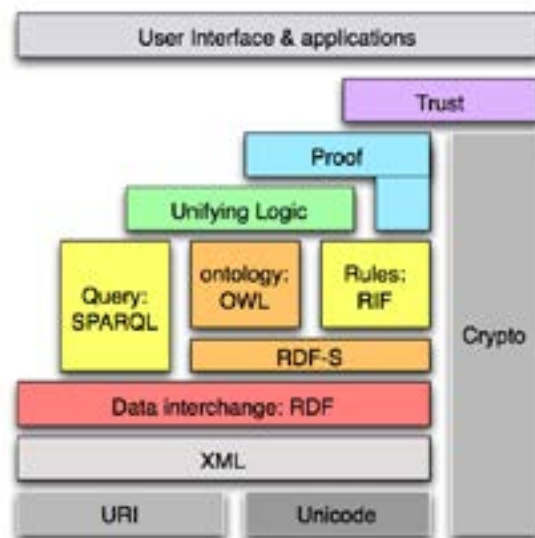
Tegenwoordig staat Web 2.0 vooral synoniem voor een zogenaamde "rich internet application" (RIA). Webapplicaties die asynchroon met de server communiceren en de traditionele *look and feel* van een desktop applicatie naar een webpagina brengen, vaak inclusief de grafische effecten en mogelijkheden die daarbij horen. Het moet dus gezegd worden dat een webapplicatie met een dergelijke interface niet per definitie een Web 2.0-applicatie is, het zijn vooral de zaken als het delen van informatie (tussen gebruikers, maar ook tussen computers) en de mogelijkheid tot samenwerken die een applicatie tot een echte Web 2.0-applicatie maken.

### Hoe werkt het dan?

Veel van deze websites zijn gebaseerd op technieken die feitelijk al jaren bestaan, maar eigenlijk pas recentelijk door veel ontwikkelaars zijn omarmd. Nieuwe browsers en snellere internetverbindingen geven weliswaar meer mogelijkheden, maar feitelijk is er de laatste jaren weinig aan de technologie zelf veranderd. In dit stukje gaan we in op enkele belangrijke technieken die we bij nieuwe applicaties regelmatig tegenkomen en geven we een korte beschrijving van de werking.

### Asynchrone communicatie

Om het desktop paradigma steeds meer te laten terugkomen op het web is het belangrijk dat de applicatie gegevens kan uitwisselen met een server om zodoende dynamisch delen van de webpagina aan te passen zonder dat de gebruiker daarvoor de complete pagina moet verversen. Het principe van delen van informatie vereist immers vaak dat de informatie centraal wordt opgeslagen, en gezien de grote hoeveelheden data waar we het over hebben zou het bijzonder inefficiënt zijn om dit elke keer in zijn geheel naar de client over te sturen.



Figuur 1: W3C semantic web stack

Voor deze vorm van communicatie wordt al tijdens het buzzword 'Ajax' gebruikt. Dit staat voor "Asynchronous Javascript and XML". Wat dit in feite betekent is dat een speciale functie binnen de browser onderhuids een HTTP request doet aan de server alsof je een gewone webpagina laadt. De server geeft dan bepaalde content terug, dit kan XML zijn, maar ook platte tekst. Deze functie is in de meeste gevallen een standaard Javascript object (in Internet Explorer is het een aanroep aan

de informatie zelf indelen zodat die later gemakkelijk door anderen en computers te begrijpen is. We gaan in dit onderdeel kort kijken naar twee veelgebruikte methodes die op dit moment in opkomst zijn.

#### 'Folksonomies'

Een 'folksonomy' is een op het internet gebaseerde 'information retrieval' methode en bestaat hoofdzakelijk uit labels die gegenereerd worden door een samenwerkingsverband van mensen. Deze labels worden gebruikt om webpagina's, multimedie-

- OWL (Web Ontology Language) geeft meer mogelijkheden om objecten (uit RDF) te classificeren. Bovendien kan in dit data-model uitgebreider de relatie tussen verschillende objecten worden beschreven. Voorbeelden van zaken die met OWL gemodelleerd kunnen worden zijn: kardinaliteit, gelijkheid en karakteristieken.

In figuur 1 is de relatie tussen de drie onderdelen te zien. Projecten waar men op het moment mee bezig om informatie op deze manier te beschrijven zijn er nog niet veel. De eerste stapjes worden echter al wel gezet, bijvoorbeeld binnen W3C Scientific Publishing taskforce, die zich bezig houden met het koppelen en integreren van wetenschappelijke informatie en artikelen, wat hierdoor beter en makkelijker toegankelijk moet worden.

"Een belangrijk onderdeel van Web 2.0 is het sociale karakter ervan."

een ActiveX object) met de naam XMLHttpRequest(); Kort door de bocht bestaat dit object uit een aantal vaste functies waarmee een standaard HTTP request kan worden samengesteld. Mocht de browser data terugkrijgen dan voert het object een door de ontwikkelaar gedefinieerde functie uit waarmee de verkregen data verwerkt kan worden. De teruggegeven data is vaak een XML document-object wat dan weer heel gemakkelijk met de ingebouwde Document Object Model (DOM) functies van javascript te verwerken valt. Zie de volgende pagina voor het voorbeeld.

Uiteraard zijn er meer mogelijkheden om data asynchroon van een server te vergaren. Populaire methodes tegenwoordig omvatten het gebruik van technologieën als Flash en Java applets, maar ook het gebruik van verborgen frames waarin de server informatie kan stoppen zijn populair.

#### Information sharing

Een belangrijk onderdeel van Web 2.0 is het sociale karakter ervan. Maar om informatie tussen de vele diensten te kunnen delen moeten alle schakels in de keten begrijpen waar het over gaat. Op bekende sites verschijnen steeds meer technieken waarbij gebruikers van de website

ale content en links te omschrijven. Veel mensen zullen al bekend zijn met dit concept onder de naam van 'tagging'. Een folksonomy is in zekere zin het omgekeerde van een taxonomie, waar classificaties systematisch (hiërarchisch) worden opgeslagen en waar de labels niet door de gebruikers maar door een kleine groep mensen worden vastgesteld.

Voorbeelden waar gegevens aan de hand van een folksonomy zijn opgeslagen zijn websites als del.icio.us (weblinks), last.fm (muziek) en Flickr (foto's).

#### RDF, OWL en XML

Deze drie afkortingen zullen velen al bekend voorkomen, maar welke rol spelen ze in web 2.0? Dat is vrij simpel: samen vormen ze het hart van het semantische web. Elk hebben ze een andere taak binnen het systeem wat moet leiden tot een autonome samenwerking van informatiesystemen:

- XML geeft een vaste syntax waarin de documenten die onderling worden uitgewisseld kunnen worden opgesteld.

- RDF (Resource Descriptor Framework) is een data-model waarin objecten (resources) en hun onderlinge relatie kunnen worden beschreven. Een RDF document kan worden opgesteld in een XML syntax.

#### Valkuilen en problemen

Het klinkt allemaal heel mooi en aardig, maar (uiteraard) is niet alles even mooi als het lijkt. Met het delen van al deze informatie en het gebruik van RIA's zitten ook gevaren verborgen die voor de huis-tuin-en-keuken-gebruiker misschien minder voor de hand liggen, maar die in de toekomst wellicht een grote rol gaan spelen. We zullen hier ingaan op de problemen waar ontwikkelaars zoal tegenaan lopen, en zullen indien mogelijk enige oplossingen bekijken.

#### Toegankelijkheid

Misschien voor de hand liggend, maar helaas wordt er door veel ontwikkelaars nauwelijks aandacht besteed aan toegankelijkheid voor invalide gebruikers, en dat zal in de toekomst met dynamische pagina's alleen maar lastiger worden. Het vertalen van een standaard website naar iets wat een brailleapparaat kan gebruiken kan al duivels ingewikkeld zijn, een Ajax-based webpagina die continu veranderd vertalen is schier onmogelijk.

Toch kan een ontwikkelaar met simpele trucs zijn website makkelijker

```
var request;
function loadRequest(url) {
    request = false;
    // Is het Mozilla of Internet Explorer?
    if(window.XMLHttpRequest && !(window.ActiveXObject))
    { //Het is Mozilla, Opera etc.!
        try
        {
            request = new XMLHttpRequest();
        }
        catch(e)
        {
            request = false;
        }
    }
    else if(window.ActiveXObject)
    { //Het is Internet Explorer!
        try
        {
            request = new ActiveXObject("Msxml2.XMLHTTP");
        }
        catch(e)
        {
            try
            {
                request = new
                ActiveXObject("Microsoft.XMLHTTP");
            }
            catch(e)
            {
                request = false;
            }
        }
    }
    if(req)
    {
        //Stel de javascript functie in die de gegevens verwerkt
        request.onreadystatechange = processReqChange;
        //Open de verbinding
        req.open("GET", url, true);
        //Je kan hier eventueel parameters (of XML) meegeven
        req.send("");
    }
}

//De functie die de gegevens verwerkt
function processReqChange() {
    // Ga alleen verder als het request geladen is:
    if (req.readyState == 4) {
        // En de request 'Okay' is:
        if (req.status == 200) {
            // Doe je hier je ding
        } else {
            alert("Er was een probleem:\n" +
                req.statusText);
        }
    }
}
}
```

Figuur 2: Voorbeeldcode





toegankelijk houden. Zo is het een goede richtlijn om alle content die dynamisch (door Javascript of Ajax) wordt aangeleverd ook beschikbaar te laten zijn via een synchrone verandering van een pagina. In het geval van een link die naar een Javascript functie wijst (bijvoorbeeld om een menu uit te klappen) moet je er dus altijd voor zorgen dat de originele link behouden blijft als er geen javascript beschikbaar is. Dit kan je doen door het gebruik van het aanroepen van de javascript middels een onclick

onconsistente documenten en vaak is nuttige informatie daardoor door een computer niet makkelijk uit het document te halen.

### Privacy

In een situatie waarin alle informatie door systemen aan elkaar gekoppeld wordt zoals dat bij het semantic web het geval is, kan je vraagtekens zetten bij de privacy die overblijft voor ieder die zijn informatie publiceert op het web. Dit is niet zo zeer een technisch probleem zoals het voorgaande, maar

war gehaald wordt met het leuk aankleden van je website.

Doordat de term Web 2.0 tegenwoordig regelmatig wordt gebruikt als reclametruc en daarmee nogal een hype is geworden, heeft het bij veel kenners en ontwikkelaars een vieze smaak in de mond achtergelaten. Dat is zonde, want de kracht van het idee, de openheid en de toegankelijkheid, is iets wat het internet juist zo'n succesvol platform heeft gemaakt. Het grootste nadeel van Web 2.0 zit eigenlijk al in de naam: Het is een evolutie maar geen revolutie. Je kan belangrijkste pijlers van Web 1.0 nou eenmaal niet helemaal weggooien en compleet overnieuw beginnen als je dat nodig zou vinden.

**"Zeker op het gebied van beleving en het uiterlijk van websites is er flink wat veranderd."**

event, terwijl de URL in de href-parameter gewoon behouden blijft: zie figuur 3.

Zorg er bovendien voor dat belangrijke functies van je applicatie ook beschikbaar zijn door links toegankelijk te maken via toetsen van het toetsenbord. Dit kan bijvoorbeeld doormiddel van de `accesskey="C"` eigenschap. Als laatste moet je opletten dat de standaard werking van je browser behouden blijft. Op het moment dat er veel informatie veranderd op een pagina moet een gebruiker ook terug kunnen door gebruik te maken van de 'Terug' knop van de browser. Helaas zien we dat het gebruik van Ajax en Rich Internet Applications steeds vaker ten koste gaat van dit soort basisfunctionaliteit waardoor het voor de gewone gebruiker alleen maar ingewikkelder wordt.

Critici van Web 2.0-applicaties menen ook dat de navigatiestructuur van zulke pagina's vaak ondoorzichtig of ingewikkeld is. Buiten het feit dat zulke applicaties voor veel mensen dus lastig te bekijken zijn, is het ook voor computers lastig om informatie te halen uit zulke dynamische applicaties. De vele onbruikbare `<script>` tags en lastig (via Ajax) bereikbare informatie zorgen voor

het is wel iets waarbij ontwikkelaars rekening moeten houden. Op dit moment is het vrij simpel om anonim te blijven of om je tekst te maskeren, als alles aan elkaar gekoppeld is wordt dit een stuk lastiger. Er zijn al voorbeelden waarbij het koppelen en doorzoeken van informatie de nodige wenkbrauwen heeft doen rijzen (zoals het doorscannen door Google van de mail van hun gebruikers om gerichte advertenties te kunnen aanbieden). Een bijkomend probleem kan zijn dat het censureren van informatie een stuk makkelijker wordt doordat alle 'gevaarlijke' content veel makkelijker door een automatisch systeem kan worden herkend.

### Wat te doen met Web 2.0?

Veel mensen zullen het er mee eens zijn dat de nieuwe generatie aan internetapplicaties een stap vooruit is vergeleken met alles wat we afgelopen jaren gewend waren. Zeker op het gebied van beleving en het uiterlijk van websites is er flink wat veranderd. Helaas is het niet zo makkelijk dat we alles wat nieuw en fancy is zomaar kunnen 'taggen' met de term Web 2.0. Web 2.0 is namelijk vooral een visie van samenwerking en het delen van kennis die te vaak door de

Hopelijk heeft dit artikel (als dat überhaupt al nodig was) wat indruk gegeven van de basiselementen van het Web 2.0-principe. ■

Remko Nolten

[remko.nolten@inter-actief.net](mailto:remko.nolten@inter-actief.net)

Kijk als je meer wilt weten vooral even op de volgende sites:

- [http://en.wikipedia.org/wiki/Web\\_2.0](http://en.wikipedia.org/wiki/Web_2.0)
- <http://deli.cio.us/> (Folksonomy gebaseerde links-verzameling)
- <http://script.acolu.us/> (framework voor het ontwikkelen van Rich Internet Applications)
- <http://last.fm/> (Sociaal netwerk gebaseerd op ieders muzieksmaak)
- <http://www.w3.org/TR/WCAG10-HTML-TECHS/> (technieken voor het bruikbaar houden van websites)
- <http://www.techcrunch.com/> (Internationaal Web 2.0 blog)
- <http://www.w3.org/2001/sw/> (Uitleg en specificaties van het Semantic Web)

```
<a href="somehost.com?action=klapmenuuit" onclick="klapMenuUit(); return false;" >Menu item</a>
```

Figuur 3: Voorbeeldlink

Dit artikel is eerder verschenen in Objective 5, het magazine over innovatie en technologie van Technolution B.V.



## Nieuwe coalitie regelt ochtendspits

### Van lokaal verkeersmanagement naar proactief en gebiedsomvattend verkeersmanagement

In Nederland wordt lokaal verkeersmanagement al enkele decennia met succes ingezet om de veiligheid en doorstroming van het wegverkeer te vergroten. Door de toenemende drukte groeit de behoefte aan proactief en gebiedsomvattend verkeersmanagement. Bovenop de vele onzekerheden in het verkeer brengt gebiedsomvattend samenwerken gedeelde verantwoordelijkheden met zich mee. Dit stelt ons voor de uitdaging om tegen beheersbare kosten een flexibel inzetbaar, modulair “verkeersmanagementsysteem” op te bouwen, een uitdaging waar ook andere domeinen met vergelijkbare vraagstukken als bijvoorbeeld defensie of de logistieke sector zich voor gesteld zien. Kunstmatige intelligentie is een technologie waarmee deze uitdaging kan worden aangegaan. Deze technologie kan in doordachte modules worden ondergebracht en in de bestaande verkeersregelsystemen worden geïnstalleerd. Daarmee wordt bereikt dat met de toevoeging van deze technologie bestaande systemen en installaties kunnen worden hergebruikt.

De beperkte mogelijkheden om het wegennet uit te breiden, samen met de ieder jaar toenemende groei in het verkeer, leiden tot een situatie waar de kop en de staart van de verschil-

lende files steeds dichter naar elkaar toegroeien op het wegennet. Lokaal verkeersmanagement zal moeten doorgroeien naar proactief en gebiedsomvattend verkeersmanagement.

#### Verkeer is moeilijk voor- spelbaar

“Gebiedsomvattend verkeersmanagement” is door de vele onzekerheden in het verkeer moeilijk te plannen. Natuurlijk is daar de gewone ochtend- en avondspits, die iedere dag weer prominent de radio haalt. De aandacht gaat dan vooral uit naar de files op de autosnelwegen, omdat we daar verwachten vlot en onbelemmerd door te kunnen rijden. Door invloed van slecht weer, grote evenementen, verkeersongevallen, wegwerkzaamheden en andere geplande en niet-geplande gebeurtenissen is er in werkelijkheid zelden sprake van een reguliere spits. Verkeer kent een hoge mate van onvoorspelbaarheid.

Deze onvoorspelbaarheid wordt groot wanneer we bedenken dat de drukte op autosnelwegen niet los kan worden gezien van de drukte op provinciale en gemeentelijke wegen. Maatregelen getroffen op het ene deel van het wegennet werken, zeker bij grote drukte, door op de andere delen. Om de maatregelen zo



effectief mogelijk te laten zijn moet verkeersmanagement zich richten op het gehele wegennet in een bepaald gebied, in plaats van op de afzonderlijke delen. De onvoorspelbaarheid van het verkeer maakt echter dat het op voorhand moeilijk vast te stellen is welke verkeersmanagementsystemen van welke wegbeheerders (verkeersmanagers) wanneer moeten samenwerken.

omvattend “sturen en geleiden” van verkeer.

### Gezamenlijke besluitvorming

Zeker in Nederland draaien veel verkeersmanagementsystemen autonoom. Ze beschikken over eigen sensoren, regelalgoritmes, actuatoren en soms zelfs bedieninterfacés. Wanneer we deze systemen kunnen

managementsystemen open en interoperabel worden gemaakt.

Van openheid is in de eerste generatie(s) van verkeersmanagementsystemen nog niet echt sprake. In deze generatie werd ieder systeem als een zuil vormgegeven, vanaf sensor tot en met actuator. In de tweede generatie zijn maatwerkinterfacés aangebracht om systemen met elkaar te laten communiceren en samenwerken. De openheid die nodig is voor gezamenlijke besluitvorming gaat verder en vraagt om het volledig “netwerken” van de systemen (zie figuur 2).

Het laten netwerken van verkeersmanagementsystemen heeft tot consequentie dat de bijbehorende applicaties moeten kunnen samenwerken, data van iedere sensor kunnen afnemen, bewerkte informatie kunnen uitwisselen en actuatoren kunnen synchroniseren. Dit is nodig om tot een gezamenlijk begrip te kunnen komen, te onderhandelen over een gezamenlijk besluit en deze informatie gezamenlijk te kunnen overbrengen op de weggebruikers.

Een dergelijke vorm van netwerken vraagt om openmaken en gedeeltematig aanpassen van de bestaande systemen. Technisch beschikken we daarvoor met multi-agent en webservicetechnologie over een aardige set hulpmiddelen. Het vraagt wel om een investering: een investering die leidt tot een grotere flexibiliteit en daarmee inzetbaarheid van de bestaande (en nieuwe) verkeersmanagementsystemen. Uiteindelijk zal dit leiden tot een groter rendement uit het arsenaal aan systemen. Hoe spannend deze investering wel niet is volgt uit de figuurlijke wet van Bob Metcalfe: “de waarde van een interactief netwerk neemt toe met het kwadraat van het aantal gebruikers”.

### Evolutionair pad

Komen tot een omgeving waarin autonome verkeersmanagementsystemen spontaan coalities kunnen vormen en zo tot gezamenlijk be-

“AI is een technologie waarmee deze uitdaging kan worden aangegaan”

### Nieuwe inrichting verkeersmanagement

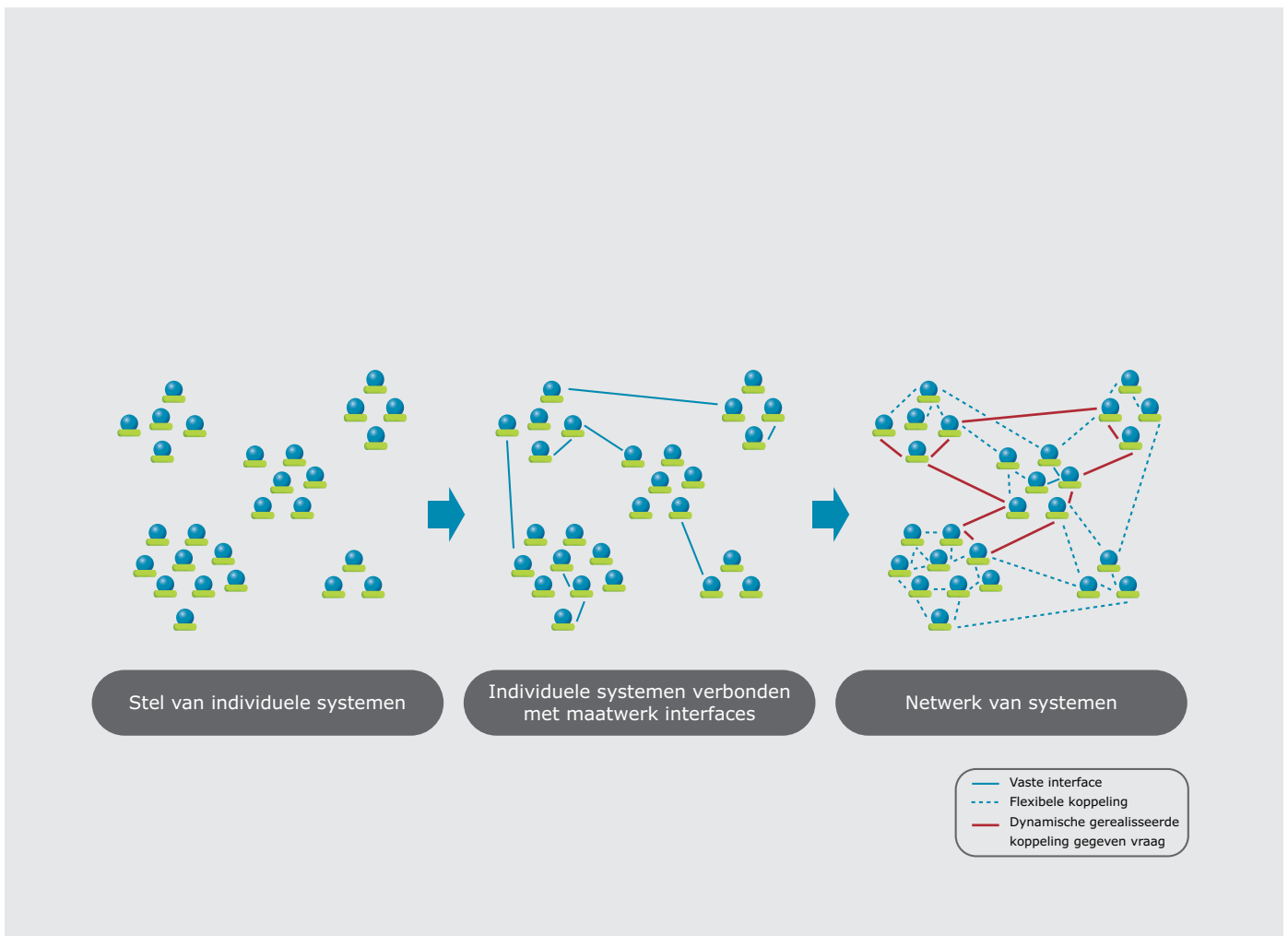
Deze onzekerheden betekenen voor proactief, gebiedsomvattend verkeersmanagement dat de waar, wat, hoe, door wie, met wie en waarmee vragen niet eenduidig kunnen worden beantwoord. Deze vragen stante pede beantwoorden is iets dat nu met het huidige arsenaal aan verkeersmanagementsystemen nog niet kan. Deze paradox vraagt om een nieuwe inrichting van de verkeersmanagementsystemen waarbij de autonome verkeersmanagementsystemen onderling samenwerken en gezamenlijk besluiten nemen rond het gebieds-

laten netwerken, kunnen ze dynamisch “coalities” vormen en binnen zo’n coalitie toewerken naar gemeenschappelijke besluiten (zie figuur 1).

Cruciaal bij de coalitievorming is welke verkeersmanagementsystemen wanneer met elkaar samenwerken. Gedurende de dag kunnen steeds andere “coalities” van verkeersmanagementsystemen nodig zijn om bijvoorbeeld het verkeer in de ochtendspits te “sturen en geleiden”, of naar het strand, naar een evenement, enzovoorts. Om zulke behoefteafhankelijke coalities te kunnen laten ontstaan, moeten de verkeersma-



Figuur 1: Proces van gezamenlijke besluitvorming.



Figuur 2: Van individuele systemen via maatwerkinterfaces naar een netwerk van systemen.

sluitvorming kunnen komen, is een evolutionair proces zonder echt begin of einde. De technische grondslagen kunnen als eerste stap worden neergezet. Een verkeersmanagementsysteem is dan niet langer sec ondersteunend aan één afgebakende verkeersregeling, maar wordt een facilitator ('enabler') voor het vernieuwen van verkeersmanagement. Er is dan ook geen sprake van een nieuw systeem dat "even" wordt gekocht en ingevoerd. Want bestaande verkeersmanagementsystemen moeten worden aangepast en opgenomen in een netwerk dat als technisch platform voor proactief, gebiedsomvattend verkeersmanagement gaat fungeren. Op deze manier kan tegen beheersbare kosten een flexibel inzetbaar, modulair verkeersmanagementsysteem worden gerealiseerd. ■

"Uiteindelijk zal dit leiden tot een groter rendement uit het arsenaal aan systemen"